



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : 3 Rev. : 0
Date : 05/08/2003
Page : i

ROSETTA AOCMS Studies Technical Note

MATRA CMT
DOC ROSETTA

Requie 27 AOUI 2003

Archivage n° 7336

Title

AOCMS On-ground Processing Definition

Part of DRL E-6

	Name and Function	Date	Signature
Prepared by	Jérémie TOUATY Rosetta AOCMS Studies Engineer	5/8/2003	
Approved by	Pascal REGNIER Rosetta AOCMS Studies Manager	14/8/2003	
Authorised by	Michel JANVIER Rosetta AOCMS Design Manager	27/8/03	
Application authorised by	Patrick LELONG Rosetta Avionics Project Manager	27/08/2003	

Document type	Nb WBS	Keywords
---------------	--------	----------

Ref. GED : DSG.NT.OB.3683153.02



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : **05/08/2003**
Page : ii

SUMMARY

This document defines and specifies the AOCMS processing which must be implemented on-ground to operate and to monitor the ROSETTA spacecraft AOCMS during its mission.

Document controlled by



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : **05/08/2003**
Page : iii

DOCUMENT CHANGE LOG

Issue/ Revision	Date	Modification Nb	Modified pages	Observations
Draft	28/06/2002		All pages	First complete issue handed over at the FAR1 review
1/0	15/07/2002		All pages	Update after internal review, with mainly : <ul style="list-style-type: none">- Completion of AFM ground processing- inversion of chapters 9 and 10 (for clarification)- correction of errors in the thruster configuration matrix computation- addition of the blow-down factor effect on the OCM disturbing torques computation The following ground processing algorithms have been added:
2/0	31/03/2003		All pages	<ul style="list-style-type: none">- solar arrays flexible modes characteristics in-flight calibration- AOCMS telemetry parameter fine datation
3/0	05/08/2003		All pages	Update for the new mission to Churyumov-Gerasimenko



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : **05/08/2003**
Page : iv

PAGE ISSUE RECORD

Issue of this document comprises the following pages at the issue shown

Page	Issue/ Rev.										
All	Draft										
All	1/0										
All	2/0										
All	3/0										

ACRONYMS AND ABBREVIATIONS

GENERAL / EQUIPMENT

AOCMS	Attitude and Orbit Control and Measurement System
ACM	Acquisition and Command Unit
AIU	AOCMS Interface Unit
APME	Antenna Pointing Mechanism Electronics
AU	Astronomical Unit
BOL	Beginning Of Life
CAM	Navigation Camera
COM	Center Of Mass
DMS	Data Management System
EOL	End Of Life
ES	Emergency Surveillance
FDIR	Failure Detection, Isolation and Reconfiguration
FM	Functional Monitoring
HGA	High Gain Antenna
IMP	Inertial Measurement Package
LGA	Low Gain Antenna
LSB	Least Significant Bit
MGA	Medium Gain Antenna
MIB	Minimum Impulse Bit
MOL	Middle Of Life
NEA	Noise Equivalent Angle
NES	Non Emergency Surveillance
ORB	On-board Reconfiguration Block
PM	Processor Module
RCS	Reaction Control System
RWA	Reaction Wheel Assembly

ACRONYMS AND ABBREVIATIONS

SA	Solar Array
SADM	Solar Array Drive Mechanism
SADE	Solar Array Drive Electronics
SAS	Sun Acquisition Sensor
S/C	Spacecraft
SM	Standard Monitoring
SRD	Software Requirements Document
STR	Star Tracker
TBC	To Be Confirmed
TBD	To Be Defined
TC	Telecommand
TM	Telemetry
WRT	With Respect To

MODES

AFM	Asteroid Fly-by Mode
NM	Normal Mode
WDP	Wheel Damping Phase
GSEP	Gyro-Stellar pointing on Ephemerides Phase
GLEP	GyroLess pointing on Ephemerides Phase
GSP	Ground Slew Phase
FPAP	Fine Pointing Accuracy Phase
FPSP	Fine Pointing Stability Phase
WOLP	Wheel Off-Loading Phase
NSH	Near Sun Hibernation / Near Sun Hibernation mode
OCM	Orbit Control Mode



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : **05/08/2003**
Page : vii

ACRONYMS AND ABBREVIATIONS

SAM	Sun Acquisition Mode
RRP	Rate Reduction Phase
SAP	Sun Acquisition Phase
SCP	Sun Capture Phase
StAP	Star Acquisition Phase
SPP	Sun Pointing Phase
BPP	Biased Pointing Phase
SBM	Stand-By Mode
SHM	Safe / Hold Mode
EAP	Earth Acquisition Phase
Hold	Hold Phase
EPIP	Earth Pointing Initialisation Phase
EPP	Earth Pointing Phase
WOL	Wheel Off-Loading
SKM	Sun Keeping Mode
RRP	Rate Reduction Phase
SAP	Sun Acquisition Phase
StAP	Star Acquisition Phase
ES	Earth Strobing sub-mode
SAR	Solar Arrays Rotation
ESM	Earth Strobing Motion
EAH	Earth Acquisition and Hold
SpM	Spin-up Mode
TTM	Thruster Transition Mode



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : 05/08/2003
Page : viii

TABLE OF CONTENTS

0	REFERENCE DOCUMENTS	1
1	INTRODUCTION	2
2	MINIMUM SPACECRAFT - EARTH DISTANCE IN SHM	4
2.1	CONTEXT AND PURPOSE	4
2.2	PROCESSING.....	4
2.2.1	<i>Principle</i>	4
2.2.2	<i>Numerical analysis</i>	5
3	RESTRICTIONS OF AUTONOMOUS GUIDING OPTIONS / CORRESPONDING REFERENCE ATTITUDE QUATERNION	8
3.1	CONTEXT AND PURPOSE	8
3.2	RESTRICTIONS OF AUTONOMOUS GUIDING OPTIONS IN NM / GSEP, GLEP & NSH	8
3.2.1	<i>Principle</i>	8
3.2.2	<i>Numerical analysis</i>	9
3.3	REFERENCE ATTITUDE QUATERNION CORRESPONDING TO SELECTED GUIDING OPTIONS	10
3.3.1	<i>Principle</i>	10
3.3.2	<i>Inputs / Outputs</i>	10
3.3.3	<i>Ground processings</i>	11
4	ATTITUDE SLEW POINTING PROFILES	15
4.1	CONTEXT AND PURPOSE	15
4.2	COMPUTATION OF LARGE ANGLE SLEW PROFILES.....	15
4.2.1	<i>Principle</i>	15
4.2.2	<i>Inputs and constraints</i>	16
4.2.3	<i>Outputs</i>	18
4.2.4	<i>Ground processings</i>	18
5	EPHEMERIDES / GROUND GUIDANCE SEGMENTS POLYNOMIALS.....	27
5.1	CONTEXT AND PURPOSE	27
5.2	PROCESSING.....	27
5.2.1	<i>Principle</i>	27
5.2.2	<i>Inputs</i>	28
5.2.3	<i>Outputs</i>	29
5.2.4	<i>Ground processings</i>	30



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : 05/08/2003
Page : ix

6	AFM PARAMETERS	43
6.1	CONTEXT AND PURPOSE	43
6.2	COMPUTATION OF THE SPACECRAFT REFERENCE FRAME TO BE ASTEROID-POINTED	43
6.2.1	<i>Principle</i>	43
6.2.2	<i>Inputs</i>	43
6.2.3	<i>Outputs</i>	44
6.2.4	<i>Ground computation</i>	44
6.3	COMPUTATION OF THE FLY-BY PLANE QUATERNION	44
6.3.1	<i>Principle</i>	44
6.3.2	<i>Inputs</i>	46
6.3.3	<i>Outputs</i>	46
6.3.4	<i>Ground computation</i>	46
6.4	COMPUTATION OF THE ATTITUDE QUATERNION AT AFM ENTRY	47
6.4.1	<i>Principle</i>	47
6.4.2	<i>Inputs</i>	47
6.4.3	<i>Outputs</i>	48
6.4.4	<i>Ground processings</i>	48
6.5	COMPUTATION OF THE ATTITUDE QUATERNION AT AFM EXIT	49
6.5.1	<i>Principle</i>	49
6.5.2	<i>Inputs</i>	49
6.5.3	<i>Outputs</i>	49
6.5.4	<i>Ground processings</i>	50
7	FREQUENCY OF SA RE-ORIENTATIONS IN NSH	52
7.1	CONTEXT AND PURPOSE	52
7.2	PROCESSING	52
7.2.1	<i>Principle</i>	52
7.2.2	<i>Numerical analysis</i>	53
8	REFERENCE ATTITUDE QUATERNION AT SPM ENTRY	57
8.1	CONTEXT AND PURPOSE	57
8.2	PROCESSING	57
8.2.1	<i>Principle</i>	57
8.2.2	<i>Inputs / Outputs</i>	57
8.2.3	<i>Ground processings</i>	58
9	RCS_ORB PARAMETERS	60
9.1	CONTEXT AND PURPOSE	60
9.2	PROCESSING	61



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : 05/08/2003
Page : x

9.2.1	Principle.....	61
9.2.2	Inputs.....	62
9.2.3	Outputs.....	63
9.2.4	Ground processing.....	64
10	ΔV MANOEUVRES PARAMETERS	72
10.1	CONTEXT AND PURPOSE.....	72
10.2	PERTURBATION TORQUES INDUCED BY AN AXIAL ΔV MANOEUVRE	72
10.2.1	Principle.....	72
10.2.2	Inputs.....	73
10.2.3	Outputs.....	73
10.2.4	Ground processings.....	73
10.3	OPTIMAL THRUST DIRECTIONS FOR VECTORED ΔV MANOEUVRES	74
10.3.1	Principle.....	74
10.3.2	Inputs.....	74
10.3.3	Outputs.....	75
10.3.4	Ground processings.....	75
10.4	THRUST RATIO VECTOR FOR VECTORED ΔV MANOEUVRES.....	77
10.4.1	Principle.....	77
10.4.2	Inputs.....	77
10.4.3	Outputs.....	78
10.4.4	Ground processings.....	78
11	ACCELEROMETERS SIZE EFFECTS COMPENSATION.....	80
11.1	CONTEXT AND PURPOSE.....	80
11.2	PROCESSING.....	80
11.2.1	Principle.....	80
11.2.2	Inputs.....	82
11.2.3	Outputs.....	83
11.2.4	Ground processing.....	83
12	STR / CAM MISALIGNMENTS IN-FLIGHT CALIBRATION	84
12.1	CONTEXT AND PURPOSE.....	84
12.2	PROCESSING.....	84
12.2.1	Principle.....	84
12.2.2	Inputs.....	85
12.2.3	Outputs.....	86
12.2.4	Ground processing.....	87
12.3	PROTOTYPING AND NUMERICAL VALIDATION	94



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : 05/08/2003
Page : xi

12.3.1 Assumptions	94
12.3.2 Results	95
12.4 APPENDIX : COMPUTATION OF LINEAR DEPENDENCE COEFFICIENTS FOR OBSERVATION MATRICES	95
13 SPACECRAFT MOMENTS OF INERTIA IN-FLIGHT CALIBRATION	97
13.1 CONTEXT AND PURPOSE	97
13.2 CONSTRAINTS ON THE SLEW	97
13.2.1 General constraints	98
13.2.2 Specific constraints	98
13.3 GROUND PROCESSING	99
13.3.1 Principle	99
13.3.2 Inputs	101
13.3.3 Outputs	102
13.3.4 Ground processing	102
13.4 PROTOTYPING AND ANALYSIS	104
13.4.1 Main characteristics of the reference slew manoeuvre	104
13.4.2 Results obtained using TM data of the reference slew manoeuvre	105
13.4.3 Typical time outputs of the reference slew manoeuvre	106
13.4.4 Sensitivity study	111
14 SOLAR ARRAYS FLEXIBLE MODES CHARACTERISTICS IN-FLIGHT CALIBRATION	113
14.1 CONTEXT AND PURPOSE	113
14.2 THEORETICAL PREDICTION OF THE SOLAR ARRAYS FLEXIBLE MODE CHARACTERISTICS	113
14.2.1 Prediction of the nominal free-free frequencies	113
14.2.2 Influence of the Cantilever frequencies and inertia dispersions on the theoretical free-free frequencies	119
14.2.3 Influence of the difference between the two solar arrays Cantilever frequencies on the theoretical free-free frequencies	120
14.2.4 Prediction of the nominal free-free flexible mode damping ratios	121
14.3 SOLAR ARRAYS FLEXIBLE MODES FREE-FREE FREQUENCIES CALIBRATION PROCESSING	122
14.3.1 Principle	122
14.3.2 Inputs	127
14.3.3 Outputs	127
14.3.4 Ground processing	127
14.4 SOLAR ARRAYS FLEXIBLE MODE FREE-FREE DAMPING RATIO ESTIMATION PROCESSING	130
14.4.1 Principle	130
14.4.2 Inputs	131
14.4.3 Outputs	131
14.4.4 Ground processing	132



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : **05/08/2003**
Page : xii

14.5	PROTOTYPING AND NUMERICAL VALIDATION	135
14.5.1	<i>Simulation of the free-free flexible mode frequencies calibration by ground processing of the measured spacecraft angular rate</i>	135
14.5.2	<i>Simulation of the free-free flexible mode damping ratios calibration by ground processing of the spacecraft angular rate</i>	149
15	AOCMS TELEMETRY PARAMETER FINE DATATION	151
15.1	CONTEXT AND PURPOSE	151
15.2	GROUND PROCESSING.....	155

0 REFERENCE DOCUMENTS

- | | |
|--|-----------------|
| [1] ROSETTA Consolidated Report on Mission Analysis
Churyumov-Gerasimenko 2004
ESOC - issue 1, February 2003 | RO-ESC-RP-5500 |
| [2] ROSETTA System Interface Requirements Specification
Astrium GmbH - issue 6, 15/10/1999 | RO-DSS-IS-1002 |
| [3] ROSETTA Mission Timeline Document
Astrium GmbH - issue 3, 30/09/2000 | RO-DSS-TN-1044 |
| [4] ROSETTA AOCMS Studies Assumptions
Astrium SAS - issue 3, 28/02/2001 | RO-MMT-TN-2029 |
| [5] ROSETTA AOCMS Application Software TM / TC ICD
Astrium SAS - issue 4, 02/08/2001 | RO-MMT-IF-2010 |
| [6] ROSETTA Tuning of AOCMS FDIR parameters
Astrium SAS - issue 3, 05/06/2002 | RO-MMT-TN-2160 |
| [7] ROSETTA Avionics User Manual
Astrium SAS - issue 2 rev. 3 | RO-MMT-MA-2025 |
| [8] ROSETTA Normal Mode Simulation Plan Results
Astrium SAS - issue 1 rev. 1, 30/05/2002 | RO-MMT-TN-2162 |
| [9] Gyro-Stellar Estimation Function Definition and Justification
Astrium SAS - issue 2 rev. 0, 26/02/2001 | RO-MMT-TN-2005 |
| [10] ROSETTA Asteroid Fly-by mode definition and justification
Astrium SAS - issue 2 rev. 0, 26/02/2001 | RO-MMT-TN-2040 |
| [11] AOCMS Software Numerical Requirement Document
Astrium SAS - issue 3 rev. 2, 04/06/2002 | RO-MMT-TN-2138 |
| [12] Change in solar arrays frequencies
Astrium GmbH - 27/07/2001 | RO-DSS-CR-1066 |
| [13] Mars Express Synthesis and trend analysis on IMU FM test results
Astrium SAS - Issue 1 | MEX-MMT-TN-0945 |

1 INTRODUCTION

This document defines and specifies the AOCMS processings which must be implemented on-ground to operate and to monitor the ROSETTA spacecraft during its mission.

On-ground processing is required to compute the parameters of relevant *TeleCommands* (TC) dedicated to AOCMS operations or to process *Telemetries* (TM). The ground processing algorithms necessary for operating the AOCMS are listed in the AOCMS procedures definition of the Avionics User Manual (ref. [7], chapter 7).

The following ground processing algorithms have been identified in these AOCMS procedures :

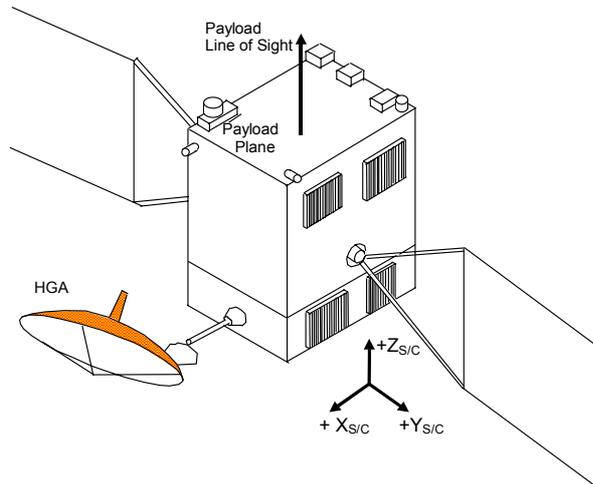
- Minimum spacecraft - Earth distance compatible with an Earth pointing guidance in SHM (§ 2) ;
- Restrictions of autonomous guiding options in NM / GSEP, GLEP & NSH, and corresponding reference attitude quaternion (§ 3) ;
- Large angle slew / pointing profiles (§ 4) ;
- Ephemerides / ground guidance segments polynomials (§ 5) ;
- AFM parameters (§ 6) ;
- Frequency of SA re-orientations in NSH (§7) ;
- Reference attitude quaternion at SpM entry (§ 8) ;
- RCS_ORB parameters (§ 9) ;
- ΔV manoeuvres parameters (§ 10) ;
- Accelerometers size effects compensation (§ 11) ;
- STR / CAM misalignments in-flight calibration (§ 12) ;
- Spacecraft inertia in-flight calibration (§ 13);
- Solar arrays flexible modes characteristics in-flight calibration (§ 14).
- AOCMS telemetry parameter fine datation (§ 15).

Notes :

- The characteristics of every AOCMS TeleCommand parameters are provided in ref. [5].
- Each on-ground processing definition includes a paragraph listing the required entry parameters which are not under AOCMS studies responsibility.
- The following spacecraft reference frame assumptions are extracted from ref.[4]. The spacecraft axes are named $X_{S/C}$, $Y_{S/C}$, $Z_{S/C}$, and form a right-hand orthogonal co-ordinate system, with the origin at the center of the spacecraft / Launcher separation plane (-Z face) :
 - $Z_{S/C}$ is perpendicular to this interface plane with positive sense towards the upper (payload) plane, and is nominally coincident with the payload line of sight ;
 - $X_{S/C}$ is defined by a mechanical reference on the spacecraft structure, and is nominally with its positive sense towards the High Gain Antenna (HGA) mounting plane ;

- $Y_{S/C}$ is such that the $(X_{S/C}, Y_{S/C}, Z_{S/C})$ frame forms a right-handed orthogonal frame, $Y_{S/C}$ being mainly parallel to the direction of the solar arrays.

All parameters in this document are defined with respect to the spacecraft reference frame.



Spacecraft mechanical reference frame

2 MINIMUM SPACECRAFT - EARTH DISTANCE IN SHM

2.1 CONTEXT AND PURPOSE

The selection of attitude guidance options in SHM is performed automatically by on-board algorithms :

- the “ecliptic” option is always selected (the spacecraft Y axis is kept perpendicular to the ecliptic plane),
- the pointing reference (celestial body pointed by the X axis) is set either to “Sun” if the spacecraft - Earth distance is lower than a specific threshold, and to “Earth” if it is larger.

The minimum spacecraft - Earth distance compatible with an Earth pointing guidance must be selected by the ground : a default upper value is stored on-board before launch, but it can be updated by TC during the mission to better match each Sun-centered orbital arc. The minimum Earth distance selection is based on Sun direction considerations with respect to the solar arrays when the Earth-pointed option is used, during the different phases of the ROSETTA mission.

The upper default value endorses all constraints, but it restricts the periods when SHM is authorized to point the Earth. The ground may thus update this parameter in-flight according to the current spacecraft orbital arc, in order to maximize the duration when the Earth-pointed option can be selected in SHM (provided that applicable constraints are fulfilled). The following Sections describe the method to be used for computing the minimum Earth distance which fulfills all AOCMS constraints, i.e. the default upper value, however the proposed method can be used as well for updating the value with respect to a specific single orbital arc.

2.2 PROCESSING

2.2.1 Principle

In order to ensure a correct illumination of the solar arrays during SHM, it is required that the Sun direction never gets more than 5° away from the spacecraft XZ plane (system specification IFPA-800, see ref. [2]). Should this specification be ignored, a maximum 10° pointing error with respect to the XZ plane would be required, in order not to trigger the Sun aspect angle surveillance. The Sun direction within this plane is not important as the solar arrays are then rotated to point towards the Sun.

Hence, the tuning of the minimum distance shall be such that during every phase of the mission where the spacecraft - Earth distance is higher than this threshold, the out-of-XZ plane angle of the Sun direction is always below 5° when the Earth-pointed option is selected. In the following Section, mission analysis data is used to support the selection of the Earth distance fulfilling that constraint.

2.2.2 Numerical analysis

The numerical analysis presented hereafter has been updated for the Churyumov-Gerasimenko mission with a comet rendez-vous at 4 AU with no asteroid fly-by.

provides the spacecraft distance to the Sun and to the Earth during the whole mission (from ref. [1], rendez-vous at 4 AU, with no asteroid fly-by). This figure can be used to check that the zooms on the Earth-spacecraft distances provided by the simulation tool (see Figure 2.2-3) for several mission cruise phases are correct.

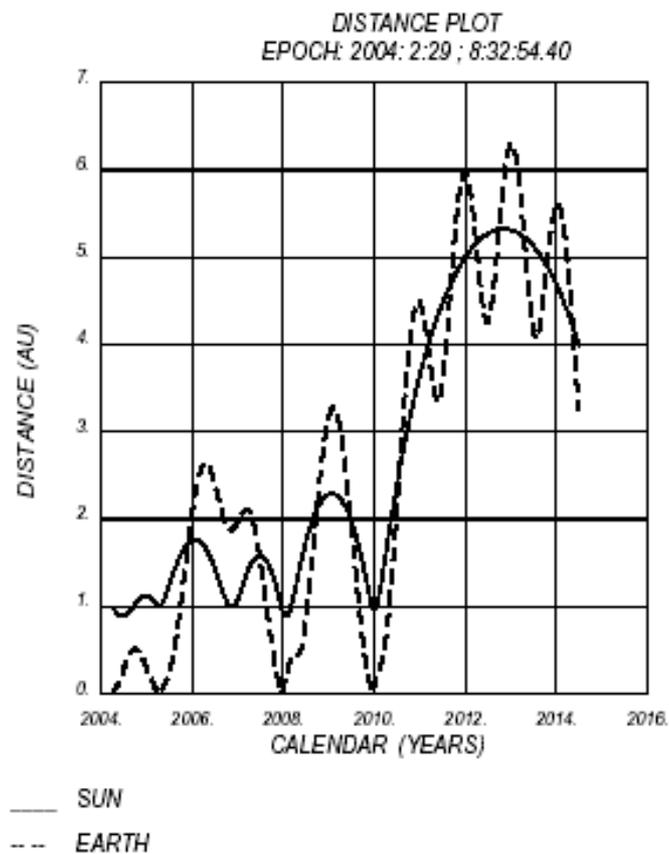


Figure 2.2-1 : Spacecraft distance to Sun and Earth (AU), rendez-vous at 4 AU with no asteroid fly-by

Figure 2.2-3 provides the Earth-spacecraft distance and the out-of-XZ plane angle of the Sun direction during the following cruise phases (obtained from a simulation tool):

- Earth to Mars cruise (from 5/3/2005 to 21/10/2006)
- Mars to Earth cruise (from 28/2/2007 to 14/11/2007)
- Earth to DSH entry cruise (from 12/11/2009 to 10/5/2011)

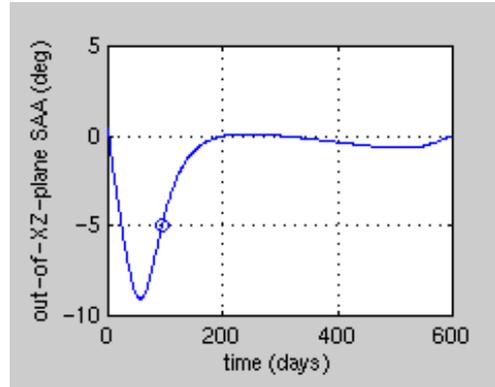
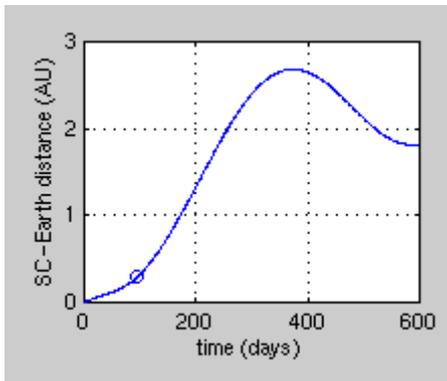
The two Earth-to-Earth cruise phases are not relevant, since the out-of-XZ plane solar aspect angle always remains lower than 5° during these two phases outside the Earth sphere of influence.

An Earth-pointed attitude is assumed, with the spacecraft Y axis perpendicular to the ecliptic plane. **This data is preliminary**, and based on early orbital assumptions provided by ESOC for the new mission to Churyumov-Gerasimenko with a comet rendez-vous at 4 AU, and no asteroid fly-by. It shall be updated by ESOC when the definitive mission profile is available, using the processing described in § 3.3 for reference attitude computations. These plots are used to determine in which conditions the Sun direction requirements mentioned above are fulfilled.

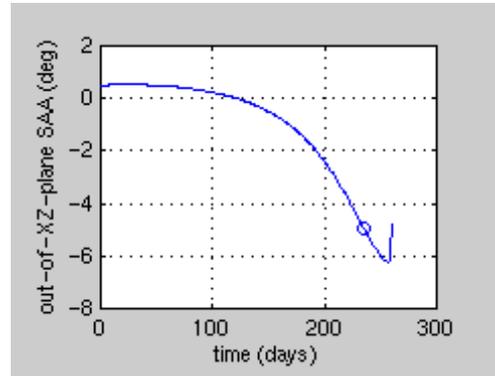
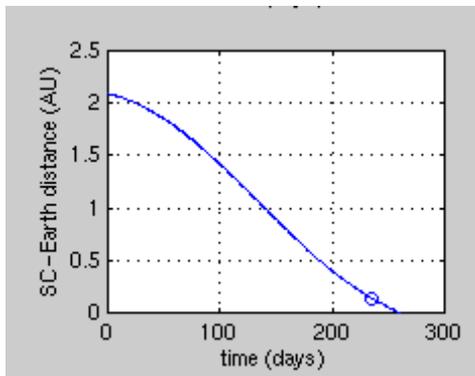
On the first plot of Figure 2.2-3 (Earth to Mars cruise phase), it appears that the out-of XZ plane solar aspect angle remains below 5° after the first 94 days following the beginning of the phase, which corresponds to a spacecraft - Earth distance at this time of 0.29 AU. This value may be selected as threshold in this case, since it ensures that any higher distance corresponds to an out-of XZ plane angle lower than 5° .

On the second plot of Figure 2.2-3 (Mars to Earth cruise phase), the out-of XZ plane solar aspect angle remains below 5° during the first 235 days following the beginning of the phase, which corresponds to a spacecraft - Earth distance at this time of 0.14 AU.

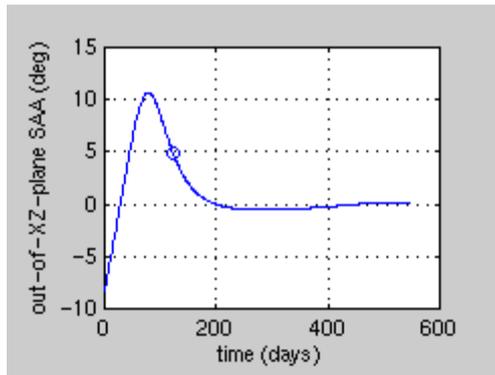
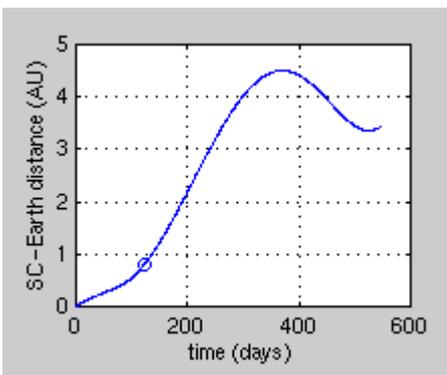
Finally, the third plot of Figure 2.2-3 (Earth to DSH entry cruise phase) shows that the out-of XZ plane solar aspect angle remains below 5° after the first 122 days following the beginning of the phase, which corresponds to a spacecraft - Earth distance at this time of 0.8 AU.



Earth to Mars cruise



Mars to Earth cruise



Earth to DSH entry cruise

Figure 2.2-3 : Earth-spacecraft distance (in AU) and out-of-XZ plane solar aspect angle (°) when the spacecraft is Earth pointed, with the Y axis perpendicular to the ecliptic plane

Hence, different values may be selected for the spacecraft - Earth distance threshold in SHM, during the different phases of the mission. The upper value ensures a correct illumination of the solar arrays as soon as the Earth-pointed option is selected, for every phase of the mission: it is thus selected as default at spacecraft launch, and is set to **0.9 AU**.

3 RESTRICTIONS OF AUTONOMOUS GUIDING OPTIONS / CORRESPONDING REFERENCE ATTITUDE QUATERNION

3.1 CONTEXT AND PURPOSE

Attitude guidance in SHM, Normal Mode (GSEP & GLEP) and NSH is based on on-board ephemerides :

- one of the spacecraft axes (nominally the X axis) is pointed towards either the Earth or the Sun,
- the spacecraft Y axis can be set perpendicular to either the ecliptic plane (“ecliptic” option) or the Sun - spacecraft - Earth plane (“SSCE” option),
- and the Y axis may be pointed towards either the North or the South side of the selected plane.

Only the North / South orientation of the Y axis is chosen by the ground in SHM : the other options are selected automatically. In NM / GSEP & GLEP and in NSH, all options can be selected by the ground ; however, there are restrictions as some options / combinations of options may be forbidden in specific Earth / Sun configurations (for example when the Sun - spacecraft - Earth angle becomes too low, or when the solar arrays cannot be adequately illuminated).

In addition, the transitions from NM / WDP or GSP to NM / GSEP for example require that the initial commanded attitude quaternion (before transition) corresponds to the attitude defined by the selected guidance options at GSEP entry : hence, the ground must be able to compute the reference attitude quaternion corresponding to any selection of autonomous guiding options, to be able to command such transitions.

3.2 RESTRICTIONS OF AUTONOMOUS GUIDING OPTIONS IN NM / GSEP, GLEP & NSH

3.2.1 Principle

The following restrictions must be considered when selecting the autonomous guiding options for NM / GSEP & GLEP and NSH :

- The “SSCE” option cannot be selected if the Sun - spacecraft - Earth plane is not properly defined, i.e. when the Sun - spacecraft - Earth angle is too low. A simulation of autonomous guidance with this option has been successfully performed during the solar opposition in Comet approach phase (conjunction n° 10, mission day 3470, see ref. [3]), for which the Sun - spacecraft - Earth angle drops down to $\cong 2^\circ$: the autonomous guidance is thus robust to such angles. It is therefore decided to prohibit the use of the “SSCE” option for Sun - spacecraft - Earth angles typically below 2° .

This option shall also be avoided if the Earth distance is too low, as the rotation of the Sun - spacecraft - Earth plane may in this case be too fast for the pointing reference to be held. The minimum distance required to use this option is assessed hereafter.

- The “ecliptic” plane reference for the Y axis selected along with the “Earth pointed” option, may cause the Sun to be excessively out of the spacecraft XZ plane, i.e. in a position where the SA cannot be rotated to get enough illumination. Hence, this combination of options is forbidden if the Sun out-of-XZ plane angle is higher than typically 5° (system specification IFPA-800, see ref. [2]). The phases of the mission (and Earth distances) corresponding to this restriction are detailed in §

2.2. Should this specification be ignored, a maximum 10° pointing error constraint with respect to the XZ plane would still be applicable, in order not to trigger the Sun aspect angle surveillance.

3.2.2 Numerical analysis

The distance to the Earth below which the SSCE normal vector motion is too fast has been assessed by simulation : **Figure 3.2-1** shows the simulation results of an Earth fly-by approach performed in NM / GSEP, using the X axis Earth pointed and the “SSCE” guiding options. The simulation is started about 4 hours before the closest approach ($t = 15\,000$ s), as the spacecraft is $\cong 140\,000$ km away from the Earth. The attitude control performance is shown through the X axis pointing accuracy towards the Earth and the Y axis pointing accuracy perpendicular to the Sun - spacecraft - Earth plane.

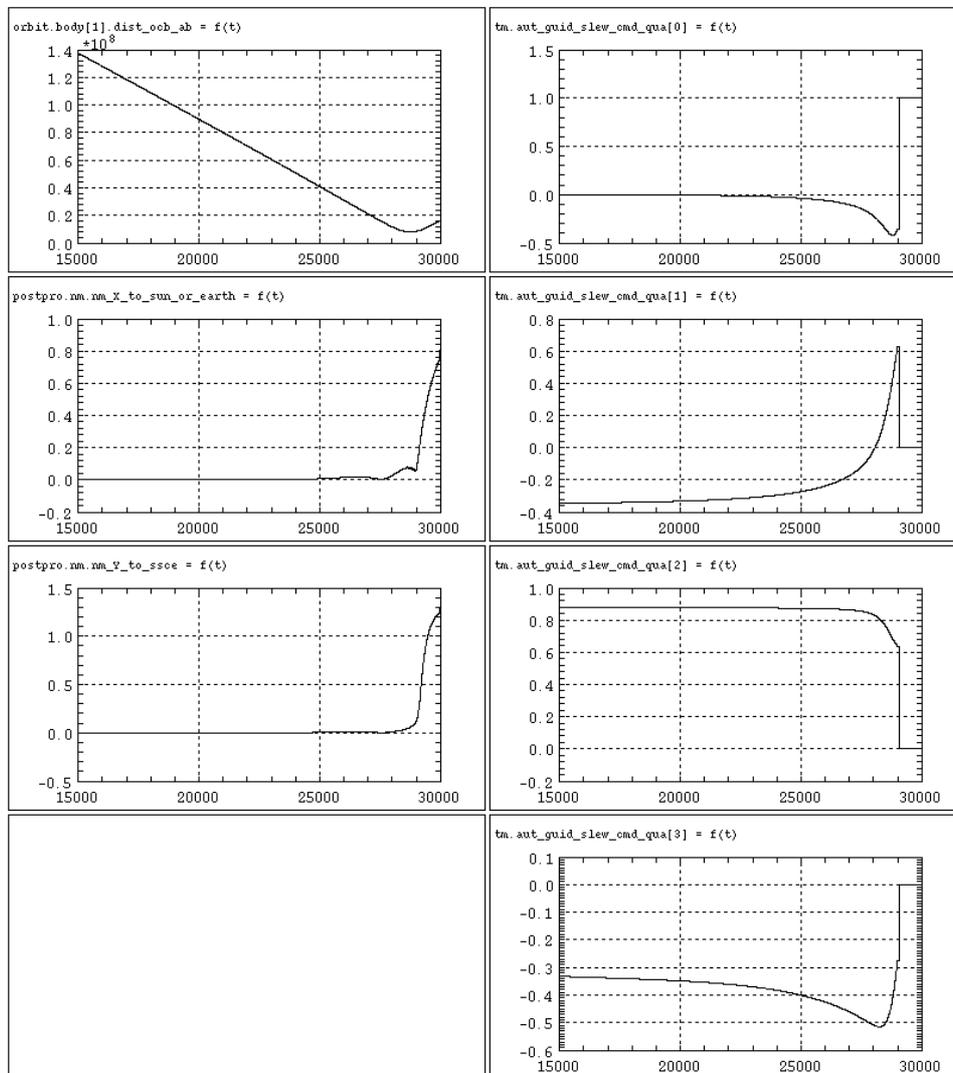


Figure 3.2-1 : Earth fly-by approach in NM /GSEP (X axis Earth pointing, “SSCE” option) spacecraft - Earth distance (top left, m), pointing errors of the X axis / Earth and Y axis / SSCE plane (left, rad), and commanded guidance quaternion (right, rad)

This simulation shows that the spacecraft attitude cannot be controlled accurately with these options when getting close to the Earth. At the very end of the simulation, the autonomous guidance algorithms are not able to compute the commanded reference quaternion. But the attitude control starts diverging about 1 hour before the closest approach ($t=25000$ s), the control errors already reaching levels about 1° . The corresponding Earth distance at this time is about $40\,000$ km : this combination of options is thus forbidden for Earth distances typically below **40 000 km**. **This numerical value is still valid for the new mission to Churyumov-Gerasimenko.**

The only solution in case of near Earth cruise is the “ecliptic” plane reference, along with the “Sun pointed” option, with the recommendation to avoid the HGA being in Tracking mode.

3.3 REFERENCE ATTITUDE QUATERNION CORRESPONDING TO SELECTED GUIDING OPTIONS

3.3.1 Principle

The selection of the different available autonomous guiding options defines a reference attitude to be held by the spacecraft. This reference attitude may also be expressed as an attitude quaternion with respect to the J2000 inertial frame.

This quaternion is required by the ground for the transitions from NM / WDP or GSP to NM / GSEP for example, where the initial commanded attitude quaternion (before transition) must correspond to the attitude defined by the selected guidance options at GSEP entry.

3.3.2 Inputs / Outputs

The spacecraft attitude quaternion is computed from the selected autonomous guidance options, from the Earth / Sun to spacecraft directions in the inertial frame, and possibly from the normal to the Sun ecliptic plane in the inertial frame.

Inputs :

The required inputs for the reference attitude quaternion processing are :

- $[x \ y \ z]$: coordinates (in the spacecraft frame) of the normalized spacecraft axis to be pointed towards either the Earth or the Sun (nominally $[1 \ 0 \ 0]$).
- X_axis_option : body to be pointed by the spacecraft axis defined above (either “Earth” or “Sun”).
- Y_axis_option : plane to be set perpendicular to the Y axis (either “ecliptic” or “SSCE”).
- Y_axis_north : *TRUE* if the Y axis must be pointed towards the North side of the selected plane, *FALSE* for the South side.
- \vec{E}_j : Earth to spacecraft normalized direction expressed in the J2000 inertial frame, at the relevant date (e.g. transition from / to GSEP)
- \vec{S}_j : Sun to spacecraft normalized direction expressed in the J2000 inertial frame at the relevant date (e.g. transition from / to GSEP).

- \vec{Z}_{ecl} : Normal to the Sun ecliptic plane expressed in the J2000 inertial frame at the relevant date (e.g. transition from / to GSEP).

Outputs :

The processing output is the spacecraft attitude quaternion (transformation quaternion from the inertial frame to the spacecraft frame) : Q_{att} .

Note : in the case of the “SSCE” reference plane, the “North” / “South” option actually means “positive” / “negative” side of the $\vec{E}_i \wedge \vec{S}_i$ vector (where “ \wedge ” represents the vector cross product).

3.3.3 Ground processings

3.3.3.1 Main processing

- Step 1 : Inertial direction \vec{u}_{sc} of the spacecraft axis to be pointed

If $X_axis_option = \text{“Earth”}$

Then $\vec{u}_{sc} = -\vec{E}_i$

Else If $X_axis_option = \text{“Sun”}$

Then $\vec{u}_{sc} = -\vec{S}_i$

End If

- Step 2 : Inertial direction \vec{Y}_{sc} of the spacecraft Y axis

If $Y_axis_option = \text{“ecliptic”}$

Then $\vec{Y}_{sc} = (\vec{u}_{sc} \wedge \vec{Z}_{ecl}) \wedge \vec{u}_{sc}$

Else If $Y_axis_option = \text{“SSCE”}$

Then $\vec{Y}_{sc} = \vec{E}_i \wedge \vec{S}_i$

End If

- Step 3 : North / South direction of the spacecraft Y axis

If $Y_axis_north = \text{FALSE}$

Then $\vec{Y}_{sc} = -\vec{Y}_{sc}$

End If

$$\vec{Y}_{sc} = \frac{\vec{Y}_{sc}}{\|\vec{Y}_{sc}\|} \quad \text{-- Normalize vector}$$

- Step 4 : Correction of \vec{Y}_{sc} to compensate for the spacecraft pointed axis (\vec{u}_{sc}) misalignment along the Y axis (y)

$$\vec{Y}_{sc} = \left(1 - \frac{y^2}{2}\right) \cdot \vec{Y}_{sc} + y \cdot \vec{u}_{sc}$$

$$\vec{Y}_{sc} = \frac{\vec{Y}_{sc}}{\|\vec{Y}_{sc}\|} \quad \text{-- Normalize vector}$$

- Step 5 : Inertial direction of the spacecraft X axis

$$\vec{X}_{sc} = x \cdot \vec{u}_{sc} - z \cdot (\vec{u}_{sc} \wedge \vec{Y}_{sc}) - x \cdot y \cdot \vec{Y}_{sc}$$

$$\vec{X}_{sc} = \frac{\vec{X}_{sc}}{\|\vec{X}_{sc}\|} \quad \text{-- Normalize vector}$$

- Step 6 : Inertial direction of the spacecraft Z axis

$$\vec{Z}_{sc} = \vec{X}_{sc} \wedge \vec{Y}_{sc}$$

$$\vec{Z}_{sc} = \frac{\vec{Z}_{sc}}{\|\vec{Z}_{sc}\|} \quad \text{-- Normalize vector}$$

- Step 7 : Final attitude quaternion derived from the spacecraft axes directions

$$Q_{att} = \mathbf{mat2qua} \left(\begin{bmatrix} \vec{X}_{sc} & \vec{Y}_{sc} & \vec{Z}_{sc} \end{bmatrix} \right) \quad \text{-- Input = matrix made of 3 columns } \vec{X}_{sc}, \vec{Y}_{sc}, \vec{Z}_{sc}$$

3.3.3.2 Procedure mat2qua

Description :

This procedure changes a rotation matrix into a quaternion, which enables to represent the orientation of frame *B* with respect to frame *A*.

Input :

M rotation matrix (dimension 3 x 3) from frame *A* to frame *B*.

Output :

Q quaternion ($[Q_0, Q_1, Q_2, Q_3]$, where Q_0 is the scalar term and $[Q_1, Q_2, Q_3]$ is the vectorial term) representing the orientation of frame *B* with respect to frame *A*.

Processing :

- Step 1 : Pivot method

$$P_0 = |1 + M(1, 1) + M(2, 2) + M(3, 3)|$$

$$P_1 = |1 + M(1, 1) - M(2, 2) - M(3, 3)|$$

$$P_2 = |1 - M(1, 1) + M(2, 2) - M(3, 3)|$$

$$P_3 = |1 - M(1, 1) - M(2, 2) + M(3, 3)|$$

- Step 2 : Maximum pivot selection

$$i_0 = 0$$

For $i = 1$ to 3

If $P_i > P_{i_0}$

Then $i_0 = i$

End If

End For

- Step 3 : Attitude quaternion

If $i_0 = 0$

Then $Q_0 = \frac{1}{2} \cdot \sqrt{P_0}$

$$Q_1 = \frac{M(3, 2) - M(2, 3)}{4 \cdot Q_0}$$

$$Q_2 = \frac{M(1, 3) - M(3, 1)}{4 \cdot Q_0}$$

$$Q_3 = \frac{M(2, 1) - M(1, 2)}{4 \cdot Q_0}$$

Else If $i_0 = 1$

Then $Q_1 = \frac{1}{2} \cdot \sqrt{P_1}$

$$Q_0 = \frac{M(3, 2) - M(2, 3)}{4 \cdot Q_1}$$

$$Q_2 = \frac{M(2, 1) + M(1, 2)}{4 \cdot Q_1}$$

$$Q_3 = \frac{M(1, 3) + M(3, 1)}{4 \cdot Q_1}$$

Else If $i_0 = 2$

Then $Q_2 = \frac{1}{2} \cdot \sqrt{P_2}$

$$Q_0 = \frac{M(1,3) - M(3,1)}{4.Q_2}$$

$$Q_1 = \frac{M(2,1) + M(1,2)}{4.Q_2}$$

$$Q_3 = \frac{M(3,2) + M(2,3)}{4.Q_2}$$

Else If $i_0 = 3$

Then $Q_3 = \frac{1}{2} \cdot \sqrt{P_3}$

$$Q_0 = \frac{M(2,1) - M(1,2)}{4.Q_3}$$

$$Q_1 = \frac{M(3,1) + M(1,3)}{4.Q_3}$$

$$Q_2 = \frac{M(3,2) + M(2,3)}{4.Q_3}$$

End If

4 ATTITUDE SLEW POINTING PROFILES

4.1 CONTEXT AND PURPOSE

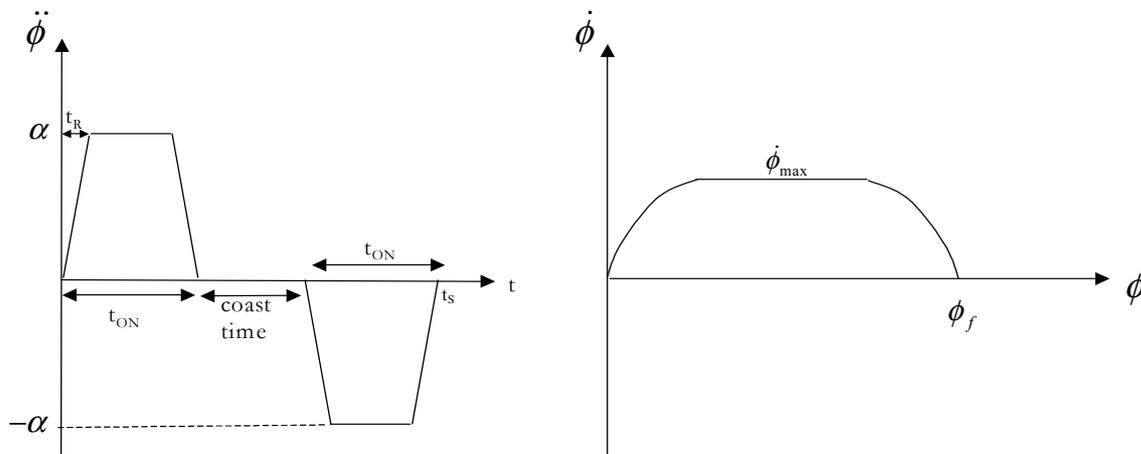
Attitude slew manoeuvres can be commanded in Normal Mode, during the Ground Slew Phase (GSP) or the Fine Pointing Accuracy Phase (FPAP). The spacecraft attitude is required to follow a ground-commanded profile, i.e. an attitude quaternion profile to be uploaded in the form of several guidance polynomials segments (see § 5 “Ephemerides / Ground Guidance segments polynomials”). This ground-commanded guidance is used as reference profile in both NM / GSP and FPAP.

This processing describes how to compute a ground-commanded slew profile to bring the spacecraft attitude in GSP / FPAP from an initial orientation to the desired orientation, while fulfilling constraints on the applied wheel angular momentum and torque. Large angle slew profiles can be performed in NM / GSP only, whereas pointing profiles must be smoothed to be used in NM / FPAP, where additional constraints apply. The proposed method supports the design of single axis slews only (i.e. one Euler axis remains constant during the rotation), and proposes to smooth the acceleration / deceleration phases by a linear ramp, in order to fulfil the pointing stability performance specifications applicable in NM-FPAP. ESOC might have other solutions to compute smooth slewing profiles fulfilling also reaction wheels constraints, but this processing reflects the Astrium-SAS in-house tool which has been used for AOCMS studies needs, and is described for information.

4.2 COMPUTATION OF LARGE ANGLE SLEW PROFILES

4.2.1 Principle

Attitude slew manoeuvres are assumed to be performed through a constant single axis rotation of the spacecraft about the equivalent Euler axis (the Euler axis of rotation is noted \vec{E} and the Euler angle ϕ_f), bringing the spacecraft from its initial quaternion Q_i to the desired quaternion Q_f . The commanded angular acceleration $\ddot{\phi}$ and rate $\dot{\phi}$ about the Euler axis are chosen as shown on the following figure :



where :

- t_{ON} (called on-time) is the time during which the acceleration and the deceleration are applied ;
- t_S (called slew-time) is the total duration of the slew ;
- t_R (called the ramp-time) is recommended in FPAP to smooth the zero-to-maximum reaction wheel torque increase ;
- α is the maximum acceleration about the Euler axis. This parameter is computed from the maximum allowable reaction wheel torque ;
- ϕ_f the final Euler angle.

The on-time (t_{ON}) and slew-time (t_S) are computed such that the reaction wheel speeds remain within prescribed bounds. The wheel speed time-profile during the slew is determined from the conservation of angular momentum, knowing the initial wheel angular momentum, the acceleration profile (depending on t_{ON} and t_S) and the spacecraft inertia.

4.2.2 Inputs and constraints

4.2.2.1 Rotation data

The following spacecraft rotation data are required :

- Q_i : the initial spacecraft attitude quaternion (line vector 1x4) ;
- Q_f : the desired final spacecraft attitude quaternion (line vector 1x4).

4.2.2.2 Reaction wheel data

The following reaction wheel data are required:

- I_W : reaction wheel inertia in kg.m^2 (3x3 diagonal matrix with $I_{W1} = I_{W2} = I_{W3}$, see ref. [4]) ;
- B : transformation matrix from spacecraft frame to wheel frame (3x3 matrix, see ref.[4]), considering the thress wheels to be used during the slew ;
- H_{Wmax} : maximum allowable reaction wheel angular momentum (scalar $\leq 37.8 \text{ N.m.s}$, corresponding to the warning limit tuned in ref. [6]) with margins described in § 4.2.2.5 ;
- $\overline{\Omega}_{W0}$: initial reaction wheel speed in the reaction wheel frame in rad/s (column vector 3x1) ; to be checked in TM
- T_{Wmax} : maximum allowable reaction wheel torque (scalar $\leq 0.15 \text{ N.m}$, corresponding to net reaction wheel torque in ref.[4]) with margins described in § 4.2.2.5 ;
- t_R : ramp time, i.e. minimum time to increase the reaction wheel torque from 0 to T_{Wmax} (scalar = typically 15 s in FPAP, 0 s in GSP ; note that a smoother profile could be used instead of a linear ramp). This ramp time aims at smoothing the acceleration / deceleration phases in order to better comply with the pointing stability requirements applicable in NM-FPAP.

4.2.2.3 Spacecraft data

The following spacecraft data are required :

- I_{SC} : spacecraft inertia in kg.m^2 (see ref.[4], 3x3 diagonal matrix).

4.2.2.4 Processing parameters

The following processing parameters are required:

- dt_S : increment of slew time in s (scalar which defines at which time step the loop on slew time t_S is to be iterated, depending on the desired slew time resolution) ;
- dt : increment of time in s (scalar which defines at which time step the loop on time t is to be iterated : dt is lower than or equal to dt_S , and depends on the desired dimension n of the quaternion table, see § 4.2.3. A resolution of 1s is classical).

4.2.2.5 Specific constraints

Several assumptions are made for the computation of the attitude slew profile. No external disturbance torques, and no SA or HGA rotation are taken into account, and the fundamental equation of dynamics is linearised, i.e. the gyroscopic torques are neglected. Furthermore there are uncertainties on the spacecraft inertia. Therefore margins shall be taken on the maximum allowable reaction wheel angular momentum (typically $H_{Wmax} \approx 30 \text{ N.m.s}$, i.e. margin of $\approx 8 \text{ N.m.s}$) and on the maximum allowable reaction wheel torque (typically $T_{Wmax} \approx 0.1 \text{ N.m}$, i.e. margin of $\approx 0.05 \text{ N.m}$).

Since the SADMs are commanded in autonomous tracking during GSP, it is recommended that the attitude slew never causes SADE speeds higher than level 6 in order to avoid large rate oscillations about the Y axis and therefore large interface torque levels; this corresponds to a maximum angular rate about the spacecraft Y axis of $0.5^\circ/\text{s}$. In addition, the SADMs shall not enter prohibited zones when autonomously commanded during the slew : if the slew profile is such that this constraint is not fulfilled, the SADMs must be driven “off-line” to their target positions at the end of the slew (using procedure FCP-AC0410 “SADE Rotation and Hold”).

If the HGA autonomous tracking is maintained during the slew, the APME maximum speed shall never be exceeded if Earth tracking must be maintained during the slew; this corresponds to a maximum angular rate of $0.5^\circ/\text{s}$ about every spacecraft axis, along with a maximum angular acceleration of $0.1^\circ/\text{s}/\text{min}$. In addition, the HGA shall not enter prohibited zones when autonomously commanded during the slew : if the slew profile is such that this constraint is not fulfilled, the HGA must be driven “off-line” to the target position at the end of the slew (using procedure FCP-AC0450 “APME Rotation and Hold”).

The solar aspect angle as seen from the arrays-mounted SAS must not exceed 10° during the whole slew. Otherwise the linear range Sun pointing surveillance (G4) shall be disabled, and the large range Sun pointing surveillance (G5) shall be used instead.

The above SADM, APME, and solar aspect angle constraints shall be handled at system level, for example by predicting their violation during the slew and correcting the slew afterwards in order to avoid those violations.

4.2.3 Outputs

The following outputs are then used as input parameters for the AOCMS on-ground processing “Ephemerides/Ground Guidance Segments Polynomials” (see §5.2.2):

- *TabTemps(:)* : time (vector of n steps of dt) ;
- *TabQ(:,:)* : attitude quaternion table ($n \times 4$ matrix).

4.2.4 Ground processings

4.2.4.1 Description

The structure of this on-ground processing is described in the following figure:

$$\phi_f = 2\pi - \phi_f$$

$$\vec{E} = -\vec{E}$$

End If

4.2.4.3 Calculation of the maximum allowable reaction wheel rate

$$\Omega_{W \max} = \frac{H_{W \max}}{I_w}$$

4.2.4.4 Calculation of the maximum acceleration about the Euler axis

-- The fundamental equation of dynamics in the inertial frame is expressed as follows:

$$\overline{H}_{SC} = I_{SC} \cdot \overline{\omega}_{SC} + \overline{H}_W = I_{SC} \cdot \overline{\omega}_{SC} + B^{-1} \cdot I_w \cdot \overline{\Omega}_{W/W}$$

-- where \overline{H}_{SC} = total spacecraft angular momentum

-- $\overline{\omega}_{SC} = \dot{\phi} \cdot \vec{E}$ = spacecraft rate vector

-- $\overline{\Omega}_{W/W}$ = reaction wheel rate vector in the reaction wheel frame

-- Assuming that the spacecraft non-linear dynamics are negligible and that the external
-- disturbing torque \overline{T}_{ext} is null, the derivation of this equation in the inertial frame gives:

$$\overline{\frac{dH_{SC}}{dt}} = \overline{T}_{ext} = \vec{0} = I_{SC} \cdot \overline{\dot{\omega}}_{SC} + B^{-1} \cdot I_w \cdot \overline{\dot{\Omega}}_{W/W} \quad (\text{because vectorial products are neglected})$$

$$\overline{\dot{\Omega}}_{W/W} = -I_w^{-1} \cdot B \cdot I_{SC} \cdot \overline{\dot{\omega}}_{SC} = -I_w^{-1} \cdot B \cdot I_{SC} \cdot \ddot{\phi} \cdot \vec{E}$$

-- The reaction wheel torque can be expressed as follows:

$$\overline{T}_{W/W} = I_w \cdot \overline{\dot{\Omega}}_{W/W} = -B \cdot I_{SC} \cdot \ddot{\phi} \cdot \vec{E}$$

-- Therefore the maximum acceleration α about the Euler axis can be calculated from the
-- maximum allowable reaction wheel torque as follows:

$$\alpha = \ddot{\phi}_{\max} = \frac{T_{W \max}}{\max |B \cdot I_{SC} \cdot \vec{E}|}$$

As explained in § 4.2.2.5 and written above, the expression of α does not account for non-linear cross-coupling dynamics terms, but a margin on the maximum available reaction wheel torque allows to compensate for this limitation when the slew is achieved by the AOCMS.

4.2.4.5 Initialisation

- Initialisation of the slew time t_s (corresponding to the function $t_s = f_1(t_R, \phi_f, \alpha)$ defined in §4.2.4.1):

-- Assuming the coast time to be zero (i.e. $t_s = 2 \cdot t_{on}$), the slew time is:

$$t_s = t_R + \sqrt{\left(t_R^2 + 4 \frac{\phi_f}{\alpha}\right)} \quad \text{i.e. } t_s \text{ is initialized to the minimum achievable value}$$

- Initialisation of the on time

$$t_{on} = 0$$

- Initialisation of processing parameters

-- The time is initialised to zero

$$t = 0$$

-- The table index k used to save the output data is initialised to zero

$$k = 0$$

-- The tables used to store the output data are initialised to zero

-- Time vector (n components)

$$TabTime(:) = 0$$

-- Attitude quaternion table (n x 4 matrix)

$$TabQ(:, :) = 0$$

4.2.4.6 Main processing

- Loop on the time

While $t \leq t_s$ with a step dt

- Calculation of the on-time

-- The on-time t_{on} is calculated as follows (corresponds to the function $t_{on} = f_2(t_s, t_R, \phi_f, \alpha)$ defined in §4.2.4.1):

$$t_{on} = \frac{1}{2} \left[t_S + t_R - \sqrt{\left((t_R + t_S)^2 - 4 \cdot \left(t_R \cdot t_S + \frac{\phi_f}{\alpha} \right) \right)} \right]$$

If $t_{on} < 2 t_R$ Then

error message = "impossible determination of $t_{on} \geq 2 t_R$ fulfilling the constraint on the maximum allowable reaction wheel torque T_{Wmax} , so the computation shall be resumed with a lower T_{Wmax} "

exit execution

End If

- Calculation of the spacecraft acceleration, rate and angle about the Euler axis:

If $t_R = 0$ Then

-- Case of large attitude slew manoeuvre in GSP

If $t < t_{on}$

$$\ddot{\phi}(t) = \alpha$$

$$\dot{\phi}(t) = \alpha \cdot t$$

$$\phi(t) = \alpha \cdot \frac{t^2}{2}$$

Else If $t < t_S - t_{on}$

$$\ddot{\phi}(t) = 0$$

$$\dot{\phi}(t) = \alpha \cdot t_{on}$$

$$\phi(t) = \alpha \cdot t_{on} \cdot \left(t - \frac{t_{on}}{2} \right)$$

Else If $t \leq t_S$

$$\ddot{\phi}(t) = -\alpha$$

$$\dot{\phi}(t) = -\alpha \cdot t$$

$$\phi(t) = -\frac{\alpha}{2} \cdot (t_S - t)^2 + \phi_f$$

Else

-- Case of smooth attitude slew manoeuvre in FPAP

If $t < t_R$

$$\ddot{\phi}(t) = \alpha \cdot \frac{t}{t_R}$$

$$\dot{\phi}(t) = \alpha \cdot \frac{t^2}{2 \cdot t_R}$$

$$\phi(t) = \alpha \cdot \frac{t^3}{6 \cdot t_R}$$

Else If $t < t_{on} - t_R$

$$\ddot{\phi}(t) = \alpha$$

$$\dot{\phi}(t) = \alpha \cdot \left(t - \frac{t_R}{2} \right)$$

$$\phi(t) = \alpha \cdot \left[\frac{1}{2} \cdot \left(t - \frac{t_R}{2} \right)^2 + \frac{1}{24} \cdot t_R^2 \right]$$

Else If $t < t_{on}$

$$\ddot{\phi}(t) = -\alpha \cdot \frac{t - t_{on}}{t_R}$$

$$\dot{\phi}(t) = -\alpha \cdot \left[\frac{(t - t_{on})^2}{2 \cdot t_R} - t_{on} + t_R \right]$$

$$\phi(t) = -\alpha \cdot \left[\frac{(t - t_{on})^3}{6 \cdot t_R} - (t_{on} - t_R) \cdot \left(t - \frac{t_{on}}{2} \right) \right]$$

Else If $t < t_S - t_{on}$

$$\ddot{\phi}(t) = 0$$

$$\dot{\phi}(t) = \alpha \cdot (t_{on} - t_R)$$

$$\phi(t) = \alpha \cdot (t_{on} - t_R) \cdot \left(t - \frac{t_R}{2} \right) + \frac{\phi_f}{2}$$

Else If $t < t_S - t_{on} + t_R$

$$\ddot{\phi}(t) = -\alpha \cdot \frac{t - t_S + t_{on}}{t_R}$$

$$\dot{\phi}(t) = -\alpha \cdot \left[\frac{(t - t_S + t_{on})^2}{2 \cdot t_R} - t_{on} + t_R \right]$$

$$\phi(t) = -\alpha \cdot \left[\frac{(t - t_S + t_{on})^3}{6 \cdot t_R} - (t_{on} - t_R) \cdot \left(t - \frac{t_S}{2} \right) \right] + \frac{\phi_f}{2}$$

Else If $t < t_S - t_R$

$$\ddot{\phi}(t) = -\alpha$$

$$\dot{\phi}(t) = -\alpha \cdot \left(t - t_S + \frac{t_R}{2} \right)$$

$$\phi(t) = -\frac{\alpha}{2} \cdot \left[\left(t - t_S + \frac{t_R}{2} \right)^2 + \frac{1}{12} \cdot t_R^2 \right] + \phi_f$$

Else If $t \leq t_S$

$$\ddot{\phi}(t) = \alpha \cdot \frac{t - t_S}{t_R}$$

$$\dot{\phi}(t) = \alpha \cdot \left[\frac{(t - t_S)^2}{2 \cdot t_R} \right]$$

$$\phi(t) = \alpha \cdot \left[\frac{(t - t_S)^3}{6 \cdot t_R} \right] + \phi_f$$

End If

- Calculation of the current relative attitude quaternion $Q = (q_1, q_2, q_3, q_4)$ from the current spacecraft angle $\phi(t)$ and the Euler axis $\vec{E} = (E_1, E_2, E_3)$, with respect to the initial attitude Q_i :

$$q_1 = \cos \frac{\phi(t)}{2}$$

$$q_2 = E_1 \cdot \sin \frac{\phi(t)}{2}$$

$$q_3 = E_2 \cdot \sin \frac{\phi(t)}{2}$$

$$q_4 = E_3 \cdot \sin \frac{\phi(t)}{2}$$

- Calculation of the rotation matrix from the current attitude to the initial attitude :

$$C(Q) = \begin{pmatrix} q_1^2 + q_2^2 - q_3^2 - q_4^2 & 2 \cdot (q_2 \cdot q_3 + q_1 \cdot q_4) & 2 \cdot (q_2 \cdot q_4 - q_1 \cdot q_3) \\ 2 \cdot (q_2 \cdot q_3 - q_1 \cdot q_4) & q_1^2 + q_3^2 - q_2^2 - q_4^2 & 2 \cdot (q_3 \cdot q_4 + q_1 \cdot q_2) \\ 2 \cdot (q_2 \cdot q_4 + q_1 \cdot q_3) & 2 \cdot (q_3 \cdot q_4 - q_1 \cdot q_2) & q_1^2 + q_4^2 - q_2^2 - q_3^2 \end{pmatrix}$$

- Calculation of the reaction wheel speed rate $\overrightarrow{\Omega_w(t)}$:

-- According to the fundamental equation of dynamics, the angular momentum defined in the inertial frame R_I but projected in the spacecraft frame $R_S(t)$ is :

$$\overrightarrow{(H_{SC}(t))}_{R_S(t)} = I_{SC} \cdot \dot{\phi}(t) \cdot \overrightarrow{E} + B^{-1} \cdot I_W \cdot \overrightarrow{\Omega_w(t)}$$

-- Assuming that no external disturbing torque applies, the angular momentum remains constant in the inertial frame :

$$\overrightarrow{(H_{SC}(t))}_{R_S(t)} = \overrightarrow{(H_{SC}(t_0))}_{R_S(t_0)} = C(Q) \cdot \overrightarrow{(H_{SC}(t_0))}_{R_S(t_0)}$$

$$\text{-- where } \overrightarrow{(H_{SC}(t_0))}_{R_S(t_0)} = I_{SC} \cdot \dot{\phi}(t_0) \cdot \overrightarrow{E} + B^{-1} \cdot I_W \cdot \overrightarrow{\Omega_w(t_0)}$$

$$\text{-- and since the initial spacecraft rate } \dot{\phi}(t_0) \text{ is null, } \overrightarrow{(H_{SC}(t_0))}_{R_S(t_0)} = B^{-1} \cdot I_W \cdot \overrightarrow{\Omega_w(t_0)}$$

$$\text{-- Hence } I_{SC} \cdot \dot{\phi}(t) \cdot \overrightarrow{E} + B^{-1} \cdot I_W \cdot \overrightarrow{\Omega_w(t)} = C(Q) \cdot B^{-1} \cdot I_W \cdot \overrightarrow{\Omega_w(t_0)}$$

-- So the reaction wheel rate projected in the reaction wheel frame can be expressed as

-- (corresponds to $\Omega_w = f_3(\phi(t), \dot{\phi}(t))$ defined in § 4.2.4.1) :

$$\overrightarrow{\Omega_w(t)} = B \cdot C(Q(t)) \cdot B^{-1} \cdot \overrightarrow{\Omega_w(t_0)} - I_W^{-1} \cdot B \cdot I_{SC} \cdot \dot{\phi}(t) \cdot \overrightarrow{E}$$

- Evaluation of the reaction wheel speed rate constraint :

$$\text{If } \max |\overrightarrow{\Omega_w(t)}| > \Omega_{w \max} \text{ Then}$$

-- Loop on the slew time t_S : the main processing is restarted with a new value of slew time

$$t_S = t_S + dt_S$$

-- Reinitialisation of the time and table index

$t = 0$

$k = 0$

Else

-- Calculation of the absolute current attitude $Q_{abs}(t)$ (line vector 1x4), taking into account the non-null initial attitude

$$Q_{abs}(t) = (Q_i \cdot Q(t))^T$$

-- Saving of the output data

-- Time vector

$TabTemps(k+1) = t$

-- Attitude quaternion table (n x 4 matrix)

$TabQ(k+1,:) = Q_{abs}(t)$

-- Loop on the time t : incrementation of the time and table index

$t = t + dt$

$k = k + 1$

Endif

End While

5 EPHEMERIDES / GROUND GUIDANCE SEGMENTS POLYNOMIALS

5.1 CONTEXT AND PURPOSE

Earth / Sun ephemerides, as well as ground guidance attitude profiles for NM / GSP and FPAP, are provided by the ground in the form of Chebishev polynomial series. These polynomials are then used by the on-board guidance functions to expand each component of the Sun / Earth position vectors, or the attitude reference quaternion to be commanded.

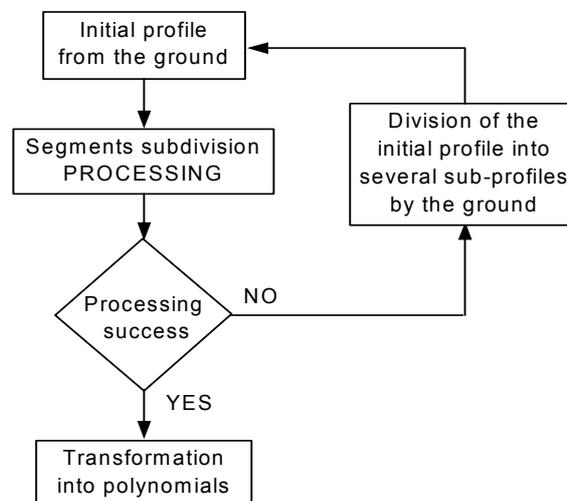
Depending on the interpolation accuracy requirements, each profile may be divided into several segments (each characterised by a time domain), especially when the corresponding data varies rapidly. For each scalar term to propagate, the ground shall find the best compromise between the segment duration and the order of Chebishev polynomials, to be compliant with the required accuracy.

5.2 PROCESSING

5.2.1 Principle

This processing transforms an initial ephemerides / ground guidance time profile provided by the ground into segments polynomials which can be directly uploaded to the flight software. In the case of attitude guidance profile for NM / GSP or FPAP, the processing described in § 4 may be used to obtain the correct profile leading the spacecraft from its initial attitude to the desired final attitude, while fulfilling specific constraints on the spacecraft angular rates / accelerations, RW angular momentum / torque capability, or the SA solar aspect angle.

This initial profile is first divided into segments which fulfill the approximation accuracy requirements : if the automatic segments subdivision described in the following processing is not successful, the ground shall perform a preliminary division of its initial profile into several sub-profiles, and then apply the segments subdivision processing on each of them. After this subdivision is successful, each segment is transformed into Chebishev polynomials. This logic is recalled in the following figure :



5.2.2 Inputs

5.2.2.1 Ephemerides propagation

On-board ephemerides are obtained through the propagation of Chebishev polynomials for each component of the Earth-to-spacecraft and Sun-to-spacecraft position vectors.

The required inputs for the segments polynomials processing are :

- User inputs :
 - *seg_ini* : initial ephemerides profile provided by the ground. It contains the components of the Earth - to - spacecraft and Sun - to - spacecraft position vectors (in *km*), gathered in a single vector $X = [X_{Ex}(t), X_{Ey}(t), X_{Ez}(t), X_{Sx}(t), X_{Sy}(t), X_{Sz}(t)]$, where the time t is in the range $[t_{seg_ini}^i, t_{seg_ini}^f]$ by steps of dt (in *s*).
 - *thd_earth* : maximum error on the Earth-to-spacecraft direction approximation (*rad*).
 - *thd_sun* : maximum error on the Sun-to-spacecraft direction approximation (*rad*).
- Processing parameters :
 - *max_order* : maximum degree of the polynomial expansion (≤ 7 , flight software limit).
 - *nb_max* : maximum number of dichotomy iterations, used to limit the number of segments (nominally $0.01^\circ \cong 0.00017 \text{ rad}$).

5.2.2.2 Ground guidance profile

In NM / GSP or FPAP, the attitude is controlled about a reference attitude profile, which is obtained through the propagation of Chebishev polynomials for each component of the commanded quaternion.

The required inputs for the segments polynomials processing are :

- User inputs :
 - *seg_ini* : initial attitude profile provided by the ground. It contains the 4 components of the reference attitude quaternion $Q = [Q_0(t), Q_1(t), Q_2(t), Q_3(t)]$, where the time t is in the range $[t_{seg_ini}^i, t_{seg_ini}^f]$ by steps of dt (in *s*).
 - *thd_angle* : maximum pointing error induced by the attitude quaternion polynomial approximation (nominally $0.01^\circ \cong 0.00017 \text{ rad}$).
 - *thd_rate* : maximum angular rate error induced by the attitude quaternion polynomial approximation (nominally $0.0001^\circ/\text{s} \cong 1.7 \cdot 10^{-6} \text{ rad/s}$).
- Processing parameters :
 - *max_order* : maximum degree of the polynomial expansion (≤ 7 , flight software limit).
 - *nb_max* : max number of dichotomy iterations, used to limit the number of segments.
 - *acc_limit* : acceleration limit for the feed-forward torque application ; if the maximum angular acceleration during a segment is higher than this limit, a feed-forward guidance torque shall be computed by the AOCMS and applied in the control

loop during this segment ; this limit is set to typically $3.10^{-5} \text{ rad/s}^2$, which corresponds to a 0.05 N.m torque applied on the spacecraft lowest inertia.

5.2.3 Outputs

5.2.3.1 Ephemerides propagation

The outputs of the segments polynomials processing are directly used as parameters for the operational TC(172,97) “*EPH_MGR.set_segments_initialisation*” (see ref. [5]) :

- *nb_segments* : number of segments the initial profile has been divided into.
- And for each segment :
 - *tc_date* : sending date of the operational telecommand ;
 - *segment_start_time* : segment start time ;
 - *segment_end_time* : segment end time ;
 - *deg_earth* : degree of the polynomial expansion for each component of the Earth - spacecraft position vector ;
 - *deg_sun* : degree of the polynomial expansion for each component of the Sun - spacecraft position vector ;
 - *coeff_earth_to_sc_pos_X* (X_{Ex}) coefficients of the Chebyshev polynomials used to expand each component of the Earth-to-spacecraft position vector (*km*), truncated to the 3.1750 numerical accuracy.
 - *coeff_earth_to_sc_pos_Y* (X_{Ey})
 - *coeff_earth_to_sc_pos_Z* (X_{Ez})
 - *coeff_sun_to_sc_pos_X* (X_{Sx}) coefficients of the Chebyshev polynomials used to expand each component of the Sun-to-spacecraft position vector (*km*), truncated to the 3.1750 numerical accuracy.
 - *coeff_sun_to_sc_pos_Y* (X_{Sy})
 - *coeff_sun_to_sc_pos_Z* (X_{Sz})

5.2.3.2 Ground guidance profile

The outputs of the segments polynomials processing are directly used as parameters for the operational TC(172,100) “*GRD_GUID.set_segments_initialisation*” (see ref. [5]) :

- *nb_segments* : number of segments the initial profile has been divided into.
- And for each segment :
 - *tc_date* : sending date of the operational telecommand ;
 - *segment_start_time* : segment start time ;
 - *segment_end_time* : segment end time ;
 - *deg_poly* : degree of the polynomial expansion for each component of the reference attitude quaternion ;
 - *coeff_scal* (Q_0) coefficients of the Chebyshev polynomials used to expand each component of the reference attitude quaternion ;
 - *coeff_vect_X* (Q_1)
 - *coeff_vect_Y* (Q_2)
 - *coeff_vect_Z* (Q_3)



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : **05/08/2003**
Page : 30

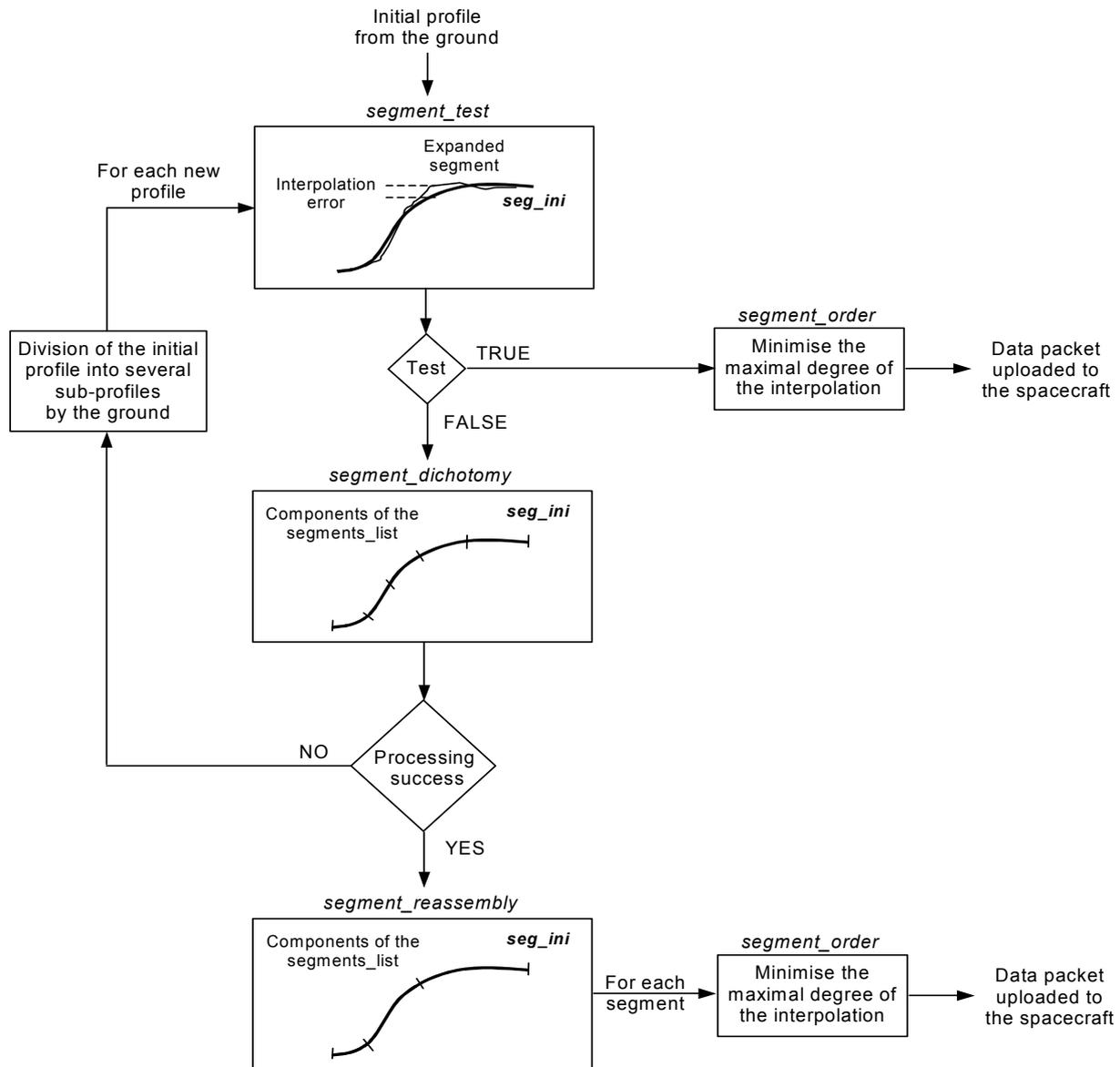
- *is_acceleration_computed* : flag stating if the feed-forward guidance torque shall be computed and applied during this segment.

5.2.4 Ground processings

5.2.4.1 Description

The following processing achieves the subdivision of an initial ground-provided profile into several time segments, so that the interpolation errors of each segment are lower than ground-defined thresholds. Each segment shall be defined as a compromise between its duration and the order of the Chebyshev polynomial. If this processing is not successful (after many dichotomy iterations) with an initial stiff guidance profile for example, the ground may be required to perform a preliminary division of its initial profile into several sub-profiles, and then apply the segments subdivision processing on each of them.

The structure of this processing is described in the following figure :



The initial profile (*seg_ini*) is first tested by procedure *segment_test*, which computes the Chebyshev polynomials interpolation of degree *max_order*, and then compares the interpolation errors with the ground-defined thresholds. If this test is successful, only 1 segment is necessary to represent the profile, and the polynomial interpolation is then optimised by procedure *segment_order* : the lowest degree of the polynomial expansion which is compatible with the accuracy requirements is selected.

If this first test is not successful, the profile is divided into several segments by procedure *segments_dichotomy* : each segment is then tested again by procedure *segment_test*. Up to *nb_max* iterations are performed, in order to find a set of segments which all fulfill the accuracy requirements. If the maximum number of iterations is reached without converging towards an adequate subdivision of the initial profile, the processing is aborted : the ground must then increase *nb_max*, or perform a

preliminary division of its initial profile into several sub-profiles, and then apply the processing on each of them.

If the processing is successful, the final number of segments is reduced by procedure *segment_reassembly*, which concatenates segments as far as the accuracy requirements are fulfilled, up to the minimal list of segments which can be issued from the initial profile. Finally, the polynomial interpolation of each segment is then optimised by procedure *segment_order*, which selects the lowest degree of the polynomial expansion which is compatible with the accuracy requirements.

Note :

The variable *constraints* used as input parameter of the different procedures of this processing actually gathers all user-defined thresholds :

- the maximum authorised error on the Earth-to-spacecraft direction approximation *thd_earth* ;
- the maximum authorised error on the Sun-to-spacecraft direction approximation *thd_sun* ;
- the maximum authorised pointing error induced by the guidance quaternion approximation *thd_angle* ;
- the maximum authorised angular rate error induced by the guidance quaternion approximation *thd_rate* ;
- the acceleration limit for the feed-forward torque application *acc_limit*.

5.2.4.2 Main processing

- Step 1 : Initialisation

segments_list = []

nb_segments_list = 0

nb_iterations = 0

- Step 2 : Test of the initial profile

If *segment_test* (*seg_ini*, *max_order*, *constraints*) = TRUE

Then [*segment_start_time*, *segment_end_time*, *deg*, *coeff*, {*is_acceleration_computed*}] = *segment_order* (*seg_ini*, *max_order*, *constraints*)

Else

- Step 3 : Subdivision of the initial profile by dichotomy

[*success*, *nb_iterations*, *segments_list*, *nb_segments_list*] = *segment_dichotomy* (*seg_ini*, *nb_max*, *max_order*, *nb_iterations*, *segments_list*, *nb_segments_list*, *constraints*)

If *success* = TRUE

Then

- Step 4 : Optimisation of the segments size

[*final_segments_list*, *nb_segments_list*] = **segment_reassembly** (*segments_list*,
nb_segments_list, *max_order*, *constraints*)

For each *segment* **of** *final_segments_list*

- Step 5 : Polynomial interpolation for each selected segment

[*segment_start_time*, *segment_end_time*, *deg*, *coeff*, {*is_acceleration_computed*}]
 = **segment_order** (*segment*, *max_order*, *constraints*)

previous_segment_start_time = *segment_start_time*

End For

Else *error_message* = "Dichotomy has failed : try to split the initial segment or to release accuracy thresholds."

End If

End If

5.2.4.3 Procedure *segment_test*

Description :

This procedure computes the polynomial interpolation of the input segment, using Chebyshev polynomials of a chosen maximum order. Interpolation errors are then computed and compared with the angle / rate thresholds : the test is successful if the errors are lower than the corresponding thresholds.

Inputs :

segment input segment containing the following parameters :

t_{seg}^i , t_{seg}^f , dt = start / end times of the segment & time-step (in s), such that $t_{seg}^f - t_{seg}^i = n \cdot dt$,
 M = matrix (dim $n \times i$) describing the segment i components during the time vector, i.e. either
 $X(n,6) = [X_{Ex}(n), X_{Ey}(n), X_{Ez}(n), X_{Sx}(n), X_{Sy}(n), X_{Sz}(n)]$ or $Q(n,4) = [Q_0(n), Q_1(n), Q_2(n), Q_3(n)]$.

degree maximum degree of the Chebyshev polynomials used to expand the input segment.

constraints user-defined thresholds : *thd_earth* = Earth direction maximum approximation error,
thd_sun = Sun direction maximum approximation error,
thd_angle = quaternion approximation maximum pointing error,
thd_rate = quaternion approximation maximum rate error,
acc_lim = acceleration limit for the feed-forward torque.

Outputs :

segment_test flag stating if the approximation of the input segment with Chebyshev polynomials of maximum order *degree* fulfills the accuracy requirements.

Processing for an ephemerides segment :

- Step 1 : Polynomial interpolation

[coeff, $X^{exp}(n,6)$] = **segment_interpolation** (segment, degree)

- Step 2 : Computation of interpolation errors

$$\Delta\theta_E(n) = \arcsin \left(\frac{\left\| \left[X_{Ex}(n), X_{Ey}(n), X_{Ez}(n) \right] \wedge \left[X_{Ex}^{exp}(n), X_{Ey}^{exp}(n), X_{Ez}^{exp}(n) \right] \right\|}{\left\| \left[X_{Ex}(n), X_{Ey}(n), X_{Ez}(n) \right] \right\| \cdot \left\| \left[X_{Ex}^{exp}(n), X_{Ey}^{exp}(n), X_{Ez}^{exp}(n) \right] \right\|} \right)$$

$$\Delta\theta_S(n) = \arcsin \left(\frac{\left\| \left[X_{Sx}(n), X_{Sy}(n), X_{Sz}(n) \right] \wedge \left[X_{Sx}^{exp}(n), X_{Sy}^{exp}(n), X_{Sz}^{exp}(n) \right] \right\|}{\left\| \left[X_{Sx}(n), X_{Sy}(n), X_{Sz}(n) \right] \right\| \cdot \left\| \left[X_{Sx}^{exp}(n), X_{Sy}^{exp}(n), X_{Sz}^{exp}(n) \right] \right\|} \right)$$

-- Approximation error for each step of the time vector

- Step 3 : Test

If MAX(ABS($\Delta\theta_E$)) \geq thd_earth **or** MAX(ABS($\Delta\theta_S$)) \geq thd_sun

Then segment_test = FALSE

Else segment_test = TRUE

End If

Processing for an attitude guidance segment :

- Step 1 : Polynomial interpolation

[coeff, $Q^{exp}(n,4)$] = **segment_interpolation** (segment, degree)

- Step 2 : Computation of interpolation errors

$\Delta Q(n) = Q^{-1}(n) \times Q_{exp}(n)$ -- Product of quaternions for each step of the time vector

$\Delta\theta(n) = 2 \times [\Delta Q_1(n), \Delta Q_2(n), \Delta Q_3(n)]$

$\Delta\omega(n-1) = \frac{2}{dt} \cdot (Q^{-1}(1:n-1) \times Q(2:n) - Q_{exp}^{-1}(1:n-1) \times Q_{exp}(2:n))$

- Step 3 : Test

If MAX(ABS($\Delta\theta$)) \geq thd_angle **or** MAX(ABS($\Delta\omega$)) \geq thd_rate

Then segment_test = FALSE

Else segment_test = TRUE

End If

5.2.4.4 Procedure segment_order

Description :

This procedure computes the lowest degree of a segment polynomial interpolation, which is compliant with the accuracy requirements. In addition to this optimised order, it provides the coefficients of the Chebyshev polynomials used to expand the input segment, and for a ground guidance profile the flag stating whether the feed-forward guidance torque shall be computed or not.

Inputs :

- segment** input segment containing the following parameters :
- t_{seg}^i, t_{seg}^f, dt = start / end times of the segment & time-step (in s), such that $t_{seg}^f - t_{seg}^i = n \cdot dt$,
 - M = matrix (dim $n \times i$) describing the segment i components during the time vector, i.e. either $X(n,6) = [X_{Ex}(n), X_{Ey}(n), X_{Ez}(n), X_{Sx}(n), X_{Sy}(n), X_{Sz}(n)]$ or $Q(n,4) = [Q_0(n), Q_1(n), Q_2(n), Q_3(n)]$.
- max_order** maximum authorised degree for the Chebyshev polynomials interpolation (≤ 7).
- constraints** user-defined thresholds :
- thd_earth = Earth direction maximum approximation error,
 - thd_sun = Sun direction maximum approximation error,
 - thd_angle = quaternion approximation maximum pointing error,
 - thd_rate = quaternion approximation maximum rate error,
 - acc_lim = acceleration limit for the feed-forward torque.

Outputs :

- degree** optimised degree of the Chebyshev polynomials interpolation.
- coeff** coefficients of the Chebyshev polynomials used to expand the input segment.
- is_acceleration_computed** flag stating if the feed-forward guidance torque shall be computed and applied during this segment.

Processing :

- Step 1 : Initialisation
 - $d = max_order - 1$
- Step 2 : Optimisation of the polynomials order
 - While** $segment_test(segment, d, constraints) = TRUE$ **and** $d > 1$
 - Repeat** $d = d - 1$
 - End While**
 - $degree = d + 1$
 - $[coeff, M] = segment_interpolation(segment, degree)$
- Step 3 : Computation of the acceleration flag (case of attitude guidance segment only)
 - $is_acceleration_computed = grd_guid_flag(segment, acc_lim)$

5.2.4.5 Procedure *segment_dichotomy*

Description :

This procedure divides the input profile into several segments, which must each fulfill procedure *segment_test* verification. If the number of iterations becomes higher than *nb_max*, the processing is aborted and declared unsuccessful.

Inputs :

profile input profile to be split, containing the following parameters :
 t_{seg}^i, t_{seg}^f, dt = start / end times of the segment & time-step (in s) : $t_{seg}^f - t_{seg}^i = n \cdot dt$,
 M = matrix (dim. $n \times i$) describing the segment i components during the time vector,
 i.e. either $X(n,6) = [X_{Ex}, X_{Ey}, X_{Ez}, X_{Sx}, X_{Sy}, X_{Sz}]$ or $Q(n,4) = [Q_0, Q_1, Q_2, Q_3]$.

max_order maximum authorised degree for the Chebyshev polynomials interpolation (≤ 7).

nb_max maximum authorised number of dichotomy iterations.

nb_iterations current number of dichotomy iterations (initialised at 0 in the main processing).

segments_list current list of segments which have fulfilled accuracy requirements (initialised at []).

nb_segments_list current number of segments in *segments_list* (initialised at 0).

constraints user thresholds : *thd_earth* = Earth direction maximum approximation error,
thd_sun = Sun direction maximum approximation error,
thd_angle = quaternion approximation maximum pointing error,
thd_rate = quaternion approximation maximum rate error,
acc_lim = acceleration limit for the feed-forward torque.

Outputs :

success flag indicating whether the dichotomy has reached the maximum authorised number of iterations.

nb_iterations updated number of dichotomy iterations.

segments_list updated list of segments which have fulfilled accuracy requirements.

nb_segments_list updated number of segments in *segments_list*.

Processing :

- Step 1 : Subdivision of the input profile into 2 segments of equal size

$$n_1 = \text{INT} \left(\frac{t_{seg}^f - t_{seg}^i}{2 \cdot dt} \right)$$

$$t_1 = t_{seg}^i + n_1 \cdot dt$$

$$M_1 = M(1:n_1, i)$$

$$M_2 = M(n_1:n, i)$$

$$seg_1 = \{ t_{seg}^i ; t_1 ; dt ; M_1 \}$$

$$seg_2 = \{ t_1 ; t_{seg}^f ; dt ; M_2 \}$$

- Step 2 : Next iteration

$nb_iterations = nb_iterations + 1$

If $nb_iteration \geq nb_max$

Then $success = FALSE$

Else

- Step 3 : Test of the 2 segments

If $segment_test(seg_1, max_order, constraints) = TRUE$

Then $segments_list = \{segments_list, seg_1\}$

$nb_segments_list = nb_segments_list + 1$

$success_1 = TRUE$

Else $[success_1, nb_iterations, segments_list, nb_segments_list] = segment_dichotomy(seg_1, nb_max, max_order, nb_iterations, segments_list, nb_segments_list, constraints)$

End If

If $segment_test(seg_2, max_order, constraints) = TRUE$

Then $segments_list = \{segments_list, seg_2\}$

$nb_segments_list = nb_segments_list + 1$

$success_2 = TRUE$

Else $[success_2, nb_iterations, segments_list, nb_segments_list] = segment_dichotomy(seg_2, nb_max, max_order, nb_iterations, segments_list, nb_segments_list, constraints)$

End If

$success = success_1 \text{ and } success_2$

End If

5.2.4.6 Procedure *segment_reassembly*

Description :

This procedure concatenates the segments provided by procedure *segment_dichotomy*, as far as the accuracy requirements are fulfilled, up to the minimal list of segments which can be issued from the initial profile.

Inputs :

segments_list list of segments provided by procedure *segment_dichotomy*, sorted in *chronological order*. Each segment includes the start time t_{seg}^i and the end time t_{seg}^f , and the list is such that for 2 successive segments seg_i and seg_{i+1} : $t_{seg(i+1)}^i = t_{seg(i)}^f$.

nb_segments_list number of segments in *segments_list*.

max_order maximum authorised degree for the Chebyshev polynomials interpolation (≤ 7).

constraints user thresholds : *thd_earth*, *thd_sun*, *thd_angle*, *thd_rate*, *acc_lim*.

Outputs :

final_segments_list final list of the initial profile segments subdivision provided by the processing

nb_segments number of segments in *final_segments_list*.

Processing :

- Step 1 : Initialisation

i = 1 -- Index of the current segment used as reference for concatenation

k = 1 -- Index of the current segment being added to segment *i*

nb_segments = 0 -- Number of segments after reassembly

- Step 2 : Concatenation of successive segments

While *i* < *nb_segments_list*

Repeat *current_segment* = [*seg_i*, *seg_{i+1}*... *seg_{i+k}*] -- Concatenation of segments *i* to *i+k*

prev_segment = [*seg_i*, *seg_{i+1}*... *seg_{i+k-1}*] -- Concatenation of segments *i* to *i+k-1*

If **segment_test** (*current_segment*, *max_order*, *constraints*) = TRUE

Then

If *i* + *k* \geq *nb_segments_list*

Then *final_segments_list* = { *final_segments_list*, *current_segment* }

nb_segments = *nb_segments* + 1

i = *nb_segments_list* + 1

Else *k* = *k* + 1

End If

Else *final_segments_list* = { *final_segments_list*, *previous_segment* }

nb_segments = *nb_segments* + 1

i = *i* + *k*

k = 1

End If

End While

- Step 3 : Case of isolated last segment

If $i = nb_segments$
Then $final_segments_list = \{final_segments_list, seg_{nb_segments_list}\}$
 $nb_segments = nb_segments + 1$
End If

5.2.4.7 Procedure segment_interpolation

Description :

This procedure computes the polynomial interpolation of an input segment, and provides the coefficients of the Chebyshev polynomials used to expand the segment, as well as the result of the polynomial expansion.

Inputs :

segment input segment containing the following parameters :
 t_{seg}^i, t_{seg}^f, dt = start / end times of the segment & time-step (in s), such that $t_{seg}^f - t_{seg}^i = n \cdot dt$,
 M = matrix (dim $n \times i$) describing the segment i components during the time vector, i.e. either
 $X(n,6) = [X_{Ex}(n), X_{Ey}(n), X_{Ez}(n), X_{Sx}(n), X_{Sy}(n), X_{Sz}(n)]$ or $Q(n,4) = [Q_0(n), Q_1(n), Q_2(n), Q_3(n)]$.
degree maximum degree of the Chebyshev polynomials used to expand the input segment.

Outputs :

coeff_i coefficients of the Chebyshev polynomials used to expand each column i of matrix M , truncated to the 3.1750 numerical accuracy.
 M^{exp} result of the polynomial expansion (dim. $n \times i$).

Processing :

- Step 1 : Creation of the Chebyshev polynomials T_r ($r = 0$ to **degree**)

$$T_0(t) = 1$$

$$T_1(t) = t$$

$$T_{r+1}(t) = 2 \cdot t \cdot T_r(t) - T_{r-1}(t)$$

- Step 2 : Computation of the transformed time-vector t_n

$$t_n = \frac{t_{seg}^i + n \cdot dt - \frac{t_{seg}^f + t_{seg}^i}{2}}{\frac{t_{seg}^f - t_{seg}^i}{2}} = \frac{2 \cdot n \cdot dt - (t_{seg}^f - t_{seg}^i)}{t_{seg}^f - t_{seg}^i}$$

- Step 3 : Computation of the Chebyshev interpolation matrix H (dim. $n \times degree + 1$)

$$H = \begin{bmatrix} T_0(t_0) & T_1(t_0) & \dots & T_{degree}(t_0) \\ T_0(t_1) & T_1(t_1) & \dots & T_{degree}(t_1) \\ \dots & \dots & \dots & \dots \\ T_0(t_n) & T_1(t_n) & \dots & T_{degree}(t_n) \end{bmatrix}$$

- Step 4 : Computation of the Chebyshev polynomials and result of the polynomial expansion

For each component i of the segment (6 for an ephemerides segment, 4 for a guidance segment)

$$coeff_i = \text{INV}(H^T \times H) \times H^T \times M(i)$$

$$coeff_i = \mathbf{fvalue1750}(coeff_i)$$

$$M^{exp}(i) = H \times coeff_i$$

End For

where $fvalue1750$ is a function which returns the numerical value of a float number at the 3.1750 numerical accuracy format (see § 5.2.4.9).

5.2.4.8 Procedure `grd_guid_flag`

Description :

This procedure computes the flag stating whether a feed-forward guidance torque shall be computed and commanded during the current segment, by comparing the maximum acceleration during the segment with the corresponding threshold.

Inputs :

segment attitude guidance input segment containing the following parameters :

t_{seg}^i, t_{seg}^f, dt = start / end times of the segment & time-step (in s), such that $t_{seg}^f - t_{seg}^i = n \cdot dt$,
 M = matrix (dim. $n \times 4$) describing the segment 4 components during the time vector, i.e.
 $M = Q(n,4) = [Q_0(n), Q_1(n), Q_2(n), Q_3(n)]$.

acc_lim acceleration limit for the feed-forward torque application : if the maximum acceleration during the segment is higher than this limit, the guidance feed-forward torque shall be computed and applied during attitude control.

Outputs :

is_acceleration_computed flag stating whether the feed-forward torque shall be computed or not.

Processing :

- Step 1 : Maximum acceleration among the 3 axes during the segment

$$acc_{max} = \text{MAX} \left(\frac{2}{dt^2} \cdot \left(Q^{-1}(2:n-1) \times Q(3:n) - Q^{-1}(1:n-2) \times Q(2:n-1) \right) \right)$$

- Step 2 : Computation of the acceleration flag

```

If     $acc_{max} \geq acc_{lim}$ 
Then  $is\_acceleration\_computed = TRUE$ 
Else  $is\_acceleration\_computed = FALSE$ 
End If
  
```

5.2.4.9 Procedure fvalue1750

Description :

This procedure computes the translation of a float number into the 3.1750 simple precision format, therefore accounting for the numerical resolution of the ROSETTA AOCMS Processor Module.

Inputs :

value the numerical value to be formatted to the 3.1750 simple precision standard

Outputs :

mantissa the mantissa of the input value in 3.1750 format
exponent the exponent of the input value in 3.1750 format
value1750 the numerical value of the input value after truncation to the 3.1750 format

Processing :

```

If (value ≠ 0) then
  sign = sign(value)
  • Step 1 : computation of the exponent
  If log(value) ≤ 0,
    temp = int( log(value) / log(2) )
  Else
    temp = int( log(value) / log(2) ) + 1
  End
  If (temp = log(value) / log(2) ) & sign>0,
    temp = temp + 1
  End
  exponent = temp
  • Step 2 : computation of the mantissa
  y = value/2^exponent
  If sign>0,
  
```

```

      u = y - 0.5
Else
      u = 1 - y
End
BitsFieldMant = zeros(1,22)
Coef = zeros(22,1)
i = 0
Coef(0) = 1/2
While i<22
      i = i + 1
      Coef(i) = Coef(i-1)/2
      BitsFieldMant(i) = int(u/Coef(i))
      u = u - BitsFieldMant(i) * Coef(i)
Endwhile
Mantissa = BitsFieldMant * Coef
If sign>0
      Mantissa = Mantissa + 0.5
Else
      Mantissa = Mantissa - 1
End
• Step 3 : computation of the truncated value
Value1750 = Mantissa * (2^exponent)
Else
Value1750 = Mantissa = exponent = 0
End

```

6 AFM PARAMETERS

6.1 CONTEXT AND PURPOSE

Several AFM parameters must be computed by the ground to support the flight procedure FCP-AC0030 “NM/FPAP mode transition to Asteroid Fly-by Mode” (see §7.22 in ref.[7]).

Before entering the AFM, the ground must provide the following parameters:

- an estimation of the spacecraft reference frame (Rast) defining the spacecraft axis to be pointed towards the asteroid, with respect to the spacecraft mechanical reference frame (to be sent as a quaternion, see §6.2)
- an estimation of the fly-by plane inertial reference frame (Ri) with respect to the J2000 reference frame (to be sent as a quaternion, see §6.3)
- an estimation of the initial commanded attitude quaternion in AFM with respect to the J2000 reference frame (see §6.4)

Before exiting the AFM, the ground must also provide an estimation of the commanded attitude quaternion corresponding to the AFM exit with respect to the J2000 reference frame (see §0).

6.2 COMPUTATION OF THE SPACECRAFT REFERENCE FRAME TO BE ASTEROID-POINTED

6.2.1 Principle

Before entering the AFM, the ground must provide an estimation of the spacecraft reference frame (Rast) which provides the orientation of the spacecraft axis to be pointed to the asteroid, with respect to the spacecraft mechanical reference frame (to be sent as a quaternion noted $Q_{SC^{cmd} / Rast}$).

This frame is introduced in order to account for any misalignment between the navigation camera bore-sight axis, defined as the AOCMS pointing reference, and the main scientific payload used during the asteroid fly-by, e.g. the Narrow Angle Camera.

The Rast frame has its Z axis defined by the payload bore-sight axis \vec{Z}_{ast} which must be pointed towards the asteroid center, and its Y axis such that it is as close as possible to the spacecraft mechanical Y axis, in order to achieve the fly-by slew manoeuvre about the axis of minimum inertia.

6.2.2 Inputs

\vec{Z}_{ast} = spacecraft axis to be pointed towards the asteroid center, expressed in the spacecraft mechanical reference frame. This axis is assumed to be close to the spacecraft mechanical Z axis, which nearly coincides with the navigation camera bore-sight axis

6.2.3 Outputs

$Q_{SC^{cmd}/Rast}$ = quaternion defining the rotation between the S/C mechanical reference frame and the Rast reference frame

6.2.4 Ground computation

This ground computation shall be done after the in-flight calibration of the navigation camera with respect to the main payload equipment to be used during AFM:

$$\vec{X}_{ast} = \frac{\vec{Y}_{SC} \otimes \vec{Z}_{ast}}{\|\vec{Y}_{SC} \otimes \vec{Z}_{ast}\|}, \quad \vec{Y}_{ast} = \vec{Z}_{ast} \otimes \vec{X}_{ast}$$

The matrix corresponding to the rotation from the (Rast) reference frame to the S/C mechanical reference frame is made of these three line vectors:

$$M_{SC/Rast} = \begin{pmatrix} \vec{X}_{ast} \\ \vec{Y}_{ast} \\ \vec{Z}_{ast} \end{pmatrix}$$

The quaternion defining the rotation between the S/C mechanical reference frame and the Rast reference frame can be calculated using procedure *mat2qua* (described in §3.3.3.2):

$$Q_{SC^{cmd}/Rast} = \text{mat2qua}(M_{SC/Rast})$$

6.3 COMPUTATION OF THE FLY-BY PLANE QUATERNION

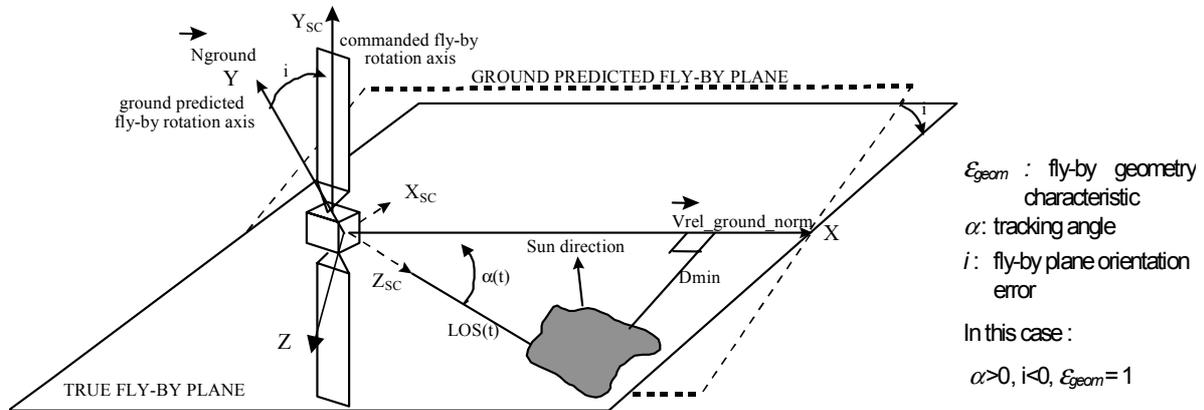
6.3.1 Principle

Before entering the AFM, the ground must provide an estimation of the fly-by plane inertial reference frame (R_i) with respect to the J2000 reference frame (to be sent as a quaternion noted $Q_{R_i/J2000}$).

The fly-by slewing plane is an inertially fixed plane defined by the Asteroid - spacecraft relative velocity \vec{V}_{rel} (assumed constant in direction and magnitude) and by the inertial direction of any Line Of Sight vector (\vec{LOS}) between the spacecraft and the Asteroid center of mass (for example the LOS vector at the minimum distance is \vec{D}_{min}).

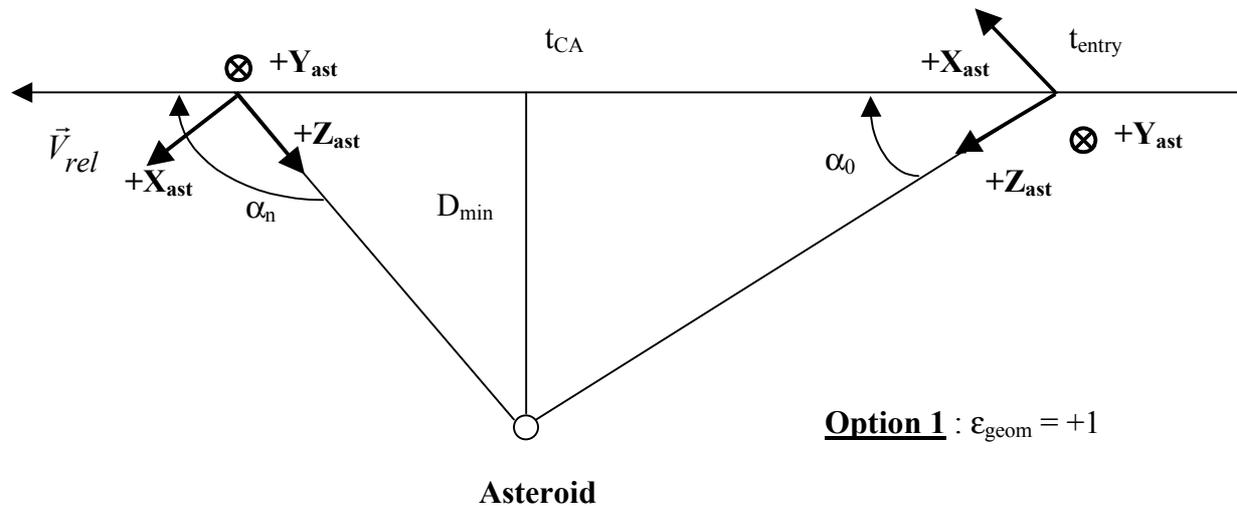
The estimated fly-by plane inertial reference frame R_i is defined by the ground estimates of the relative velocity unit vector $\vec{V}_{rel_ground_norm}$ and of the direction perpendicular to the fly-by plane \vec{N}_{ground} (defined by **plus or minus** the vectorial product between the Line Of Sight vector and the relative

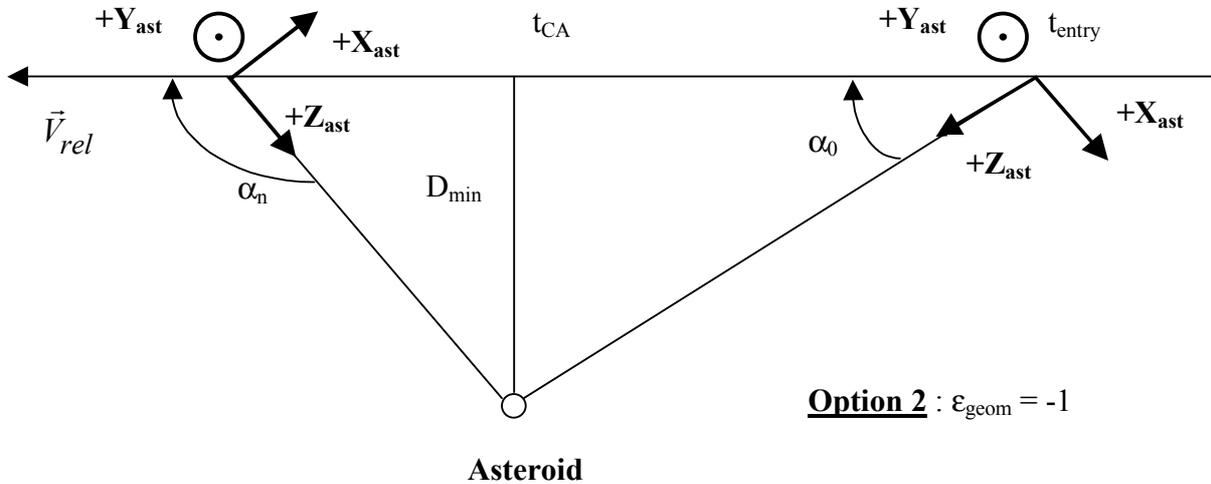
velocity vector, in that order). The X axis is defined by $\vec{V}_{rel_ground_norm}$, the Y axis by \vec{N}_{ground} and the Z axis is such as to complete the orthogonal direct frame.



Fly-by geometry

The actual definition of the fly-by plane reference frame R_i will depend on the fly-by geometry. There are two possible options as shown in the figures here after. The selection of a given option by the ground depends on thermal constraints on -X satellite face.





6.3.2 Inputs

\vec{V}_{rel} = Asteroid - spacecraft relative velocity in the J2000 inertial reference frame (assumed constant in direction and magnitude)

$$= \vec{V}_{S/C} - \vec{V}_{asteroid} \quad (\text{both velocity vectors being defined in J2000})$$

$\overline{LOS}_{/J2000}$ = direction of any Line Of Sight vector between the spacecraft and the Asteroid in the J2000 inertial reference frame (e.g. as measured by the navigation camera LOS measurement to the asteroid just before entering AFM)

6.3.3 Outputs

$Q_{Ri/J2000}$ = quaternion defining the orientation of the fly-by plane inertial reference frame in the J2000 reference frame

6.3.4 Ground computation

This ground computation shall be done after the last ΔV manoeuvre on the final free trajectory arc. Each axis (X, Y, Z) of the ground-estimated fly-by plane inertial reference frame (Ri) is equal to:

$$X = \vec{V}_{rel_ground_norm} = \frac{\overline{V_{rel}}}{\|\overline{V_{rel}}\|}, \quad Y = \overline{N_{ground}} = \epsilon_{geom} \cdot \frac{\overline{LOS} \otimes \overline{V_{rel}}}{\|\overline{LOS} \otimes \overline{V_{rel}}\|}, \quad Z = X \otimes Y$$

where ϵ_{geom} ($= \pm 1$) depends on the fly-by geometry selected by the ground. The fly-by geometry is determined by the ground in order to avoid that the sun comes too close to a given spacecraft face (-X face in principle).

The matrix corresponding to the rotation from the J2000 inertial reference frame to the ground estimated fly-by plane inertial reference frame (Ri) is made of these three line vectors:

$$M_{Ri/J2000} = M_{J2000 \rightarrow Ri} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

The quaternion defining the rotation between the ground-estimated fly-by plane inertial reference frame (Ri) and the J2000 reference frame can be calculated using procedure *mat2qua* (described in §3.3.3.2):

$$Q_{Ri/J2000} = \text{mat2qua}(M_{Ri/J2000})$$

6.4 COMPUTATION OF THE ATTITUDE QUATERNION AT AFM ENTRY

6.4.1 Principle

In order to verify the constraint that the commanded attitude quaternion processed through the on-board guidance function at NM / FPAP exit must correspond to the commanded attitude quaternion computed through the on-board guidance function at AFM entry, it is necessary to estimate this latter commanded quaternion.

Knowing the AFM attitude guidance parameters ($Q_{S/C^{cmd}/R_{ast}}$, \vec{V}_{rel} , \vec{D}_{min} and $Q_{Ri/J2000}$) and the time of flight data, the ground can compute the initial commanded attitude quaternion in AFM with respect to the J2000 reference frame (noted Q_{AFM_init}).

6.4.2 Inputs

$\|\vec{V}_{rel}\|$ = norm of the Asteroid - spacecraft relative velocity in the J2000 inertial reference frame
 (assumed constant)

$\|\vec{D}_{min}\|$ = minimum distance between the spacecraft and the Asteroid in the J2000 inertial reference frame

t_{CA} = time at the closest approach (assumed as reference)

t_{entry} = time at the AFM entry ($t_{entry} - t_{CA} < 0$)

$Q_{SC^{cmd}/R_{ast}}$ = ground-commanded quaternion corresponding to the rotation from the spacecraft mechanical reference frame to a spacecraft frame R_{ast} associated to an axis Z_{ast} (near the spacecraft Z axis) to be pointed towards the Asteroid optical center (computed in §6.2)

$Q_{Ri/J2000}$ = quaternion corresponding to the rotation from the ground-estimated fly-by plane inertial reference frame (Ri) to the J2000 reference frame (computed in §6.3)

6.4.3 Outputs

Q_{AFM_init} = quaternion defining the spacecraft attitude at AFM entry with respect to the J2000 reference frame

6.4.4 Ground processings

The in-plane tracking angle α at the AFM entry can be computed as follows (see §4.2 in ref.[10]):

$$\alpha_0 = \alpha(t_{entry}) = \mathcal{E}_{geom} \cdot \arctan \frac{\| \overline{D_{min}} \|}{\| \overline{V_{rel}} \| \cdot |t_{entry} - t_{CA}|} \quad (\alpha \text{ sign depends on the fly-by geometry})$$

The quaternion corresponding to the rotation from the spacecraft reference frame at the AFM entry to the J2000 reference frame can be calculated as follows:

$$Q_{SC^{cmd}/J2000} = Q_{SC^{cmd}/Rast} \cdot Q_{Rast/Ri} \cdot Q_{Ri/J2000}$$

where $Q_{Rast/Ri}$ = quaternion corresponding to the rotation from the R_{ast} reference frame to the ground-estimated fly-by plane inertial reference frame R_i , as computed below

The matrix corresponding to the rotation from the ground-estimated fly-by plane inertial reference frame R_i to the R_{ast} reference frame is equal to:

$$M_{Rast/Ri} = M_{Ri \rightarrow Rast} = \begin{pmatrix} \sin \alpha_0 & 0 & -\cos \alpha_0 \\ 0 & 1 & 0 \\ \cos \alpha_0 & 0 & \sin \alpha_0 \end{pmatrix}$$

The definition of the above matrix shows that vectors \vec{Y}_{ast} (which is close to $\vec{Y}_{S/C}$) and \vec{Y} are identical at the beginning of the asteroid fly-by. Note that all along the asteroid fly-by both vectors are supposed to remain very close.

Therefore the quaternion corresponding to the rotation from the R_{ast} reference frame to the ground-estimated fly-by plane inertial reference frame R_i can be calculated using procedure *mat2qua* (described in §3.3.3.2):

$$Q_{Rast/Ri} = \text{mat2qua}(M_{Rast/Ri})$$

The initial commanded attitude quaternion in AFM with respect to the J2000 reference frame can then be computed as follows:

$$Q_{AFM_init} = \overline{Q_{SC^{cmd}/J2000}} = \overline{Q_{SC^{cmd}/Rast} \cdot Q_{Rast/Ri} \cdot Q_{Ri/J2000}}$$

6.5 COMPUTATION OF THE ATTITUDE QUATERNION AT AFM EXIT

6.5.1 Principle

Since NM / WDP is entered at AFM exit, the WDP inertial commanded attitude must correspond to the attitude at AFM exit. Hence, it is necessary to estimate this AFM exit attitude quaternion with respect to the J2000 reference frame (noted Q_{AFM_exit}).

If this computation can be done after the AFM entry, the fly-by plane on-board estimated inclination error (i) (available through the TM AFM_GUID.pred_state_vect) can be taken into account (along with the AFM attitude guidance parameters ($Q_{S/C^{cmd}/R_{ast}}$, \vec{V}_{rel} , \vec{D}_{min} and $Q_{R_i/J2000}$)) in the calculation of the AFM exit attitude quaternion. Otherwise this inclination error will be assumed to be null ($i = i_0 = 0$).

6.5.2 Inputs

$\|\vec{V}_{rel}\|$ = norm of the Asteroid - spacecraft relative velocity in the J2000 inertial reference frame
 (assumed constant)

$\|\vec{D}_{min}\|$ = minimum distance between the spacecraft and the Asteroid in the J2000 inertial reference frame

t_{CA} = time at the closest approach

t_{exit} = time at the AFM exit

$Q_{SC^{cmd}/R_{ast}}$ = ground-commanded quaternion corresponding to the rotation from the spacecraft mechanical reference frame to a spacecraft frame R_{ast} associated to an axis Z_{ast} (near the spacecraft Z axis) to be pointed towards the Asteroid optical center (computed in §6.2)

$Q_{R_i/J2000}$ = quaternion corresponding to the rotation from the ground-estimated fly-by plane inertial reference frame (Ri) to the J2000 reference frame (computed in §6.3)

$|i| = errY \leq 4^\circ$ = fly-by plane estimated inclination error, i.e. angle between the on-board estimated inertial direction of the perpendicular to the slewing plane and the initial ground estimated one

6.5.3 Outputs

Q_{AFM_exit} = quaternion defining the spacecraft attitude at AFM exit with respect to the J2000 reference frame

6.5.4 Ground processings

The in-plane tracking angle α_n at the AFM exit can be computed as follows:

$$\alpha_n = \alpha(t_{exit}) = \mathcal{E}_{geom} \cdot \left(\pi - \arctan \frac{\|D_{min}\|}{\|V_{rel}\| \cdot |t_{exit} - t_{CA}|} \right) \quad \left(\frac{\pi}{2} < |\alpha_n| < \pi, \alpha_n \text{ sign depends on the fly-by geometry ; see §6.3.1} \right)$$

The quaternion corresponding to the rotation from the spacecraft reference frame at the AFM exit to the J2000 reference frame can be calculated as follows:

$$Q_{SC^{cmd}/J2000} = Q_{SC^{cmd}/R_{ast}} \cdot Q_{R_{ast}/R_{i0}} \cdot Q_{R_{i0}/J2000}$$

where $Q_{R_{ast}/R_{i0}}$ = quaternion corresponding to the rotation from the R_{ast} reference frame to the **current on-board** estimated fly-by plane reference frame R_{i0} , as computed below

The matrix corresponding to the rotation from the **on-board** estimated fly-by plane reference frame R_{i0} to the R_{ast} reference frame can be written as:

$$M_{R_{ast}/R_{i0}} = M_{R_{ast}/R_i} \cdot M_{R_i/R_{i0}}$$

where M_{R_{ast}/R_i} = matrix of rotation from the **initial ground-estimated** fly-by plane reference frame R_i to the R_{ast} reference frame (see § 6.4.4)

$M_{R_i/R_{i0}}$ = matrix of rotation from the **on-board** estimated fly-by plane inertial reference frame R_{i0} to the **ground-estimated** fly-by plane **inertial** reference frame R_i

$$M_{R_{ast}/R_{i0}} = \begin{pmatrix} \sin \alpha_n & 0 & -\cos \alpha_n \\ 0 & 1 & 0 \\ \cos \alpha_n & 0 & \sin \alpha_n \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos i & -\sin i \\ 0 & \sin i & \cos i \end{pmatrix}$$

$$= \begin{pmatrix} \sin \alpha_n & -\cos \alpha_n \cdot \sin i & -\cos \alpha_n \cdot \cos i \\ 0 & \cos i & -\sin i \\ \cos \alpha_n & \sin \alpha_n \cdot \sin i & \sin \alpha_n \cdot \cos i \end{pmatrix}$$

(here $i < 0$; if $i = 0$, the above equations still apply, but then R_{i0} and R_i are identical and thus $M_{R_i/R_{i0}}$ is the identity matrix).

Warning: As the angle 'i' is determined on-board the spacecraft in real-time during the asteroid fly-by, the calculation of $M_{R_{ast}/R_{i0}}$ by the ground would require the use of current on-board data. Given the short AFM duration and the actual communication delays between ground and S/C, such a situation is not acceptable from an operations point of view. It is therefore advised to assume the angle 'i' is negligible and consider that R_{i0} and R_i are identical.



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : **05/08/2003**
Page : 51

Therefore, assuming R_{i0} and R_i are identical, the quaternion corresponding to the rotation from the R_{ast} reference frame to the current on-board estimated fly-by plane reference frame R_{i0} can be calculated using procedure *mat2qua* (described in §3.3.3.2):

$$Q_{Rast/Ri_0} \square Q_{Rast/Ri} = mat2qua(M_{Rast/Ri})$$

The commanded attitude quaternion at the AFM exit with respect to the J2000 reference frame can then be computed as follows:

$$Q_{AFM_exit} = \overline{Q_{SC^{cmd}/J2000}} \square \overline{Q_{SC^{cmd}/Rast}} \cdot Q_{Rast/Ri} \cdot Q_{Ri/J2000}$$

Remark : the approximation $R_i = R_{i0}$ may induce a small attitude transient at AFM exit which is compatible with NM/WDP operating conditions.

7 FREQUENCY OF SA RE-ORIENTATIONS IN NSH

7.1 CONTEXT AND PURPOSE

If the “X axis Earth pointed” option is selected in NSH, to keep the communications with the ground, the solar arrays will not be oriented towards the Sun continuously : due to the evolution of the Sun - spacecraft - Earth angle during the long NSH periods, periodic adjustment of the SA rotation angles is necessary from the ground to ensure that the solar aspect angle (as seen from the arrays-mounted SAS) remains compatible with the linear range Sun pointing surveillance.

The number of such necessary ground interventions depends on the Sun / Earth configuration during each NSH phase, and on the selected guidance options and angular dead-band widths.

This Section presents a method to predict when the SA re-orientations are necessary and how to handle their positioning.

7.2 PROCESSING

7.2.1 Principle

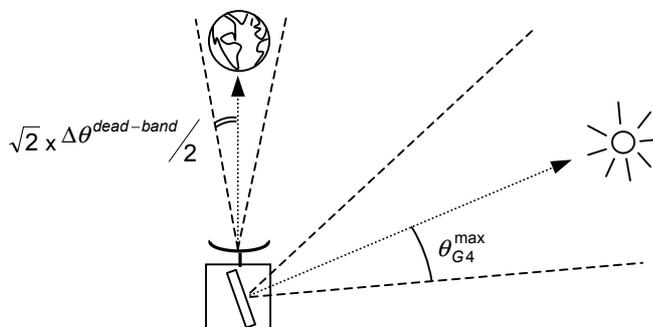
The solar arrays orientation must remain such that during the whole NSH phases, the solar aspect angle as seen from the SA-mounted SAS always remains lower than the linear Sun pointing surveillance (G4) limit : $\theta_{G4}^{\max} = 15^\circ$ (from ref. [6]).

When the “X axis Earth pointed” option is selected in NSH, using an angular dead-band width $\Delta\theta^{\text{dead-band}}$ on each axis, the angular excursion from the nominal pointing can reach up to (half cone error) :

$$\sqrt{2} \times \Delta\theta^{\text{dead-band}} / 2 \quad \text{in case of combined } \pm \Delta\theta^{\text{dead-band}} / 2 \text{ pointing errors about 2 different axes.}$$

Hence, the minimum requirement on the dead-band width selection is that :

$$\sqrt{2} \times \Delta\theta^{\text{dead-band}} / 2 < \theta_{G4}^{\max} \Rightarrow \Delta\theta^{\text{dead-band}} < \sqrt{2} \times \theta_{G4}^{\max} \cong \pm 10^\circ.$$



But selecting a $\pm 10^\circ$ dead-band would require to update the solar arrays orientation almost continuously. Hence, taking into account sufficient margins in order to limit the frequency of ground

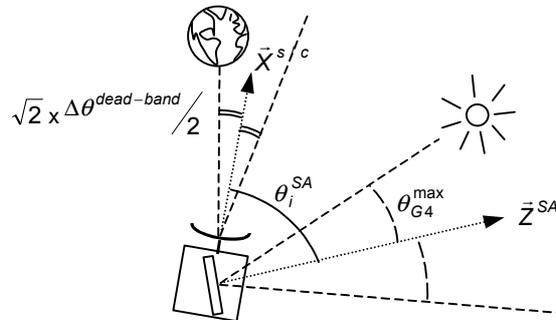
interventions, it is recommended to select a dead-band width below typically $\pm 5^\circ$ (this recommendation applies only when the “Earth pointing” option is selected).

In the following Section, mission analysis data is used to determine what are the SA re-orientation constraints as a function of the selected Earth pointing control dead-band.

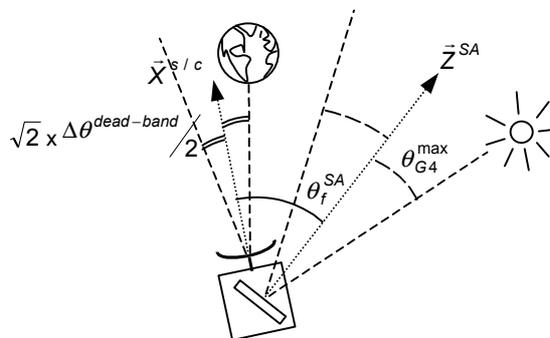
7.2.2 Numerical analysis

The extreme situation which requires to update the solar arrays orientation in NSH is when the maximum angular excursion from the nominal pointing ($\frac{\sqrt{2} \times \Delta\theta^{dead-band}}{2}$) causes the solar aspect angle (as seen from the arrays-mounted SAS) to reach exactly surveillance G4 maximum limit

θ_{G4}^{max} :



The solar arrays (which are in an initial angular position θ_i^{SA}) must then be rotated towards a final angular position θ_f^{SA} , such that the maximum angular excursion *on the opposite side* of the nominal pointing causes the SA solar aspect angle to reach surveillance G4 maximum limit θ_{G4}^{max} *on the opposite side* :



Hence, a ground intervention is necessary to adjust the SA rotation angles as soon as the Sun - spacecraft - Earth angle SSCE becomes such that :

$$SSCE = \theta_i^{SA} + \sqrt{2} \times \Delta\theta^{dead-band} / 2 - \theta_{G4}^{max} \quad \text{or} \quad SSCE = \theta_f^{SA} - \sqrt{2} \times \Delta\theta^{dead-band} / 2 + \theta_{G4}^{max}$$

The delay between 2 such ground interventions thus corresponds to the delay needed by the Sun - spacecraft - Earth angle to increase or decrease by :

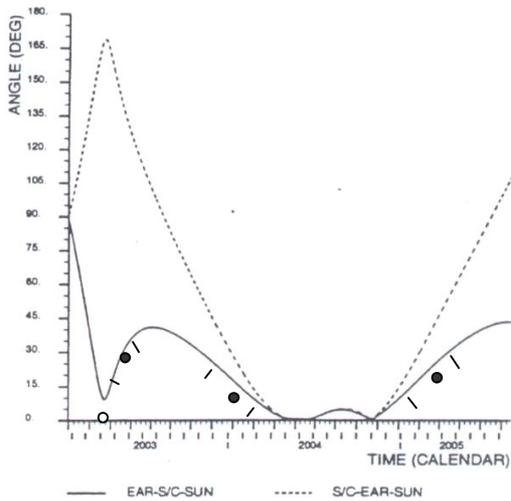
$$\Delta\theta = \left| \sqrt{2} \times \Delta\theta^{dead-band} - 2 \cdot \theta_{G4}^{max} \right|$$

The ground interventions necessary with a $\pm 5^\circ$ dead-band correspond to each variation of the Sun - spacecraft - Earth angle by : $\Delta\theta_{\pm 5^\circ} \cong 16^\circ$.

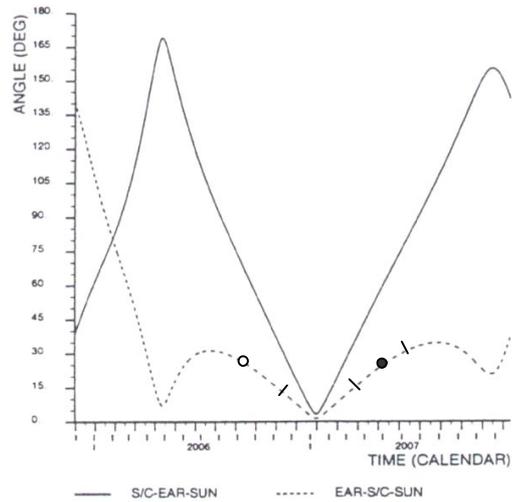
The minimum number of ground interventions is obtained using the minimum authorized dead-band width of $\pm 1^\circ$, and corresponds to each variation of the Sun - spacecraft - Earth angle by : $\Delta\theta_{\pm 1^\circ} \cong 27^\circ$.

Figure 7.2-1 provides the evolution of the Sun - spacecraft - Earth angle during the different phases of the mission where NSH is to be used. **This data is preliminary**, and based on early orbital assumptions provided by ESOC at the beginning of the project **when the baseline mission was still a rendez-vous with Wirtanen. It shall be updated by ESOC** according to the latest mission profile **and once it is decided in what phases of the mission the NSH will be activated.**

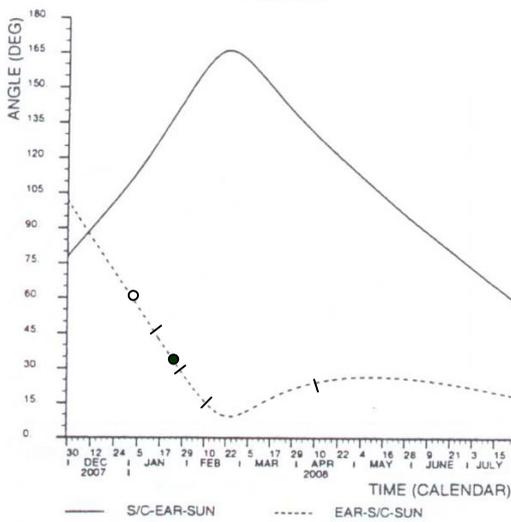
These plots showing the evolution of the SSCE angle (for the original Wirtanen mission, these ones being not yet available for the new baseline mission to Churyumov-Gerasimenko) are used to assess the number of SA re-orientations which will be necessary in NSH, during the following phases of the mission :



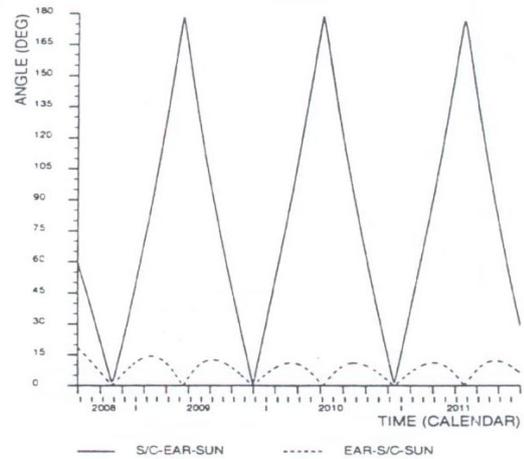
Earth to Mars cruise #1



Earth - Otawara - Earth cruises #3 & #4



Earth to Siwa cruise #5



Siwa to Wirtanen & DSH cruises #6 & #7

Figure 7.2-1 : Sun - spacecraft - Earth angle (°)

The dark dots (●) are used to mark the required SA re-orientations when NSH is used with a $\pm 1^\circ$ dead-band, which corresponds to each variation of the Sun - spacecraft - Earth angle by 27° :

- at least 3 ground interventions are necessary during Earth to Mars cruise #1, after the initial SA orientation at NSH entry (○) ;
- the Otawara to Earth cruise #4 requires about 1 ground intervention after the initial SA orientation ;
- about 1 ground intervention is required in the first months of Earth to Siwa cruise #5 (after the initial SA orientation) ;

- finally, no ground intervention is necessary during Siwa to Wirtanen cruise #6 (nor during Deep Space Hibernation cruise #7, if NSH is used in backup), once the SA have been adequately oriented at NSH entry.

The dashes (-) are used to mark the required SA re-orientations when NSH is used with a $\pm 5^\circ$ dead-band, which corresponds to each variation of the Sun - spacecraft - Earth angle by 16° :

- up to approximately 6 ground interventions are necessary during Earth to Mars cruise #1 ;
- the Otawara to Earth cruise #4 requires about 3 ground interventions ;
- about 4 ground interventions are required in the first months of Earth to Siwa cruise #5 ;
- finally, no ground intervention is necessary during Siwa to Wirtanen cruise #6 (nor during Deep Space Hibernation cruise #7, if NSH is used in backup), since the Sun - spacecraft - Earth angle remains between 0 and 15° .

The above method should be used to determine at which date the solar arrays need to be re-oriented during NSH when the Earth pointing option is selected, on each orbital arc of the **new** Rosetta mission to **Churyumov-Gerasimenko**.

8 REFERENCE ATTITUDE QUATERNION AT SPM ENTRY

8.1 CONTEXT AND PURPOSE

Deep Space Hibernation is entered through the Spin-up Mode (SpM), which provides a spin-stabilization of the spacecraft attitude, such that its principal axis of inertia remains inertially fixed during the whole hibernation phase. The spacecraft is spun about a specific inertial direction, which is commanded by the ground.

Before entering SpM, the ground must also provide an estimation of the spacecraft principal inertia axes orientation with respect to the spacecraft reference frame (see ref. [7], § 7.26). The spacecraft attitude at SpM entry must be such that the ground-estimated principal axis of maximum inertia is pointed towards the commanded inertial spin direction. This attitude is reached through an angular slew in Normal Mode, before the AOCMS is switched to TTM and then to SpM. To be able to command this manoeuvre and these transitions, the ground must compute the quaternion corresponding to this SpM entry initial attitude.

8.2 PROCESSING

8.2.1 Principle

The main constraint on the SpM initial attitude is that the spacecraft principal axis of maximum inertia, as estimated by the ground, must be pointed towards the commanded inertial spin direction.

In addition, the HGA must be kept pointing towards the Earth, using only one articulation drive : indeed, it is expected that the azimuth angle is close to 0° , while the elevation angle is negative. Hence, the Earth direction must be in the spacecraft XZ plane.

Both constraints enable to define a single reference attitude, expressed as a quaternion with respect to the J2000 inertial frame.

8.2.2 Inputs / Outputs

The spacecraft attitude quaternion at SpM entry is computed from the ground-estimated principal axis of maximum inertia, from the commanded inertial spin direction, and from the spacecraft to Earth direction in the inertial frame.

Inputs :

The required inputs for the reference attitude quaternion processing are :

- $[x \ y \ z]$: coordinates (in the spacecraft frame) of the ground-estimated principal axis of maximum inertia.
- \vec{D}_i^c : spin direction selected by the ground, in the J2000 inertial frame.
- \vec{E}_j : spacecraft to Earth direction expressed in the J2000 inertial frame.

The spin-up geometry for the new mission to Churyumov-Gerasimenko is not yet available (TBD by Astrium GmbH and ESOC), but the spin direction \vec{D}_i^c is expected to be close to the Sun direction at orbit aphelion, and the Earth direction \vec{E}_i is expected to be included in the spacecraft X/Z plane, thus allowing to set the HGA boresight axis in that plane using only one HGA articulation drive.

Outputs :

The processing output is the spacecraft attitude quaternion to be commanded at SpM entry (transformation quaternion from the inertial frame to the spacecraft frame) : Q_{att} .

8.2.3 Ground processings

- Step 1 : The spacecraft principal axis of inertia \vec{X}^p must be pointed towards the spin direction

$$\vec{X}^p = \vec{D}_i^c$$

- Step 2 : The spacecraft Y axis must be perpendicular to the Earth - spacecraft - spin direction plane

$$\vec{Y}_{sc} = \vec{E}_i \wedge \vec{D}_i^c$$

$$\vec{Y}_{sc} = \frac{\vec{Y}_{sc}}{\|\vec{Y}_{sc}\|} \quad \text{-- Normalize vector}$$

- Step 3 : Correction of \vec{Y}_{sc} to compensate for the spacecraft principal axis of inertia y misalignment

$$\vec{Y}_{sc} = \left(1 - \frac{y^2}{2}\right) \cdot \vec{Y}_{sc} + y \cdot \vec{X}^p$$

$$\vec{Y}_{sc} = \frac{\vec{Y}_{sc}}{\|\vec{Y}_{sc}\|} \quad \text{-- Normalize vector}$$

- Step 4 : Inertial direction of the spacecraft X axis

$$\vec{X}_{sc} = x \cdot \vec{X}^p - z \cdot (\vec{X}^p \wedge \vec{Y}_{sc}) - x \cdot y \cdot \vec{Y}_{sc}$$

$$\vec{X}_{sc} = \frac{\vec{X}_{sc}}{\|\vec{X}_{sc}\|} \quad \text{-- Normalize vector}$$

- Step 5 : Inertial direction of the spacecraft Z axis

$$\vec{Z}_{sc} = \vec{X}_{sc} \wedge \vec{Y}_{sc}$$

$$\bar{Z}_{sc} = \frac{\bar{Z}_{sc}}{\|\bar{Z}_{sc}\|}$$

-- Normalize vector

- Step 6 : Final attitude quaternion derived from the spacecraft axes directions

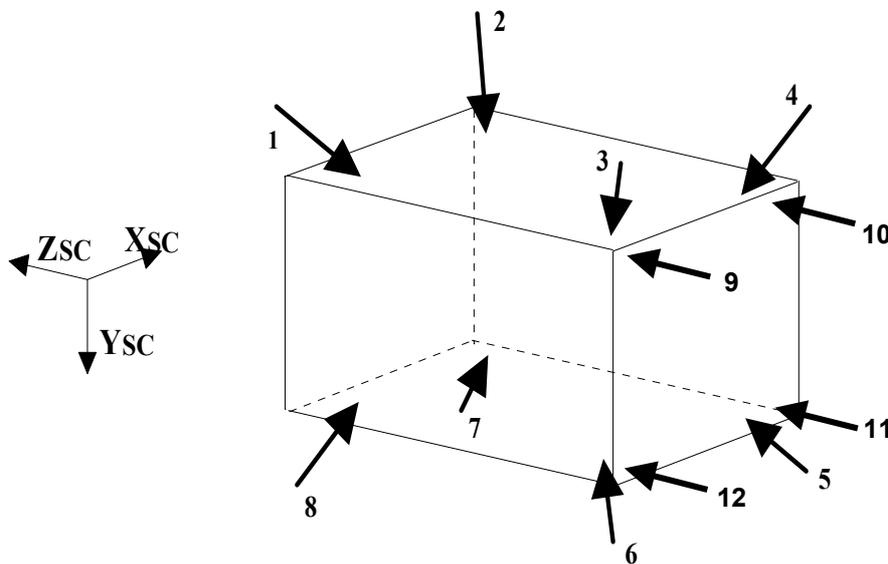
$$Q_{att} = \mathbf{mat2qua} \left(\begin{bmatrix} \bar{X}_{sc} & \bar{Y}_{sc} & \bar{Z}_{sc} \end{bmatrix} \right)$$

-- Input = matrix made of 3 columns \bar{X}_{sc} , \bar{Y}_{sc} , \bar{Z}_{sc}
 -- Procedure *mat2qua* is described in § 3.3.3.2.

9 RCS_ORB PARAMETERS

9.1 CONTEXT AND PURPOSE

The RCS_ORB SW object is in charge of computing thrusters on-times commands as a function of the torques and forces commanded by the control laws. The ROSETTA thrusters configuration derived from [4] is depicted below. It consists in 2x12 thrusters (each thruster pod holds one nominal “A” and one redundant “B” thruster), i.e. each thruster is duplicated. The 2x4 thrusters located on the $-Z$ side of the spacecraft central body are used during OCM Burn manoeuvres in off-modulation, while the 2x8 tilted thrusters are used for 3-axis attitude control and for Vectored Thrust Manoeuvres.



The main characteristic of this thruster configuration is that pure axial torques can be provided by the eight tilted thrusters, i.e. no parasitic force is generated (apart thrusters misalignments effects), which is essential while orbiting the comet. However only four tilted thrusters would be enough to produce 3-axis torque capacity, would multiple failures prevent the use of the full set.

The thrusters are commanded through the AIU, by intermediate Thruster Driver Control Modules (TDC) inside each ACM, the connections being as follows :

Nominal ACM			Redundant ACM		
TDC1A	TDC2A	TDC3A	TDC1B	TDC2B	TDC3B
1A	5A	9A	1B	5B	9B
2A	6A	10A	2B	6B	10B
3A	7A	11A	3B	7B	11B
4A	8A	12A	4B	8B	12B

By default the nominal thruster configuration contains all thrusters of branch A, while the redundant configuration contains all thrusters of branch B. However the on-board software offers the capacity of changing these default configurations by discarding one or several failed thrusters (one TDC failure

discards four thrusters), as well as mixing thrusters from branches A and B in the nominal or the redundant configurations, in order to accommodate several thruster failures while still trying to maintain a full independence between the nominal and the redundant configurations.

The computation of thrusters on-times from commanded forces and torques requires the use of several on-board parameters which must be computed on-ground, some of them being dependent on the spacecraft center of mass position, therefore requiring frequent updates. Furthermore these parameters need to be up-linked every time the nominal / redundant thruster configuration contents are changed by the ground, e.g following in-orbit thruster failures. This chapter deals with the algorithms to be processed on-ground in order to compute these parameters.

9.2 PROCESSING

9.2.1 Principle

The RCS_ORB SW object handles the thrusters modulator and aims at computing thruster on-times from commanded 3-axis torques and forces. It first computes a thrusters on-times ratio vector from ground uplinked actuation tables (which link each axial torque to a given thruster combination), which complies to the commanded torques. It then adds the force demand given by a specific thruster combination, either uplinked by the ground (case of Vectored thrust manoeuvres), or defined a priori (case of burn manoeuvres). Finally it minimizes the sum of the thrusters on-times ratio vector in order to minimize the fuel consumption, through a one step optimization process based on an iterative Simplex algorithm.

In order to perform the previous operations, many parameters linked to the thrusters configuration must be uplinked by the ground. These parameters depend on :

- The physical characteristics of the thrusters nominal and redundant configurations, i.e. number of thrusters, thrusters positions, directions, force level
- The spacecraft center of mass coordinates with respect to thrusters positions
- The thrusters Minimum Impulse Bit characteristics.

Some parameters need to be updated only when the thrusters nominal / redundant configurations are changed by the ground (e.g. after failure passivation), see Section 9.2.4.3, some other parameters need to be updated before each Burn manoeuvre, as soon as the spacecraft center of mass position has significantly varied (Section 9.2.4.2).

One major parameter which must be uplinked with a new thruster configuration is the physical assignment of each of the twelve thrusters defined previously to branch A, branch B or to an empty assignment : *nom/red_used_thr* is a 12-element vector containing A, B, or 0 values (0 means that no physical thruster is assigned to a numbered thruster pod). Nominally, *nom_used_thr* elements are all A, and *red_used_thr* elements are all B. This parameter shall be defined at system level only if a failure prevents the use of the default values, after failure investigation to identify the faulty thrusters. However some recommendations can be provided at avionics level concerning that point :

- it is recommended to discard a thruster which has been identified by the ground as being failed or presenting malfunctions.
- it is recommended to select distinct thrusters for nominal and redundant configurations, in order to enable an adequate thruster failure passivation when a reconfiguration is commanded
- as far as possible, it is recommended to select distinct TDC Modules for nominal and redundant configurations, in order to enable an adequate TDC failure passivation when a reconfiguration is commanded
- a thruster configuration shall contain at least four thrusters mounted on the same spacecraft body side to allow 3-axis attitude control : these thrusters quadruplets are 1/2/3/4, 3/4/5/6, 5/6/7/8, 1/2/7/8, 1/3/6/8, 2/4/5/7. However using only four thrusters means that parasitic forces will be generated along with axial torques. Therefore before selecting a reduced thruster configuration, it shall be checked at system / ESOC level that parasitic forces do not endanger the spacecraft safety by impacting too much its orbit around the comet.

9.2.2 Inputs

The following system inputs are necessary :

- *Nom/red_used_thr* : 1x12 vectors of 0, A, B elements defining the physical selection of the twelve thrusters for the nominal and redundant thrusters configurations (e.g. *nom_used_thr* = [A 0 A A B B 0 A A A A])
- *Com_pos* : 1x3 position vector of the spacecraft center of mass with respect to the spacecraft geometrical reference frame.

The following system constant parameters are necessary :

- *Thrust* : 1x24 vector containing the default nominal thrust level of each thruster (A branch first, then B branch) with pressurized tanks, assumed 10N. This data corresponds to a blow-down factor (*bdf*) of 1, this parameter being regularly up-linked to correct the thrusters parameters with the adequate thrust level depending on the tank pressure. The default nominal thrust of each thruster can be determined from on-ground calibration results or from in-flight calibration, if applicable.
- *Thrust_pos* : 3x24 array containing the position vectors of the 24 thrusters (A branch first, then B branch) with respect to the spacecraft geometrical reference frame. This data is expected to be issued from S/C integration results, or from [4])
- *Thrust_dir* : 3x24 array containing the direction cosine vectors of the 24 thrusters (A branch first, then B branch) with respect to the spacecraft geometrical reference frame. This data is expected to be issued from S/C integration results, or from [4])

The following AOCMS SW constant data are necessary :

- *Min_on_time* : the thruster minimum on-time defined on-board to manage the thrusters Minimum Impulse Bit (defined as 5ms in [4])

- *Alpha_cold, beta_cold, alpha_hot, beta_hot* : on-board coefficients of thrusters cold and hot efficiency models (defined in [11])

9.2.3 Outputs

The following parameters are computed by the present ground processing algorithms :

- the number of tilted thrusters to be processed for attitude control : *nom/red_numb_thr* ; this integer should be equal to 8 minus the number of tilted thrusters (i.e. thrusters numbered 1 to 8) defined as zero in *nom/red_used_thr*. It indicates to the on-board SW the number of thrusters on which the on-times shall be optimised with respect to fuel consumption. Indeed the RCS-ORB SW always tries to minimise the sum of thrusters on-times by driving at least one thruster on-time to zero. In the case where one thruster is discarded in the configuration, the RCS_ORB SW shall not take into account its null on-time in the optimisation process.
- the maximum positive / negative torque levels about each spacecraft axis, for the nominal / redundant configuration, using tilted thrusters only (first row), and using -Z thrusters and tilted thrusters #3, #4, #5, #6, during burn manoeuvres (second row) :
nom/red_max_neg/pos_trq; i.e. four 2x3 matrices
- the positive / negative minimum torque impulses about each spacecraft axis using tilted thrusters, for the nominal / redundant configuration, using the thrusters cold pulse efficiency on-board model (first row), and the thrusters hot pulse efficiency on-board model (second row) :
nom/red_min_neg/pos_imp, i.e. four 2x3 matrices
- the force / torque thrusters participation matrices, for the nominal / redundant configuration :
nom/red_thr_frc/trq_conf_mat ; 3x12 matrices
- the thrusters ratios to be used to create pure axial torques (-X,-Y,-Z,+X,+Y,+Z in that order) with 8 tilted thrusters, for the nominal / redundant configuration :
nom/red_act_table ; i.e. two 8x6 matrices
- the off-modulation thruster ratio to be used during Burn manoeuvres to create transverse control torques (-X, -Y, +X, +Y in that order) with -Z thrusters, for the nominal / redundant configuration :
nom/red_burn_act_table_xy ; i.e. two 4x4 vectors
- the thrusters ratios to be used during Burn manoeuvres to create Z-axis control torques, for the nominal / redundant configuration :
nom/red_burn_act_table_z ; i.e. two 8x2 matrices
- the steepest descent vectors for the tilted thrusters on-times optimisation, for the nominal / redundant configuration :
nom/red_optim_1st_vect : i.e. two 1x8 vectors
- the steepest descent matrices for the -Z thrusters on-times optimisation, for the nominal / redundant configuration, each column corresponding to the steepest descent vector of the reduced thrusters configuration when :

nom/red_optim_2nd_mat : 3x4 vectors

- the Z-axis torque deadband for Burn manoeuvres (scalar), which adapts the minimum torque impulse provided about Z with the 8-tilted-thruster configuration to the one provided by the reduced thrusters configuration in Burn, i.e. only tilted thrusters 3/4/5/6 : *Ocm_mgr.BFP_deadband_factor(3)*

9.2.4 Ground processing

The following algorithms are aimed at computing the data listed in the previous Section They are organized as follows :

- A first batch of equations computes the geometrical characteristics of the thrusters configurations, namely the force and torques configuration matrices, in order to prepare for further computations
- A second batch computes the parameters specific to the burn manoeuvres, which must be updated before each Burn manoeuvre, if the predicted spacecraft center of mass has evolved.
- A last batch computes all the other parameters, which must be updated only when the thrusters nominal / redundant configurations are changed (note that the preceding parameters must also be updated in that case).

9.2.4.1 Thrusters configurations geometrical data

The following algorithms apply :

- Compute the direction and position of each thruster in each thrusters configuration :

```

For   i       from 1       to   12

      If    nom_used_thr(i) = A       then
          dir_nom(1:3,i) = thrust_dir(1:3,i)
          pos_nom (1:3,i) = thrust_pos(1:3,i)
          thrust_nom (1:3,i) = thrust(1:3,i)

      Elseif nom_used_thr (i) = B       then
          dir_nom(1:3,i) = thrust_dir(1:3,12+i)
          pos_nom (1:3,i) = thrust_pos (1:3,12+i)
          thrust_nom (1:3,i) = thrust(1:3,12+i)

      Elseif nom_used_thr (i) = 0       then
          dir_nom(1:3,i) = 0
          pos_nom (1:3,i) = 0
          thrust_nom (1:3,i) = 0

      Endif

      If    red_used_thr(i) = A       then
          dir_red(1:3,i) = thrust_dir(1:3,i)
  
```

```
pos_red (1:3,i) = thrust_pos (1:3,i)
thrust_red (1:3,i) = thrust(1:3,i)
```

```
Elseif red_used_thr (i) = B then
dir_red(1:3,i) = thrust_dir(1:3,12+i)
pos_red(i,1:3) = thrust_pos (1:3,12+i)
thrust_red (1:3,i) = thrust(1:3,12+i)
```

```
Elseif red_used_thr (i) = 0 then
dir_red(1:3,i) = 0
pos_red (1:3,i) = 0
thrust_red (1:3,i) = 0
```

Endif

Endfor

- Compute the force / torque configuration matrices in each thrusters configuration :

```
For config from ["nom", "red"] -- For each thruster configuration, the corresponding
inputs shall be taken into account.
```

```
For i from 1 to 12
A_config (1:3,i)= dir_config (1:3,i) * thrust_config(i) ---force components
A_config (4:6,i)= ( pos_config (1:3,i) – com_pos(1:3) ) ⊗ ( dir_config (1:3,i) * thrust_config(i) )
--- torque components
```

Endfor

where ⊗ stands for the vectorial product

Endfor

9.2.4.2 RCS_ORB parameters required for Burn manoeuvres

The following algorithms apply to update the parameters necessary for the Burn manoeuvre (i.e. *nom/red_burn_act_table_xy*, *nom/red_optim_2nd_mat*, *nom/red_max_neg/pos_trq*), if the spacecraft center of mass has moved since the last Burn manoeuvre. Note that the following algorithms are valid only if *config_used_thr(i) ≠ 0* for each i=9 to 12, i.e. each of the four –Z thrusters is allocated to a real thruster.

- Compute the thrusters influence torque matrices when considering off-times of the four –Z thrusters used in off-modulation during Burn (only X and Y-axis torques are generated) :

```
M = [ 0 1 1 1 ;
      1 0 1 1 ;
      1 1 0 1 ;
```

1 1 1 0]

For config **from** ["nom", "red"]

For i **from** 1 **to** 4

Aburn_config (1:2,i) = A_config (4:5,9:12) * M(1:4,i)

Endfor

(Aburn_nom and Aburn_red are 2x4 matrices)

Endfor

- Compute the optimal actuation table for the off-times of the four –Z thrusters used in off-modulation during Burn (generation of negative / positive torques about X and Y axes) :

M = [-1 1 0 0 ;
 0 0 -1 1]

f = [1 1 1 1]

LB = [0 0 0 0]

For config **from** ["nom", "red"]

For i **from** 1 **to** 4

X(1:4,i) = Simplex (f, Aburn_config, M(1:2,i), LB)

X(1:4,i) = X(1:4,i) / max(X(1:4,i))

Endfor

config_burn_act_table_xy = X

Endfor

where Simplex (f, A, b, LB) is a simplex optimization algorithm (such as the "linprog" Matlab function) which finds a solution X for the following linear programming problem :

Minimize f*X subject to A*X=b and X≥LB (lower bounds)

- Compute the maximum positive / negative torque levels about X/Y axes corresponding to the off-times actuation tables of the four –Z thrusters used in off-modulation during Burn (generation of the first two elements of the second row of matrices *nom/red_max_neg/pos_trq*) :

For config **from** ["nom", "red"]

M(1:2,1:4) = Aburn_config * *config_burn_act_table_xy*

config_max_neg_trq (2,1) = M(1,1)

config_max_neg_trq (2,2) = M(2,3)

config_max_pos_trq (2,1) = M(1,2)

$config_max_pos_trq(2,2) = M(2,4)$

Endfor

Meanwhile other coefficients of $config_max_neg/pos_trq$ remain unchanged as long as the thrusters configurations have not been changed by the ground (see their computation in Section 9.2.4.3)

- Compute the steepest descent vectors corresponding to the four $-Z$ thrusters used in off-modulation during Burn :

$$\begin{array}{cccc}
 M1 = [& 0 & 0 & 0 ; \\
 & 1 & 0 & 0 ; \\
 & 0 & 1 & 0 ; \\
 & 0 & 0 & 1] &
 M2 = [& 1 & 0 & 0 ; \\
 & 0 & 0 & 0 ; \\
 & 0 & 1 & 0 ; \\
 & 0 & 0 & 1] &
 M3 = [& 1 & 0 & 0 ; \\
 & 0 & 1 & 0 ; \\
 & 0 & 0 & 0 ; \\
 & 0 & 0 & 1] &
 M4 = [& 1 & 0 & 0 ; \\
 & 0 & 1 & 0 ; \\
 & 0 & 0 & 1 ; \\
 & 0 & 0 & 0]
 \end{array}$$

For config **from** ["nom", "red"]

For i **from** 1 **to** 4
 Aburn_reduced_conf(1:2,1:3) = Aburn_config * Mi
 NU = null (Aburn_reduced_conf)
 $config_optim_2^{nd}_mat(1:3,i) = NU * NU^T * [1 ; 1 ; 1]$

Endfor

Endfor

Where null(A) is a function which outputs an orthogonal basis for the null space of matrix A (e.g. "null" function in Matlab).

9.2.4.3 RCS_ORB parameters required at each thruster configuration change

The following algorithms apply to update the RCS_ORB parameters listed in Section 9.2.3 when the thrusters configuration is modified :

- Determine the number of tilted thrusters to be processed for attitude control :

For config **from** ["nom", "red"]

$config_numb_thr = 8$
For i **from** 1 **to** 8
If config_used_thr(i) = 0 **then**
 $config_numb_thr = config_numb_thr - 1$

Endif

Endfor

Endfor

- Compute the thrusters configurations force and torque matrices :

For config **from** ["nom", "red"]

```
config_thr_frc_conf_mat = A_config (1:3,1:12)
config_thr_trq_conf_mat = A_config (4:6,1:12)
```

Endfor

- Compute the steepest descent vectors used for 3-axis control using the tilted thrusters (1 to 8) :

For config **from** ["nom", "red"]

```
NU = null ( A_config (1:6,1:8) )
descent_vect(1:8) = NU * NUT * [ 1 ; 1 ; 1 ; 1 ; 1 ; 1 ; 1 ; 1 ]
```

j = 1

For i **from** 1 **to** 8

If config_used_thr(i) ≠ 0 **then** -- move the empty thrusters to the end

config_optim_1st_vect(j) = descent_vect(i)

j = j + 1

Endif

Endfor

config_optim_1st_vect(j:8) = 0

Endfor

The need to move the empty thrusters to the end of the steepest descent vector stems from the on-board SW implementation which handles an optimization routine processing only the first non zero elements of the thrusters ratio vector.

- Compute the optimal actuation tables for 3-axis attitude control using the tilted thrusters :

```
M = [ 0 0 0 0 0 0 ;
      0 0 0 0 0 0 ;
      0 0 0 0 0 0 ;
      -1 0 0 1 0 0 ;
      0 -1 0 0 1 0 ;
      0 0 -1 0 0 1 ]
```

```
f = [ 1 1 1 1 1 1 1 1 ]
LB = [ 0 0 0 0 0 0 0 0 ]
```

```
For config from ["nom", "red"]
```

```
  For i from 1 to 6
    config_temp_table (1:8,i) = Simplex ( f, A_config, M(1:6,i), LB )
```

```
  If the simplex algorithm failed to find a feasible solution (this may happen when
  thrusters are missing), then re-iterate the algorithm with only torque constraints, i.e.
```

```
    config_temp_table (1:8,i) = Simplex ( f, A_config(4:6,1:8), M(4:6,i), LB )
```

```
  Endif
```

```
  config_temp_table (1:8,i) = config_temp_table (1:8,i) / max( config_temp_table (1:8,i) )
```

```
Endfor
```

```
j = 1
```

```
For i from 1 to 8
```

```
  If config_used_thr(i) ≠ 0 then -- move the empty thrusters to the end
```

```
    config_act_table(j,1:6) = config_temp_table (j,1:6)
```

```
    j = j + 1
```

```
  Endif
```

```
Endfor
```

```
config_act_table(j:8,1:6) = 0
```

```
Endfor
```

The need to move the empty thrusters to the end of the actuation tables stems from the on-board SW implementation which handles an optimization routine processing only the first non zero elements of the thrusters ratio vector.

- Compute the maximum positive / negative torque levels about X/Y/Z axes using tilted thrusters :

```
For config from ["nom", "red"]
```

```
  M(1:3,1:6) = A_config (4:6,1:8) * config_temp_table
```

```
  config_max_neg_trq (1,1) = M(1,1)
```

```
  config_max_neg_trq (1,2) = M(2,2)
```

```
  config_max_neg_trq (1,3) = M(3,3)
```

```
  config_max_pos_trq (1,1) = M(1,4)
```

```
  config_max_pos_trq (1,2) = M(2,5)
```

```
  config_max_pos_trq (1,3) = M(3,6)
```

Endfor

- Compute the optimal actuation tables for Z axis attitude control using the tilted thrusters during burn manoeuvres :

If all thrusters 3/4/5/6 are available (i.e. $config_used_thr(i) \neq 0$ for each $i = 3$ to 6), then they are selected such as to generate Z axis torques with a residual thrust along Z (transverse forces remain null but transverse torques might be non zero because of thrusters mismatches). Else an analysis at system level must be performed to decide whether it is preferable to use a subset of thrusters 3/4/5/6, therefore creating parasitic transverse ΔV s, or to use thrusters 1/2/7/8 instead, therefore degrading the fuel consumption, because those thrusters generate a residual force opposite to the burn thrust.

```
M = [ 0 0 ;
      0 0 ;
      -1 1 ]
f = [ 1 1 1 1 ]
LB = [ 0 0 0 0 ]
```

```
For config from ["nom", "red"]
```

```
For i from 1 to 2 (this is assuming that thrusters 3/4/5/6 are available)
  config_temp_table (3:6,i) = Simplex ( f, A_config([1:2 6],3:6), M(1:3,i), LB )
  ---- optimization with respect to X/Y-axes torques and Z-axis force only
  config_temp_table (3:6,i) = config_temp_table (3:6,i) / max( config_temp_table (3:6,i) )
  config_temp_table ([1:2 7:8],i) = 0
```

```
Endfor
```

```
j = 1
```

```
For i from 1 to 8
  If config_used_thr(i)  $\neq$  0 then -- move the empty thrusters to the end
    config_burn_act_table_z(j,1:2) = config_temp_table (j,1:2)
    j = j + 1
```

```
Endif
```

```
Endfor
```

```
config_burn_act_table_z (j:8,1:2) = 0
```

```
Endfor
```

- Compute the maximum positive / negative torque levels about Z axis using tilted thrusters during Burn manoeuvres :

```
For config from ["nom", "red"]
```

```
M(1,1:2) = A_config (6,1:8) * config_temp_table
```

config_max_neg_trq (2,3) = M(1,1)
config_max_pos_trq (2,3) = M(1,2)

Endfor

- Compute the minimum positive / negative torque impulses about each spacecraft axis using tilted thrusters, with the thrusters cold and hot pulses efficiency on-board models :

First compute the inverted thruster efficiency model, i.e. search t_{on} such that $\min_{on_time} = t_{on} \cdot \frac{t_{on} + \alpha}{t_{on} + \beta}$ (note that this can be done once for all as it does not depend on the thruster configurations).

For model **from** ["cold", "hot"]

Delta = min_on_time – alpha_model
 $t_{on_model} = (\text{Delta} + \sqrt{ \text{Delta}^2 + 4 * \text{beta_model} * \text{min_on_time} }) / 2$

Endfor

Then compute the minimum negative and positive torque impulses for each model and each thruster configuration from maximum torque capacities with tilted thrusters (for Burn manoeuvres with thrusters 9 to 12, there is no minimum impulse bit considered as the thrusters are used in off-modulation).

For config **from** ["nom", "red"]

config_min_neg_imp (1,1:3) = *config_max_neg_trq* (1,1:3) * *t_on_cold*
config_min_pos_imp (1,1:3) = *config_max_pos_trq* (1,1:3) * *t_on_cold*
config_min_neg_imp (2,1:3) = *config_max_neg_trq* (1,1:3) * *t_on_hot*
config_min_pos_imp (2,1:3) = *config_max_pos_trq* (1,1:3) * *t_on_hot*

Endfor

- Compute the Z axis torque deadband for Burn manoeuvres (to be updated before the next burn manoeuvre) :

Ocm_mgr.BFP_deadband_factor (3) = min (nom_max_neg_trq (2,3) / nom_max_neg_trq (1,3) ,
 nom_max_pos_trq (2,3) / nom_max_pos_trq (1,3))

10 ΔV MANOEUVRES PARAMETERS

10.1 CONTEXT AND PURPOSE

This chapter deals with the ground processings required for preparing the parameters used to achieve Burn and Vectored Thrust manoeuvres in the Orbit Control Mode, or in Normal Mode WDP / SKM Earth Strobing phases. The corresponding procedures described in [7] are :

- FCP-AC0032 : NM/FPAP mode transition to Orbit Control Mode
- FCP-AC0640 : Thrusting in NM-WDP and SKM-EAH
- FCP-AC0300 : Axial Delta-V manoeuvre
- FCP-AC0310 : Vectored Thrust Delta-V manoeuvre

There are three types of ground computations involved in these four procedures :

- Computation of the disturbing torques created by the $-Z$ thrusters in Burn manoeuvres. This concerns FCP-AC0300 and aims at enabling an on-board feed-forward compensation of thrusters disturbing torques. This ground processing is addressed in Section 10.2.
- Prediction of optimized thrust directions in spacecraft axes with respect to fuel consumption for Vectored Thrust manoeuvres : this concerns FCP-AC0032 and FCP-AC0640 and consists in finding the spacecraft axes which present the maximum thrusting efficiency, i.e. which maximize the ratio of produced longitudinal force over sum of thruster delivered thrusts. These directions are recommended for selection at system level but this is not mandatory. This ground processing is addressed in Section 10.3.
- Computation of the thrusters ratio vector which will enable to implement the thrust direction desired by the ground (for example the one computed at the step before) : this concerns FCP-AC0310 and FCP-AC0640 and consists in finding the thrusters ratios such that the deterministic disturbing torque generated by the thrusters is zero, while maximizing the longitudinal force in the desired direction. This ground processing is addressed in Section 10.4.

10.2 PERTURBATION TORQUES INDUCED BY AN AXIAL ΔV MANOEUVRE

10.2.1 Principle

Before each burn manoeuvre, the 3-axis disturbing torques generated by opening all $-Z$ thrusters must be predicted in order to uplink the corresponding on-board values which is used as a feed-forward term in the control loop. These values depend on the thruster configuration and on the spacecraft center of mass position with respect to the spacecraft geometrical frame. The disturbing torques must be uplinked with the same sign in order to be properly compensated on-board.

10.2.2 Inputs

The following system inputs are necessary :

- *Nom/red_used_thr* : 1x12 vectors of 0, A, B elements defining the physical selection of the twelve thrusters for the nominal and redundant thrusters configurations
- *config* : the RCS_ORB configuration (nominal or redundant) currently in use
- *Com_pos* : 1x3 position vector of the spacecraft center of mass with respect to the spacecraft geometrical reference frame.
- *Blow_down_factor* : blow-down factor reflecting the tank pressure (*blow_down_factor* = 1 corresponds to thruster force levels of 10N)

The following system parameters are necessary :

- *Thrust* : 1x24 vector containing the default nominal thrust level of each thruster (A branch first, then B branch) with pressurized tanks, assumed 10N. This data corresponds to a blow-down factor (*bdf*) of 1, this parameter being regularly up-linked to correct the thrusters parameters with the adequate thrust level depending on the tank pressure.
- *Thrust_pos* : 3x24 array containing the position vectors of the 24 thrusters (A branch first, then B branch) with respect to the spacecraft geometrical reference frame
- *Thrust_dir* : 3x24 array containing the direction cosine vectors of the 24 thrusters (A branch first, then B branch) with respect to the spacecraft geometrical reference frame

10.2.3 Outputs

The output from the ground processing is a 3-element vector *disturb_trq* corresponding to the predicted -Z thrusters disturbing torques on each spacecraft axis.

10.2.4 Ground processings

- Computation of the thrusters geometrical data : the same algorithms as in Section 9.2.4.1 shall be performed, with as output the (6x12) matrix *A_config* which contains the torque and force signatures of each thruster, accounting for the spacecraft center of mass current position.
- Computation of the disturbing torques when thrusters 9 to 12 are fired :

$$Disturb_trq(1:3) = A_config(4:6,:) * [0;0;0;0;0;0;0;0;1;1;1;1]$$

- Multiplication by the blow-down factor :

$$Disturb_trq(1:3) = Disturb_trq(1:3) * blow_down_factor$$

10.3 OPTIMAL THRUST DIRECTIONS FOR VECTORED ΔV MANOEUVRES

10.3.1 Principle

This ground processing aims at finding the optimal thrust directions in spacecraft axes, with respect to fuel consumption, to perform a Vectored Thrust Manoeuvre in OCM, in NM-WDP or in SKM-EAH.

10.3.2 Inputs

The following system inputs are necessary :

- *Nom/red_used_thr* : 1x12 vectors of 0, A, B elements defining the physical selection of the twelve thrusters for the nominal and redundant thrusters configurations
- *config* : the RCS_ORB configuration (nominal or redundant) currently in use
- *are_Z_thrusters_used* : flag indicating whether the $-Z$ thrusters (normally used for Burn) shall be used for computing the optimal thrust directions (0 means “no”, 1 means “yes”) : this might concern the small ΔV s in NM-WDP and SKM-EAH only, where the ground has the possibility to fire all 12 thrusters to achieve the manoeuvre. In OCM vectored thrust manoeuvres, only the 8 tilted thrusters can be used.
- *min_frc_ratio* : force ratio nominally equal to 1 in the case of a thrust manoeuvre in NM-WDP or SKM-EAH, and nominally equal to 0.05 in the case of a Vectored Thrust in OCM (initial thrust ratio of the thrust modulation ramp)
- *Com_pos* : 1x3 position vector of the spacecraft center of mass with respect to the spacecraft geometrical reference frame.

The following system parameters are necessary :

- *Thrust* : 1x24 vector containing the default nominal thrust level of each thruster (A branch first, then B branch) with pressurized tanks, assumed 10N. This data corresponds to a blow-down factor (*bdf*) of 1, this parameter being regularly up-linked to correct the thrusters parameters with the adequate thrust level depending on the tank pressure.
- *Thrust_pos* : 3x24 array containing the position vectors of the 24 thrusters (A branch first, then B branch) with respect to the spacecraft geometrical reference frame
- *Thrust_dir* : 3x24 array containing the direction cosine vectors of the 24 thrusters (A branch first, then B branch) with respect to the spacecraft geometrical reference frame

The following AOCMS SW data are necessary :

- *Min_on_time* : the thruster minimum on-time defined on-board to manage the thrusters Minimum Impulse Bit

- *Alpha_cold, beta_cold* : on-board coefficients of the thrusters efficiency cold model (the only one used in OCM, NM-WDP and SKM-EAH).
- *RCS_ORB.ctrl_period* (TM) : thruster actuation frequency in the mode where the Vectored Thrust Manoeuvre shall be performed. This parameter is equal to 1.5s in OCM Vectored Thrust and 1s in SKM-EAH. In NM-WDP, it might be equal to 1s or 0.125s, depending on the preceding thruster controlled mode, or to a multiple of 0.125s potentially patched by the ground.

10.3.3 Outputs

The output from the ground processing is an efficiency matrix representing the efficiency in terms of fuel consumption for a set of thrust directions in spacecraft axes defined through azimuth / elevation angles. This efficiency matrix then allows to find optimal thrust directions in spacecraft axes. The number of retained optimal directions (i.e. local optimal ones) depends on the system strategy for vectored thrust manoeuvres, i.e. :

- Is it acceptable to perform a large angle slew with reaction wheels before the thrust in order to align the desired inertial ΔV direction with the optimal thrust direction in spacecraft axes (in that case only the very best optimal direction is needed) ?
- Is it acceptable to perform small angle slews with reaction wheels before the thrust in order to align the desired inertial ΔV direction with the local optimal thrust direction in spacecraft axes (in that case all locally optimal directions are needed) ?
- Otherwise local optimal directions could still be used by decomposing the desired thrust direction into several small manoeuvres in local optimal directions, but it shall then be verified (through the same algorithm as described below) whether a direct thrust would be less or more efficient.

10.3.4 Ground processings

- Computation of the thrusters geometrical data : the same algorithms as in Section 9.2.4.1 shall be performed, with as output the (6x12) matrix *A_config* which contains the torque and force signatures of each thruster.
- Computation of the minimum **commandable** on-time ratio *min_on_ratio*, in order to specify lower bounds on the thruster ratios which are achievable by the thruster modulator function implemented in the on-board software : compute the inverted thruster efficiency cold model, i.e. search *min_on_ratio* such that

$$\min_frc_ratio * ctrl_period * \min_on_ratio - \frac{\min_frc_ratio * ctrl_period * \min_on_ratio + \alpha}{\min_frc_ratio * ctrl_period * \min_on_ratio + \beta} = \min_on_time$$

$$\Delta = \min_on_time - \alpha_cold$$

$$\min_on_ratio = (\Delta + \sqrt{ \Delta^2 + 4 * \beta_cold * \min_on_time }) / 2$$

$$min_on_ratio = min_on_ratio / (min_frc_ratio * ctrl_period)$$

- Computation of the thrust efficiencies in all spacecraft axes defined through azimuth / elevation angle in steps of 1 deg :

f = [1 1 1 1 1 1 1 1 1 1 1 1] --- 1x12 vector

LB = [0 0 0 0 0 0 0 0 0 0 0 0] --- 1x12 vector

If are_Z_thrusters_used = 0 **then** --- only thrusters 1 to 8 are considered

A_config = A_config (1:6,1:8)

f = f(1:8)

LB = LB(1:8)

Endif

For I (azimuth) **from** 1 **to** 360 in steps of 1°

For J (elevation) **from** -89 **to** 90 in steps of 1°

Fx = cos(I*π/180)*cos(J*π/180)

Fy = sin(I*π/180)*cos(J*π/180)

Fz = sin(J*π/180)

tau = Simplex (f, A_config, [Fx;Fy;Fz;0;0;0], LB) --- tau is a 1x12 vector

If tau (1:12) / max(tau(1:12)) ≠ 0 **and** tau (1:12) / max(tau(1:12)) < min_on_ratio **then**
 eff (I,J+90) = 0 -- the corresponding direction can not be achieved with the
 computed thrusters ratio vector

Else

eff (I,J+90) = 1 / (f * tau)

Endif

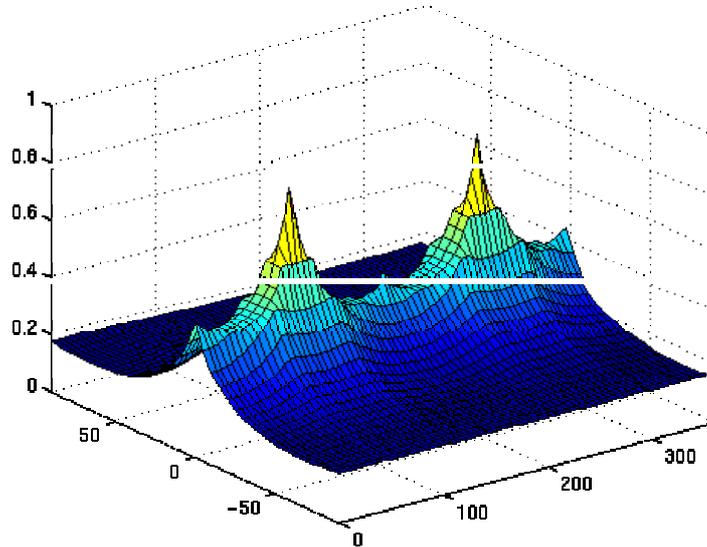
Endfor

Endfor

where Simplex (f, A, b, LB) is a simplex optimization algorithm (such as the "linprog" Matlab function) which finds a solution X for the following linear programming problem :

Minimize f*X subject to A*X=b and X≥LB (lower bounds)

- Computation of the optimal thrust directions in spacecraft axes : the efficiency matrix *eff* found at the step before provides the thrust efficiency as a function of the azimuth and elevation angles in the spacecraft reference frame. An example of results is shown on the graph below. From this it is possible to extract elevation and azimuth angles of roughly local optimum directions (e.g. near +/- X, near +/-Y), and then to run a dichotomy or a locally more precise version (with a better resolution) of the previous algorithm in order to refine the optimal directions.



10.4 THRUST RATIO VECTOR FOR VECTORED ΔV MANOEUVRES

10.4.1 Principle

This ground processing aims at computing the thruster ratios for achieving a thrust in a specified direction defined with respect to the spacecraft reference frame (e.g. one of the optimal directions computed in Section 10.3, or directly the thrust direction desired at navigation level, accounting for the current spacecraft inertial attitude).

10.4.2 Inputs

The following system inputs are necessary :

- *Nom/red_used_thr* : 1x12 vectors of 0, A, B elements defining the physical selection of the twelve thrusters for the nominal and redundant thrusters configurations
- *are_Z_thrusters_used* : flag indicating whether the $-Z$ thrusters (normally used for Burn) shall be used for computing the thruster ratios : this might concern the small ΔV s in NM-WDP and SKM-EAH only, where the ground has the possibility to fire all 12 thrusters to achieve the manoeuvre. In OCM vectored thrust manoeuvres, only the 8 tilted thrusters can be used.
- *min_frc_ratio* : force ratio nominally equal to 1 in the case of a thrust manoeuvre in NM-WDP or SKM-EAH, and nominally equal to 0.05 in the case of a Vectored Thrust in OCM (initial thrust ratio of the thrust modulation ramp)

- *Com_pos* : 1x3 position vector of the spacecraft center of mass with respect to the spacecraft geometrical reference frame.
- *Thrust_vect* : 1x3 thrust desired direction vector in spacecraft axes

The following system parameters are necessary :

- *Thrust* : 1x24 vector containing the default nominal thrust level of each thruster (A branch first, then B branch) with pressurized tanks, assumed 10N. This data corresponds to a blow-down factor (*bdf*) of 1, this parameter being regularly up-linked to correct the thrusters parameters with the adequate thrust level depending on the tank pressure.
- *Thrust_pos* : 3x24 array containing the position vectors of the 24 thrusters (A branch first, then B branch) with respect to the spacecraft geometrical reference frame
- *Thrust_dir* : 3x24 array containing the direction cosine vectors of the 24 thrusters (A branch first, then B branch) with respect to the spacecraft geometrical reference frame

The following AOCMS SW data are necessary :

- *Min_on_time* : the thruster minimum on-time defined on-board to manage the thrusters Minimum Impulse Bit
- *Alpha_cold*, *beta_cold* : on-board coefficients of the thrusters efficiency cold model (the only one used in OCM, NM-WDP and SKM-EAH).
- *RCS_ORB.ctrl_period* (TM) : thruster actuation frequency in the mode where the Vectored Thrust Manoeuvre shall be performed. This parameter is equal to 1.5s in OCM Vectored Thrust and 1s in SKM-EAH. In NM-WDP, it might be equal to 1s or 0.125s, depending on the preceding thruster controlled mode, or to a multiple of 0.125s potentially patched by the ground.

10.4.3 Outputs

The outputs from the ground processing are the (1x12) thrust ratio vectors *nom/red_frc_thr_ratio* which must be sent by TC to the AOCMS SW (RCS_ORB object) before the manoeuvre.

10.4.4 Ground processings

- Computation of the thrusters geometrical data : the same algorithms as in Section 9.2.4.1 shall be performed, with as outputs the (6x12) matrices *A_config* which contain the torque and force signatures of each thruster, for each nominal / redundant configuration.
- Computation of the minimum **commandable** on-time ratio *min_on_ratio*, in order to specify lower bounds on the thruster ratios which are achievable by the thruster modulator function implemented in the on-board software : compute the inverted thruster efficiency cold model, i.e. search *min_on_ratio* such that

$$\min_frc_ratio * ctrl_period * \min_on_ratio \cdot \frac{\min_frc_ratio * ctrl_period * \min_on_ratio + \alpha}{\min_frc_ratio * ctrl_period * \min_on_ratio + \beta} = \min_on_time$$

Delta = min_on_time – alpha_cold
 $\min_on_ratio = (\text{Delta} + \text{sqrt} (\text{Delta}^2 + 4 * \beta_{\text{cold}} * \min_on_time)) / 2$
 $\min_on_ratio = \min_on_ratio / (\min_frc_ratio * ctrl_period)$

- Computation of the thrusters ratio vectors :

f = [1 1 1 1 1 1 1 1 1 1 1 1] --- 1x12 vector
 LB = [0 0 0 0 0 0 0 0 0 0 0 0] --- 1x12 vector

For config **from** ["nom", "red"]

If are_Z_thrusters_used = 0 **then**
 A_config = A_config (1:6,1:8)
 f = f(1:8)
 LB = LB(1:8)

Endif

config_frc_thr_ratio = Simplex (f, A_config, [Thrust_vect;0;0;0], LB)

config_frc_thr_ratio = *config_frc_thr_ratio* / max (*config_frc_thr_ratio* (:))

If *config_frc_thr_ratio*(1:12) ≠ 0 **and** *config_frc_thr_ratio*(1:12) < min_on_ratio **then**

Warning : the specified thrust direction is not achievable.

In this case the *config_frc_thr_ratio* element can be forced to 0 or to min_on_ratio (whatever the closest), and the corresponding thrust direction can be computed through A_config*min_on_ratio, but the residual torques would not be exactly zero.

Endif

Endfor

11 ACCELEROMETERS SIZE EFFECTS COMPENSATION

11.1 CONTEXT AND PURPOSE

Accelerometers are used in OCM to provide an on-board measurement of the ΔV manoeuvre amplitude, allowing to stop it exactly when the ground commanded amplitude has been reached. Three accelerometers are mounted in each Inertial Measurement Package which contains also three gyros. Because the accelerometers are not located at the spacecraft center of mass, they sense not only a linear acceleration but also a centrifugal and angular acceleration when the spacecraft body experiences a rotation, which depends on the distance to the spacecraft center of mass. This “size effect” must be removed from the accelerometers measurements in order to isolate the measured linear acceleration, which is used to assess the end of the OCM. The accelerometer size effect is computed on-board the AOCMS SW by using three matrices of constant parameters which must be up-linked each time the spacecraft body center of mass varies significantly, i.e. after large ΔV manoeuvres depleting a lot of fuel.

11.2 PROCESSING

11.2.1 Principle

The acceleration measured by each accelerometer is the sum of :

- the linear acceleration of the spacecraft ;
- the centrifugal accelerations due to the position of each accelerometer away from the spacecraft CoG position. These accelerations are due to the spacecraft angular acceleration and angular velocity.

The following equation gives the acceleration measured along any accelerometer axis :

$$\text{Acceleration}_i^{\text{measured}} = \vec{A}_i \left[\vec{\gamma}_l + \vec{\Omega} \wedge \overrightarrow{\text{Pos}_{SC-Acc_i}} + \dot{\vec{\Omega}} \wedge \left(\vec{\Omega} \wedge \overrightarrow{\text{Pos}_{SC-Acc_i}} \right) \right] + \text{errors}$$

Where : \vec{A}_i is the accelerometer sensitive axis

$\vec{\gamma}_l$ is the linear non gravitational acceleration at the spacecraft CoG

$\vec{\Omega}$ is the angular rate of the spacecraft

$\dot{\vec{\Omega}}$ is the angular acceleration of the spacecraft

$\overrightarrow{\text{Pos}_{SC-Acc_i}}$ is the accelerometer i position vector with respect to the S/C center of mass



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
 Issue : **3** Rev. : 0
 Date : **05/08/2003**
 Page : 81

The size effect correction consists in subtracting the centrifugal and angular accelerations of the spacecraft to the velocity measured by the IMP at each time-step by the on-board software. The velocity correction to be subtracted to the velocity measured by the IMP and expressed in the IMP reference frame (as in the AOCMS on-board SW, inside the IMP_SWR object) is:

$$\begin{pmatrix} \Delta V_x^{IMP} \\ \Delta V_y^{IMP} \\ \Delta V_z^{IMP} \end{pmatrix} = [AAC] \begin{pmatrix} \Delta \omega_x^{IMP} \\ \Delta \omega_y^{IMP} \\ \Delta \omega_z^{IMP} \end{pmatrix} + \Delta T \cdot [RSC] \begin{pmatrix} (\omega_x^{IMP})^2 \\ (\omega_y^{IMP})^2 \\ (\omega_z^{IMP})^2 \end{pmatrix} + \Delta T \cdot [RPC] \begin{pmatrix} \omega_x^{IMP} \omega_z^{IMP} \\ \omega_x^{IMP} \omega_y^{IMP} \\ \omega_y^{IMP} \omega_z^{IMP} \end{pmatrix}$$

where ΔT is the on-board software sampling period (0.125 sec)

$[AAC]$ is the Angular Acceleration Coefficients (3x3) matrix

$[RSC]$ is the Rate Square Coefficients (3x3) matrix

$[RPC]$ is the Rate Product Coefficients (3x3) matrix

$\omega_{x,y,z}^{IMP}$ is the angular velocity vector measured by the IMP in the IMP reference frame

$\Delta \omega_{x,y,z}^{IMP} = \omega_{x,y,z}^{IMP} - \omega_{x,y,z}^{IMP \text{ prior}}$ where $\omega_{x,y,z}^{IMP \text{ prior}}$ is the angular velocity measured at the

prior cycle ($\Delta \omega_{x,y,z}^{IMP}$ is proportional to the spacecraft angular acceleration)

The detailed general expressions of $[AAC]$, $[RSC]$ and $[RPC]$ matrices are :

$$[AAC] = \begin{bmatrix} A_{1z}^{IMP} * POSSC - Acc1_y^{IMP} & -A_{1y}^{IMP} * POSSC - Acc1_z^{IMP} & A_{1x}^{IMP} * POSSC - Acc1_z^{IMP} & -A_{1z}^{IMP} * POSSC - Acc1_x^{IMP} & A_{1y}^{IMP} * POSSC - Acc1_x^{IMP} & -A_{1x}^{IMP} * POSSC - Acc1_y^{IMP} \\ A_{2z}^{IMP} * POSSC - Acc2_y^{IMP} & -A_{2y}^{IMP} * POSSC - Acc2_z^{IMP} & A_{2x}^{IMP} * POSSC - Acc2_z^{IMP} & -A_{2z}^{IMP} * POSSC - Acc2_x^{IMP} & A_{2y}^{IMP} * POSSC - Acc2_x^{IMP} & -A_{2x}^{IMP} * POSSC - Acc2_y^{IMP} \\ A_{3z}^{IMP} * POSSC - Acc3_y^{IMP} & -A_{3y}^{IMP} * POSSC - Acc3_z^{IMP} & A_{3x}^{IMP} * POSSC - Acc3_z^{IMP} & -A_{3z}^{IMP} * POSSC - Acc3_x^{IMP} & A_{3y}^{IMP} * POSSC - Acc3_x^{IMP} & -A_{3x}^{IMP} * POSSC - Acc3_y^{IMP} \end{bmatrix}$$

$$[RSC] = \begin{bmatrix} A_{1y}^{IMP} * POSSC - Acc1_z^{IMP} & +A_{1z}^{IMP} * POSSC - Acc1_y^{IMP} & A_{1x}^{IMP} * POSSC - Acc1_z^{IMP} & +A_{1z}^{IMP} * POSSC - Acc1_x^{IMP} & A_{1y}^{IMP} * POSSC - Acc1_x^{IMP} & +A_{1x}^{IMP} * POSSC - Acc1_y^{IMP} \\ A_{2y}^{IMP} * POSSC - Acc2_z^{IMP} & +A_{2z}^{IMP} * POSSC - Acc2_y^{IMP} & A_{2x}^{IMP} * POSSC - Acc2_z^{IMP} & +A_{2z}^{IMP} * POSSC - Acc2_x^{IMP} & A_{2y}^{IMP} * POSSC - Acc2_x^{IMP} & +A_{2x}^{IMP} * POSSC - Acc2_y^{IMP} \\ A_{3y}^{IMP} * POSSC - Acc3_z^{IMP} & +A_{3z}^{IMP} * POSSC - Acc3_y^{IMP} & A_{3x}^{IMP} * POSSC - Acc3_z^{IMP} & +A_{3z}^{IMP} * POSSC - Acc3_x^{IMP} & A_{3y}^{IMP} * POSSC - Acc3_x^{IMP} & +A_{3x}^{IMP} * POSSC - Acc3_y^{IMP} \end{bmatrix}$$

$$[RPC] = \begin{bmatrix} A_{1y}^{IMP} * POSSC - Acc1_z^{IMP} & +A_{1z}^{IMP} * POSSC - Acc1_y^{IMP} & A_{1x}^{IMP} * POSSC - Acc1_z^{IMP} & +A_{1z}^{IMP} * POSSC - Acc1_x^{IMP} & A_{1y}^{IMP} * POSSC - Acc1_x^{IMP} & +A_{1x}^{IMP} * POSSC - Acc1_y^{IMP} \\ A_{2y}^{IMP} * POSSC - Acc2_z^{IMP} & +A_{2z}^{IMP} * POSSC - Acc2_y^{IMP} & A_{2x}^{IMP} * POSSC - Acc2_z^{IMP} & +A_{2z}^{IMP} * POSSC - Acc2_x^{IMP} & A_{2y}^{IMP} * POSSC - Acc2_x^{IMP} & +A_{2x}^{IMP} * POSSC - Acc2_y^{IMP} \\ A_{3y}^{IMP} * POSSC - Acc3_z^{IMP} & +A_{3z}^{IMP} * POSSC - Acc3_y^{IMP} & A_{3x}^{IMP} * POSSC - Acc3_z^{IMP} & +A_{3z}^{IMP} * POSSC - Acc3_x^{IMP} & A_{3y}^{IMP} * POSSC - Acc3_x^{IMP} & +A_{3x}^{IMP} * POSSC - Acc3_y^{IMP} \end{bmatrix}$$

The above matrices must be computed on-ground and up-linked to the AOCMS SW for each of the three IMPs.

11.2.2 Inputs

The inputs to the ground processing are the following :

- Accelerometers sensitive axes direction vectors in each IMP reference frame : actually the accelerometers are aligned with the IMP reference frame, i.e. the accelerometer axes are :

$$\vec{A}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}_{IMP}, \vec{A}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}_{IMP}, \vec{A}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_{IMP}$$

- Accelerometers position vectors with respect to the spacecraft geometrical reference, expressed in the SC reference frame (\vec{POS}_{Acc}^{SC}) : these have been computed in [4] from the IMP reference position vectors with respect to the spacecraft geometrical reference, and from the accelerometer centers of percussion position vectors with respect to their IMP reference frame. Note that these nominal positions should be corrected at system level by on-ground positioning data, if available.

in [mm]	IMP A			IMP B			IMP C		
	Accel. X	Accel. Y	Accel. Z	Accel. X	Accel. Y	Accel. Z	Accel. X	Accel. Y	Accel. Z
$X_{s/c}$	-542.42	-559.46	-523.86	-807.70	-872.55	-876.15	-749.89	-668.00	-700.00
$Y_{s/c}$	-787.80	-826.33	-848.96	-787.80	-826.33	-848.96	-787.80	-826.33	-848.96
$Z_{s/c}$	466.63	381.91	398.31	380.22	437.34	398.31	653.16	680.77	703.40

- The rotation matrices from the spacecraft to the IMP reference frames are necessary in order to transform the accelerometers position vectors with respect to the spacecraft center of mass from the spacecraft to the IMP reference frame. These rotation matrices are also derived from [4]. Note that these nominal directions should be corrected at system level by on-ground alignment data, if available

$$T_{SC/IMPA} = \begin{bmatrix} \frac{\sqrt{3}}{2\sqrt{2}} & \frac{1}{2} & \frac{\sqrt{3}}{2\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{2\sqrt{2}} & \frac{\sqrt{3}}{2} & \frac{1}{2\sqrt{2}} \end{bmatrix} \quad T_{SC/IMPB} = \begin{bmatrix} \frac{\sqrt{3}}{2\sqrt{2}} & \frac{1}{2} & \frac{\sqrt{3}}{2\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{2\sqrt{2}} & \frac{\sqrt{3}}{2} & \frac{1}{2\sqrt{2}} \end{bmatrix} \quad T_{SC/IMPC} = \begin{bmatrix} 0 & -1 & 0 \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \end{bmatrix}$$

- Finally the spacecraft center of mass position vector with respect to the spacecraft geometrical reference must be provided and expressed in spacecraft axes. This is assumed to be computed at system level from spacecraft mass properties models, according to the current position of appendages and to the current fuel ratio. This data is noted \vec{POS}_{com}^{SC} .

11.2.3 Outputs

The outputs from the ground processing are the three matrices $[AAC]$ (*ang_acc_coef_mat*), $[RSC]$ (*rate_square_coef_mat*), and $[RPC]$ (*rate_product_coef_mat*), for each IMP, which must be sent by TC to the AOCMS SW (IMP_SWR object).

11.2.4 Ground processing

The following computation steps must be performed *for each IMP* :

- o Compute the position vectors of the accelerometers with respect to the current spacecraft center of mass, expressed in the IMP reference frame :

$$\begin{bmatrix} \overrightarrow{POS_{SC_Acc1}}^{\rightarrow IMP} & \overrightarrow{POS_{SC_Acc2}}^{\rightarrow IMP} & \overrightarrow{POS_{SC_Acc3}}^{\rightarrow IMP} \end{bmatrix} = T_{SC/IMP}^t \cdot \begin{bmatrix} \overrightarrow{POS_{Acc1}}^{\rightarrow SC} - \overrightarrow{POS_{com}}^{\rightarrow SC} & \overrightarrow{POS_{Acc2}}^{\rightarrow SC} - \overrightarrow{POS_{com}}^{\rightarrow SC} & \overrightarrow{POS_{Acc3}}^{\rightarrow SC} - \overrightarrow{POS_{com}}^{\rightarrow SC} \end{bmatrix}$$

- o Compute the $[AAC]$, $[RSC]$ and $[RPC]$ matrices, according to the formula given in Section 11.2.1, however with simplifications because of the alignment of accelerometers sensitive axes with the IMP reference frame :

$$[AAC] = \begin{bmatrix} 0 & POSSC-Acc1_z^{IMP} & -POSSC-Acc1_y^{IMP} \\ -POSSC-Acc2_z^{IMP} & 0 & POSSC-Acc2_x^{IMP} \\ POSSC-Acc3_y^{IMP} & -POSSC-Acc3_x^{IMP} & 0 \end{bmatrix}$$

$$[RSC] = \begin{bmatrix} 0 & POSSC-Acc1_x^{IMP} & POSSC-Acc1_x^{IMP} \\ POSSC-Acc2_y^{IMP} & 0 & POSSC-Acc2_y^{IMP} \\ POSSC-Acc3_z^{IMP} & POSSC-Acc3_z^{IMP} & 0 \end{bmatrix}$$

$$[RPC] = \begin{bmatrix} 0 & POSSC-Acc1_z^{IMP} & POSSC-Acc1_y^{IMP} \\ POSSC-Acc2_z^{IMP} & 0 & POSSC-Acc2_x^{IMP} \\ POSSC-Acc3_y^{IMP} & POSSC-Acc3_x^{IMP} & 0 \end{bmatrix}$$

12 STR / CAM MISALIGNMENTS IN-FLIGHT CALIBRATION

12.1 CONTEXT AND PURPOSE

Nominally, the AOCMS uses the STR for attitude estimation. However, AOCMS pointing requirements apply to the navigation camera reference frame. In order to fulfill these requirements, and especially the attitude measurement accuracy requirement, the misalignments between both sensors must be calibrated in-flight.

Even though an on-ground calibration of STR / CAM alignments is performed, there is a need to elaborate an in-flight calibration in order to compensate additional biases such as shocks during launch and thermo-elastic effects during the orbital life.

The in-flight calibration of the STR / CAM misalignments shall be executed prior to every mission phase where stringent attitude estimation performance needs apply at the navigation camera level, i.e. namely before each asteroid fly-by and before starting the comet in-orbit observation. The corresponding procedure (FCP-AC0220) is described in [7], and covers the calibration of STR A as well as STR B with respect to CAM A. It is also assumed here that CAM B is calibrated with respect to CAM A which stands as the reference with respect to other scientific payloads. This CAM A / CAM B calibration is also dealt with in this Chapter.

The algorithms described hereunder specify the on-ground operations to perform the STR/CAM calibration.

12.2 PROCESSING

12.2.1 Principle

The STR/CAM misalignment calibration aims at estimating the 3-axis angular biases between the on-ground calibrated frames of the two optical sensors, from in-orbit stellar measurements.

The calibration method consists in processing stellar measurements from the CAM and the STR acquired on different star patterns in order to filter the random bias errors. This is achieved by slewing the spacecraft body in-between several inertial attitudes where the sensor measurements are stored on-board or directly down-linked (depending on system constraints). The mode supporting these operations is the NM-FPAP which offers fine pointing and slewing capacities. Furthermore several measurements must be acquired on the same star pattern in order to filter the noise errors.

The AOCMS pointing budget has shown that up to 100 different star patterns were necessary in order to achieve sufficient filtering of the STR random bias, whereas noise filtering requires only 10 successive CAM measurements (once every 10s) and 100 STR successive measurements (performed once every 0.5s, but acquired only once every 1s, to simplify the TM configuration). The reason to hold 100 inertial attitudes instead of acquiring STR / CAM measurements during a continuous slew is because the CAM can not track faint stars with an apparent motion (e.g. the maximum tracking rate for stars of magnitude 10 is between 1 and 5arcsec/s), which would require too long manoeuvres. Furthermore holding inertial attitudes simplifies the required processing as there is no need to

synchronize the STR / CAM measurements. More constraints and characteristics of the attitude profile to be used for the STR/CAM calibration process are described in ref. [7].

The CAM/STR calibration method consists in computing stellar innovations from the four optical sensors (two STR and two CAM), using the commanded spacecraft attitude quaternion for predicting the star coordinates in the sensor reference frames. These innovations are then averaged over the measurements acquired during inertial attitudes hold phases, and then combined in a least square estimation process in order to derive the 3-axis angular misalignments between each STR/CAM A sensor pair, and between CAM A and CAM B (i.e. three calibrations in total).

Whereas the STR is autonomous when it tracks stars, predicted stars coordinates and magnitudes must be commanded to the CAM to perform measurements. An on-ground selection allows the CAM to track selected stars in specified windows and send back stars measurements, the CAM being in the "Point Target Tracking Mode". A star catalogue with stars of magnitude up to 11 (which is the CAM magnitude sensitivity) is necessary on ground to provide the CAM with predicted stars positions on its CCD.

The ground processing therefore requires the following TM data :

- Commanded spacecraft body inertial attitude quaternion with respect to J2000 reference frame (this data is available on-ground but could be cross-checked with the TM data)
- STR A, STR B, CAM A and CAM B reference frame orientations in the spacecraft frame (this data is available on-ground but should be cross-checked with the on-board data through a memory dump)
- STR A and STR B Autonomous Acquisition / Tracking Mode TM data packet at 1Hz (for an easier TM configuration) during inertial attitude hold phases, for a duration of 100s, in order to get 100 measurements in a row. The following elements will be processed : tracked stars (up to 9) direction cosines in the J2000 inertial frame, tracked stars measured coordinates on STR transverse axes
- CAM A and CAM B Point Target Tracking Mode TM data packet at 0.1Hz during inertial attitude hold phases, for a duration of 100s, in order to get 10 measurements in a row. The following elements will be processed : tracked stars (up to 5) coordinates on CAM transverse axes.

The following algorithms specify the different processes to be done in order to carry out the STR/CAM in-flight calibration.

12.2.2 Inputs

The following data are necessary to perform the specified ground computations:

- Ground segment data :

- A star catalogue containing stars up to magnitude 11, i.e. compatible with the navigation camera tracking performance. The star catalogue should be such that, as far as possible, five stars shall always be visible in the CAM FOV whatever the inertial direction of the CAM bore-sight axis
- CAM FOV size ($5^\circ \times 5^\circ$), reduced to $4.9^\circ \times 4.9^\circ$ for margins (to avoid selecting a star too close to the FOV edge)
- TM data acquired only once to cross-check with ground segment data
 - Commanded spacecraft attitude quaternion with respect to the J2000 reference frame, at each inertial attitude hold phase : $ref_cmd_qua (Q_{cmd}^n)$, where $n=1$ to 100
 - STR A, STR B, CAM A and CAM B reference frame attitude quaternion with respect to the spacecraft reference frame : $str_to_sat_qua_A (Q_{STR/SC}^A)$, $str_to_sat_qua_B (Q_{STR/SC}^B)$, $cam_to_sat_qua_A (Q_{CAM/SC}^A)$, $cam_to_sat_qua_B (Q_{CAM/SC}^B)$
- TM data acquired from the optical sensors during each inertial attitude hold phase ($n=1$ to 100) :
 - For each star tracked inside the STR A / STR B Autonomous Acquisition / Tracking Mode TM data packet, elements acquired 100 times ($i=1$ to 100):
 - $star_direction_cosine (\vec{u}_{star/J2000}^{STR}(1:3))$: direction cosines of the tracked star in the J2000 reference frame (this should remain the same over the 100 acquisitions)
 - $star_coordinate (X_{m/STR}^i, Y_{m/STR}^i)$: measured star coordinates on the STR transverse axes, at each of the 100 acquisitions
 - For each star tracked inside the CAM A / CAM B Point Target Tracking Mode TM data packet, elements acquired 10 times ($i=1$ to 10):
 - $star_coordinate (X_{m/CAM}^i, Y_{m/CAM}^i)$: measured star coordinates on the CAM transverse axes, at each of the 10 acquisitions

12.2.3 Outputs

The following data is output from the ground processing :

- Updated STR A, STR B, CAM B reference frame attitude quaternion with respect to the spacecraft frame : $str_to_sat_qua_A (Q_{STR/SC}^A)$, $str_to_sat_qua_B (Q_{STR/SC}^B)$, $cam_to_sat_qua_B (Q_{CAM/SC}^B)$, which should be updated by TC to the AOCMS SW

12.2.4 Ground processing

12.2.4.1 CAM Data pre-processing

This process consists in preparing the inputs to be up-linked to the CAM (through the MTL), which will also be used in further ground computations. Indeed it is necessary to command the stars to be tracked by the CAM during each inertial attitude hold phase, by extracting adequate stars from the on ground catalogue and predicting their coordinates on the CAM CCD matrix :

The following process must be repeated for each CAM (only if the nominal orientations of the two sensors are not identical following ground alignment procedures) and for each of the n=1 to 100 inertial attitude hold phases.

12.2.4.1.1 Computation of the CAM bore-sight axis with respect to the J2000 reference frame :

The CAM bore-sight inertial direction $\vec{Z}_{CAM/J2000}^n$ needs to be computed for selecting the stars visible in the CAM FOV. This is computed from the spacecraft commanded attitude instead of the on-board estimated one, given that the slight difference between the two (spacecraft control error) is negligible compared to the CAM tolerance on the tracking window positioning error.

The CAM bore-sight axis is the CAM internal Z axis which is processed through the quaternion $Q_{CAM/SC}^{A/B}$ representing the orientation of the CAM reference frame in spacecraft axes and then through the quaternion Q_{cmd}^n representing the orientation of the spacecraft reference frame in the J2000 inertial frame :

$$\begin{bmatrix} 0 \\ \vec{Z}_{CAM/J2000}^n \end{bmatrix} = Q_{cmd}^n \otimes \overline{Q}_{CAM/SC}^{A/B} \otimes \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \otimes Q_{CAM/SC}^{A/B} \otimes \overline{Q}_{cmd}^n$$

12.2.4.1.2 Declination of the CAM bore-sight inertial direction vector

From the previous vector, the inertial declination angle δ_{Zcam} of the CAM bore-sight axis is computed for later use in the extraction of the visible stars from the catalogue :

$$\text{With } \vec{Z}_{CAM/J2000}^n = \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \end{bmatrix}, \delta_{Zcam} \text{ is such that } \begin{cases} \sin(\delta_{Zcam}) = Z_3 \\ \cos(\delta_{Zcam}) = \sqrt{Z_1^2 + Z_2^2} \end{cases}$$

12.2.4.1.3 Selection of visible stars :

The CAM Point Target Tracking Mode needs to be guided through the predicted coordinates on the CAM CCD of the stars to be tracked. These stars must be selected from a ground star catalogue

containing stars up to magnitude 11 which is the CAM sensitivity. Actually the star catalogue contents should be such that as far as possible, five stars are visible in the 5°x5° CAM FOV whatever the CAM bore-sight inertial direction.

In the catalogue, stars are represented by :

- either their direction cosine vector \vec{u}_{J2000} , or the 2 angles α and δ representing their declination and right ascension. In the latter case, more computations will be necessary, especially *sine* et *cosine* calculations, but this may not be too much constraining for ground processing. The two solutions will be considered in the algorithms.
- their visual magnitude, which must also be sent by TC to the CAM to enable stars tracking.

Stars should be ordered in ascending declination or ascending third coordinate of the direction cosine ($u_z = \sin \delta$) in order to facilitate the stars selection (for faster processing).

The selection of visible stars within the CAM FOV is then performed from three successive tests :

- star pre-selection from its declination (in order to reduce the processing of the next test) :

If the declination is available in the catalogue, the pre-selection can be expressed simply :

$$\delta_{Zcam} - \frac{FOV}{2} \leq \delta(star) \leq \delta_{Zcam} + \frac{FOV}{2} \Rightarrow \text{star pre-selected}$$

where δ_{Zcam} is the declination of the CAM bore-sight inertial direction vector computed before.

If the catalogue contains the direction cosine vector, the pre-selection is performed on the 3rd coordinate, $u_z = \sin(\delta)$:

$$z_{min} = \sin\left(\delta_{Zcam} - \frac{FOV}{2}\right) \text{ or } z_{min} = -1 \quad \text{if} \quad \delta_{Zcam} - \frac{FOV}{2} < -\frac{\pi}{2}$$

$$z_{max} = \sin\left(\delta_{Zcam} + \frac{FOV}{2}\right) \text{ or } z_{min} = +1 \quad \text{if} \quad \delta_{Zcam} + \frac{FOV}{2} > +\frac{\pi}{2}$$

$$\text{if } (z_{min} \leq u_z(star) \leq z_{max}) \Rightarrow \text{star pre-selected}$$

- FOV test :

If the catalogue contains the declination δ and the right ascension α for each star, the cosine direction vector must be computed :

$$\vec{u}_{J2000} = \begin{bmatrix} \cos\delta \cdot \cos\alpha \\ \cos\delta \cdot \sin\alpha \\ \sin\delta \end{bmatrix}$$

Afterwards, the pre-selected stars are checked with respect to the CAM FOV :

$$\text{if } \vec{Z}_{CAM/J2000}^n \cdot \vec{u}_{J2000}(star) > \cos(FOV/2) \Rightarrow \text{star in the FOV}$$

- Final selection according to a distance criteria

The final selection is performed only if the number of stars is higher than the maximum stars tracked at the same time, i.e. 5. In that case only the five most distant stars from the CAM bore-

sight axis are retained in order to maximize the observability, i.e. the five stars minimizing the following expression :

$$\vec{Z}_{CAM/J2000}^n \cdot \vec{u}_{J2000}(\text{star}), \text{ among all pre-selected stars.}$$

The direction cosine vectors of the five retained stars in the J2000 reference frame are named $\vec{u}_{star/J2000}^{CAM}$ in the following sections. There are 5 x100 such vectors (five stars for each inertial attitude scanned).

12.2.4.1.4 Prediction of star coordinates on the CAM CCD :

The predicted coordinates $(X_{p/CAM}^{A/B}, Y_{p/CAM}^{A/B})$ on CAM transverse axes of the stars to be tracked are computed as follows :

$$Q_{star/CAM}^{A/B} = Q_{CAM/SC}^{A/B} \otimes \bar{Q}_{cmd}^n \otimes \begin{bmatrix} 0 \\ \vec{u}_{star/J2000}^{CAM}(1) \\ \vec{u}_{star/J2000}^{CAM}(2) \\ \vec{u}_{star/J2000}^{CAM}(3) \end{bmatrix} \otimes Q_{cmd}^n \otimes \bar{Q}_{CAM/SC}^{A/B}$$

and $\begin{cases} X_{p/CAM}^{A/B} = \frac{Q_{star/CAM}^{A/B}(2)}{Q_{star/CAM}^{A/B}(4)} \\ Y_{p/CAM}^{A/B} = \frac{Q_{star/CAM}^{A/B}(3)}{Q_{star/CAM}^{A/B}(4)} \end{cases}$ for each of the five stars and for each of the 100 inertial attitudes

12.2.4.2 STR/CAM misalignment estimation

The STR/CAM misalignment can be calibrated using stars measurements from the two sensors as follows :

The CAM and the STR generate the same type of measures, i.e. direction cosines of the tracked stars on their transverse axes. By comparing those direction cosines with the predicted ones, the following measurement equation with respect to the spacecraft attitude can be expressed :

$$H \cdot \Delta\theta = \Delta Z$$

with

ΔZ : innovation ([2x1] vector) which represents the difference between the actual and the predicted stellar measurements on the CCD matrix, (in the (\underline{u} , \underline{v}) plane defined below),

$\Delta\theta$: attitude deviation ([3x1] vector) which is the difference between the predicted spacecraft attitude and the true one, in the spacecraft frame,

H : observation matrix ([2x3] matrix) defined below,

The observation matrix is defined by :

$$H = \begin{bmatrix} 0 \\ -\underline{w}^T \end{bmatrix} Xp + \begin{bmatrix} \underline{w}^T \\ 0 \end{bmatrix} Yp + \begin{bmatrix} \underline{v}^T \\ -\underline{u}^T \end{bmatrix}$$

in which:

\underline{u} , \underline{v} , \underline{w} are the direction vectors of the sensor frame (CAM or STR) in the spacecraft frame, assuming \underline{w} is the bore-sight direction, which can be derived as follows from the quaternions $Q_{STR/SC}^{A/B}$, $Q_{CAM/SC}^{A/B}$, which provide the orientation of the sensor frames in the spacecraft frame :

$$\begin{pmatrix} 0 \\ \underline{u}(1) \\ \underline{u}(2) \\ \underline{u}(3) \end{pmatrix} = \overline{Q}_{STR_or_CAM/SC}^{A/B} * \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} * Q_{STR_or_CAM/SC}^{A/B}$$

$$\begin{pmatrix} 0 \\ \underline{v}(1) \\ \underline{v}(2) \\ \underline{v}(3) \end{pmatrix} = \overline{Q}_{STR_or_CAM/SC}^{A/B} * \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} * Q_{STR_or_CAM/SC}^{A/B}$$

$$\begin{pmatrix} 0 \\ \underline{w}(1) \\ \underline{w}(2) \\ \underline{w}(3) \end{pmatrix} = \overline{Q}_{STR_or_CAM/SC}^{A/B} * \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} * Q_{STR_or_CAM/SC}^{A/B}$$

Xp , Yp are the predicted coordinates on the CCD matrix, in the (\underline{u} , \underline{v}) plane, corresponding to a given stellar measurement,

$$\Delta Z \text{ must be written as : } \begin{cases} \Delta Z_x = X_m - R \cdot X_p \\ \Delta Z_y = Y_m - R \cdot Y_p \end{cases} \text{ with } R = \sqrt{\frac{1+X_m^2+Y_m^2}{1+X_p^2+Y_p^2}}, \text{ where } X_m \text{ and } Y_m$$

are the star direction cosine measurements on the sensor transverse axes ($\underline{u}, \underline{v}$).

The above expressions have been derived from equations provided in [9] which are not repeated here.

If $\underline{\delta}_{STR}$ et $\underline{\delta}_{CAM}$ are STR and CAM 3-axis angular misalignment vectors with respect to their nominal direction vectors (\underline{u} , \underline{v} , \underline{w}), assuming these vectors are representing small rotations, this leads to (for each star):

$$\Delta Z_{STR} = H_{STR} \cdot (\Delta \underline{\theta} + \underline{\delta}_{STR})$$

$$\Delta Z_{CAM} = H_{CAM} \cdot (\Delta \underline{\theta} + \underline{\delta}_{CAM})$$

with

$\Delta \underline{\theta}$ attitude deviation at the time of measurement

H_{STR} and H_{CAM} are the observation matrices for each observed star, for the STR and the CAM.

The spacecraft attitude deviation, which has no influence on the bias calibration, can be eliminated from the measurement equations by combination, supposing there is a linear dependence between each line of the observation matrices :

$$A_{STR} \cdot H_{STR} + A_{CAM} \cdot H_{CAM} = 0$$

with

$A_{STR} = [A_{STR\ 1}, A_{STR\ 2}]$ and $A_{CAM} = [A_{CAM\ 1}, A_{CAM\ 2}]$ in which $A_{STR\ i}$ and $A_{CAM\ i}$ are scalar values, which can be computed as described in Section 12.4.

Indeed, the use of these coefficients leads to:

$$A_{STR} \cdot Z_{STR} + A_{CAM} \cdot Z_{CAM} = (A_{STR} \cdot H_{STR} + A_{CAM} \cdot H_{CAM}) \Delta\theta + A_{STR} \cdot H_{STR} \cdot \underline{\delta}_{STR} + A_{CAM} \cdot H_{CAM} \cdot \underline{\delta}_{CAM}$$

And finally :

$$A_{STR} \cdot Z_{STR} + A_{CAM} \cdot Z_{CAM} = A_{STR} \cdot H_{STR} \cdot (\underline{\delta}_{STR} - \underline{\delta}_{CAM})$$

This equation stands for one star pair (one star on each optical sensor), and can be re-written as :

$$Z_i = H_i \cdot (\underline{\delta}_{STR} - \underline{\delta}_{CAM}) \text{ where } Z_i \text{ is a scalar and } H_i \text{ is a } 1 \times 3 \text{ vector}$$

This expression multiplied by the number of star pairs and by the number of inertial attitudes held by the spacecraft enables to estimate the relative misalignment $\Delta\underline{\delta}$ between the two optical sensors, through the least square estimation formula :

$$\Delta\underline{\delta} = (\underline{\delta}_{STR} - \underline{\delta}_{CAM}) = (H^t \cdot H)^{-1} \cdot H^t \cdot Z$$

The following computation steps are necessary to achieve the above result :

12.2.4.2.1 STR/CAM innovations and observation matrices pre-processing

This first step consists in pre-processing the innovations from all optical sensors as follows.

For each of the 100 inertial attitudes :

- At CAM level (for each CAM) :

The tracked stars predicted coordinates ($X_{p/CAM}, Y_{p/CAM}$) have been computed according to Section 12.2.4.1.4, for each of the five visible stars. It has been assumed that 10 TM packet measurements $j=1$ to 10 are available which were acquired over 100s. These measurements need to be filtered (i.e. averaged) in order to derive mean innovation values over that period :

$$\Delta Z_{CAM}^{mean} = \frac{1}{10} \begin{bmatrix} \sum_{j=1}^{10} \Delta Z_x(j) \\ \sum_{j=1}^{10} \Delta Z_y(j) \end{bmatrix} \quad \text{where} \quad \begin{cases} \Delta Z_x(j) = X_{m/CAM}(j) - \sqrt{\frac{1 + X_{m/CAM}^2(j) + Y_{m/CAM}^2(j)}{1 + X_{p/CAM}^2 + Y_{p/CAM}^2}} \cdot X_{p/CAM} \\ \Delta Z_y(j) = Y_{m/CAM}(j) - \sqrt{\frac{1 + X_{m/CAM}^2(j) + Y_{m/CAM}^2(j)}{1 + X_{p/CAM}^2 + Y_{p/CAM}^2}} \cdot Y_{p/CAM} \end{cases},$$

for each tracked star

Note that the predicted star coordinates are constant over the 10 measurements as the attitude is inertial.

The CAM measurement matrix for each star remains constant over the 10 measurements and is equal to :

$$H_{CAM} = \begin{bmatrix} 0 \\ -\underline{w}_{CAM}^T \end{bmatrix} X_{p/CAM} + \begin{bmatrix} \underline{w}_{CAM}^T \\ 0 \end{bmatrix} Y_{p/CAM} + \begin{bmatrix} \underline{v}_{CAM}^T \\ -\underline{u}_{CAM} \end{bmatrix} \text{ for each tracked star}$$

- At STR level (for each STR) :

The STR is assumed to produce 100 measurements of the same star pattern, including up to 9 stars, during 100s of inertial attitude hold phases.

These 100 measurements must be processed as for the CAM in order to produce average innovation values. The first step is to compute the predicted star coordinates on STR transverse axes as follows :

For each of the nine stars :

$$Q_{star/STR} = Q_{STR/SC}^{A/B} \otimes \bar{Q}_{cmd}^n \otimes \begin{bmatrix} 0 \\ \bar{u}_{star/J2000}^{STR}(1) \\ \bar{u}_{star/J2000}^{STR}(2) \\ \bar{u}_{star/J2000}^{STR}(3) \end{bmatrix} \otimes Q_{cmd}^n \otimes \bar{Q}_{STR/SC}^{A/B} \text{ where the } \bar{u}_{star/J2000}^{STR} \text{ vector of the star}$$

direction cosines in the J2000 inertial frame are provided in the STR Autonomous Tracking TM data packet, and should be constant over the 100 measurements because of the inertial attitude.

$$\text{Then } \begin{cases} X_{p/STR} = \frac{Q_{star/STR}(2)}{Q_{star/STR}(4)} \\ Y_{p/STR} = \frac{Q_{star/STR}(3)}{Q_{star/STR}(4)} \end{cases}, \text{ for each of the nine stars.}$$

When combining STR and CAM measurements, the same number of stars must be retained on each side. Therefore five from the nine stars tracked by the STR must be selected for further processing. As for the stars selected in the on-ground catalogue to be up-linked to the CAM, the observability of STR/CAM misalignments is maximized when the five most distant stars from the STR bore-sight are selected. This is performed by retaining the five stars which maximize the value of $X_{p/STR}^2 + Y_{p/STR}^2$, based on the predicted star coordinates.

Then mean innovations are computed on the five remaining stars as for the CAM :

$$\Delta Z_{STR}^{mean} = \frac{1}{100} \begin{bmatrix} \sum_{j=1}^{100} \Delta Z_x(j) \\ \sum_{j=1}^{100} \Delta Z_y(j) \end{bmatrix} \text{ where } \begin{cases} \Delta Z_x(j) = X_{m/STR}(j) - \sqrt{\frac{1+X_{m/STR}^2(j)+Y_{m/STR}^2(j)}{1+X_{p/STR}^2+Y_{p/STR}^2}} \cdot X_{p/STR} \\ \Delta Z_y(j) = Y_{m/STR}(j) - \sqrt{\frac{1+X_{m/STR}^2(j)+Y_{m/STR}^2(j)}{1+X_{p/STR}^2+Y_{p/STR}^2}} \cdot Y_{p/STR} \end{cases}$$

The STR measurement matrix for each star remains constant over the 100 measurements and is equal to :

$$H_{STR} = \begin{bmatrix} 0 \\ -\underline{w}_{STR}^T \end{bmatrix} X_{p/STR} + \begin{bmatrix} \underline{w}_{STR}^T \\ 0 \end{bmatrix} Y_{p/STR} + \begin{bmatrix} \underline{v}_{STR}^T \\ -\underline{u}_{STR} \end{bmatrix} \text{ for each tracked star}$$

12.2.4.2.2 STR/CAM measurements combination and least square processing

The previous computation steps allowed to gather, for each of the 100 inertially fixed attitudes :

- Five 2x1 innovation vectors ΔZ_{CAM}^{mean} and five 2x3 measurement matrices H_{CAM} , for each CAM
- Five 2x1 innovation vectors ΔZ_{STR}^{mean} and five 2x3 measurement matrices H_{STR} , for each STR

For each sensor pair STR A / CAM A, STR B / CAM A, CAM A / CAM B, these measurements must now be combined in the following way :

- Computation of linear dependence coefficients :

Star pairs must be formed using one star on each side, the grouping order having no influence. Then linear dependence coefficients vectors A_{STR} (1x2) and A_{CAM} (for example, assuming one STR and one CAM are considered) must be computed as described in Section 12.4, such that :

$$A_{STR} \cdot H_{STR} + A_{CAM} \cdot H_{CAM} = 0$$

Note that in the particular case of calibrating CAM B versus CAM A, and if both sensors have exactly the same nominal orientations in the spacecraft frame, then linear dependence coefficients are equal to [1 1] and [-1 -1], i.e. the CAM innovations can be directly subtracted in order to provide an observation of the misalignments between both cameras.

- Least square processing:

For each star pair, compute the observable and the observation matrix which will constitute the data to be processed by the least square algorithm (shown here for a STR / CAM pair) :

$$Z_{STR/CAM} = [Z_{STR/CAM} ; A_{STR} \cdot Z_{STR} + A_{CAM} \cdot Z_{CAM}] , H_{STR/CAM} = [H_{STR/CAM} ; A_{STR} \cdot H_{STR}]$$

$Z_{STR/CAM}$ being at the end a 500x1 vector and $H_{STR/CAM}$ being a 500x3 matrix.

The least square result then writes :

$$\Delta \underline{\delta} = (\underline{\delta}_{STR} - \underline{\delta}_{CAM}) = (H_{STR/CAM}^t \cdot H_{STR/CAM})^{-1} \cdot H_{STR/CAM}^t \cdot Z_{STR/CAM}$$

12.2.4.3 STR/CAM alignment matrix correction

The CAM A is considered as the reference and all other optical sensors are corrected with the estimated misalignments.

The relative misalignment $\Delta \underline{\delta}$ must be transformed into a corrected STR A, STR B and CAM B orientation in the spacecraft frame expressed as a quaternion Q_{new} assuming the calibrated misalignments are small angles, as a function of the nominal orientation Q_{old} :

$$Q_{new} = Q_{old} \cdot \begin{bmatrix} \sqrt{1 - \frac{\Delta\delta(1)^2 + \Delta\delta(2)^2 + \Delta\delta(3)^2}{4}} \\ \frac{\Delta\delta(1)}{2} \\ \frac{\Delta\delta(2)}{2} \\ \frac{\Delta\delta(3)}{2} \end{bmatrix}$$

12.3 PROTOTYPING AND NUMERICAL VALIDATION

Previous algorithms have been partly prototyped and tested in order to validate the calibration method, focusing on the STR/CAM bias estimation, which must converge towards the true misalignment. One typical simulation illustrates the on-ground computations from stellar measurements.

12.3.1 Assumptions

One STR and one CAM have been considered with their representative orientation in the spacecraft frame. One hundred different star patterns have been considered on each sensor, with 100 different CAM measurements performed every 10 seconds, and 2000 STR measurements performed every 0.5s, i.e. the star pattern holds during 1000s. This is not fully representative of the real situation, where less measurements will be processed on each star pattern, however the calibration performance primarily depends on the star patterns measurements, not on the noise filtering. Furthermore the aim of the present activity is to validate the algorithms, not the performance. Another major difference with respect to reality is that only one star will be simulated for each star pattern, not five, in order to simplify the prototype program. Furthermore the least square processing algorithm has been coded in a recursive form instead of a direct global processing. This allows to figure out an approximate convergence time for the algorithm, as a function of the number of measurements processed.

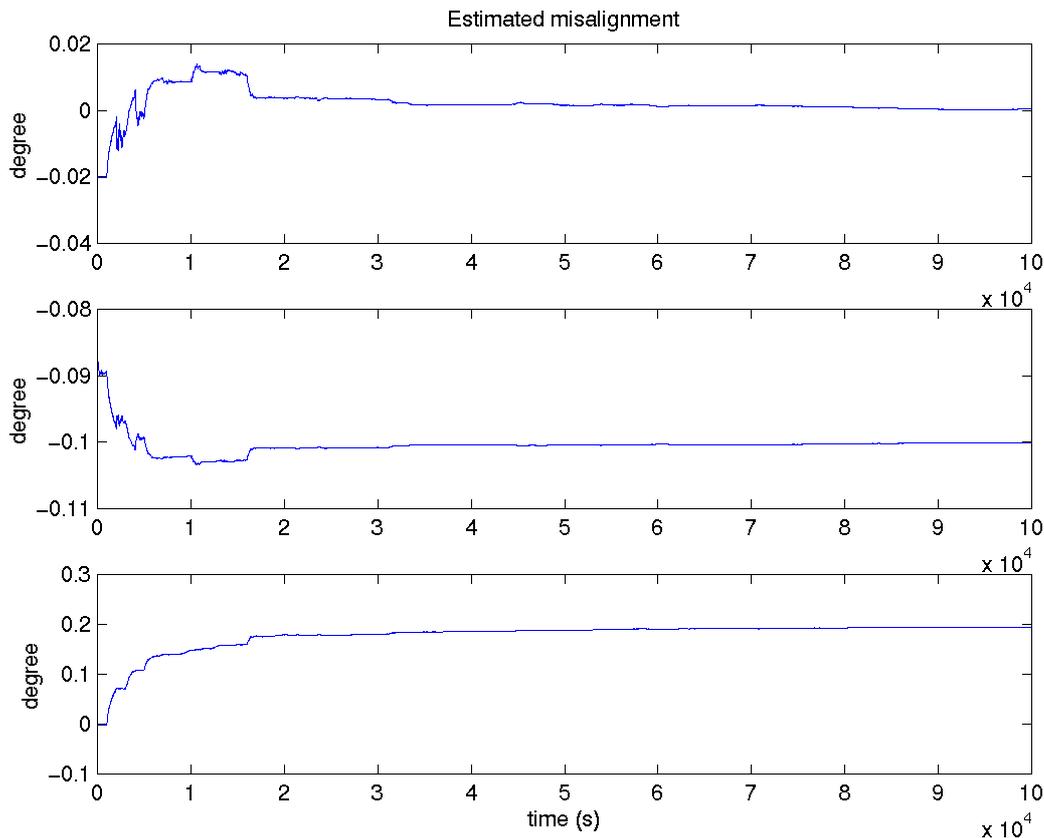
Other main assumptions are :

- A typical misalignment of -0.1 deg is considered on Y-axis and +0.2 deg on Z-axis (no misalignment on X-axis),
- Measurement Noise : 9% of the pixel dimension (17 arcsec) for the CAM and the STR
- A random bias for each star measurement varying within [-17,+17] arcsec
- Random positions for the 100 star positions on the CCD matrices

Note that the real star selection will provide better performance, because the most distant stars from the CCD matrix center will be retained (increased observability).

12.3.2 Results

The following graphs represent the outputs of the recursive least square procedure: Estimated misalignment corresponds to $\Delta\delta$, the estimated STR/CAM misalignment :



Evolution of the misalignment vector (X, Y, Z) estimation (named $\Delta\delta$ in the equations)

As confirmed by the previous graph, the misalignment estimator converges towards the real misalignment : (0,-0.1,0.2) degrees, in about 17000s, i.e. after 17 different stars have been acquired on each sensor. This demonstrates that the proposed calibration process will definitely converge towards accurate values in the real situation, where up to 500 different stars (100 different patterns of five stars) will be acquired by each sensor.

12.4 APPENDIX : COMPUTATION OF LINEAR DEPENDENCE COEFFICIENTS FOR OBSERVATION MATRICES

Observation matrices are [2x3] matrices which are noted :

$$H_{STR} = \begin{bmatrix} L_{STR_1} \\ L_{STR_2} \end{bmatrix} \quad \text{and} \quad H_{CAM} = \begin{bmatrix} L_{CAM_1} \\ L_{CAM_2} \end{bmatrix}$$

with

L_{STR_1} a [1x3] vector representing the first line of the observation matrix for STR

L_{STR_2} a [1x3] vector representing the second line of the observation matrix

Idem for L_{CAM_1} and L_{CAM_2}

The coefficients $A_{STR} = \begin{bmatrix} A_{STR_1} \\ A_{STR_2} \end{bmatrix}$ and $A_{CAM} = \begin{bmatrix} A_{CAM_1} \\ A_{CAM_2} \end{bmatrix}$, [2x1] vectors, are solutions of the following relation:

$$A_{STR}^T \cdot H_{STR} + A_{CAM}^T \cdot H_{CAM} = 0$$

It leads to 3 equations with 4 unknown variables; i.e. with one degree of freedom.

The previous relation can be written as, using the defined notations:

$$\begin{bmatrix} L_{STR_1}^T & L_{STR_2}^T & L_{CAM_1}^T \end{bmatrix} \cdot \begin{bmatrix} A_{STR_1} & A_{STR_2} & A_{CAM_1} \end{bmatrix}^T = -A_{CAM_2} \cdot \begin{bmatrix} L_{CAM_1} \end{bmatrix}^T$$

Assuming $A_{CAM_2} = -\det\left(\begin{bmatrix} L_{STR_1}^T & L_{STR_2}^T & L_{CAM_1}^T \end{bmatrix}\right)$, the other coefficients can be deduced easily. Indeed, one solution for the linear dependence coefficients expresses as:

$$\begin{cases} A_{STR_1} = +\det\left(\begin{bmatrix} L_{STR_2}^T & L_{CAM_1}^T & L_{CAM_2}^T \end{bmatrix}\right) \\ A_{STR_2} = -\det\left(\begin{bmatrix} L_{STR_1}^T & L_{CAM_1}^T & L_{CAM_2}^T \end{bmatrix}\right) \\ A_{CAM_1} = +\det\left(\begin{bmatrix} L_{STR_1}^T & L_{STR_2}^T & L_{CAM_2}^T \end{bmatrix}\right) \\ A_{CAM_2} = -\det\left(\begin{bmatrix} L_{STR_1}^T & L_{STR_2}^T & L_{CAM_1}^T \end{bmatrix}\right) \end{cases}$$

This formulation is used in the least square algorithm, to estimate the STR/CAM biases.

13 SPACECRAFT MOMENTS OF INERTIA IN-FLIGHT CALIBRATION

13.1 CONTEXT AND PURPOSE

For pointing performance optimisation reasons, the S/C inertia matrix is used on-board in SHM, NM, NSHM and AFM modes to compute feed-forward torques and to compensate for gyroscopic torques. The on-board inertia matrix must be up-linked regularly by the ground by TC(172,78) "AOCMS_COMMON.set_sc_inertia_matrix" in order to fulfill an on-board estimation accuracy of at least +/-15% in NM outside comet observation, SHM, NSH, and AFM, and at least +/-3% in NM during comet observation as more stringent pointing performance requirements apply. The +/-15% estimation accuracy constraint is supposed to be fulfilled by classical spacecraft mass balancing prediction tools, whereas the +/-3% estimation accuracy requires an in-flight S/C inertia calibration. Those constraints apply on the diagonal moments of inertia. No constraint has been identified on the cross-products of inertia, but typical figures of 50kgm² can be retained as a goal.

According to ref.[8], the +/-3% estimation accuracy of S/C moments of inertia is necessary in order to guarantee AOCMS pointing performances while orbiting the comet. In previous mission phases, the pointing requirements are less stringent so that the on-ground prediction of the S/C moments of inertia is precise enough to ensure adequate AOCMS pointing performances. Furthermore, in these early mission phases, the large amount of fuel inside the tanks would prohibit an accurate in-flight inertia calibration.

The in-flight S/C inertia calibration only needs to be done once over the mission and will be performed after the large ΔV manoeuvres to rendez-vous with the comet but before the scientific pointing sequences in comet observation. Indeed the fuel left in the tanks after this event is low enough not to alter too much the spacecraft inertia configuration afterwards.

The calibration process is divided into two steps:

- an adequate slew manoeuvre using reaction wheels is first performed in NM-GSP and the resulting gyro measurements, reaction wheel speeds and estimated S/C attitude quaternion data are stored on-board
- all these data are then down-linked to the ground where they are used as inputs of a least-square estimation algorithm to provide an evaluation of the S/C moments of inertia.

Section 13.2 details the constraints which apply on the slew manoeuvre (first step), whereas section 13.3 describes the ground-processing algorithm itself (second step). Some results obtained when prototyping the proposed algorithms are then documented in section 13.4.

13.2 CONSTRAINTS ON THE SLEW

As stated above, the calibration ground-processing is preceded by a slew manoeuvre which has to conform to the constraints detailed in this paragraph.

13.2.1 General constraints

The slew guidance profile must be generated according to several general constraints, as listed in the procedure FCP-AC0044 described in [7] :

- The slew must be performed in NM-GSP, using reaction wheels and gyros,
- Continuity of the slew attitude profile with respect to the previous and future commanded attitude must be ensured, within an accuracy of 0.01°
- Attitude segments defining the slew must be synchronized with respect to the on-board time and with respect to the GSP entry and exit TCs (see the procedure for details)
- Attitude segments must represent the desired profile within an accuracy of 0.01° , including at transitions between successive segments. Note that this constraint could be relaxed up to typically 0.1° for the slew aimed at the in-flight calibration of spacecraft inertia, as no pointing performance requirement apply, and only the rate profile is of interest for the ground processing
- reaction wheels speeds / torques capacities must be fulfilled
- the SA solar aspect angle constraint must be fulfilled, i.e. either the predicted out-of-XZ-plane solar aspect angle remains below 15° , or the Linear Range Sun Pointing Monitoring must be disabled (see also specific constraints listed below)
- the SA interface torque limitation constraint must be fulfilled, i.e. the SADE speed level shall not exceed level 6. Furthermore the SADM end stops shall not be reached during the slew

13.2.2 Specific constraints

The slew aimed at supporting the spacecraft inertia in-flight calibration process shall also fulfill specific constraints in order to achieve the optimal conditions for restituting the spacecraft moments of inertia from the available TMs :

- the attitude profile shall generate rates about each of the 3 S/C axes, by sequencing for instance 3 single-axis rotations
- the minimum rate values to be reached on each S/C axis are $0.05^\circ/\text{s}$ about X/Z axes and $0.2^\circ/\text{s}$ about the Y-axis (note anyway that there is no duration nor angular amplitude constraint, since the important issue is to reach but not to hold the above rate levels). With these constraints fulfilled, there will be enough measurements to be processed for the calibration algorithm convergence.
- There is no specific constraint on separating the slews about each axis, i.e. the three axial slewing rates can be combined in order to reduce the slew duration and avoid transient phases with no imparted rate, which are useless for the calibration algorithm
- the maximum SADE speed level has to be limited to 'level 4' ($0.05^\circ/\text{s}$) in order to avoid oscillations between level 4 and 5 ($0.3^\circ/\text{s}$) which induce important perturbations around the S/C Y axis. As a consequence, the Linear Range Sun pointing angle monitoring has therefore to be inhibited in order to avoid any alarm triggering when the SA pointing error increases

- The HGA must be set to a fixed position during the manoeuvre to avoid large inertia changes due to HGA motion. Furthermore the position chosen should be either an average position in terms of inertia impact or a specific position representative of the range to be experienced during scientific pointing profiles. Since the in-flight calibration algorithm assumes the external perturbation torque to be negligible, it is also better to set the HGA position such that the solar pressure torque is minimised.
- The distance from the comet to the S/C must be large enough to ensure low outgassing disturbance torque levels.

13.3 GROUND PROCESSING

The ground-processing algorithm itself is described in this section.

13.3.1 Principle

The S/C inertia estimation algorithm described here below, is based on the conservation of the total angular momentum (spacecraft and wheels) in the inertial reference frame. In Normal Mode, the S/C is indeed controlled by the wheels and during the spacecraft inertia in-flight calibration process, the external disturbance torques are assumed to be very low and are therefore neglected.

The following notations are used:

- $[J]$ S/C inertia matrix in the S/C reference frame R_{ref} ($kg.m^2$),
- $\omega(t)$ inertial S/C body rate vector in S/C axes measured by the gyros at the date t (rad/s),
- $H_{Rref}^{RW}(t)$ RW angular momentum vector in the S/C reference frame R_{ref} , at the date t (Nms),
- $Q_{RI/Rref}(t)$ S/C attitude quaternion corresponding to the rotation from the inertial frame R_i to R_{ref} , at the date t .

The projection of the angular momentum computed in the S/C frame provides the total angular momentum H_{RI} in the inertial frame:

$$H_{RI}(t) = Q_{RI/Rref}(t) \otimes ([J]\omega(t) + H_{Rref}^{RW}(t)) \otimes \tilde{Q}_{RI/Rref}(t)$$

where

- \tilde{Q} is the inverse quaternion of Q
- \otimes is the quaternion multiplication
- $Q \otimes \vec{v}$ is actually the quaternion multiplication of quaternion Q and quaternion Q_v associated to vector \vec{v} so that: $Q_v = \begin{pmatrix} 0 \\ -\vec{v} \end{pmatrix}$

It is assumed that, at $t = 0$ s, the initial S/C rate is well converged and the S/C attitude is inertial, that is to say $\omega(0) \approx 0$. As a result, for each date t , the previous equation can be written (assuming no external disturbing torque) :

$$Q_{RI/Rref}(t) \otimes ([J]\omega(t) + H_{Rref}^{RW}(t)) \otimes \tilde{Q}_{RI/Rref}(t) = Q_{RI/Rref}(0) \otimes H_{Rref}^{RW}(0) \otimes \tilde{Q}_{RI/Rref}(0)$$

The spacecraft inertia matrix $[J]$ is thus extracted from the previous formula:

$$[J]\omega(t) = \tilde{Q}_{RI/Rref}(t) \otimes Q_{RI/Rref}(0) \otimes H_{Rref}^{RW}(0) \otimes \tilde{Q}_{RI/Rref}(0) \otimes Q_{RI/Rref}(t) - H_{Rref}^{RW}(t) \quad (E1)$$

where $[J]\omega(t)$ denotes the quaternion associated to vector $[J]\omega(t)$

When the time profiles of the S/C angular rate, of the RW angular momentum and of the S/C attitude quaternion are known, the inertia matrix $[J]$ can therefore be estimated using the linear equation :

$$[J]\omega(t) = h(t) . \quad (E1_{bis})$$

where $h(t)$ stands for the right-hand term of equation (E1)

The algorithm used for the estimation of the S/C inertia matrix is based on a least square identification. Taking into account the whole slew profile, N gyro measurements of the S/C angular rates are assumed available, and N "inertial angular momentums" values h_n can be computed according to equation (E1). The matrix $[J]$ must verify the following equation for each measurement n between 1 and N .

$$[J]\omega_n = h_n, \text{ with } n \text{ from } 1 \text{ to } N.$$

The matrices $[\omega]$ and $[h]$ are two 3-by- N matrices defined as follows:

$$[\omega] = [\omega_1 \quad \omega_2 \quad \dots \quad \omega_N]$$

$$[h] = [h_1 \quad h_2 \quad \dots \quad h_N]$$

$$\text{with } \omega_n = \begin{pmatrix} \omega_{nx} \\ \omega_{ny} \\ \omega_{nz} \end{pmatrix} \quad \text{and} \quad h_n = \begin{pmatrix} h_{nx} \\ h_{ny} \\ h_{nz} \end{pmatrix}$$

Using the least square identification, the best estimation of matrix $[J]$ is:

$$[J^*] = [h][\omega]^T ([\omega][\omega]^T)^{-1}$$

In order to improve the algorithm performances, the symmetry of the inertia matrix $[J]$ is used to formulate the least-square problem in an other way. The proposed algorithm estimates the vector j , defined by:

$$j = [j_1 \quad j_2 \quad j_3 \quad j_4 \quad j_5 \quad j_6] \quad \text{with} \quad [J] = \begin{bmatrix} j_1 & j_4 & j_5 \\ j_4 & j_2 & j_6 \\ j_5 & j_6 & j_3 \end{bmatrix}$$

So, considering the unknown vector j , the equation $[J]\omega_n = h_n$ is written:

$$\begin{bmatrix} \omega_{nx} & 0 & 0 & \omega_{ny} & \omega_{nz} & 0 \\ 0 & \omega_{ny} & 0 & \omega_{nx} & 0 & \omega_{nz} \\ 0 & 0 & \omega_{nz} & 0 & \omega_{nx} & \omega_{ny} \end{bmatrix} \begin{pmatrix} j_1 \\ j_2 \\ j_3 \\ j_4 \\ j_5 \\ j_6 \end{pmatrix} = \begin{pmatrix} h_{nx} \\ h_{ny} \\ h_{nz} \end{pmatrix}$$

Then, the 3-by-6 matrix $[\Omega_n]$ is defined as follows :

$$[\Omega_n] = \begin{bmatrix} \omega_{nx} & 0 & 0 & \omega_{ny} & \omega_{nz} & 0 \\ 0 & \omega_{ny} & 0 & \omega_{nx} & 0 & \omega_{nz} \\ 0 & 0 & \omega_{nz} & 0 & \omega_{nx} & \omega_{ny} \end{bmatrix}, \quad (E2)$$

Finally the vector j of 6 elements is estimated, which minimizes the quadratic norm of:

$$[\Omega] j - \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_N \end{pmatrix} \quad \text{with} \quad [\Omega] = \begin{bmatrix} [\Omega_1] \\ [\Omega_2] \\ \vdots \\ [\Omega_N] \end{bmatrix}$$

Using the least square identification, this leads to :

$$j^* = ([\Omega]^T [\Omega])^{-1} [\Omega]^T \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_N \end{pmatrix} \quad (E3)$$

In the ground process, the spacecraft body rates are replaced by the gyros measurements, the inertial attitude quaternion by the gyro-stellar estimated attitude quaternion, and the reaction wheels angular momentum vector by the reaction wheels measured speeds.

13.3.2 Inputs

The following names are those provided in the Software Requirement Document.

stl_est_gys_pred_qua(t) ($Q_{Ri \rightarrow Rsat}^{est}(t)$) : attitude quaternion predicted on-board by the gyrostellar estimation process at the date t. It is a 1-by-4 vector of floats.

imp_orb_rate_meas(t) ($\omega_{SC}^{meas}(t)$) : vector of inertial S/C body rate measurements around S/C axes at the date t (rad/s). It is a 1-by-3 vector of floats.

rwao_sc_meas_ang_mom_vect(t) ($h_{SC}^{est}(t)$) : on-board estimated RW angular momentum vector expressed in S/C reference frame at the date t (Nms). It is a 1-by-3 vector of floats.

13.3.3 Outputs

estim_inertia_matrix (J_{SC}^{est}) : estimated S/C inertia matrix (kg.m²). It is a 3-by-3 matrix of floats.

13.3.4 Ground processing

Step 1 : Initialisation

N = 0

W_sc = 0

Hwheel_sc = 0

Qest = 0

Step 2 : Creation of the input vectors

Only the time steps corresponding to the slew are used to determine the inertia matrix estimation

For t **from** t_start **to** t_end

W_sc = [W_sc, $\omega_{SC}^{meas}(t)$]

Hwheel_sc = [Hwheel_sc, $h_{SC}^{est}(t)$]

Qest = [Qest, $Q_{Ri \rightarrow Rsat}^{est}(t)$]

N = N + 1

Endfor

Comments

N is the number of valid time steps kept for the processing.

Finally W_sc is a N-by-3 matrix. W_sc(n) is a 1-by-3 vector, which corresponds to the gyro rate measurement at a given time step.

Hwheel_sc is a N-by-3 matrix. Hwheel_sc(n) is a 1-by-3 vector, which corresponds to the wheel angular momentum in spacecraft frame at the same time step as W_sc(n).

Qest is a N-by-4 matrix. Qest(n) is a 1-by-4 vector, which corresponds to the estimated attitude quaternion at the same time step as W_sc(n) and Hwheel_sc(n).

Step 3 : Computation of the vector h from the equation (E1_{bis})

For n **from** 1 **to** N

$$\begin{pmatrix} 0 \\ h_n \end{pmatrix} = Q_{est}(n)^{-1} * Q_{Ri \rightarrow Rsat}^{est}(0) * \begin{pmatrix} 0 \\ h_{SC}^{est}(0)^T \end{pmatrix} * Q_{Ri \rightarrow Rsat}^{est}(0)^{-1} * Q_{est}(n) - \begin{pmatrix} 0 \\ (Hwheel_sc(n))^T \end{pmatrix}$$

Endfor

where the initial values of the spacecraft estimated attitude quaternion and the reaction wheel speeds are used

Then, build the 3*N-by-1 vector h :

$$h = \begin{bmatrix} h_1 \\ \dots \\ h_n \\ \dots \\ h_N \end{bmatrix}$$

Step 4 : Computation of the matrix Ω from the equation (E2)

The matrix Ω is initialised to a null 3*N-by-6 matrix.

Ω = zeros (3*N , 6);

For n **from** 1 **to** N

Ω(1+3*(n-1),1) = W_sc(n,1);

Ω(1+3*(n-1),4) = W_sc(n,2);

Ω(1+3*(n-1),5) = W_sc(n,3);

Ω(2+3*(n-1),2) = W_sc(n,2);

Ω(2+3*(n-1),4) = W_sc(n,1);

Ω(2+3*(n-1),6) = W_sc(n,3);

Ω(3+3*(n-1),3) = W_sc(n,3);

Ω(3+3*(n-1),5) = W_sc(n,1);

$$\Omega(3+3*(n-1),6) = W_sc(n,2);$$

end

Step 5 : Estimation of the 6-by-1 vector j^* from the equation (E3)

$$j^* = (\Omega^T * \Omega)^{-1} * \Omega^T * h$$

Step 6 : Creation of the 3-by-3 matrix estim_inertia_matrix (J_{SC}^{est})

$$J_{SC}^{est} = \begin{pmatrix} j(1) & j(4) & j(5) \\ j(4) & j(2) & j(6) \\ j(5) & j(6) & j(3) \end{pmatrix}$$

13.4 PROTOTYPING AND ANALYSIS

The proposed ground processing algorithm has been prototyped on Xmath using ROSACE simulation outputs of a reference slew manoeuvre in order to validate the equations and to assess the spacecraft inertia estimation performance. The main characteristics of the slew as well as the main results obtained are described in the next Sections. The conclusions drawn from a sensitivity analysis are then provided.

13.4.1 Main characteristics of the reference slew manoeuvre

During the first part of the simulation, the S/C is in WDP with an inertial pointing attitude in order to satisfy the initial conditions of the algorithm: *the initial S/C rate is well converged and the S/C attitude is inertial ($\omega(0) \approx 0$).*

Then the S/C body is slewed around S/C X, Y and Z axes from simulation time 585 sec to 1200 sec, performing the following slews:

- **X-axis** slew: from 585 sec to 892.5 sec with a maximum rate of **1.5×10^{-3} rad/s (0.086°/s)**
- **Y-axis** slew: from 738.75 sec to 1046.25 sec with a maximum rate of **3.5×10^{-3} rad/s (0.2°/s)**
- **Z-axis** slew: from 892.5 sec to 1200 sec with a maximum rate of **1.5×10^{-3} rad/s (0.086°/s)**

Simulation time (sec)	AOCMS phase transition
0	SBM entry
100.875	SBM → NM_WDP
584.875	NM_WDP → NM_GSP
1200	End of simulation

Table 13.4-1: AOCMS modes and phases

The SADE is kept ON but a TC is sent to limit its maximum speed level to level 4.

The HGA is set OFF and therefore kept to its initial position.

No model of liquid dynamics is simulated in these calibration runs.

Other simulation conditions (gyro measurements errors, reaction wheel tachometers errors, misalignments,...) are typical of assumptions described in [4].

13.4.2 Results obtained using TM data of the reference slew manoeuvre

The results obtained using a slew conforming to the optimised constraints detailed in paragraph 13.2 are the following:

$$J_{SC}^{est} = \begin{pmatrix} 17425.3 & 29.9 & 171.8 \\ 29.9 & 1705.2 & -1.8 \\ 171.8 & -1.8 & 17451.7 \end{pmatrix} \text{ kg m}^2$$

Then the estimation error $[\Delta J]$ is computed with respect to the inertia values at the beginning of simulation $[J_{real}]$:

$$|\Delta J| = \begin{pmatrix} 20.9 & 5.2 & 3.0 \\ 5.2 & 11.9 & 2.6 \\ 3.0 & 2.6 & 59.8 \end{pmatrix} \text{ kg.m}^2$$

As the off-diagonal matrix elements are of much lower order of magnitude than the diagonal ones, the absolute error values are assessed for the off-diagonal elements whereas the error percentage is computed for the diagonal ones:

$$\frac{\Delta J_{11}}{J_{11}} = 0.1\%, \quad \frac{\Delta J_{22}}{J_{22}} = 0.7\% \quad \frac{\Delta J_{33}}{J_{33}} = 0.3\%$$

These values comply to the 3% accuracy requirement.

Note also that due to solar array motions the actual S/C inertia matrix coefficients vary along the simulation run duration to the following extend:

$$\frac{J_{11_final} - J_{11_initial}}{J_{11_initial}} = 0.06\%, \quad \frac{J_{22_final} - J_{22_initial}}{J_{22_initial}} = 0.002\%, \quad \frac{J_{33_final} - J_{33_initial}}{J_{33_initial}} = 0.06\% .$$

In the same time, the off-diagonal matrix elements experience the following variation:

$$J_{12_final} - J_{12_initial} = 1.2 \text{ kg.m}^2, \quad J_{13_final} - J_{13_initial} = 4.5 \text{ kg.m}^2, \quad J_{23_final} - J_{23_initial} = 0.7 \text{ kg.m}^2.$$

The order of magnitude of these variations define obviously a limit for the calibration algorithm precision. However the 3% estimation accuracy requirement is not challenged by those values.

13.4.3 Typical time outputs of the reference slew manoeuvre

The following sections show some time simulation outputs of interest extracted from the ROSACE simulation of the reference slew manoeuvre.

13.4.3.1 S/C attitude and rates

The first column of the following figure gives the actual S/C rates encountered during the attitude manoeuvre, the second column gives the attitude control errors around S/C X, Y and Z-axes, and finally, the third column provides the attitude guidance rate profile generated on-board:

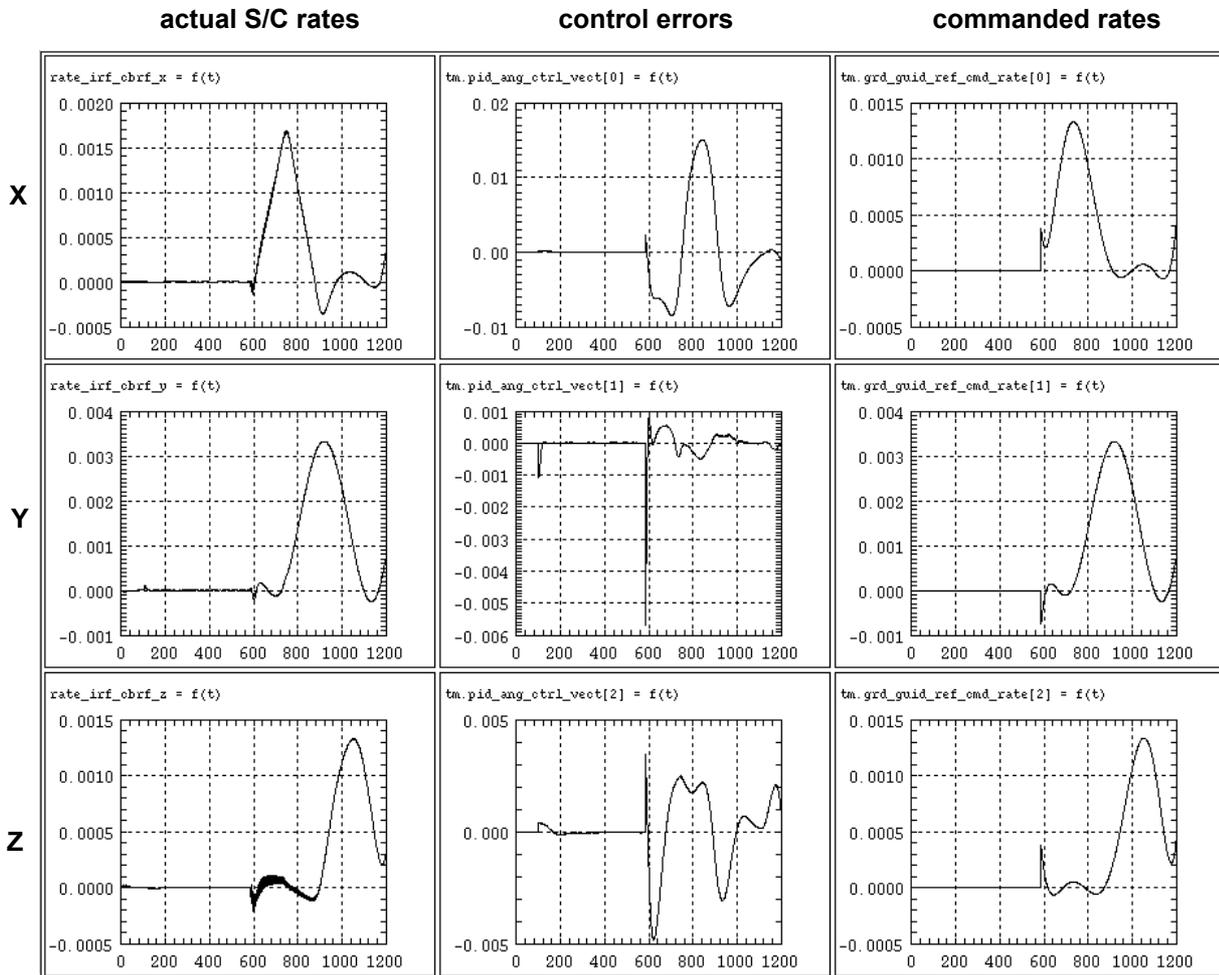


Figure 13.4-1: S/C actual rates, S/C angular control errors and S/C commanded rates (rad/s)

Note that since no particular accuracy was required for generating the attitude guidance profile, only one polynomial function of order 7 was used to interpolate the whole manoeuvre, thus generating non zero rate guidance steps at the beginning of the slew. More polynomial functions can obviously be used to reach a better accuracy on the guidance profile, thus allowing to reduce the angular control errors. However angular control errors are not detrimental to the spacecraft inertia estimation performance because the on-board estimated attitude quaternion, which is used as input to the ground processing algorithm, tracks these errors.

Note also that the spacecraft angular rates exhibit axial slew profiles which are not fully separated, i.e. the slew about one axis starts before the preceding slew is over. This is neither mandatory nor detrimental to the spacecraft inertia estimation performance, but allows to minimize the slew duration and therefore the TM data volume to be stored.

13.4.3.2 Telemetry used as ground process inputs

The following figures plot all the TM variables used as inputs to the calibration algorithm. The first column gives the S/C attitude quaternion predicted on-board by the gyro-stellar estimation process, the second column provides the S/C rates around X, Y and Z axes as measured by the gyros, and the third column gives the on-board estimated values of the RW angular momentum also expressed in S/C X, Y and Z-axes.

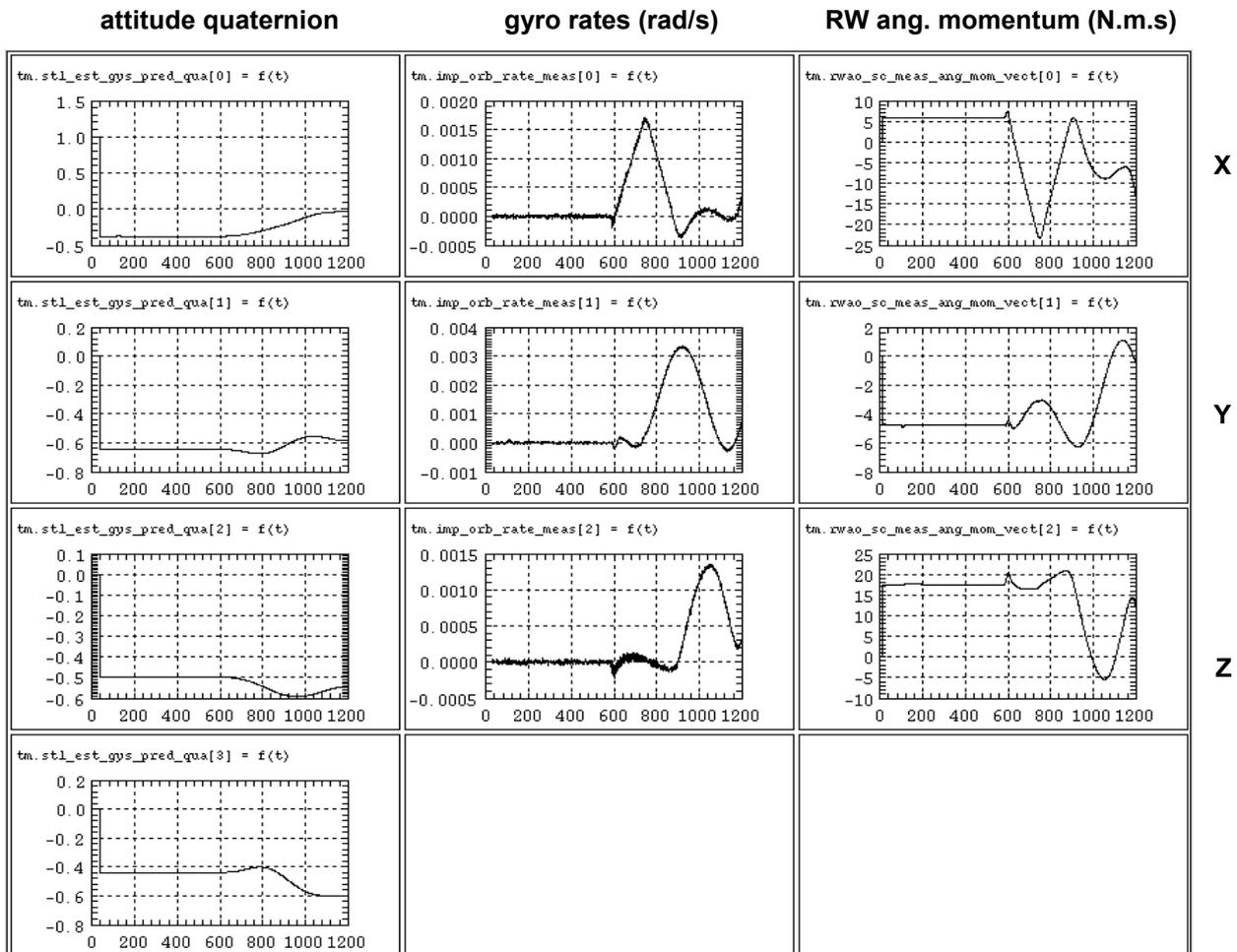


Figure 13.4-2: TM data used as calibration inputs

Despite the SADE speed limit sent by TC, the gyro measurements are still noisy which will slightly affect the inertia estimation accuracy.

13.4.3.3 External disturbance torques

Figure 13.4-3 below provides the disturbance torque levels encountered during the manoeuvre expressed in S/C axes: the first column gives the 3 components of the solar pressure torque, the

second column shows the gravity gradient torques, the third column shows the aerodynamic torques and the last columns finally gives the 3 components of the total disturbance torque resulting from each contribution.

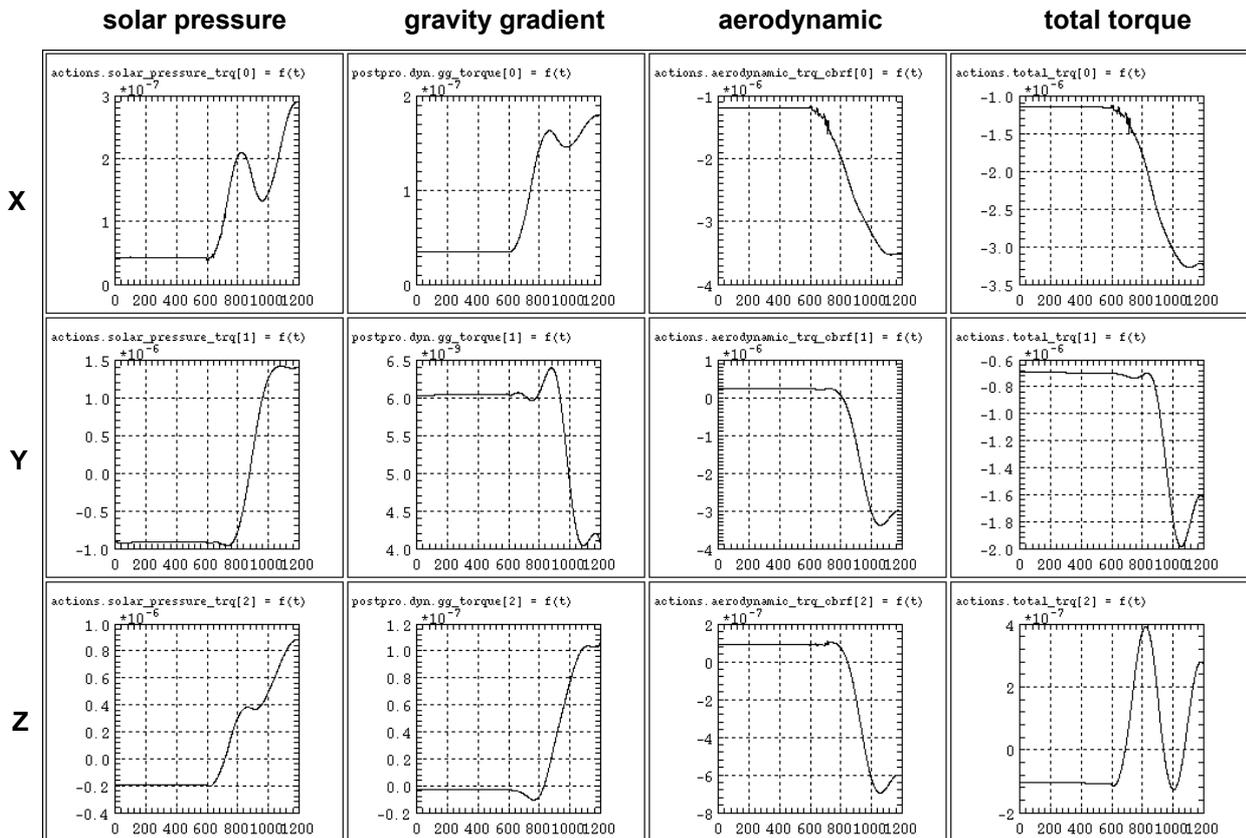


Figure 13.4-3: External disturbance torques (N.m)

As stated before, the calibration algorithm is based on the assumption that the external disturbance torques can be neglected in the conservation of the spacecraft angular momentum.

From [Figure 13.4-3](#) the disturbance torque levels encountered are lower than 3.5×10^{-6} N.m which is negligible owing at their impact on the reaction wheels angular momentum variation during the slew duration. It therefore justifies the assumption made in the calibration algorithm using the outputs from the simulation.

13.4.3.4 SA pointing error and SA commanded speed levels

In the first column of the figure below are given the speed level commanded to SADM1: the first line gives the sign of the commanded speed level, the second line, the speed level itself, and the third line, the number of commanded motor micro steps.

The second column shows the first (+Y) and second (-Y) SA angular positions (rad), while the third line gives the pointing error (rad) between the SA normal and the Sun direction projection on the plane normal to the SA Y-axis for the fictitious SAS.

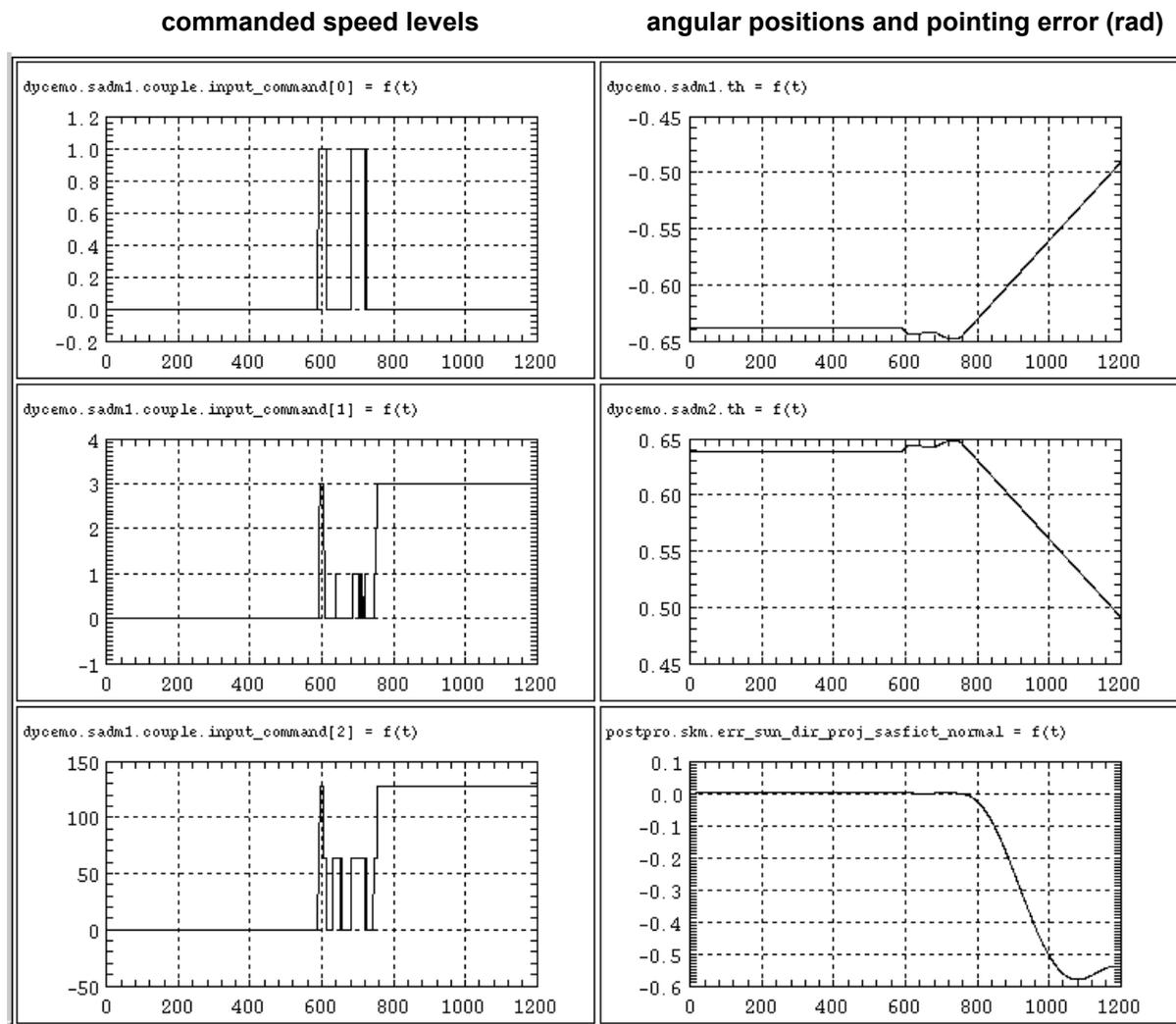


Figure 13.4-4: SA commanded speed levels and SA pointing errors

The SA pointing error (Column 2, line 3) reaches a value close to 35°, because of the SADE speed limitation imposed by TC, which prevents the SADM from tracking the Sun. Note that the Linear Range Sun Pointing alarm has been disabled in the simulation in order to prevent an FDIR alarm triggering in this situation.

13.4.4 Sensitivity study

A sensitivity study has been carried out in order to assess the major contributors to the spacecraft inertia estimation performance, especially for what concerns the influence of :

- the rate levels reached during the slew
- the SADE speed oscillations
- the gyros configuration noise transmission factor

Other potential error sources like gyros / reaction wheel misalignments and reaction wheel speeds measurement errors have been assessed to be negligible and are not described hereafter.

Table 13.4-2 provides the S/C inertia estimation errors obtained using TM data provided by different simulation runs and compare them to the reference case (denoted **Run 1**) described in section 13.4.2 and 13.4.3.

	Relative estimation error on J_{11}	Relative estimation error on J_{22}	Relative estimation error on J_{33}	Absolute estimation error on J_{12} (kg.m ²)	Absolute estimation error on J_{13} (kg.m ²)	Absolute estimation error on J_{23} (kg.m ²)
Run 1: reference run	0.1%	0.7 %	0.3 %	5.2	3.0	2.6
Run 2: separated slews & increased X/Z S/C rates	0.5 %	0.5 %	0.6 %	4.5	11.8	12.4
Run 3: SADE OFF	0.1 %	0.2 %	0.3 %	3.6	5.0	5.6
Run 4: SADE ON with no speed limitation	0.1 %	4.8 %	0.5 %	6	4.2	5.3
Run 5: using a worst-case gyro configuration	1 %	1.1 %	1.4 %	78.7	6.6	8.1
Run 6: lower slew rate on S/C Y-axis	0.1 %	1.5 %	0.4 %	8.2	19.7	10.3

Table 13.4-2: estimation results comparison

The attitude slew simulation runs listed in the first column of Table 13.4-2 above have the following characteristics:

- **Run 2:** the attitude profile has been generated so that the 3 S/C axis slews happen one after the other, instead of being superposed in the reference case: X-axis slew first, then Y-axis slew and

finally Z-axis slew, with an inertial pointing phase between each manoeuvre. Due to guidance profile generation needs, the total duration of these manoeuvres has been enlarged to 4415 sec and the maximum S/C rates have been decreased to 10^{-3} rad/s instead of $1.5 \cdot 10^{-3}$ rad/s

- **Run 3:** this manoeuvre is exactly the same as the reference one, but the SADE is set OFF at GSP entering.
- **Run 4:** this manoeuvre is exactly the same as the reference one, but no TC is sent to limit the maximum SADE speed level.
- **Run 5:** this manoeuvre is exactly the same as the reference one, but the gyro configuration selected is changed to the one numbered 13 in Table 1.2-2 of ref.[8], whose gyro triplet noise amplification factor is 16 instead of 1 for the configuration selected in the reference manoeuvre (numbered 1 in Table 1.2-2 of ref.[8]).
- **Run 6:** this manoeuvre is exactly the same as the reference one, but the Y-axis slew profile has been generated using a maximum rate value of $1.5 \cdot 10^{-3}$ rad/s instead of $3.5 \cdot 10^{-3}$ rad/s in the reference case.

From the results of [Table 13.4-2](#) several conclusions can be drawn on the influence of the attitude slew simulation parameters:

- There is no particular need for separating the 3 axial slews as the order of magnitude of the accuracy obtained on the estimated inertia coefficients is the same with separated or superposed slews. Note anyway that the S/C rates are lower in Run 2 than in the reference run: this difference probably explains the worse values obtained for off-diagonal coefficients with the separated slews.
- When the SADE is set OFF, the results are improved thanks to gyros measurements of better quality, but the SA experience a larger Sun pointing error. As a result it is preferred to limit the maximum SADE speed value to level 4 so that SADE oscillations are minimised. Indeed results from Run 4, where the SADE is in autonomous tracking with no speed limitation, exhibit a substantial degradation of the spacecraft Y-axis inertia estimation performance, because of polluted gyros measurements about the Y-axis. The 3% accuracy requirement is even exceeded.
- The Gyros configuration noise transmission also degrades the estimation accuracy, as shown by results from run 5, especially off-diagonal inertia terms estimations. It will therefore be recommended to use the best gyro configuration still available which minimizes the noise amplification factor. Note anyway that even with a worst-case gyro configuration selected as in run 5, the accuracy obtained fulfils the 3% requirement value on the S/C inertia matrix diagonal coefficients.
- Finally, the influence of lower slew rate values increase the estimation error, as shown by results from run 6 (with a decreased slewing rate about S/C Y axis), and Run 2 (with decreased slewing rates about X/Z axes). It is therefore recommended to achieve a slew manoeuvre with rate levels reaching $1.5 \cdot 10^{-3}$ rad/s about X/Z axes and $3.5 \cdot 10^{-3}$ rad/s about the Y axis.

14 SOLAR ARRAYS FLEXIBLE MODES CHARACTERISTICS IN-FLIGHT CALIBRATION

14.1 CONTEXT AND PURPOSE

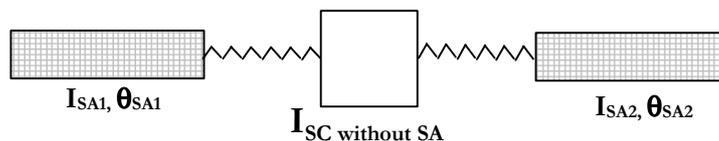
This on-ground processing describes how to calibrate the solar arrays flexible modes characteristics, i.e. the free-free frequencies and the free-free damping ratios. There are four main flexible modes (see ref. [4]) likely to be calibrated: the 1st out-of-plane, the 2nd out-of-plane, the 1st in-plane and the 1st torsion flexible modes. This calibration is of interest, because if these free-free frequencies turned out to be quite different from the expected ones, there could be an impact on the controller tuning and on the performances of all AOCMS modes. Furthermore this calibration can help to confirm that the solar arrays deployment has been well performed with no hinge failure.

In section 14.2, the solar arrays flexible modes characteristics (free-free frequencies and damping ratios) are theoretically predicted. The ground processing used to calibrate the free-free frequencies is described in section 14.3. The ground processing used to estimate the free-free damping ratios is described in section 14.4. In section 14.5, typical results of these two ground processings are shown.

14.2 THEORETICAL PREDICTION OF THE SOLAR ARRAYS FLEXIBLE MODE CHARACTERISTICS

14.2.1 Prediction of the nominal free-free frequencies

The following spacecraft configuration is considered:



For a given solar array flexible mode, the two solar arrays are oscillating in phase ($\theta_{SA1} = \theta_{SA2} = \theta_{SA}$) at the same free-free frequency, and the two solar array inertias can be reasonably considered as identical ($I_{SA1} = I_{SA2} = I_{SA}$). So the free-free frequency $f_{free-free}$ of the solar array flexible mode can be computed as follows, assuming both solar arrays have the same Cantilever frequency f_c for each flexible mode (the case where the two solar arrays have different Cantilever frequencies is treated in section 14.2.3):

$$f_{free-free} = \lambda \cdot f_c$$

where $\lambda = \sqrt{1 + \frac{2 \cdot (\epsilon \sqrt{I_{SA}})^2}{I_{SC\ without\ SA}}}$ is a ratio specific to each SA flexible mode

and $(\epsilon\sqrt{I_{SA}})$ is the SA inertia modal participation of the flexible mode

$I_{SC\ without\ SA}$ is the inertia of the spacecraft without the solar arrays (i.e. only the central body and the liquids)

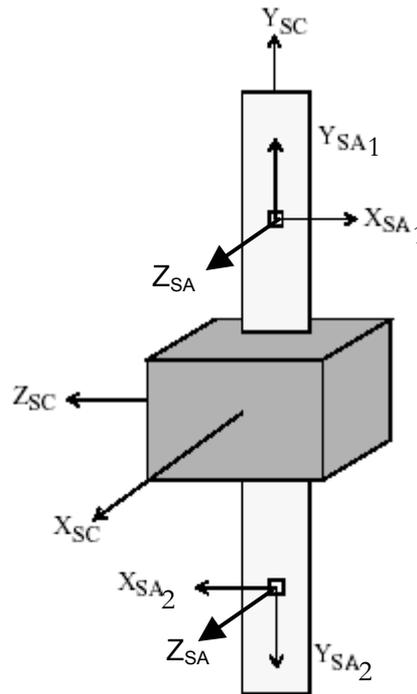
Both inertias are expressed wrt the spacecraft center of mass in the spacecraft reference frame.

The nominal Cantilever frequencies shown in the following table are taken from ref. [12] (they are not up-to-date in ref. [4] for the two out-of-plane modes). The matrix $L_{A/SA}$ of the modal participation factors are taken from §4.2.5 in ref. [4] and are expressed in the SA reference frame wrt the SA attachment point.

Flexible mode	Nominal Cantilever frequency f_c (Hz)	$\epsilon\sqrt{M_{SA}^X}$	$\epsilon\sqrt{M_{SA}^Y}$	$\epsilon\sqrt{M_{SA}^Z}$	$\epsilon\sqrt{I_{SA}^X}$	$\epsilon\sqrt{I_{SA}^Y}$	$\epsilon\sqrt{I_{SA}^Z}$
1 st Out-of-plane	0.064	0	-0.002	7.311	80.055	-0.045	0.004
2 nd Out-of-plane	0.42	-0.174	0.006	-3.934	-11.068	-0.044	1.777
1 st In-plane	0.296	7.942	-0.001	-0.086	-0.239	0.022	-81.144
1 st Torsion	0.55	0.09	0	0.022	0.042	5.218	-0.882

Table 14.2-1 Nominal Cantilever frequency and modal participation factors of each SA flexible mode in the SA reference frame wrt the SA attachment point

When the solar arrays are in canonical position (i.e. the spacecraft X axis being perpendicular to the solar arrays surface), the solar arrays axes are as shown on the following figure:



**Figure 14.2-1 Solar array axes with respect to the spacecraft axes
 ("1" for Y+ SA and "2" for Y- SA)**

So the spacecraft axes can be expressed wrt the Y+ SA axes as follows:

$$\begin{cases} X_{SC} = Z_{SA} \\ Y_{SC} = Y_{SA} \\ Z_{SC} = -X_{SA} \end{cases}$$

A pure axial torque about the spacecraft X axis will excite the solar array in-plane flexible mode, a pure axial torque about the spacecraft Y axis will excite the solar array torsion flexible mode and a pure axial torque about the spacecraft Z axis will excite the solar array out-of-plane flexible mode.

The transformation matrix from the Y+ SA reference frame to the SC reference frame is:

$$M_{SA \rightarrow SC} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

So the matrix of the Y+ SA modal participation factors can be expressed in the spacecraft reference frame wrt the Y+ SA attachment point as follows:

$$L_{A/SC} = L_{A/SA} \cdot \begin{pmatrix} M_{SA \rightarrow SC} & 0 \\ 0 & M_{SA \rightarrow SC} \end{pmatrix}$$

$$L_{A/SC} = \begin{pmatrix} -7.311 & -0.002 & 0 & -0.004 & -0.045 & 80.055 \\ 3.934 & 0.006 & -0.174 & -1.777 & -0.044 & -11.068 \\ 0.086 & -0.001 & 7.942 & 81.144 & 0.022 & -0.239 \\ -0.022 & 0 & 0.09 & 0.882 & 5.218 & 0.042 \end{pmatrix}$$

The Y+ SA attachment point A mean coordinates wrt the spacecraft mechanical reference frame with its origin at the launcher interface plane are (see §4.2.5 in ref. [4]):

$$\begin{cases} X_A = 0 \\ Y_A = 1.0645 \text{ m} \\ Z_A = 1.32113 \text{ m} \end{cases}$$

The spacecraft center of mass G mean coordinates wrt the spacecraft mechanical reference frame are at BOL (see §4.2.7.4 in ref. [4]):

$$\begin{cases} X_G = 0.028 \text{ m} \\ Y_G = 0.009 \text{ m} \\ Z_G = 1.2545 \text{ m} \end{cases}$$

The SA attachment point – spacecraft center of mass vector \overrightarrow{AG} coordinates are therefore:

$$\begin{cases} X = X_G - X_A = 0.028 \text{ m} \\ Y = Y_G - Y_A = -1.0555 \text{ m} \\ Z = Z_G - Z_A = -0.0666 \text{ m} \end{cases}$$

The matrix of the Y+ SA modal participation factors can be expressed in the spacecraft reference frame wrt the spacecraft center of mass as follows:

$$L_{G/SC} = L_{A/SC} \cdot \Delta$$

where $\Delta =$

$$\begin{pmatrix} 1 & 0 & 0 & 0 & -Z & Y \\ 0 & 1 & 0 & Z & 0 & -X \\ 0 & 0 & 1 & -Y & X & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

The modal participation factors $L_{G,SC}$ of each flexible mode of the Y+ SA in the spacecraft reference frame with respect to the spacecraft center of mass are shown in the following table (in bold are the numerical values that will be used in the further computations):

Flexible mode	$\epsilon\sqrt{M_{SA/G}^X}$	$\epsilon\sqrt{M_{SA/G}^Y}$	$\epsilon\sqrt{M_{SA/G}^Z}$	$\epsilon\sqrt{I_{SA/G}^X}$	$\epsilon\sqrt{I_{SA/G}^Y}$	$\epsilon\sqrt{I_{SA/G}^Z}$
1st Out-of-plane	-7.311	-0.002	0	-0.0039	-0.5321	87.7718
2nd Out-of-plane	3.934	0.006	-0.174	-1.9611	0.3013	-15.2205
1st In-plane	0.086	-0.001	7.942	89.5268	0.2501	-0.3297
1st Torsion	-0.022	0	0.09	0.977	5.2191	0.0652

Table 14.2-2 Modal participation factors of each SA flexible mode in the SC reference frame wrt the SC center of mass

The inertias of the spacecraft without the solar arrays are taken here equal to the mean inertia values of the stowed spacecraft at BOL (see §4.2.7.1 in ref. [4]):

$$I_{SC\ without\ SA} = \begin{pmatrix} 2863.5 \\ 2917.7 \\ 1906.6 \end{pmatrix} \text{ kg.m}^2 \text{ (these values correspond to (max + min) / 2)}$$

A more detailed analysis can be done at system level by considering the spacecraft central body (including liquids).

The SA being in their canonical positions, the ratio corresponding to the first out-of-plane flexible mode is:

$$\lambda_{z_1} = \sqrt{1 + \frac{2 \cdot \left(\varepsilon \sqrt{I_{SA/G}^Z} \right)_{1^{st} \text{ out-of-plane}}^2}{I_{SC \text{ without SA}}^Z}} = \sqrt{1 + \frac{2 \cdot (87.7718)^2}{1906.6}} = 3.0135$$

The ratio corresponding to the second out-of-plane flexible mode is:

$$\lambda_{z_2} = \sqrt{1 + \frac{2 \cdot \left(\varepsilon \sqrt{I_{SA/G}^Z} \right)_{2^{nd} \text{ out-of-plane}}^2}{I_{SC \text{ without SA}}^Z}} = \sqrt{1 + \frac{2 \cdot (-15.2205)^2}{1906.6}} = 1.1149$$

The ratio corresponding to the first in-plane flexible mode is:

$$\lambda_X = \sqrt{1 + \frac{2 \cdot \left(\varepsilon \sqrt{I_{SA/G}^X} \right)_{in-plane}^2}{I_{SC \text{ without SA}}^X}} = \sqrt{1 + \frac{2 \cdot (89.5268)^2}{2863.5}} = 2.5687$$

The ratio corresponding to the first torsion flexible mode is:

$$\lambda_Y = \sqrt{1 + \frac{2 \cdot \left(\varepsilon \sqrt{I_{SA/G}^Y} \right)_{torsion}^2}{I_{SC \text{ without SA}}^Y}} = \sqrt{1 + \frac{2 \cdot (5.2191)^2}{2917.7}} = 1.0093$$

Considering these nominal λ ratios and Cantilever frequencies, the nominal free-free frequencies are:

Flexible mode	Free-free frequency $f_{free-free}^{nom}$
1st Out-of-plane	$0.064 \times 3.0135 = 0.19 \text{ Hz}$
2nd Out-of-plane	$0.42 \times 1.1149 = 0.47 \text{ Hz}$
1st In-plane	$0.296 \times 2.5687 = 0.76 \text{ Hz}$
1st Torsion	$0.55 \times 1.0093 = 0.555 \text{ Hz}$

Table 14.2-3 Expected nominal free-free frequencies assuming the two solar arrays have the same Cantilever frequency

14.2.2 Influence of the Cantilever frequencies and inertia dispersions on the theoretical free-free frequencies

In the computations performed in §14.2.1, nominal Cantilever frequencies and mean inertia values have been considered.

But, as indicated in §4.2.5 of ref. [4], the dispersions on the nominal Cantilever frequencies (shown in [Table 14.2-1](#)) are up to ±15% for the 1st in-plane flexible mode and ±7.5% for the other flexible modes. So the following minimum and maximum Cantilever frequencies shall be taken into account:

Flexible mode	Min. Cantilever frequency f_c^{\min}	Max. Cantilever frequency f_c^{\max}
1 st Out-of-plane	0.059 Hz	0.069 Hz
2 nd Out-of-plane	0.39 Hz	0.45 Hz
1 st In-plane	0.25 Hz	0.34 Hz
1 st Torsion	0.51 Hz	0.59 Hz

Table 14.2-4 Minimum and maximum Cantilever frequencies

The inertia of the spacecraft without the solar arrays are taken equal to the inertia values of the stowed spacecraft at BOL. Mean values have been taken into account in the prediction of the nominal free-free frequencies, but in fact these inertia values can vary (see §4.2.7.1 in ref. [4]):

$$\text{from } I_{SC\text{without } SA}^{\min} = \begin{pmatrix} 2317.04 \\ 2365.1 \\ 1701.3 \end{pmatrix} \text{ kg.m}^2 \text{ to } I_{SC\text{without } SA}^{\max} = \begin{pmatrix} 3409.97 \\ 3470.38 \\ 2111.98 \end{pmatrix} \text{ kg.m}^2.$$

With these minimum and maximum inertia values (and with the same modal participation factors as in [Table 14.2-2](#)), the following minimum and maximum λ ratios are obtained:

Flexible mode	Minimum ratio λ^{\min}	Maximum ratio λ^{\max}
1 st Out-of-plane	2.8802	3.1712
2 nd Out-of-plane	1.1043	1.1280
1 st In-plane	2.3877	2.8140
1 st Torsion	1.0078	1.0114

Table 14.2-5 Minimum and maximum λ ratios

Taking into account these extreme values for the Cantilever frequencies (see [Table 14.2-4](#)) and for the λ ratios (see [Table 14.2-5](#)), the following minimum and maximum free-free frequencies can be reached:

Flexible mode	Min. free-free frequency $f_{free-free}^{\min} = \lambda^{\min} \cdot f_c^{\min}$	Max. free-free frequency $f_{free-free}^{\max} = \lambda^{\max} \cdot f_c^{\max}$
1st Out-of-plane	0.17 Hz	0.22 Hz
2nd Out-of-plane	0.43 Hz	0.51 Hz
1st In-plane	0.60 Hz	0.96 Hz
1st Torsion	0.51 Hz	0.60 Hz

Table 14.2-6 Expected minimum and maximum free-free frequencies taking into account the dispersions on the Cantilever frequencies and on the inertia values

The difference between the minimum and the maximum free-free frequency is particularly high for the in-plane flexible mode, where a $\pm 15\%$ dispersion on the Cantilever frequency has been taken into account.

14.2.3 Influence of the difference between the two solar arrays Cantilever frequencies on the theoretical free-free frequencies

In the computations performed in §14.2.1, it has been assumed that the two solar arrays have the same Cantilever frequency ($f_{c_1} = f_{c_2}$). However, as indicated in §4.2.5 of ref. [4], the frequency difference between the two solar array wings can reach up to $\alpha = 7.5\%$ for the 1st in plane flexible mode and up to $\alpha = 5\%$ for the other flexible modes. When the two solar arrays do not have the same Cantilever frequency ($f_{c_1} \neq f_{c_2}$), they both oscillate at the same free-free frequency equal to:

$$f_{free-free} = \lambda \cdot \sqrt{f_{c_1} \cdot f_{c_2}}$$

The two extreme cases are thus reached when:

$$f_{c_1} = f_c^{nom} \text{ and } f_{c_2} = (1 - \alpha) \cdot f_c^{nom} \Rightarrow f_{free-free}^{\min} = \sqrt{1 - \alpha} \cdot (\lambda \cdot f_c^{nom}) = \sqrt{1 - \alpha} \cdot f_{free-free}^{nom}$$

or $f_{c_1} = f_c^{nom} \text{ and } f_{c_2} = (1 + \alpha) \cdot f_c^{nom} \Rightarrow f_{free-free}^{\max} = \sqrt{1 + \alpha} \cdot (\lambda \cdot f_c^{nom}) = \sqrt{1 + \alpha} \cdot f_{free-free}^{nom}$

So the nominal free-free frequencies $f_{free-free}^{nom}$ (shown in [Table 14.2-3](#)) can be multiplied by a factor going from $\sqrt{1-\alpha}$ to $\sqrt{1+\alpha}$. Considering these extreme factors, the minimum and maximum free-free frequencies are:

Flexible mode	Frequency difference α	Min. free-free frequency $f_{free-free}^{min}$	Max. free-free frequency $f_{free-free}^{max}$
1st Out-of-plane	7.5 %	0.18 Hz	0.20 Hz
2nd Out-of-plane	5 %	0.46 Hz	0.48 Hz
1st In-plane	5 %	0.74 Hz	0.78 Hz
1st Torsion	5 %	0.54 Hz	0.57 Hz

Table 14.2-7 Expected minimum and maximum free-free frequencies taking into account the maximum difference between the two solar arrays cantilever frequencies

This table shows that the difference between the two solar arrays Cantilever frequencies only have a slight influence on the global free-free frequency.

14.2.4 Prediction of the nominal free-free flexible mode damping ratios

The free-free damping ratio $\xi_{free-free}$ of the solar array flexible mode is related to the Cantilever one ξ_c by the same λ factor as for the frequency:

$$\xi_{free-free} = \lambda \cdot \xi_c$$

The Cantilever damping ratio is expected to be $\xi_c = 2.5e-3$ for all solar arrays flexible modes (see §4.2.5 in ref. [4]). So applying the nominal λ factors calculated in §14.2.1, the following free-free damping ratios are expected:

Flexible mode	Free-free damping ratio $\xi_{free-free}^{th}$
1st Out-of-plane	7.5e-3
2nd Out-of-plane	2.8e-3
1st In-plane	6.4e-3
1st Torsion	2.5e-3

Table 14.2-8 Expected nominal free-free damping ratios

14.3 SOLAR ARRAYS FLEXIBLE MODES FREE-FREE FREQUENCIES CALIBRATION PROCESSING

14.3.1 Principle

The free-free frequency calibration requires the excitation of the solar array flexible modes. The solar arrays are assumed to be in their canonical position (i.e. the spacecraft X axis perpendicular to the solar arrays surface). Hence, in an inertial Sun pointing phase (SHM/EPP, NM/GSEP), since the automatic guidance aims at pointing the X axis towards the Sun (during the commissioning phase), the solar arrays will be facing the Sun in their canonical position.

A thrust pulse has to be fired (also see [Figure 14.2-1](#)):

- about the spacecraft Z axis (i.e. about the solar array X axis) to excite the two out-of-plane flexible modes
- about the spacecraft X axis (i.e. about the solar array Z axis) to excite the 1st in-plane flexible mode
- about the spacecraft Y axis (i.e. about the solar array Y axis) to excite the 1st torsion flexible mode.

The calibration is recommended to take place during a wheel-off-loading in an inertial Sun pointing phase (i.e. SHM/EPP or NM/GSEP, with the spacecraft X axis pointed towards the Sun): a wheel angular momentum variation can be commanded to create a pure torque about the chosen spacecraft axis. Thus the thrust pulse fired to compensate the wheel angular momentum variation will not create any off-pointing angles.

Assuming the chosen inertial Sun pointing phase is the NM/GSEP, the calibration of each solar array flexible mode shall be performed as shown in the following figure:

	Guidance	Steps
NM/GSEP WOLP NM/GSEP	(Autonomous guidance) Inertial Sun pointing	Acquisition of the estimated wheel angular momentum from the telemetry data Computation and sending of the wheel off-loading reference angular momentum to create a pure torque Triggering of the wheel off-loading phase
		Acquisition of the telemetry data required by the calibration Return in NM/GSEP
	(Autonomous guidance) Inertial Sun pointing	On-ground processing to estimate the solar array flexible mode free-free frequency

Figure 14.3-1 Main steps in the calibration of a solar array flexible mode

These steps are more detailed in the following paragraphs, and are also presented in procedure FCP-AC0600 “In-flight calibration of solar arrays flexible modes characteristics” (see ref. [7], Vol. 2, Ch. 7).

Acquisition of the estimated wheel angular momentum

First the ground shall acquire the estimated wheel angular momentum in the reaction wheel reference frame $\overline{H}_{RW\ init}$ (available through TM NACW0G0*, where * stands for 6, I or U for respectively RWA A, B or C).

Computation and sending of the wheel off-loading reference angular momentum

Then it shall compute the reaction wheel off-loading reference angular momentum in order to create a pure torque about the spacecraft X, Y or Z axis.

A constraint applies on the reaction wheel angular momentum variation: it shall be lower or equal than 2 Nms for each wheel (in the reaction wheel reference frame) in order to ensure that only one wheel off-loading cycle is performed, and thus only one thruster actuation is fired.

The reaction wheel angular momentum variation in the reaction wheel reference frame can be computed as follows:

$$\overline{\Delta H}_{RW} = M_{SC \rightarrow RW} \cdot \overline{\Delta H}_{SC}$$

where $M_{SC \rightarrow RW} = \begin{pmatrix} 0 & 1 & 0.82 \\ 0.82 & -1 & 0 \\ 0.82 & 0 & -0.82 \end{pmatrix}$ is the rotation matrix from the spacecraft reference frame to the reaction wheel reference frame (composed of RWA A, B and C axes)

$\overrightarrow{\Delta H_{SC}}$ is the reaction wheel angular momentum variation in the spacecraft reference frame

For the calibration of the two out-of-plane flexible modes, the reaction wheel angular momentum variation in the spacecraft reference frame can be taken as:

$$\overrightarrow{\Delta H_{SC}} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \text{ Nms.}$$

With this value, the reaction wheel angular momentum variation in the reaction wheel reference frame to create a pure torque about the spacecraft Z axis is:

$$\overrightarrow{\Delta H_{RW}} = \begin{pmatrix} 0.82 \\ 0 \\ -0.82 \end{pmatrix} \text{ Nms, which fulfill the 2 Nms constraint for each wheel.}$$

For the calibration of the 1st in-plane flexible mode, the reaction wheel angular momentum variation in the spacecraft reference frame can be taken as:

$$\overrightarrow{\Delta H_{SC}} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \text{ Nms.}$$

With this value, the reaction wheel angular momentum variation in the reaction wheel reference frame to create a pure torque about the spacecraft X axis is:

$$\overrightarrow{\Delta H_{RW}} = \begin{pmatrix} 0 \\ 0.82 \\ 0.82 \end{pmatrix} \text{ Nms, which fulfill the 2 Nms constraint for each wheel.}$$

For the calibration of the 1st torsion flexible mode, the reaction wheel angular momentum variation in the spacecraft reference frame shall be taken as its maximum possible value. Otherwise it is hard to excite the torsion flexible mode (with low inertia engaged, about the spacecraft Y axis) and thus hard to observe the rate oscillations which have the same order of magnitude as the gyro noise and gyro lsb. So the reaction wheel angular momentum variation in the spacecraft reference frame is taken as:

$$\overline{\Delta H_{SC}} = \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix} \text{ Nms.}$$

With this value, the reaction wheel angular momentum variation in the reaction wheel reference frame to create a pure torque about the spacecraft Y axis corresponds to the maximum possible one:

$$\overline{\Delta H_{RW}} = \begin{pmatrix} 2 \\ -2 \\ 0 \end{pmatrix} \text{ Nms, which fulfill the 2 Nms constraint for each wheel.}$$

For the calibration of all flexible modes, the off-loading reference angular momentum to be sent through TC ZAC200* (where * stands for 35, 38 or 41 for respectively RWA A, B or C) "RWA_X.set_cmd_ang_mom" (X = A, B or C) to create a pure torque is:

$$\overline{H_{RWref}} = \overline{H_{RWinit}} + \overline{\Delta H_{RW}}$$

Triggering of the reaction wheel off-loading phase

The transition to the reaction wheel off-loading phase is performed upon reception of TC ZAC20188 "WOL_MGR.switch_on". It is recommended to switch-on the RCS ORB before entering the WOL in order to fasten the transition to the WOL phase.

Acquisition of the telemetry data required by the calibration

The following telemetry data are to be acquired during a duration T^{calib} (determined hereafter) and at a TM storage frequency of 8 Hz:

- the on-board time (see on-ground processing described in section 15)
- the spacecraft angular rate measured by the IMP ORB about the spacecraft Z axis (TM NACW0P02) for the two out-of-plane flexible modes calibration
- the spacecraft angular rate measured by the IMP ORB about the spacecraft X axis (TM NACW0P00) for the 1st in-plane flexible mode calibration
- the spacecraft angular rate measured by the IMP ORB about the spacecraft Y axis (TM NACW0P01) for the 1st torsion flexible mode calibration

The minimum free-free frequency to be characterized is the first out-of-plane one: 0.17 Hz (see [Table 14.2-6](#)), with a frequency accuracy goal better than 7.5%, corresponding to the dispersion on the Cantilever frequency (see §4.2.5 in ref. [4]).

So the required frequency resolution is: $f^{accuracy} < 7.5\% \times 0.17 = 0.01275$ Hz. This frequency resolution is directly linked to the calibration duration T^{calib} , i.e. the duration during which the spacecraft angular rate is recorded.

The calibration duration has to be : $T^{calib} = \frac{1}{f^{accuracy}} > \frac{1}{0.01275} = 78.4s$.

A calibration duration of $T^{calib} = 100s$, which corresponds to a frequency resolution of $f^{accuracy} = 0.01Hz$, is sufficient (corresponding to a maximum relative accuracy of 6%). Note that all frequencies lower than $2 \cdot f^{accuracy} = 0.02$ Hz cannot be characterized.

The telemetry data shall be acquired between t^{start} and $(t^{start} + T^{calib})$, with $t^{start} = t^{WOL} + \Delta t$

where: – t^{WOL} is the effective date of entry into the WOL phase

- Δt is a margin to ensure that the thruster actuation has already been performed and that the spacecraft rates have converged after the angular momentum variation: this margin can be $\Delta t = 10s$ for the 1st in-plane and the two out-of-plane flexible modes, and $\Delta t = 40s$ for the torsion flexible mode (where the inertia engaged –about the spacecraft Y axis- are lower, and thus the convergence is longer). Note that if the RCS ORB has not been switched-on before entering the WOL, an additional duration of 10s shall be taken into account in the margin Δt to ensure that all thrusters have been well switched-on.

The telemetry acquisition frequency is fixed to $f^{telemetry} = 8$ Hz, thus the maximum frequency that can be calibrated by the proposed ground processing is $f^{telemetry} / 2 = 4$ Hz (which is much higher than the maximum expected nominal free-free frequency, i.e. the 1st in-plane one : 0.76 Hz, see [Table 14.2-3](#)).

So all frequencies between 0.02 Hz and 4 Hz can be characterized with the ground processing.

Return in NM/GSEP

The transition from the reaction wheel off-loading phase to NM/GSEP is automatic, once the reaction wheel angular momentum of each wheel has reached its reference angular momentum.

On-ground processing

The ground processing used to calibrate the solar arrays free-free flexible mode frequencies is based on the discrete Fourier transform of the measured spacecraft angular rate. It is explained in the following paragraphs. Note that this ground processing can be used to calibrate the four “main” flexible modes (see ref. [4]): the 1st out-of-plane, the 2nd out-of-plane, the 1st in-plane and the 1st torsion flexible modes.

14.3.2 Inputs

The constant $T^{calib} = 100s$ has been determined in the previous section.

The inputs are the following telemetry data acquired at $f^{telemetry} = 8$ Hz between t^{start} and $(t^{start} + T^{calib})$:

- t = the on-board time (s)
- $\omega_{axis}^{SC}(t)$ = the spacecraft angular rate (rad/s) measured at the date t by the IMP ORB about the spacecraft axis (with axis = Z for out-of-plane, X for in-plane and Y for torsion)

14.3.3 Outputs

The output is the estimated free-free frequency of the flexible mode f^{est} (Hz), which is:

- a scalar for the in-plane and the torsion flexible mode, where only the 1st flexible mode is to be calibrated
- a 2-by-1 vector for the out-of-plane, where the 1st and the 2nd flexible modes are to be calibrated

14.3.4 Ground processing

Step 1: Initialisation

$N = 0$

$W_{sc} = 0$

Step 2 : Creation of the input vectors

The telemetry data must be valid without lack of measurements during the whole calibration period (T^{calib}). This can be checked visually by plotting the spacecraft measured rate as a function of the time

For t **from** t^{start} **to** $t^{start} + T^{calib}$

$W_{sc} = [W_{sc}; \omega_{axis}^{SC}(t)]$

$N = N + 1$

Endfor

At the end of the input vector creation, N is expected to be equal to $T^{calib} \times f^{telemetry} + 1$

N is the length of the input vector taken into account for the processing, i.e. W_{sc} is a N -by-1 vector, # which corresponds to the measured spacecraft rate about the chosen spacecraft axis

Delete the last point of W_sc if N is not an even number

If N is not an even number **then**

W_sc = W_sc(1:N-1)

N = N-1

Endif

Step 3 : Discrete Fourier transform of the spacecraft rate

The Discrete Fourier Transform of the vector X of length N is defined by the following expression :

for k from 1 to N,
$$DFT(X(k)) = \sum_{n=1}^N \left(X(n) * \exp(-j * 2 * \pi * (k-1) * \frac{(n-1)}{N}) \right).$$

real is the function that extracts the real part of a complex number.

abs is the function that provides the absolute value of a number.

exp is the function that provides the exponential of a number.

DFT_W_sc = abs(real(DFT(W_sc)))

DFT_W_sc is thus a N-by-1 vector

Step 4 : Creation of the associated frequency vector

The associated frequency N-by-1 vector (freq) has to contain first the positive frequencies (from 0 to $f^{\text{telemetry}}/2$) and then the negative ones (from $-f^{\text{telemetry}}/2$ to $-f^{\text{telemetry}}/N$) to correspond to DFT_W_sc

For k **from** 1 **to** $\frac{N}{2}$

$$freq(k) = f^{\text{telemetry}} * \frac{k-1}{N}$$

Endfor

For k **from** $\frac{N}{2} + 1$ **to** N

$$freq(k) = -f^{\text{telemetry}} * \frac{(N-k+1)}{N}$$

Endfor

Step 5 : Deletion of the frequencies that are inferior to twice the frequency resolution

Computation of the frequency resolution $f^{accuracy}$

$$f^{accuracy} = \frac{1}{T_{calib}}$$

For k **such as** $abs(freq(k)) \leq 2 * f^{accuracy}$

DFT_W_sc(k) = 0

Endfor

Step 6 : Estimation of the first flexible mode free-free frequency

$f^{est} = abs(freq(k))$ **such as** $DFT_W_sc(k) = \max_i(DFT_W_sc(i))$

Step 7 (for out-of-plane only) : Estimation of the second flexible mode free-free frequency

The 2nd out-of-plane free-free frequency is expected to be at least twice higher than the 1st one

(see theoretical prediction in section 14.2), so the 2nd out-of-plane free-free frequency corresponds to

the # local maximum in the frequency domain $\left[2 * f^{est}, \frac{f^{telemetry}}{2} \right]$

$f_2^{est} = abs(freq(k))$ **such as** $\begin{cases} abs(freq(k)) > 2 * f^{est} \\ DFT_W_sc(k) = \max_i(DFT_W_sc(i)) \end{cases}$

Create the vector containing the two free-free frequencies (1st and 2nd out-of-plane)

$$f^{est} = [f^{est}; f_2^{est}]$$

Step 8 : Manual verification of the flexible mode free-free frequency using the diagram of the discrete Fourier transform (DFT)

plot is a function that plots the diagram of the discrete Fourier transform (DFT) of the spacecraft rate

as a function of the frequency

plot(freq, DFT_W_sc)

Typical results of this ground processing are presented in section 14.5.1.

14.4 SOLAR ARRAYS FLEXIBLE MODE FREE-FREE DAMPING RATIO ESTIMATION PROCESSING

14.4.1 Principle

The theoretically predicted free-free damping ratios have been calculated in section 14.2.4. This calibration aims at confirming that the real free-free damping ratios have the same order of magnitude as the expected ones. It is not considered as mandatory to estimate these damping ratios, but it enables to complete the ground knowledge of the solar array flexible modes characteristics.

The principle presented below is as simple as possible and can allow the ground to estimate the free-free damping ratios of the 1st out-of-plane, 1st in-plane and 1st torsion flexible modes. The free-free damping ratio 2nd out-of-plane cannot be estimated by this processing, and is anyway not of special interest. The same telemetry data as for the frequencies, i.e. the measured spacecraft angular rates about a chosen axis during a wheel off-loading, are used to estimate the free-free damping ratio.

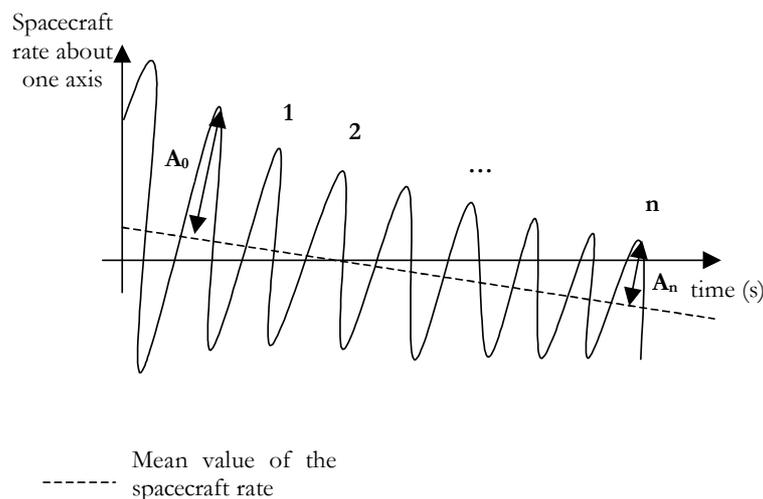


Figure 14.4-1 : Principle of the flexible mode damping ratio estimation

The estimated free-free damping ratio is equal to :

$$\zeta_{free-free}^{est} = \frac{\ln(A_0/A_n)}{2\pi.n}$$

with :

A_0 Amplitude of the rate oscillation number 0, which is equal to the difference in absolute value between the maximum rate and the mean rate.

A_n , Amplitude of the rate oscillation number n, which is equal to the difference in absolute value between the maximum rate and the mean rate.

$n + 1$ Number of rate oscillations taken into account to estimate the damping ratio

Here the mean spacecraft rate is assumed to be zero, because the spacecraft is in an inertial pointing phase and the spacecraft rates have had enough time to converge to zero when the pure axial torque is commanded.

The measured rate profiles are not as regular as on [Figure 14.4-1](#), so in order to have a more precise calculation, A_0 is taken as a mean value between the first three maximum amplitudes, and A_n is taken as a mean value between the last three maximum amplitudes.

The number n is related to the desired time domain ΔT and to the estimated free-free frequencies f^{est} (determined by the ground processing) as follows: $n = \text{round}(\Delta T \cdot f^{est})$ (where “round” is a function that rounds towards the nearest integer).

If the time domain is taken as a value such that the amplitude has been divided by two ($A_n/A_0 = 2$), it can be evaluated using the expected free-free damping ratios $\xi_{free-free}^{th}$ (determined in section 14.2.4) as follows:

$$\Delta T = \frac{\ln 2}{2\pi \cdot \xi_{free-free}^{th} \cdot f^{est}}$$

The calibration duration (during which the required telemetry data are acquired) used for the flexible modes free-free frequency calibration ($T^{calib} = 100s$, see section 14.3.1) is sufficiently long because the time domain ΔT is expected to be lower than 100s with some margin (see).

14.4.2 Inputs

The same telemetry data as for the flexible modes frequency calibration are required, i.e. the following data at $f^{telemetry} = 8$ Hz between t^{start} and $(t^{start} + T^{calib})$:

- t = the on-board time (s)
- $\omega_{axis}^{SC}(t)$ = the spacecraft angular rate (rad/s) measured at the date t by the IMP ORB about the spacecraft axis (with axis = Z for out-of-plane, X for in-plane and Y for torsion)

These additional inputs are also required for the estimation of the free-free damping ratio:

- $\xi_{free-free}^{th}$ = expected free-free damping ratio of the solar array flexible mode (no unit)
- f^{est} = estimated free-free frequency of the solar array flexible mode (Hz)

14.4.3 Outputs

The output is the estimated damping ratio $\xi_{free-free}^{est}$, which is a scalar (no unit).

14.4.4 Ground processing

Step 1: Initialisation

N = 0

W_sc = 0

Step 2 : Computation of the required time interval to have an amplitude divided by 2

$$\Delta T = \frac{\ln 2}{2\pi \cdot \xi_{free-free}^{th} \cdot f^{est}}$$

Computation of the end time

$$t^{end} = t^{start} + \Delta T$$

Step 3 : Creation of the input vectors

The telemetry data must be valid without lack of measurements during the whole time interval
 # domain (ΔT). This can be checked visually by plotting the spacecraft measured rate as a function
 # of the time

For t **from** t^{start} **to** t^{end}

$$W_sc = [W_sc ; \omega_{axis}^{SC}(t)]$$

N = N + 1

Endfor

At the end of the input vector creation, N is expected to be equal to $\Delta T \times f^{telemetry} + 1$

N is the length of the input vector taken into account for the processing, i.e. W_sc is a N-by-1 vector,
 # which corresponds to the measured spacecraft rate about the chosen spacecraft axis

Computation of the free-free period of the flexible mode

$$T^{est} = \frac{1}{f^{est}}$$

Step 4 : Computation of the initial and final maximum amplitudes

For k **from** 1 **to** 3

Calculation of the three first maximum rates (over the first three periods)

$$\omega_{\max}^{init}(k) = \max_{t \in [t^{start} + (k-1) \cdot T^{est}, t^{start} + k \cdot T^{est}]} (abs(W_{sc}))$$

Calculation of the last three maximum rates (over the last three periods)

$$\omega_{\max}^{final}(k) = \max_{t \in [t^{end} - (4-k) \cdot T^{est}, t^{end} - (3-k) \cdot T^{est}]} (abs(W_{sc}))$$

Endfor

Calculation of the initial maximum amplitude (the mean spacecraft rate being 0)

$$A_0 = mean(\omega_{\max}^{init})$$

This initial maximum amplitude is associated to the following date (mean value of
the $[t^{start}, t^{start} + 3 \cdot T^{est}]$ time interval):

$$\# t^{init} = t^{start} + \frac{3 \cdot T^{est}}{2}$$

Calculation of the final maximum amplitude (the mean spacecraft rate being 0)

$$A_n = mean(\omega_{\max}^{final})$$

This final maximum amplitude is associated to the following date (mean value of
the $[t^{end} - 3 \cdot T^{est}, t^{end}]$ time interval):

$$\# t^{final} = t^{end} - \frac{3 \cdot T^{est}}{2}$$

Step 5 : Computation of the number of oscillations

The time interval between A_0 and A_n is:

$$\# t^{final} - t^{init} = \Delta T - 3 \cdot T^{est}$$

The number of oscillations $n = round\left(\frac{t^{final} - t^{init}}{T^{est}}\right)$ can be computed as follows:

$$n = round\left(\frac{\Delta T}{T^{est}}\right) - 3$$

Step 6 : Computation of the free-free damping ratio

$$\xi_{free-free}^{est} = \frac{\ln(A_0/A_n)}{2.\pi.n}$$

Typical results of this ground processing are presented in section 14.5.2.

14.5 PROTOTYPING AND NUMERICAL VALIDATION

14.5.1 Simulation of the free-free flexible mode frequencies calibration by ground processing of the measured spacecraft angular rate

14.5.1.1 Introduction

The ground processing that evaluates the free-free flexible mode frequencies from the spacecraft angular rate is described in section 14.3.

Three simulations have been performed on the Rosace simulation tool with AOCMS prototype SW version VBVA4.1.1 (equivalent to EEPROM 1 SW version) and the corresponding Rosace version (V12.2_BVA4.1.1). One simulation is dedicated to each flexible mode: out-of-plane, in-plane and torsion. In every simulation, the TC file follows procedure FCP-AC0600 "In-flight calibration of solar arrays flexible modes characteristics" described in Vol. 2 Ch. 7 of ref. [7].

In every simulation, a wheel off-loading is commanded by TC at t=1200s in a Sun pointing SHM/EPP. Three reaction wheels are used: A, B and C. The solar arrays are in their canonical position, i.e. the spacecraft X axis is perpendicular to the solar arrays surface. For each flexible mode, both solar arrays have the same Cantilever frequency, corresponding to the values shown in [Table 14.2-1](#).

14.5.1.2 Evaluation of the first and second out-of-plane flexible mode free-free frequencies

The current AOCS phase is the SHM/EPP with the X axis pointed towards the Sun (1st line of [Figure 14.5-1](#)).

The initial angular momentum in the reaction wheel reference frame is (at t=1200s):

$$\overrightarrow{H_{RW_{init}}} = \begin{pmatrix} 8.74 \\ 11.02 \\ 11.41 \end{pmatrix} \text{ Nms (acquired by TM, see 3rd line of [Figure 14.5-1](#)).$$

A wheel off-loading is commanded at t=1200s with the following reference angular momentum:

$$\overrightarrow{H_{RW_{ref}}} = \overrightarrow{H_{RW_{init}}} + \overrightarrow{\Delta H_{RW}}$$

$$\text{where } \overrightarrow{\Delta H_{RW}} = \begin{pmatrix} 0.82 \\ 0 \\ -0.82 \end{pmatrix} \text{ Nms in order to create an angular momentum variation of 1 Nms about the}$$

spacecraft Z axis (see 4th line of [Figure 14.5-1](#)) so as to excite the solar array out-of-plane flexible modes.

$$\text{So } \overrightarrow{H_{RW_{ref}}} = \begin{pmatrix} 9.56 \\ 11.02 \\ 10.59 \end{pmatrix} \text{ Nms.}$$

It can be checked on [Figure 14.5-1](#) that the reference angular momentum is approximately reached (see 3rd line of [Figure 14.5-1](#)), and that this angular momentum variation creates a high thruster torque to be commanded about the spacecraft Z axis (see 2nd line of [Figure 14.5-1](#)).

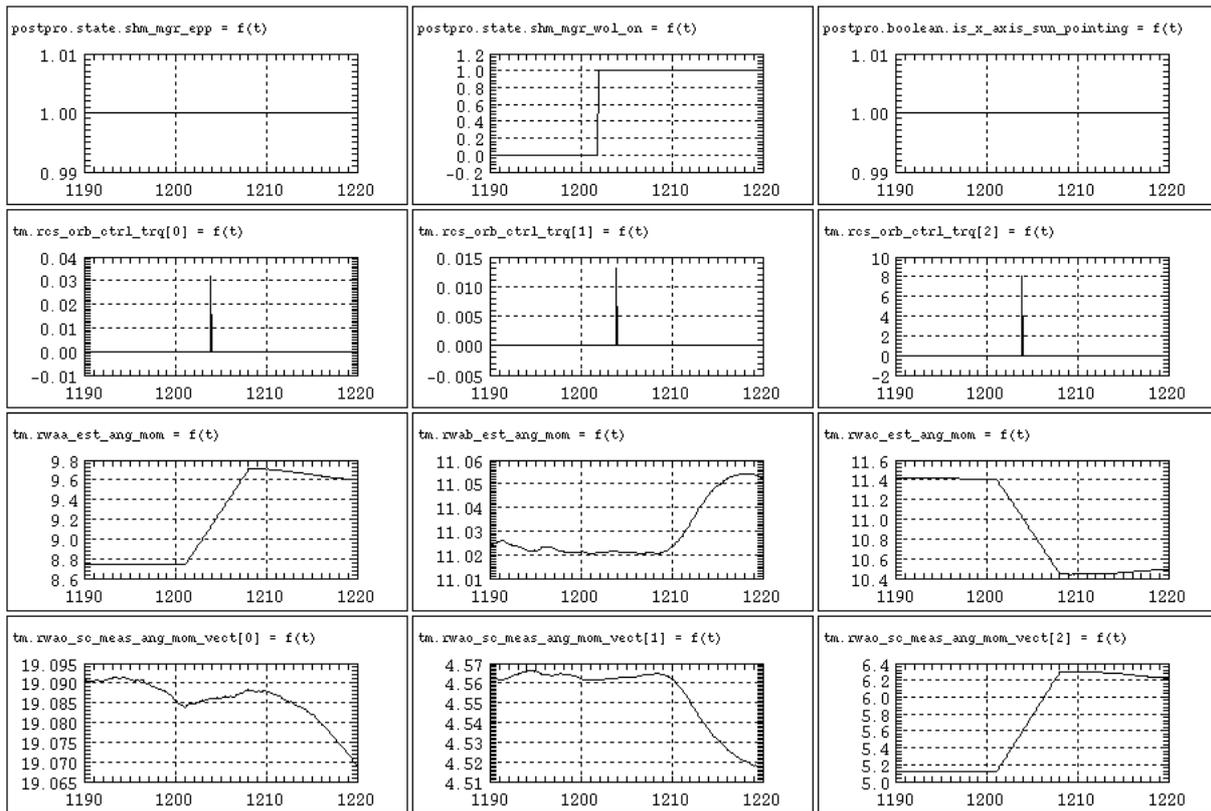


Figure 14.5-1 1st line: SHM/EPP flag, SHM/WOLP flag and Sun pointing flag
2nd line: Thruster commanded torques in the spacecraft reference frame (Nm)
3rd line: Estimated angular momentum in the reaction wheel reference frame (Nms)
4th line: Measured angular momentum in the spacecraft reference frame (Nms)

The spacecraft angular rate measured by the IMP ORB about the spacecraft Z axis (see 2nd line of [Figure 14.5-2](#)) is acquired between $t^{\text{start}}=1210\text{s}$ and $t=1310\text{s}$ (i.e. during $T^{\text{calib}} = 100\text{s}$, and at a frequency of 8 Hz). It is oscillating at the frequency of the 1st out-of-plane flexible mode (expected to be 0.19 Hz, see [Table 14.2-3](#), since the Cantilever frequency is equal to 0.064 Hz in the simulation).

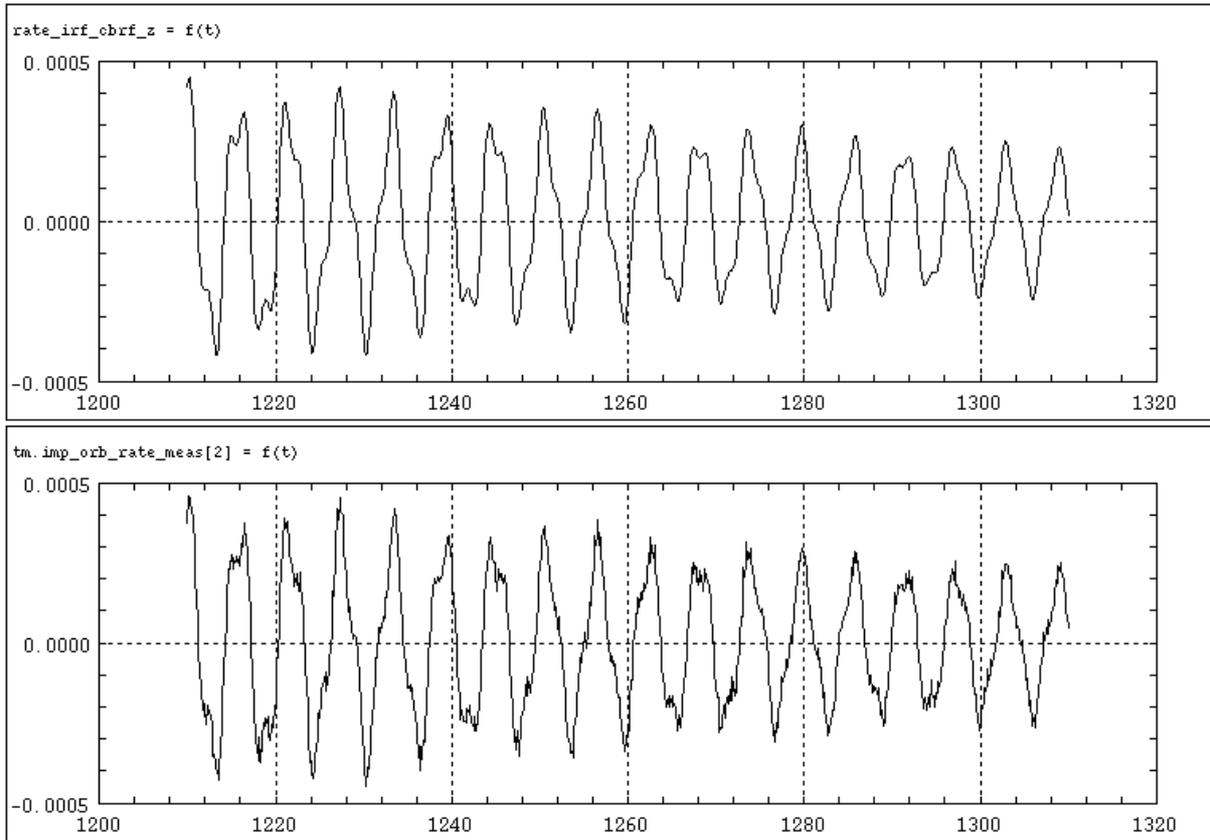


Figure 14.5-2 Real (1st line) and measured (2nd line) spacecraft angular rate about the spacecraft Z axis

The Discrete Fourier Transform of the measured spacecraft rate about the spacecraft Z axis is shown on [Figure 14.5-3](#). This DFT clearly shows that there are two main free-free frequencies, corresponding to the 1st and 2nd out-of-plane flexible modes frequencies (expected to be respectively 0.19 and 0.47 Hz, see [Table 14.2-3](#)).

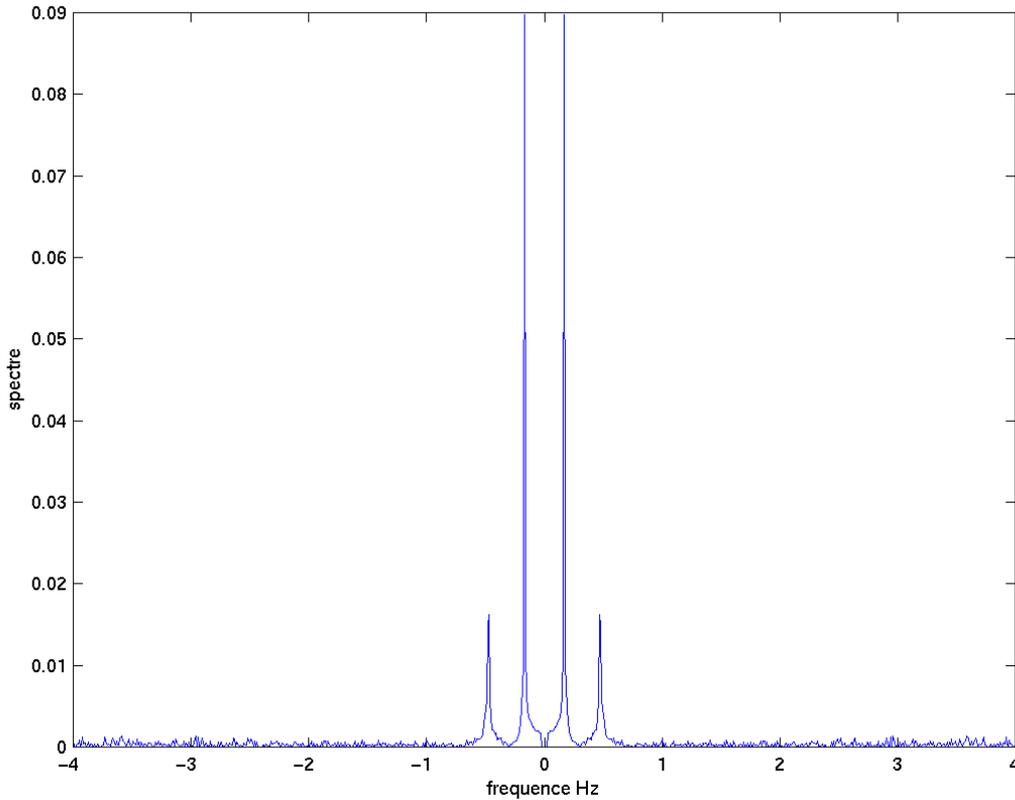


Figure 14.5-3 Discrete Fourier Transform of the spacecraft angular rate about the spacecraft Z axis

The ground processing of the spacecraft angular rate about the spacecraft Z axis (described in section 14.3.4) gives the following results: : $f^{est}=[0.17; 0.48]$ Hz for respectively the 1st and 2nd out-of-plane modes free-free frequencies, and are to be compared with the expected ones.

Flexible mode	Theoretically expected free-free frequency	Experimentally evaluated free-free frequency	Difference
1 st Out-of-plane	0.19 Hz	0.17 Hz	10.5%
2 nd Out-of-plane	0.47 Hz	0.48 Hz	2.1%

Table 14.5-1 Comparison between the theoretically expected and the experimentally evaluated free-free frequencies

The free-free frequency of the 1st out-of-plane flexible mode is 10.5% lower than the expected one, and the free-free frequency of the 2nd out-of-plane flexible mode is 2.1% higher than the expected one.

14.5.1.3 Evaluation of the first in-plane flexible mode free-free frequency

The current AOCS phase is the SHM/EPP with the X axis pointed towards the Sun (1st line of [Figure 14.5-4](#)).

The initial angular momentum in the reaction wheel reference frame is (at t=1200s):

$$\overrightarrow{H_{RW\,init}} = \begin{pmatrix} 8.74 \\ 11.02 \\ 11.41 \end{pmatrix} \text{ Nms (acquired by TM, see 3rd line of [Figure 14.5-4](#)).$$

A wheel off-loading is commanded at t=1200s with the following reference angular momentum:

$$\overrightarrow{H_{RW\,ref}} = \overrightarrow{H_{RW\,init}} + \overrightarrow{\Delta H_{RW}}$$

where $\overrightarrow{\Delta H_{RW}} = \begin{pmatrix} 0 \\ 0.82 \\ 0.82 \end{pmatrix}$ Nms in order to create an angular momentum variation of 1 Nms about the

spacecraft X axis (see 4th line of [Figure 14.5-4](#)), which will excite the solar array in-plane flexible mode.

So $\overrightarrow{H_{RW\,ref}} = \begin{pmatrix} 8.74 \\ 11.84 \\ 12.23 \end{pmatrix}$ Nms.

It can be checked that the reference angular momentum is approximately reached (see 3rd line of [Figure 14.5-4](#)), and that this angular momentum variation creates a high thruster torque to be commanded about the spacecraft X axis (see 2nd line of [Figure 14.5-4](#)).

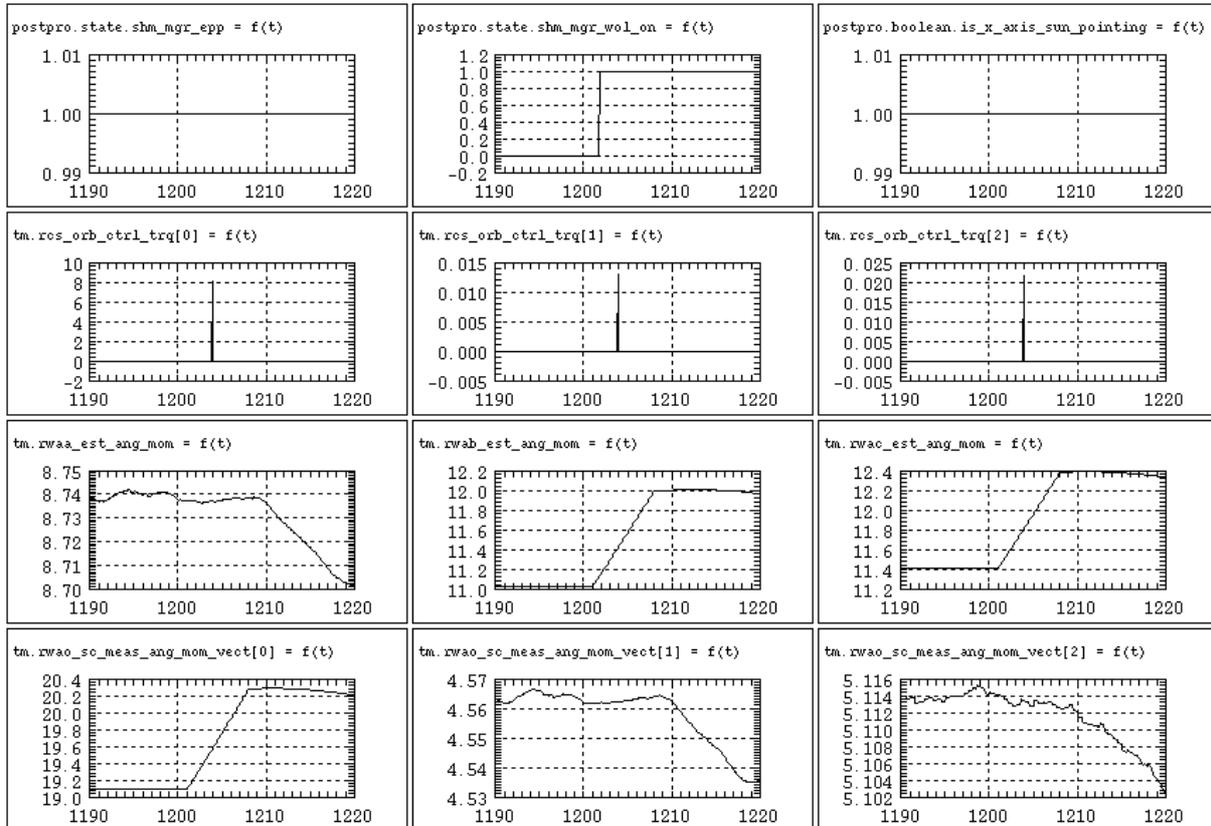


Figure 14.5-4 1st line: SHM/EPP flag, SHM/WOLP flag and Sun pointing flag
 2nd line: Thruster commanded torques in the spacecraft reference frame (Nm)
 3rd line: Estimated angular momentum in the reaction wheel reference frame (Nms)
 4th line: Measured angular momentum in the spacecraft reference frame (Nms)

The spacecraft angular rate measured by the IMP ORB about the spacecraft X axis (see 2nd line of Figure 14.5-5) is acquired between $t^{\text{start}}=1210\text{s}$ and $t=1310\text{s}$ (i.e. during $T^{\text{calib}} = 100\text{s}$, and at a frequency of 8 Hz). It is oscillating at the frequency of the 1st in-plane flexible mode (expected to be 0.76 Hz, see Table 14.2-3, since the Cantilever frequency is equal to 0.296 Hz in the simulation).

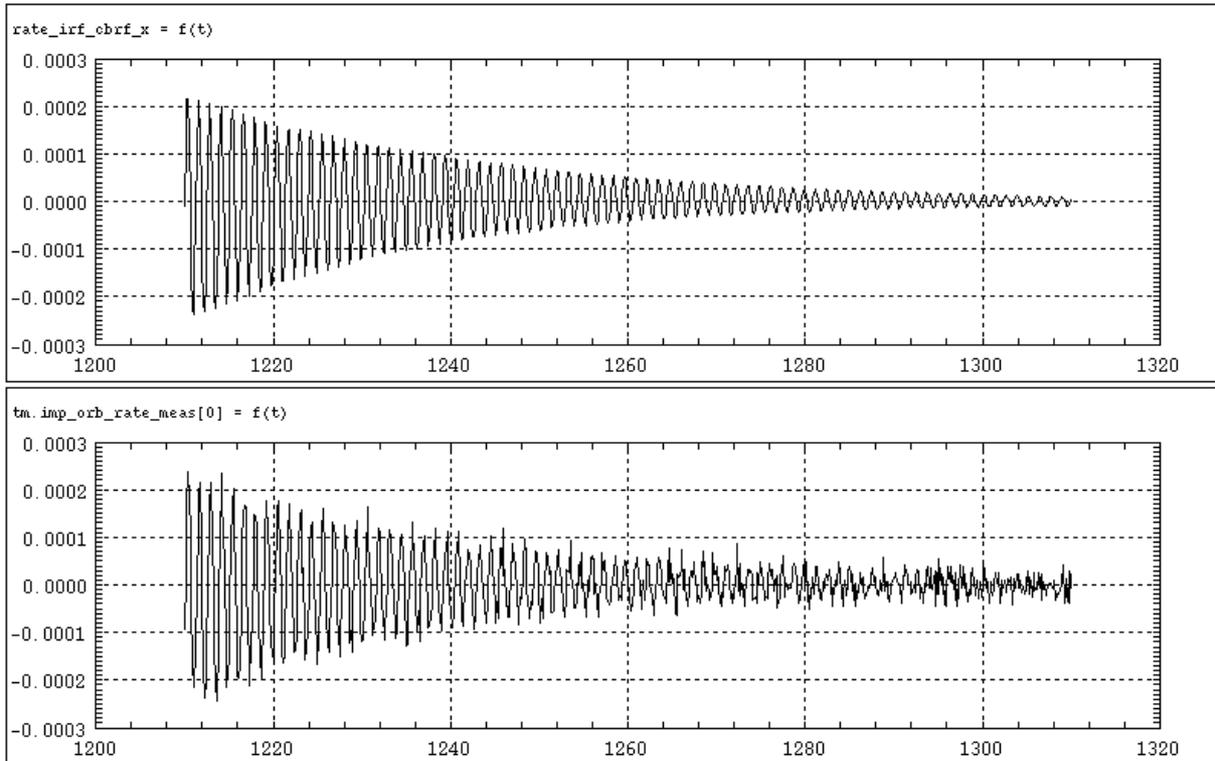


Figure 14.5-5 Real (1st line) and measured (2nd line) spacecraft angular rate about the spacecraft X axis

The Discrete Fourier Transform of the measured spacecraft rate about the spacecraft X axis is shown on [Figure 14.5-6](#). This DFT clearly shows that there is one main free-free frequency, corresponding to the 1st in-plane flexible mode frequency (expected to be 0.76 Hz, see [Table 14.2-3](#)).

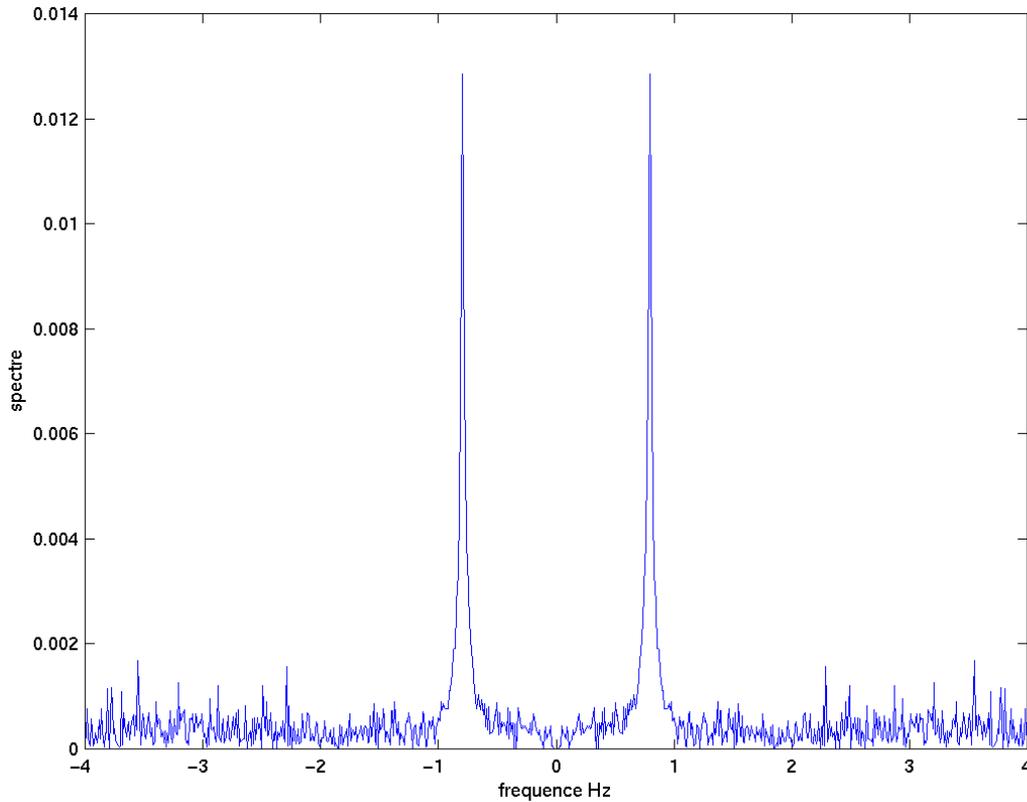


Figure 14.5-6 Discrete Fourier Transform of the spacecraft angular rate about the spacecraft X axis

The ground processing of the spacecraft angular rate about the spacecraft X axis (described in section 14.3.4) gives the following result: $f^{est}=0.8$ Hz for the 1st in-plane flexible mode free-free frequency, and is to be compared with the expected one.

Flexible mode	Theoretically expected free-free frequency	Experimentally evaluated free-free frequency	Difference
1 st in-plane	0.76 Hz	0.8 Hz	5.3%

Table 14.5-2 Comparison between the theoretically expected and the experimentally evaluated in-plane free-free frequencies

The free-free frequency of the 1st in-plane flexible mode is 5.3% higher than the expected one.

14.5.1.4 Evaluation of the first torsion flexible mode free-free frequency

The current AOCS phase is the SHM/EPP with the X axis pointed towards the Sun (1st line of [Figure 14.5-7](#)).

The initial angular momentum in the reaction wheel reference frame is (at t=1200s):

$$\overrightarrow{H_{RW\,init}} = \begin{pmatrix} 8.74 \\ 11.02 \\ 11.41 \end{pmatrix} \text{ Nms (acquired by TM, see 3rd line of [Figure 14.5-7](#)).$$

A wheel off-loading is commanded at t=1200s with the following reference angular momentum:

$$\overrightarrow{H_{RW\,ref}} = \overrightarrow{H_{RW\,init}} + \overrightarrow{\Delta H_{RW}}$$

where $\overrightarrow{\Delta H_{RW}} = \begin{pmatrix} 2 \\ -2 \\ 0 \end{pmatrix}$ Nms in order to create an angular momentum variation of 2 Nms about the

spacecraft Y axis (see 4th line of [Figure 14.5-7](#)). Unlike the other flexible modes, here it is quite difficult to excite the torsion flexible mode, so a 2 Nms angular momentum variation (instead of 1 Nms) about the spacecraft Y axis is necessary to excite the torsion flexible mode.

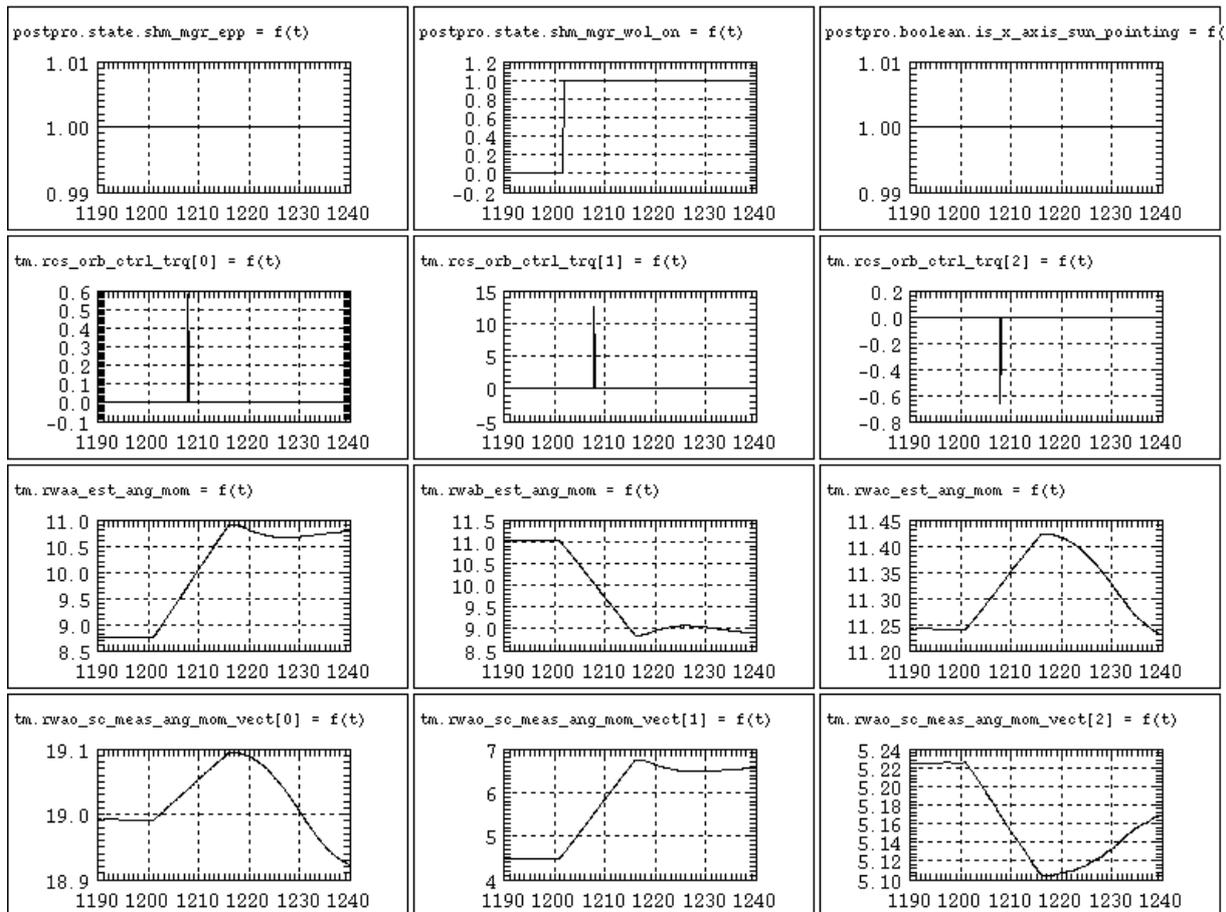
So $\overrightarrow{H_{RW\,ref}} = \begin{pmatrix} 10.74 \\ 9.02 \\ 11.41 \end{pmatrix}$ Nms.

Even with an angular momentum variation of 2 Nms, it was not possible to evaluate the first torsion flexible mode free-free frequency with the gyro noise values coming from ref. [4]. Indeed the rate oscillations were not visible because of the low rate excitation with respect to the high gyro noise. In ref. [13], the gyro noise values turned out to be much lower than the ones from ref. [4] (see [Table 14.5-3](#)), and are considered as more representative since they come from tests on the real IMP. So the values coming from ref. [13] have been used in this Rosace simulation, and with these values the evaluation of the first torsion flexible mode free-free frequency is possible.

Parameters	Values from ref. [4]	Values from ref. [13]
Read-out-noise	4.848e-6 rad	0.3e-6 rad
Angular random walk	0.01 °/√h	0.0022 °/√h

Table 14.5-3 Gyro read-out noise and angular random walk

It can be checked that the reference angular momentum is approximately reached (see 3rd line of Figure 14.5-7), and that this angular momentum variation creates a high thruster torque to be commanded about the spacecraft Y axis (see 2nd line of Figure 14.5-7).



**Figure 14.5-7 1st line: SHM/EPP flag, SHM/WOLP flag and Sun pointing flag
 2nd line: Thruster commanded torques in the spacecraft reference frame (Nm)
 3rd line: Estimated angular momentum in the reaction wheel reference frame (Nms)
 4th line: Measured angular momentum in the spacecraft reference frame (Nms)**

The spacecraft angular rate measured by the IMP ORB about the spacecraft Y axis (see 2nd line of Figure 14.5-8) is acquired between $t^{\text{start}}=1240\text{s}$ and $t=1340\text{s}$ (i.e. during $T^{\text{calib}} = 100\text{s}$, and at a frequency of 8 Hz). Here a 40s margin (instead of 10s margin for the other flexible modes) has to be taken into account wrt the entry into WOL in order to allow the rate to converge (see Figure 14.5-9), the spacecraft Y axis being the axis of lowest spacecraft inertia.

The spacecraft rate about the spacecraft Y axis is oscillating at the frequency of the 1st torsion flexible mode (expected to be 0.555 Hz, see Table 14.2-3). It can be seen that the rate levels are much lower

than for the two other flexible modes (out-of-plane and in-plane), which explains why the gyro noise (RON and ARW, see [Table 14.5-3](#)) and l_{sb} ($1e-6$ rad, see ref. [4]) have a higher effect on the measured spacecraft rate.

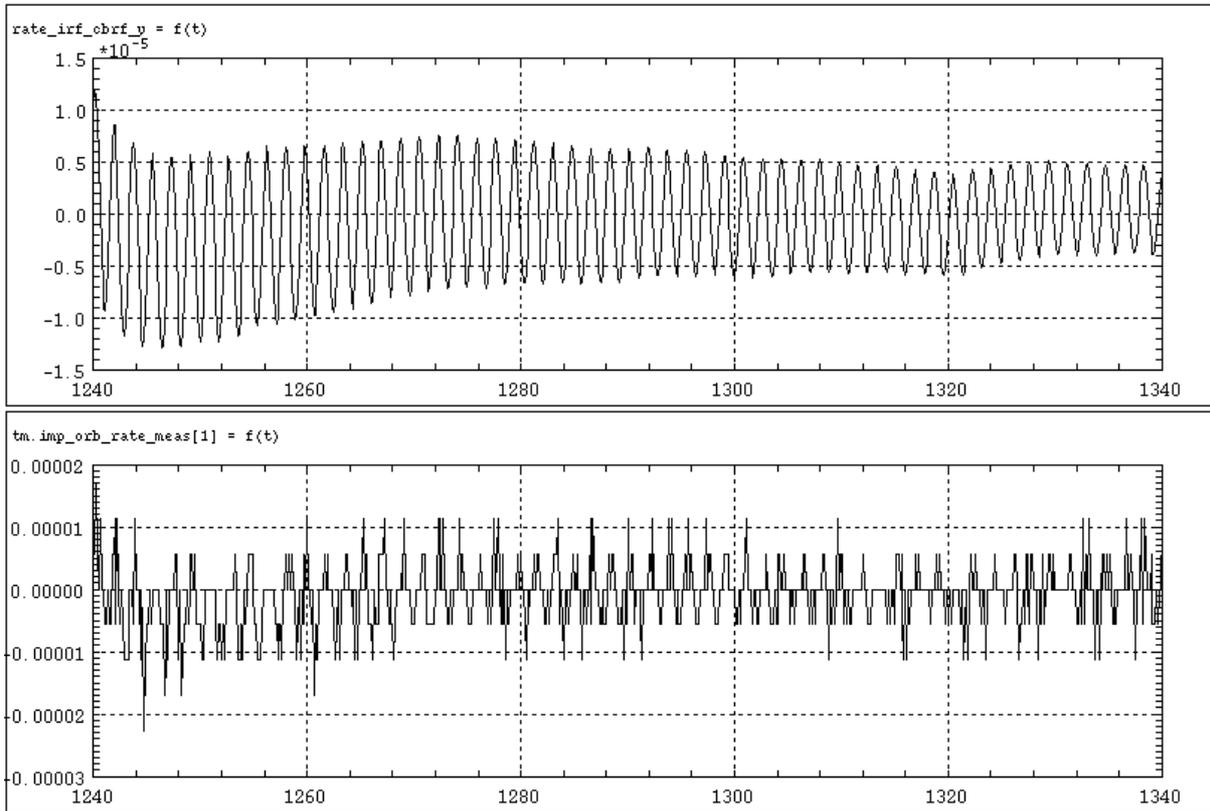


Figure 14.5-8 Real (1st line) and measured (2nd line) spacecraft angular rate about the spacecraft Y axis

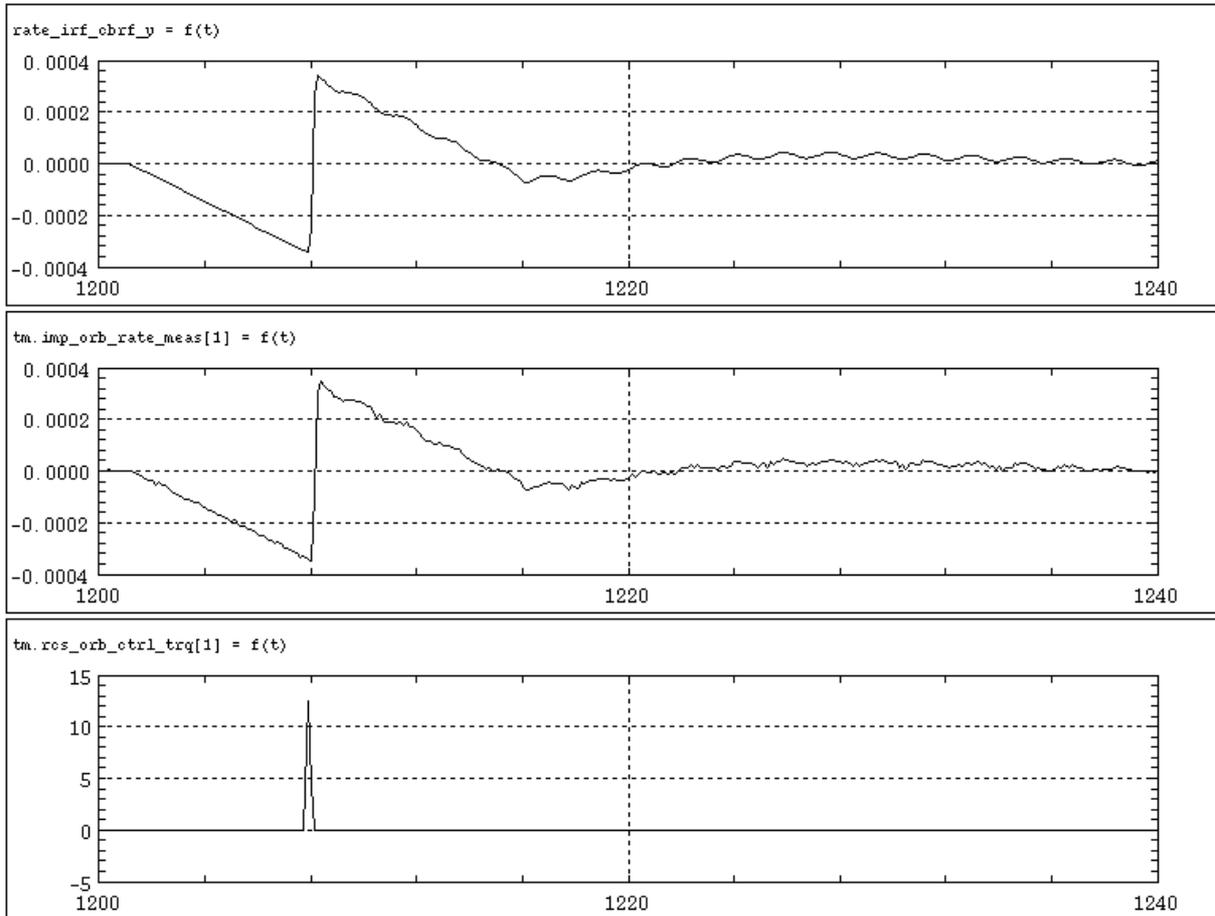


Figure 14.5-9 Convergence of the real (1st line) and measured (2nd line) spacecraft angular rate about the spacecraft Y axis after the thruster actuation (3rd line) during the wheel off-loading phase

The Discrete Fourier Transform of the measured spacecraft rate about the spacecraft Y axis is shown on [Figure 14.5-10](#). This DFT shows that there is one main free-free frequency, corresponding to the 1st torsion flexible mode frequency (expected to be 0.555 Hz, see [Table 14.2-3](#)).

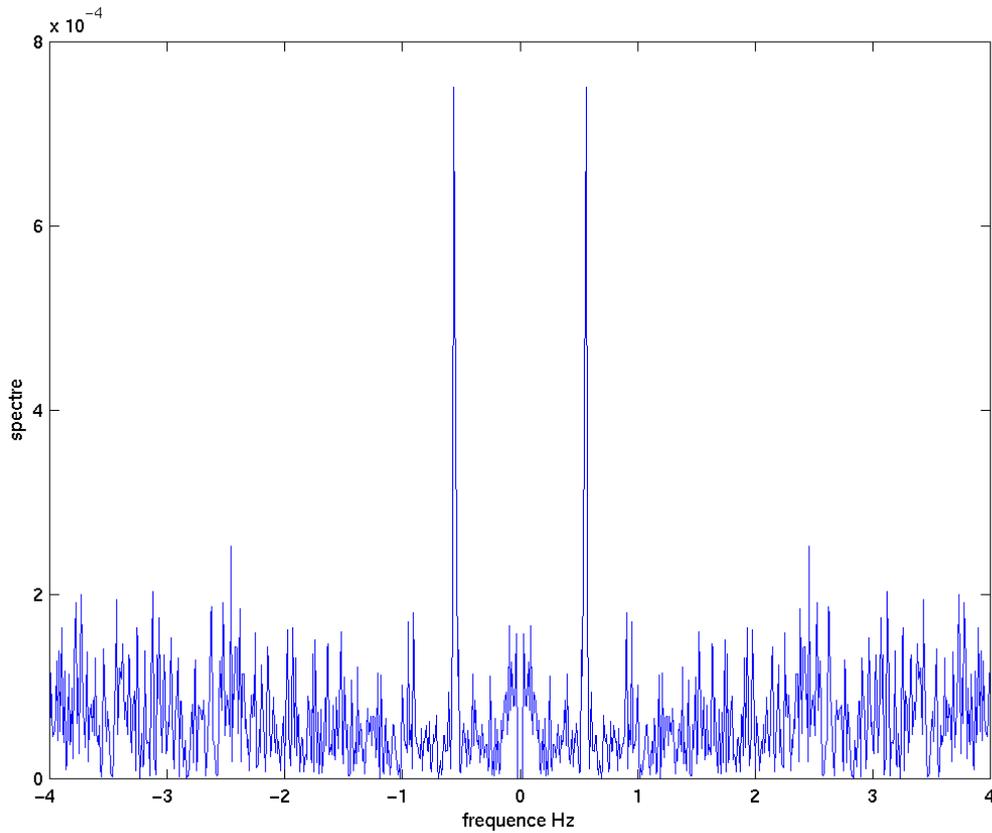


Figure 14.5-10 Discrete Fourier Transform of the spacecraft angular rate about the spacecraft Y axis

The ground processing of the spacecraft angular rate about the spacecraft Y axis (described in section 14.3.4) gives the following result: $f^{\text{est}}=0.56$ Hz for the 1st torsion flexible mode free-free frequency, and is to be compared with the expected one.

Flexible mode	Theoretically expected free-free frequency	Experimentally evaluated free-free frequency	Difference
1 st torsion	0.555 Hz	0.56 Hz	0.9%

Table 14.5-4 Comparison between the theoretically expected and the experimentally evaluated torsion free-free frequencies

The free-free frequency of the 1st torsion flexible mode is 0.9% higher than the expected one.

14.5.1.5 Conclusion

The following table recalls the free-free frequencies experimentally evaluated using the on-ground processing described in section 14.3.4. They are similar to the theoretically expected ones (less than 10.5% of difference).

Flexible mode	Experimentally evaluated free-free frequency
1 st Out-of-plane	0.17 Hz
2 nd Out-of-plane	0.48 Hz
1 st in-plane	0.8 Hz
1 st torsion	0.56 Hz

Table 14.5-5 Free-free frequencies experimentally evaluated by on-ground processing

The calibration of the torsion flexible mode will strongly depend on the gyro noise and on the level of the angular momentum variation generated by the wheel off-loading (as explained in section 14.5.1.4).

14.5.2 Simulation of the free-free flexible mode damping ratios calibration by ground processing of the spacecraft angular rate

14.5.2.1 Introduction

The same simulations results as for the free-free frequencies are used. With the ground processing principle described in section 14.4.1, only the damping ratios of the 1st out-of-plane, 1st in-plane and 1st torsion flexible modes can be determined. The damping ratio of the 2nd out-of-plane flexible mode is hard to determine and is not of special interest for the solar arrays flexible modes characteristics knowledge.

14.5.2.2 Evaluation of the first out-of-plane flexible mode free-free damping ratio

The same measured spacecraft angular rate about the spacecraft Z axis (shown on [Figure 14.5-2](#)) is used as input for the ground processing, but during a shorter time interval: between $t^{\text{start}}=1210\text{s}$ and

$t^{\text{end}}=1296.5\text{s}$, i.e. during $\Delta T = \frac{\ln 2}{2\pi \cdot \xi_{\text{free-free}}^{\text{th}} \cdot f^{\text{est}}} = 86.5\text{s}$ (this equation comes from section 14.4.1)

with $\xi_{\text{free-free}}^{\text{th}} = 7.5\text{e-}3$ (theoretical damping ratio of the 1st out-of-plane flexible mode, see [Table 14.2-8](#))

and $f^{\text{est}} = f^{\text{est}}(1) = 0.17\text{ Hz}$ (ground-processing evaluated free-free frequency of the 1st out-of-plane flexible mode, see [Table 14.5-5](#)).

The on-ground processing gives for the free-free damping ratio of the 1st out-of-plane flexible mode:

$\xi_{\text{free-free}}^{\text{est}} = 6.3\text{e-}3$, which is 16% lower than the theoretically expected one. However this damping ratio has the same order of magnitude as the expected one, and a better precision cannot be reached with the proposed ground processing.

14.5.2.3 Evaluation of the first in-plane flexible mode free-free damping ratio

The same measured spacecraft angular rate about the spacecraft X axis (shown on [Figure 14.5-5](#)) is used as input for the ground processing, but during a shorter time interval: between $t^{\text{start}}=1210\text{s}$ and

$t^{\text{end}}=1231.5\text{s}$, i.e. during $\Delta T = \frac{\ln 2}{2\pi \cdot \xi_{\text{free-free}}^{\text{th}} \cdot f^{\text{est}}} = 21.5\text{s}$ (this equation comes from section 14.4.1)

with $\xi_{\text{free-free}}^{\text{th}} = 6.4\text{e-}3$ (theoretical damping ratio of the 1st in-plane flexible mode, see [Table 14.2-8](#))

and $f^{\text{est}} = 0.8\text{ Hz}$ (ground-processing evaluated free-free frequency of the 1st in-plane flexible mode, see [Table 14.5-5](#)).

The on-ground processing gives for the free-free damping ratio of the 1st in-plane flexible mode:

$\xi_{free-free}^{est} = 5.6e-3$, which is 12.5% lower than the theoretically expected one. However this damping ratio has the same order of magnitude as the expected one, and a better precision cannot be reached with the proposed ground processing.

14.5.2.4 Evaluation of the first torsion flexible mode free-free damping ratio

The same measured spacecraft angular rate about the spacecraft Y axis (shown on [Figure 14.5-8](#)) is used as input for the ground processing, but during a shorter time interval: between $t^{start}=1240s$ and $t^{end}=1318.8s$, i.e. during $\Delta T = \frac{\ln 2}{2\pi \cdot \xi_{free-free}^{th} \cdot f^{est}} = 78.8s$ (this equation comes from section 14.4.1)

with $\xi_{free-free}^{th} = 2.5e-3$ (theoretical damping ratio of the 1st torsion flexible mode, see [Table 14.2-8](#))

and $f^{est} = 0.56$ Hz (ground-processing evaluated free-free frequency of the 1st torsion flexible mode, see [Table 14.5-5](#)).

The on-ground processing gives for the free-free damping ratio of the 1st torsion flexible mode:

$\xi_{free-free}^{est} = 2.9e-3$, which is 16% higher than the theoretically expected one. However this damping ratio has the same order of magnitude as the expected one, and a better precision cannot be reached with the proposed ground processing.

14.5.2.5 Conclusion

The following table recalls the free-free damping ratios experimentally evaluated using the on-ground processing described in section 14.4.4. They have the same order of magnitude as the theoretically expected ones (less than 16% of difference).

Flexible mode	Experimentally evaluated free-free damping ratio
1 st Out-of-plane	6.3e-3
2 nd Out-of-plane	N/A
1 st in-plane	5.6e-3
1 st torsion	2.9e-3

Table 14.5-6 Free-free damping ratios experimentally evaluated by on-ground processing

15 AOCMS TELEMETRY PARAMETER FINE DATATION

15.1 CONTEXT AND PURPOSE

The aim of this ground processing is to explain how to find the best datation for all AOCMS parameters, either belonging to YAC00001 (e.g. the AOCMS mode transition, the gyro-stellar attitude quaternion or the spacecraft measured rate) or not. In the following paragraphs, it is assumed that the TM storage frequency is 1 Hz or 8 Hz (whereas the TM building frequency is always 1 Hz).

The AOCMS TM packet is generated as follows (also see next table):

- at the end of the last 8 Hz cycle of the 1 Hz cycle, the parameters contained in the TM packet are recopied into a buffer which will be sent in the TM. If the TM storage frequency is 1 Hz, only one value is recopied for each parameter. If the TM storage frequency is 8 Hz, eight values are recopied for each parameter
- after the start of the next 1 Hz cycle, the TM packet is built and sent to the DMS, the TM packet header date corresponding to the date at which the packed is built (so the TM building frequency is 1 Hz)

In the following table, there are two parameters: TIME_DP.CURRENT_FINE_OBT (TM NAWD0500 and NAWD0501) and SEQUENCEUR_DP.CURRENT_FINE_OBT (TM NAWD0614 and NAWD0615), which are both composed of two parts:

- COARSE_OBT (TM NAWD0500 or NAWD0614) on 4 bytes with a LSB = 1 s, and corresponds to the integer part of CURRENT_FINE_OBT (ex: if CURRENT_FINE_OBT = 5.16494082, CURRENT_FINE_OBT.COARSE_OBT = INT(CURRENT_FINE_OBT) = 5)
- PRECISION (TM NAWD0501 or NAWD0615) on 2 bytes with a LSB = 2^{-16} s, and corresponds to the decimal part of CURRENT_FINE_OBT (ex: if CURRENT_FINE_OBT = 5.16494082, CURRENT_FINE_OBT.PRECISION = CURRENT_FINE_OBT - INT(CURRENT_FINE_OBT) = 0.16494082)

So, for TIME_DP or SEQUENCEUR_DP: $CURRENT_FINE_OBT = COARSE_OBT + PRECISION$.

These two parameters are updated as follows:

- TIME_DP.CURRENT_FINE_OBT (NAWD0500 and NAWD0501) is updated at 1 Hz (on the first 64 Hz cycle of 1 Hz cycle) from a reading of the true CDMU date
- SEQUENCEUR_DP.CURRENT_FINE_OBT (NAWD0614 and NAWD0615) is updated:
 - at start of every 1 Hz (on the first 64 Hz cycle of 1 Hz cycle) from a recopy of TIME_DP.CURRENT_FINE_OBT

- at every 8 Hz cycle (except the first 8 Hz cycle of the 1 Hz cycle) through an incrementation by 0.125 s

In fact, SEQUENCEUR_DP.CURRENT_FINE_OBT corresponds to the true CDMU date at 8 Hz, whereas TIME_DP.CURRENT_FINE_OBT corresponds to the true CDMU date at 1 Hz.

Since these two parameters are recopied on the last 8 Hz cycle of the 1 Hz cycle, at this recopy date they are related as follows:

$$\text{SEQUENCEUR_DP.CURRENT_FINE_OBT} = \text{TIME_DP.CURRENT_FINE_OBT} + 0.875\text{s}$$

And, generally speaking, they are related as follows at the k^{th} 8 Hz cycle:

$$\text{SEQUENCEUR_DP.CURRENT_FINE_OBT} = \text{TIME_DP.CURRENT_FINE_OBT} + (k-1)*0.125\text{s}$$

This relation is useful, because although the SEQUENCEUR_DP.CURRENT_FINE_OBT parameter is more relevant (since it corresponds to the true CDMU date at 8 Hz), the TM NAWD0614 and NAWD0615 mnemonics are not embedded in any operational packet, whereas NAWD0500 and NAWD0501 are embedded in operational packet YAC00001.

The TM packet is built after the start of the next 1 Hz cycle (but not necessarily after the start of the 1st 8 Hz cycle), and the date of the TM packet header is elaborated as follows:

- the integer part is equal to the COARSE_OBT part of TIME_DP.CURRENT_FINE_OBT
- the decimal part is taken from a 1750 timer (called "timer B"), which is incremented by 0.125 s at every 8 Hz cycle and reset to 0 at start of every 1 Hz cycle.
(Note that timer B is equal to $(k-1)*0.125\text{s}$ when the TM packet is built after the start of the k^{th} 8 Hz cycle)
- and an additional delay Δt (also called "jitter") corresponds to the delay of building the TM (lower than 50 ms, TBC)

So the date of the TM packet header is equal to: $\text{TIME_DP.CURRENT_FINE_OBT} + \text{timer B} + \Delta t$

The 7th line of the table corresponds to the date of the TM packet header if the delay of building the TM is null (i.e. $\Delta t = 0$).

In the following table, the TM packet starts being built after the start of the second 8 Hz cycle (which is not necessarily always the case), at $\text{TIME_DP.CURRENT_FINE_OBT.COARSE_OBT} + \text{timer B} = 6.125\text{ s}$, and an additional jitter delay (Δt) has to be taken into account, so the date of the TM packet header will be: $6.125 + \Delta t$.

So the date of the TM packet header will be different from the real acquisition date of the TM parameter ($\text{TIME_DP.CURRENT_FINE_OBT} + (k-1)*0.125$ s). The difference between the real acquisition date of the TM parameter ($\text{TIME_DP.CURRENT_FINE_OBT} + (k-1)*0.125$ s) and the TM packet header date will be equal to:

$$\begin{aligned} & (\text{TIME_DP.CURRENT_FINE_OBT}^n + (k-1)*0.125) \\ & - (\text{TIME_DP.CURRENT_FINE_OBT}^{(n+1)}. \text{COARSE_OBT} + \text{timer B} + \Delta t) \\ & = (k-9)*0.125 + \text{TIME_DP.CURRENT_FINE_OBT}^{(n+1)}. \text{PRECISION} - \text{timer B} - \Delta t \end{aligned}$$

In the example of the following table, the difference between the two dates is (at $k = 8$):

$$-0.125 + 0.16494082 - 0.125 - \Delta t = -0.08505918 - \Delta t$$

But it could be greater, **the difference between the real acquisition date of the TM parameter** ($\text{TIME_DP.CURRENT_FINE_OBT} + (k-1)*0.125$ s) **and the TM packet header date can vary:**

from $(k-9)*0.125 + 0 - 0.875 - 0.05 = -2.05 + k*0.125$ s to $(k-9)*0.125 + 1 - 0 - 0 = (k-1)*0.125$ s.

i.e. **from -2.05s** (with $k = 0$) to **0.875s** (with $k = 8$).

Note that the $(k = 0)^{\text{th}}$ 8 Hz cycle has to be introduced: it corresponds to the $(k = 9)^{\text{th}}$ cycle of the $(n-1)^{\text{th}}$ 1 Hz cycle.

And it is therefore better to relate the date of any AOCMS parameter to the real acquisition date of the TM parameter ($\text{TIME_DP.CURRENT_FINE_OBT} + (k-1)*0.125$ s, belonging to YAC00001) than to the date of the TM packet header.

All AOCMS TM packets are generated at the end of the last 8 Hz cycle of the 1 Hz cycle, so the values included in two different packets can be considered as simultaneous. If packet YAC00001 is always available, the real acquisition date of the TM parameter ($\text{TIME_DP.CURRENT_FINE_OBT} + (k-1)*0.125$ s) can be used not only for the AOCMS parameters belonging to the same packet YAC00001, but also for the other AOCMS parameters.



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
 Issue : **3** Rev. : 0
 Date : **05/08/2003**
 Page : 154

1 Hz cycle		n								n+1	
8 Hz cycle		1	2	3	4	5	6	7	8	1	2
True CDMU date		5.16494082	5.28994082	5.41494082	5.53994082	5.66494082	5.78994082	5.91494082	6.03994082	6.16494082	6.28994082
TIME_DP.CURRENT_FINE_OBT	NAWD0500/NAWD0501	5.16494082	5.16494082	5.16494082	5.16494082	5.16494082	5.16494082	5.16494082	5.16494082	6.16494082	6.16494082
SEQUENCER_DP.CURRENT_FINE_OBT	NAWD0614/NAWD0615	5.16494082	5.28994082	5.41494082	5.53994082	5.66494082	5.78994082	5.91494082	6.03994082	6.16494082	6.28994082
timer B		0	0.125	0.25	0.375	0.5	0.625	0.75	0.875	0	0.125
date of TM packet header (if it was generated right at the start of the 8 Hz cycle)		5	5.125	5.25	5.375	5.5	5.625	5.75	5.875	6	6.125

Value recopied in a
buffer sent in the TM

TM packet header
date = 6.125 + Δt

15.2 GROUND PROCESSING

All AOCMS parameters updated on-board at a frequency higher or equal than 1 Hz (e.g. the AOCMS mode transition, the gyro-stellar attitude quaternion or the spacecraft measured rate) are elaborated at a date between:

[TIME_DP.CURRENT_FINE_OBT - 0.125 s; TIME_DP.CURRENT_FINE_OBT + 0.875s]

where TIME_DP.CURRENT_FINE_OBT is available by TM NAWD0500 and NAWD0501 of YAC00001

In particular for parameters updated on-board at 8 Hz (with a TM storage frequency of 1 or 8 Hz):

TIME_DP.CURRENT_FINE_OBT + 0.875 s ($k = 8$) is the best estimation of their date of elaboration

In particular for parameters updated on-board at 1 Hz:

If the TM storage frequency is 1 Hz (only one value is available for each parameter in each TM packet):

TIME_DP.CURRENT_FINE_OBT + 0.375 s ($k = 4$) is the best estimation of their date of elaboration

Else if the TM storage frequency is 8 Hz (eight values are available for each parameter in each TM packet):

If the 1 Hz updated AOCMS parameter has evolved at the k^{th} 8 Hz cycle, then:

TIME_DP.CURRENT_FINE_OBT + $(k-1) * 0.125$ s is the best estimation of their date of elaboration

When the AOCMS parameter is updated on-board at a frequency f lower than 1 Hz, the TM packets shall be examined during a period higher than $1/f$, and the first TM packet where the measurement has evolved should be retained. By looking at packet YAC00001 corresponding to this TM packet, their dates of elaboration can be estimated as for the parameters updated on-board at 1 Hz (using NAWD0500 and NAWD0501).



ROSETTA AVIONICS

Ref : RO-MMT-TN-2180
Issue : **3** Rev. : 0
Date : 05/08/2003
Page : 156

DISTRIBUTION LIST

<u>ASTRIUM-F PROJECT TEAM :</u>		<u>AOCS Studies Team</u>	
J.C. ASSELINO		O. BONNAMY	1
Ch. BERGEZ		E. ECALE	1
C. BILLARD		J. TOUATY	1
R. CASPAR		R. FAYARD	1
JM. CHARBONNIER		A. INCERTI (DOC DSG)	1
E. CRABOS		J-L. GONNAUD (Summary only)	1
B. HOSTEIN			
F. FRAKSO			
B. GILLOT			
M. JANVIER	1		
P. LELONG	1		
J.L. PAGE		<u>ASTRIUM-GmbH</u>	
B. POUZET		Mr. R. BEST	2
P. REGNIER	1	<u>ESTEC</u>	
A. PEFFERKORN		Mr. J. ELLWOOD	5
P. VLIMANT		<u>ESOC</u>	
L. WOLFF		Mr. M.WARHAUT	1