RE-ENGINEERING OF MISSION ANALYSIS SOFTWARE FOR ENVISAT-1

PPF_POINTING SOFTWARE USER MANUAL

PO-IS-DMS-GS-0559

Code:	PO-IS-DMS-GS-0559
Issue:	5.9
Date:	30/05/2011

	Name	Function	Signature
Prepared by:	Noelia Sánchez-Ortiz	Project Engineer	
Checked by:	José Antonio González Abeytua	Project Manager	
Approved by:	José Antonio González Abeytua	Project Manager	

DEIMOS Space S.L.U
Ronda de Poniente, 19
Edificio Fiteni VI, Portal 2, 2ª Planta
28760 Tres Cantos(Madrid), SPAIN
Tel.: +34 91 806 34 50
Fax: +34 91 806 34 51
E-mail: deimos@deimos-space.com

Document Information

Contract Data		Classification	
Contract Number:	Contract number	Internal	
		Public	
Contract Issuer:	ESA / ESTEC	Industry	X
		Confidential	

Internal Distribution		
Name	Unit	Copies

External Distribution		
Name	Organisation	Copies

Archiving	
Word Processor:	Framemaker 6.0
File name:	PointingSum
Archive Code	P/SUM/DMS/01/043-017

Document Status Log

Issue	Change Description	Date	Approval
4.1	• First Release	15/06/97	
4.2	• pp_genatt removed from the library • variables expressed in Mean of Date CS now in True of Date CS • known problems of release 4.1 solved	30/04/97	
4.3	• Runtime performances improved • Output parameters can be selected in pp_target CFI • Special AATSR and MERIS output selection switch values in pp_target CFI to optimise runtime performances • Improved accuracy of the EF related parameters calculation in pp_target CFI • Improved accuracy of the pp_target CFI mode idir = PP_GR_RANGE • New mode in pp_target CFI idir = PP_GENERIC_TARG • New refraction models in pp_target CFI including a new executable CFI pp_converter to use a user's defined atmosphere • Enumeration values can be used to identify an element of the results vector of pp_target, pp_stavis and pp_drsvis CFIs • known problems of release 4.2 solved	18/05/98	
4.4	• Error handling vectors length modified • Port to Windows	19/11/98	
4.5	• threshold used to increase runtime performance updated. • Time and latitude ranges changed in pp_target refraction mode 30x to cover the whole Earth. • DUT1 range updated to [-1.0,1.0]	31/05/99	
4.6	• SPR57 (wrong results in PP_NO_TAR mode of pp_target) fixed.	14/05/00	
4.7	• Unreleased	22/06/01	
4.8	• Unreleased	31/07/01	
4.9	• Unreleased • New document from previous version by: - L.J. Alvarez (GMV) - J.A. González (GMV) - M. Sánchez-Gestido (GMV) - M.Sánchez-Nogales (GMV)	22/10/01	
5.0	• Document format change.	18/01/02	
5.1	• Unreleased	25/11/02	

Issue	Change Description	Date	Approval
5.2	• See change bars	26/05/03	
5.3	• New Attitude functions	13/12/04	
5.3.1	• Maintenance release	15/02/05	
5.4	• Maintenance release	17/05/05	
5.5	• Maintenance release • Library Optimization	30/01/06	
5.6	• Maintenance release • New library for LINUX 64-bits	22/12/06	
5.7	• Changes for Envisat extended life • New library for MAC OS on Intel platforms • Maintenance release	11/04/08	
5.8	• Maintenance release.	15/09/09	
5.9	• Sun position calculation improved by considering difference between UTC and TBC • Maintenance release.	30/05/11	

Table of Contents

1 SCOPE

The Software User Manual (SUM) of the Envisat-1 mission CFI software is composed of

- a general document describing the sections common to all the CFI software libraries
- a specific document for each of those libraries.

This document is the PPF_POINTING Software User Manual. It provides a detailed description of the use of the CFI functions included within the PPF_POINTING CFI software library.

2 ACRONYMS AND NOMENCLATURE

2.1 Acronyms

AOCS	Attitude and Orbit Control System
ANX	Ascending Node Crossing
CFI	Customer Furnished Item
CS	Coordinate System
DRS	Data Relay Satellite
ESA	European Space Agency
ESTEC	European Space Technology and Research Centre
FOS	Flight Operation Segment
GS	Ground Station
H/W	Hardware
I/F	Interface
LOS	Line Of Sight
LUT	Look-Up Table
PPF	Polar Platform
RAM	Random Access Memory
SBT	Satellite Binary Time
SUM	Software User Manual
S/W	Software
UTC	Universal Time Coordinated
UT1	Universal Time UT1
SSP	Sub Satellite Point

2.2 Nomenclature

CFI	A group of CFI functions, and related software and documentation, that will be distributed by ESA to the users as an independent unit
CFI function	A single function within a CFI that can be called by the user
Library	A software library containing all the CFI functions included within a CFI plus the supporting functions used by those CFI functions (transparently to the user)

3 APPLICABLE AND REFERENCE DOCUMENTS

3.1 Applicable documents

AD 1 Finalization of the re-engineering of Mission Analysis Software and of the ROP Generation Tool for Envisat: Statement of Work.PO-SW-ESA-SY-1242. ESA/ESTEC/APP. Issue 1.1. 03/10/2001.

AD 2 ESA Software Engineering Standards. ESA PSS-05-0. ESA. Issue 2. February 1991

3.2 Reference documents

RD 1 Envisat-1 Mission CFI Software Description and Interface Definition Document. PO-ID-ESA-SY-00412.

RD 2 Envisat-1 Mission CFI Software. Mission Conventions Document. PO-IS-GMV-GS-0561

RD 3 Envisat-1 Mission CFI Software. General Software User Manual. PO-IS-DMS-GS-0556.

RD 4 Envisat-1 Mission CFI Software. PPF_LIB Software User Manual. PO-IS-DMS-GS-0557

RD 5 Envisat-1 Mission CFI Software. PPF_ORBIT Software User Manual. PO-IS-DMS-GS-0558

RD 6 Envisat-1 Reference Operation Plan (ROP). EN-PL-ESA-GS-00334

RD 7 Envisat Products Specifications. Volume 16. PO-RS-MDA-GS-2009

4 INTRODUCTION

This software library contains the CFI functions required to perform accurate computation of pointing parameters from and to Envisat-1 for various types of targets.

It includes a generic target function and more specialized functions for a Ground Station and for a Data Relay Satellite, as well as some relevant supporting functions, i.e the following CFI functions:

- **pp_target**: generic target pointing calculation
- **pp_converter**: auxiliary executable for some **pp_target** computations
- **pp_stavis**: ground station pointing calculation
- **pp_genstate_drs**: DRS state vector calculation
- **pp_drsvis**: DRS pointing calculation
- **pp_init_attitude_file**: initialise attitude with file
- **pp_get_attitude_aocs**: calculate attitude for a given input time
- **pp_free_attitude**: clean-up memory reserved by **pp_init_attitude_file**

Note that:

- the possible occultation of the Ground Station or DRS link or the observation of a target due to the appendages of the PPF, the solar array or the payload antennae is NOT modeled.
- the pointing calculation functions require an Envisat-1 cartesian state vector expressed in the Earth fixed coordinate system at the user specified time. It is provided either by the **po_ppforb** or by the **po_interpol** CFI functions, included in the **PPF_ORBIT** software library.
- the **pp_drsvis** CFI function requires also the DRS cartesian state vector expressed in the Earth fixed coordinate system at the user specified time. It is provided by the **pp_genstate_drs** CFI function, also available in the **PPF_POINTING** software library.

A detailed description of each function is provided in 7.

Please refer also to:

- RD 2 for a detailed description of the time references and formats, coordinate systems, parameters and models used in this document
- RD 3 for a complete overview of the CFI, and in particular the detailed description of the error handling functions

5 LIBRARY INSTALLATION

For a detailed description of the installation of any CFI library, please refer to RD 3.

Note that to run the example program, the following other CFI is required:

- PPF_ORBIT

6 LIBRARY USAGE

Note that to use the PPF_POINTING software library, the following other CFI software libraries are required:

- PPF_LIB (version 5.9. See RD 4).

To use the PPF_POINTING software library in a user application, that application must include in his source code either:

- ppf_pointing.h (for a C application)
- ppf_pointing.inc (for a Fortran application under SOLARIS/AIX/LINUX/MacOS)
- ppf_pointing_win.inc (for a Fortran application under Windows 95/NT)

To link correctly his application, the user must include in his linking command flags like (assuming *cfi_libs_dir* and *cfi_include_dir* are the directories where respectively all CFI libraries and include files have been installed, see RD 3 for installation procedures):

- SOLARIS / AIX

  -I*cfi_include_dir* -L*cfi_lib_dir* -lppf_pointing -lppf_lib

- WINDOWS

  /I "*cfi_include_dir*" /libpath:"*cfi_lib_dir*"libppf_pointing.lib

  libppf_lib.lib

All functions described in this document have a name starting with the prefix pp_

To avoid problems in linking a user application with the PPF_POINTING software library due to the existence of names multiple defined, the user application should avoid naming any global software item beginning with either the prefix PP_ or pp_.

To preserve compatibility with the historical CFI function names, it is possible to call the CFI functions described in this document from a user application with or without the pp_ prefix. This does not apply to the error handling functions, which are described in the General SUM (see RD 3).

This is summarized in the table below.

Function Name	Enumeration value	long
Main CFI Functions		
pp_target target	PP_TARGET_ID	0
pp_stavis stavis	PP_STAVIS_ID	1
pp_genstate_drs genstate_drs	PP_GENSTATE_DRS_ID	2
pp_drsvis drsvis	PP_DRSVIS_ID	3
pp_init_attitude_file	PP_INIT_ATTITUDE_FILE_ID	5

Table 1: ppf_pointing functions.

Function Name	Enumeration value	long
pp_get_attitude_aocs	PP_GET_ATTITUDE_AOCS_ID	6
Error Handling Functions		
pp_verbose	not applicable	
pp_silent		
pp_vector_code		
pp_vector_msg		
pp_print_msg		

Table 1: ppf_pointing functions.

Notes about the table:

- to transform the status vector returned by a CFI function to either a list of error codes or list of error messages, the enumeration value (or the corresponding integer value) described in the table must be used
- the error handling functions have no enumerated values

Whenever available **it is strongly recommended to use enumeration values rather than integer values**.

6.1 Usage hints

The runtime performances of few of the CFI functions pp_target , pp_stavis, pp_drsvis and pp_genstate_drs are improved to a large extent if they are called two consecutive times keeping constant some of their inputs.

- pp_target: UT1 time, position, velocity, acceleration, aocs, mispointing and mispointing rate.
- pp_stavis: UT1 time, position, velocity, acceleration, aocs, mispointing and mispointing rate.
- pp_drsvis: UT1 time, position, velocity, acceleration, aocs, mispointing and mispointing rate.
- pp_genstate_drs: UT1 time.

Nevertheless, although the user may not need to call the CFI functions two consecutive times with the same inputs, there are internal functions that are actually called in those conditions, and thus improving the runtime performances of the former.

Thus, the runtime improvement is achieved with any sequence of calls to those CFI functions, not only with a sequence of calls to the same function.

In fact, the UT1 time, position, velocity, acceleration vectors, AOCS and mispointing angles do not need to keep exactly constant as long as the difference between two consecutive calls lays within the following thresholds:

- UT1 time: 0.0864 microsec
- Position vector: 0.6e-3 m
- Velocity vector: 0.6e-6 m/s
- Acceleration vector: 0.6e-9 m/s^2
- AOCS: 5e-9 deg
- Mispointing angles: 5e-9 deg
- Mispointing angles-rate: 5e-12 deg
- Mispointing angles-rate-rate: 5e-15 deg

7 CFI FUNCTIONS DESCRIPTION

The following sections describe each CFI function.

The calling interfaces are described both for C users and Fortran users.

Input and output parameters of each CFI function are described in tables, where C programming language syntax is used to specify:

- parameter types (e.g. long, double)
- array sizes of N elements (e.g. param[N])
- array element M (e.g. [M])

Fortran users should adapt the tables using Fortran syntax equivalent terms:

- parameter types (e.g. long <=> INTEGER*4, double <=>REAL*8)
- array sizes of N elements (e.g. param[N] <=> param (N))
- array element M (e.g. [M] <=> (M+1))

7.1 pp_target

7.1.1 Overview

The **pp_target** CFI function calculates a set of parameters related to the solution of several problems:

- Calculation of the signal path from Envisat-1 to a (moving) target
- Calculation of the signal path from Envisat-1 to the related (Earth-fixed) target
- Calculation of the light path from the Sun to that (moving) target
- Calculation of the light path from the Sun to Envisat-1
- Calculation of the light path from the Moon to Envisat-1

plus the roll, pitch and yaw angles (+ rate) of the Satellite Relative Actual Reference coordinate system

The location of the target is calculated according to certain conditions expressed in terms of a mode and a set of mode related parameters (see table and figure on next pages).

For the sake of understanding it is worth recalling from RD 2 some definitions:

The target is a point that is observed from the satellite and that satisfies certain conditions.

The look direction, or line of sight (LOS), is the direction at the satellite of the path followed by the corresponding signal in its travel from that target to the satellite.

If the target moves wrt the Earth, as a result of a change in the satellite position or a change in the look direction, it is called the moving target.

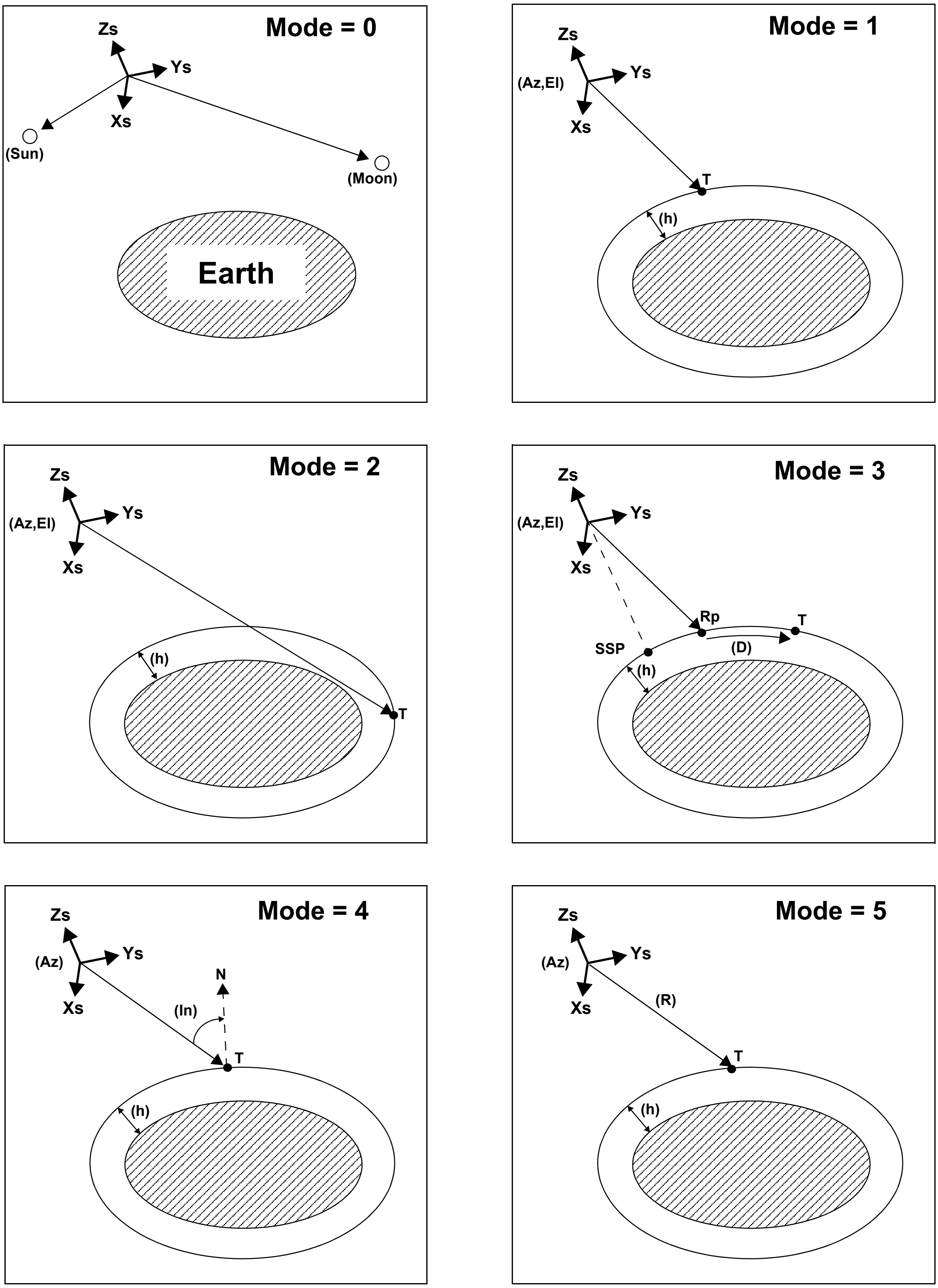
If the target is fixed wrt the Earth, which implies that if the satellite position changes then the look direction has to change in the precise way to keep looking to that particular point fixed to the Earth, it is called the Earth fixed target.

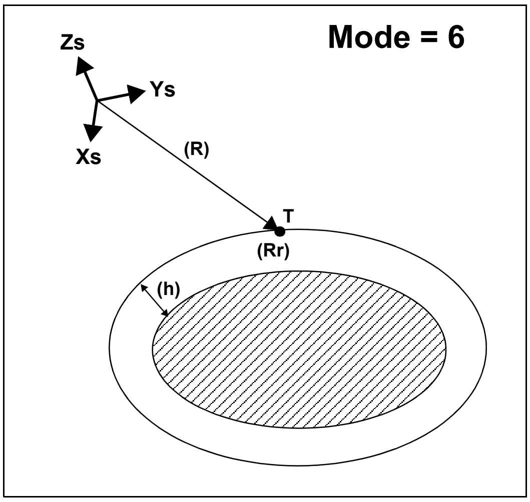
In other words, the velocity of the moving target is the result of the motion of the satellite and the change in the look direction, or in the conditions that define it, with time. On the other hand, the velocity of the Earth fixed target is only a function of the position of that point wrt the Earth and the rotation of the Earth fixed coordinate system.

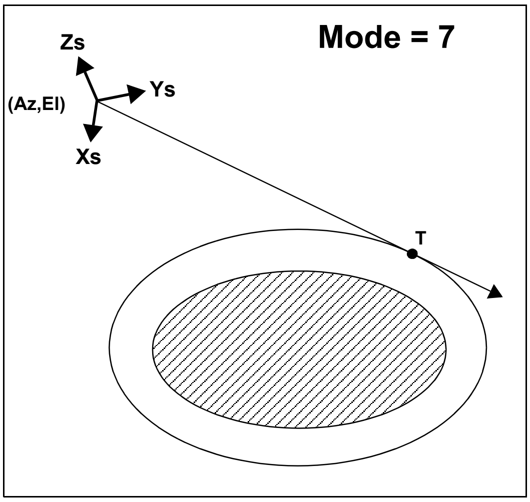
The table below and the diagram on the next page describe the various modes of **pp_target.**

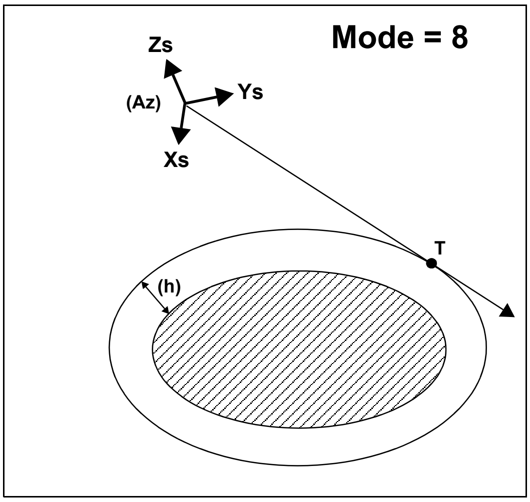
Mode (idir)	Description
-2	The user's predefined refraction LUTs are initialized (i.e. the file generated by **pp_converter** is read), thus no target point is calculated nor any other parameter, i.e. all the parameters of the results vector are zero.
-1	The user's refraction ray tracing model is initialized (i.e. the user's file is read), thus no target point is calculated nor any other parameter, i.e. all the parameters of the results vector are zero
0	No target point is calculated, since this mode only calculates the Envisat-1 attitude parameters and the look directions to the centre of the Sun and of the Moon and the tangent points over the Earth that are located on the corresponding lines of sight, i.e. only the parameters res[45:60] and res[64:65] of the results vector.
1	It calculates the first intersection point of the line of sight defined by an elevation and an azimuth angle expressed in the Satellite Relative Actual Reference coordinate system, with a surface located at a certain geodetic altitude over the Earth.
2	It calculates the second intersection point of the line of sight defined by an elevation and an azimuth angle expressed in the Satellite Relative Actual Reference coordinates system, with a surface located at a certain geodetic altitude over the Earth.
3	It calculates the location of a point that is placed on a surface at a certain geodetic altitude over the Earth, that lays on the plane defined by the Envisat-1 position, the nadir and a reference point, and that is at a certain distance or ground range measured along that surface from that reference point. This reference point is calculated being the intersection of the previous surface with the line of sight defined by an elevation and azimuth angle in the Satellite Relative Actual Reference coordinate system.
4	It calculates the location of a point that is placed on a surface at a certain geodetic altitude over the Earth and that is seen from Envisat-1 on a line of sight that forms a certain azimuth angle in the Satellite Relative Actual Reference coordinate system and that intersects that surface with a certain incidence angle.
5	It calculates the location of a point that is placed on a surface at a certain geodetic altitude over the Earth, that is seen from Envisat-1 on a line of sight that forms a certain azimuth angle in the Satellite Relative Actual Reference coordinate system, and that is at a certain range or slant-range from Envisat-1.
6	It calculates the location of a point that is placed on a surface at a certain geodetic altitude over the Earth, that is at a certain range from Envisat-1, and whose associated Earth-fixed target has a certain range-rate value.
7	It calculates the location of the tangent point over the Earth that is located on the line of sight defined by an elevation and azimuth angles expressed in the Satellite Relative Actual Reference coordinate system.
8	It calculates the location of the tangent point over the Earth that is located on a surface at a certain geodetic altitude over the Earth and that is on a line of sight that forms a certain azimuth angle in the Satellite Relative Actual Reference coordinate system.
9	It calculates the location of the tangent point over the Earth that is located on the line of sight that points to a star defined by its right ascension and declination coordinates.
10	The cartesian state vector of the target is taken as an input.

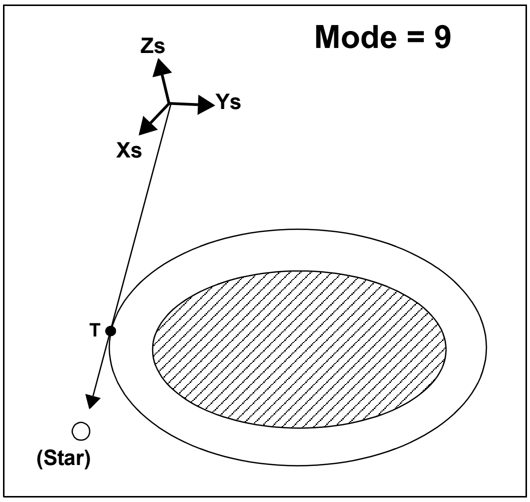
Table 2: pp_target modes.

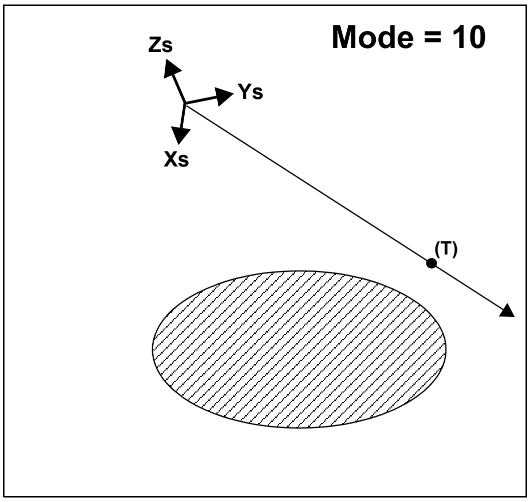

Mode = 0
Zs
Ys
Xs
(Sun)
(Moon)
Earth
Mode = 1
Zs
Ys
(Az,El)
Xs
T
(h)
Mode = 2
Zs
Ys
(Az,El)
Xs
(h)
T
Mode = 3
Zs
Ys
(Az,El)
Xs
Rp
T
SSP
(D)
(h)
Mode = 4
Zs
Ys
(Az)
Xs
N
(In)
T
(h)
Mode = 5
Zs
Ys
(Az)
Xs
(R)
T
(h)











[Xs,Ys,Zs] = Satellite Relative Actual Reference CS

() = Input data to the mode

(Az,El) = Azimuth + Elevation of the LOS
(h) = Geodetic altitude of the target
(R) = Range Satellite ⇔ Reference Point/Target
(D) = Distance or Ground range Ref. Point ⇔ Target
(In) = Incidence angle of the LOS
(Rr) = Range-rate of the Earth-fixed target

T = Target
SSP = Sub Satellite Point = Nadir of the satellite
Rp = Reference Point

N = Normal vector to the surface at a geodetic altitude = h

7.1.2 Calling interface

The calling interface of the **pp_target** CFI function is the following (input parameters are underlined):

#include <ppf_pointing.h>

```
{
       double mjdp[2], pos[3], vel[3], acc[3];
       double aocs[3], att[3], datt[3];
       double dir[8], freq, res[75];
       long status, idir, iray, ieres, ierr[15];

       status = pp_target(mjdp, pos, vel, acc,
                          aocs, att, datt,
                          &idir, dir, &iray, &freq, &ieres,
                          res, ierr);
}
```

For Fortran programs the declaration and calling procedure is as follows (input parameters are underlined, note that the C preprocessor must be used because of the presence of the `#include` statement):

```
#include <ppf_pointing.inc>

       REAL*8 MJDP(2), POS(3), VEL(3), ACC(3)
       REAL*8 AOCS(3), ATT(3), DATT(3)
       REAL*8 DIR(8), FREQ, RES(75)
       INTEGER*4 STATUS, IDIR, IRAY, IERES, IERR(15)

       STATUS = PP_TARGET(MJDP, POS, VEL, ACC,
     &                    AOCS, ATT, DATT,
     &                    IDIR, DIR, IRAY, FREQ, IERES,
     &                    RES, IERR)
```

7.1.3 Input parameters

The **pp_target** CFI function has the following input parameters:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
mjdp[2]	double	[0]	Specified UTC (UT1 time)	decimal days (Processing format)	>= -18262.0 < +36525.0
		[1]	Specified ΔUT1=UT1-UTC (UT1 time)	s (Processing format)	>= -1.0 <= +1.0
pos[3]	double	all	Satellite position vector at specified UT1 (Earth fixed CS)	m	-
vel[3]	double	all	Satellite velocity vector at specified UT1 (Earth fixed CS)	m/s	-
acc[3]	double	[0]	Satellite acceleration vector at specified UT1 (Earth fixed CS)	m/s^2	-
aocs[3]	double	[0]	AOCS Cx parameter [pitch] (Satellite Reference CS)	deg	if no better value, assume -0.167074
		[1]	AOCS Cy parameter [roll] (Satellite Reference CS)	deg	if no better value, assume +0.050233
		[2]	AOCS Cz parameter [yaw] (Satellite Reference CS)	deg	if no better value, assume +3.912987
att[3]	double	[0]	Pitch mispointing angle (Satellite Relative Reference CS)	deg	if no better value, assume 0.0
		[1]	Roll mispointing angle (Satellite Relative Reference CS)	deg	if no better value, assume 0.0
		[2]	Yaw mispointing angle (Satellite Relative Reference CS)	deg	if no better value, assume 0.0
datt[3]	double	[0]	Pitch mispointing rate (Satellite Relative Reference CS)	deg/s	if no better value, assume 0.0
		[1]	Roll mispointing rate (Satellite Relative Reference CS)	deg/s	if no better value, assume 0.0
		[2]	Yaw mispointing rate (Satellite Relative Reference CS)	deg/s	if no better value, assume 0.0

Table 3: pp_target input parameters.

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
idir	long *	-	Direction mode switch	-	-2 to 10
dir[8]	double	all	Direction parameters	-	-
iray	long *	-	Ray tracing model switch	-	0,1,2,3,10,20,30, 300,301,302, 303,304,305,4xx, 3000,3001,3002, 3003,3004,3005, 40xx
freq	double *	-	Frequency of the signal	Hz	>= 0[a]
ieres	long *	-	Extended results vector switch: 0 = Only the basic set of parameters of the results vector is calculated 1 = All the parameters of the results vector are calculated. To select the calculation of individual parameters of the result vector see next table.	-	0, 1 See the following table.

Table 3: pp_target input parameters.

a. If iray = 3/30 then the allowed range for the wavelength is
o 1000.0 nm <= c/freq <= 10000.0 nm (idir = 7/8)
o 250.0 nm <= c/freq <= 900.0 nm (idir = 9)

7.1.3.1 idir enumeration values

The following table shows the enumeration values that can be assigned to the *idir* parameter.

It is strongly recommended to use enumeration values rather than integer values.

Input	Description	Enumeration value	long
idir	Direction mode switch	PP_TAR_LUT_INIT	-2
		PP_TAR_INIT	-1
		PP_NO_TAR	0
		PP_INTER_1ST	1
		PP_INTER_2ND	2
		PP_GR_RAN	3
		PP_IN_ANG	4
		PP_RAN	5
		PP_RAN_RATE	6
		PP_TAN	7
		PP_ALT	8
		PP_STAR	9
		PP_GENERIC_TARG	10

Table 4: idir enumeration values.

7.1.3.2 iray enumeration values

The following table shows the enumeration values that can be assigned to the *iray* parameter.

It is strongly recommended to use enumeration values rather than integer values (see section 7.1.5).

Input	Description	Enumeration value	long
iray	Ray tracing model switch	PP_NO_REF	0
		PP_STD_REF	1
		PP_USER_REF	2
		PP_PRED_REF	3
		PP_STD_REF_N	10
		PP_USER_REF_N	20
		PP_PRED_REF_N	30
		PP_US76_REF	300
		PP_TROPIC_REF	301
		PP_MID_SUM_REF	302
		PP_MID_WIN_REF	303
		PP_SUBAR_SUM_REF	304
		PP_SUBAR_WIN_REF	305
		PP_LUT_REF	400
		PP_US76_REF_N	3000
		PP_TROPIC_REF_N	3001
		PP_MID_SUM_REF_N	3002
		PP_MID_WIN_REF_N	3003
		PP_SUBAR_SUM_REF_N	3004
		PP_SUBAR_WIN_REF_N	3005
		PP_LUT_REF_N	4000

Table 5: iray enumeration values.

7.1.3.3 ieres enumeration values

The following table shows the enumeration values that can be assigned to the *ieres* parameter.

It is strongly recommended to use enumeration values rather than integer values.

Input	Description	Enumeration value	long
ieres	Extended results vector switch	PP_BAS_RES	0
		PP_EXT_RES	1
	Calculation of main satellite to target related parameters (no derivatives)	PP_TARG_RES_SAT2TARG	62
	Calculation of auxiliary satellite to target related parameters (no derivatives)	PP_TARG_RES_SAT2TARG_AUX	1051840
	Calculation of main satellite to target related parameters (first derivatives)	PP_TARG_RES_SAT2TARG_D	253952
	Calculation of auxiliary satellite to target related parameters (first derivatives)	PP_TARG_RES_SAT2TARG_AUX_D	786432
	Calculation of all satellite to Earth Fixed target related parameters.	PP_TARG_RES_SAT2EF_TARG	31457280
	Calculation of target to Sun related parameters (no derivatives)	PP_TARG_RES_TARG2SUN	671088640
	Calculation of target to Sun related parameters (first derivatives)	PP_TARG_RES_TARG2SUN_D	268435456
	Calculation of satellite to Moon related parameters (no derivatives)	PP_TARG_RES_SAT2MOON	256
	Calculation of satellite to Moon related parameters (first derivatives)	PP_TARG_RES_SAT2MOON_D	33554432
	Calculation of satellite to Sun related parameters (no derivatives)	PP_TARG_RES_SAT2SUN	512
	Calculation of satellite to Sun related parameters (first derivatives)	PP_TARG_RES_SAT2SUN_D	67108864
	Special combination of elements used in AATSR Level1B processor.	PP_TARG_MODE_AATSR_0	6
	Special combination of elements used in AATSR Level1B processor.	PP_TARG_MODE_AATSR_1	134217742
	Special combination of elements used in MERIS Level1B processor.	PP_TARG_MODE_MERIS_0	38
	Special combination of elements used in MERIS Level1B processor.	PP_TARG_MODE_MERIS_1	134217774
	Extended results vector switch, taking into account sunlight travel time effect in computations	PP_BAS_RES_SUNTRAVTIME	1073741824
		PP_EXT_RES_SUNTRAVTIME	1073741825

Table 6: ieres enumeration values.

Input	Description	Enumeration value	long
	Calculation of target to Sun related parameters (no derivatives), taking into account sunlight travel time effect in computations	PP_TARG_RES_TARG2SUN_SUNTRAVTIME	1744830464
	Calculation of target to Sun related parameters (first derivatives), taking into account sunlight travel time effect in computations	PP_TARG_RES_TARG2SUN_D_SUNTRAVTIME	1342177280
	Calculation of satellite to Moon related parameters (no derivatives), taking into account sunlight travel time effect in computations	PP_TARG_RES_SAT2MOON_SUNTRAVTIME	1073742080
	Calculation of satellite to Moon related parameters (first derivatives), taking into account sunlight travel time effect in computations	PP_TARG_RES_SAT2MOON_D_SUNTRAVTIME	1107296256
	Calculation of satellite to Sun related parameters (no derivatives), taking into account sunlight travel time effect in computations	PP_TARG_RES_SAT2SUN_SUNTRAVTIME	1073742336
	Calculation of satellite to Sun related parameters (first derivatives), taking into account sunlight travel time effect in computations	PP_TARG_RES_SAT2SUN_D_SUNTRAVTIME	1140850688
	Special combination of elements used in AATSR Level1B processor, taking into account sunlight travel time effect in computations	PP_TARG_MODE_AATSR_1_SUNTRAVTIME	1207959566
	Special combination of elements used in MERIS Level1B processor, taking into account sunlight travel time effect in computations	PP_TARG_MODE_MERIS_1_SUNTRAVTIME	1207959598

Table 6: ieres enumeration values.

7.1.3.4 dir elements

For each mode defined by *idir*, the *dir* elements are the following:

Mode (idir)	Mode parameter (idir element)	Description (Reference)	Unit (Format)	Allowed Range
-2	[0:7]	*(dummy input)*	-	-
-1	[0:7]	*(dummy input)*	-	-
0	[0:7]	*(dummy input)*	-	-
1 / 2	[0]	Azimuth of the LOS (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
	[1]	Elevation of the LOS (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
	[2]	Geodetic altitude over the Earth (Earth fixed CS)	m	>= $-b_{WGS}$
	[3]	*(dummy input)*	-	-
	[4]	Azimuth-rate of the LOS (Satellite Relative Actual Reference CS)	deg/s	-
	[5]	Elevation-rate of the LOS (Satellite Relative Actual Reference CS)	deg/s	-
	[6:7]	*(dummy input)*	-	-
3	[0]	Azimuth of the LOS to the reference point (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
	[1]	Elevation of the LOS to the reference point (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
	[2]	Geodetic altitude over the Earth (Earth fixed CS)	m	>= $-b_{WGS}$
	[3]	Distance or ground range to the reference point, positive from the nadir in the azimuth direction (Earth fixed CS)	m	-
	[4]	Azimuth-rate of the LOS to the reference point (Satellite Relative Actual Reference CS)	deg/s	-
	[5]	Elevation-rate of the LOS to the reference point (Satellite Relative Actual Reference CS)	deg/s	-
	[6:7]	*(dummy input)*	-	-

Table 7: dir elements.

Mode (idir)	Mode parameter (idir element)	Description (Reference)	Unit (Format)	Allowed Range
4	[0]	Azimuth of the LOS (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
	[1]	Incidence angle of the LOS (Earth fixed CS)	deg	>= 0 <= 90
	[2]	Geodetic altitude over the Earth (Earth fixed CS)	m	>= $-b_{WGS}$
	[3]	*(dummy input)*	-	-
	[4]	Azimuth-rate of the LOS (Satellite Relative Actual Reference CS)	deg/s	-
	[5:7]	*(dummy input)*	-	-
5	[0]	Azimuth of the LOS (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
	[1]	Range to the satellite (Earth fixed CS)	m	> 0
	[2]	Geodetic altitude over the Earth (Earth fixed CS)	m	>= $-b_{WGS}$
	[3]	*(dummy input)*	-	-
	[4]	Azimuth-rate of the LOS (Satellite Relative Actual Reference CS)	deg/s	-
	[5]	Range-rate to the satellite (Earth fixed CS)	m/s	-
	[6:7]	*(dummy input)*	-	-
6	[0]	Range-rate of the related Earth-fixed target (Earth fixed CS)	m/s	-
	[1]	Range or slant-range from target to satellite (Earth fixed CS)	m	> 0
	[2]	Geodetic altitude over the Earth (Earth fixed CS)	m	>= $-b_{WGS}$
	[3]	*(dummy input)*	-	-
	[4]	Range-rate-rate of the related Earth-fixed target (Earth fixed CS)	m/s^2	-
	[5]	Range-rate from target to satellite (Earth fixed CS)	m/s	-
	[6:7]	*(dummy input)*	-	-

Table 7: dir elements.

Mode (idir)	Mode parameter (idir element)	Description (Reference)	Unit (Format)	Allowed Range
7	[0]	Azimuth of the LOS (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
	[1]	Elevation of the LOS (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
	[2:3]	*(dummy input)*	-	-
	[4]	Azimuth-rate of the LOS (Satellite Relative Actual Reference CS)	deg/s	-
	[5]	Elevation-rate of the LOS (Satellite Relative Actual Reference CS)	deg/s	-
	[6:7]	*(dummy input)*	-	-
8	[0]	Azimuth of the LOS (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
	[1]	*(dummy input)*	-	-
	[2]	Geodetic altitude over the Earth (Earth fixed CS)	m	>= $-b_{WGS}$
	[3]	*(dummy input)*	-	-
	[4]	Azimuth-rate of the LOS (Satellite Relative Actual Reference CS)	deg/s	-
	[5:7]	*(dummy input)*	-	-
9	[0]	Right ascension of the star (True of Date CS)	deg	>= 0 < 360
	[1]	Declination of the star (True of Date CS)	deg	>= -90 <= +90
	[2:3]	*(dummy input)*	-	-
	[4]	Right ascension rate of the star (True of Date CS)	deg/s	-
	[5]	Declination rate of the star (True of Date CS)	deg/s	-
	[6:7]	*(dummy input)*	-	-
10	[0:2]	Target position vector (Earth Fixed CS)	m	-
	[3:5]	Target velocity vector (Earth Fixed CS)	m/s	-
	[6:7]	*(dummy input)*	-	-

Table 7: dir elements.

7.1.3.5 Raytracing models

The ray tracing model selected by *iray* is:

iray	Parameters affected	Ray tracing model
0	All	No refraction
1	All but exceptions below	Standard atmosphere refraction
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction of course)
2	All but exceptions below	User's atmosphere refraction
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
3	All but exceptions below	Predefined refraction corrective functions. Average atmosphere.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	target-to-sun related parameters: res[66:71]	No refraction (but the target is computed using refraction)
10	All but exceptions below	Standard atmosphere refraction
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction
20	All but exceptions below	User's atmosphere refraction
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction
30	All but exceptions below	Predefined refraction corrective functions. Average atmosphere.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)

Table 8: raytracing modes.

iray	Parameters affected	Ray tracing model
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction
300	All but exceptions below	Predefined refraction corrective functions. US Standard 76.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	target-to-sun related parameters: res[66:71]	No refraction (but the target is computed using refraction)
301	All but exceptions below	Predefined refraction corrective functions. Tropical.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	target-to-sun related parameters: res[66:71]	No refraction (but the target is computed using refraction)
302	All but exceptions below	Predefined refraction corrective functions. Middle latitudes in summer.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	target-to-sun related parameters: res[66:71]	No refraction (but the target is computed using refraction)
303	All but exceptions below	Predefined refraction corrective functions. Middle latitudes in winter.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	target-to-sun related parameters: res[66:71]	No refraction (but the target is computed using refraction)
304	All but exceptions below	Predefined refraction corrective functions. Subartic latitudes in summer.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	target-to-sun related parameters: res[66:71]	No refraction (but the target is computed using refraction)

Table 8: raytracing modes.

iray	Parameters affected	Ray tracing model
305	All but exceptions below	Predefined refraction corrective functions. Subartic latitudes in winter.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	target-to-sun related parameters: res[66:71]	No refraction (but the target is computed using refraction)
400	All but exceptions below	Predefined refraction corrective functions. Average User -defined Corrective function.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	target-to-sun related parameters: res[66:71]	No refraction (but the target is computed using refraction)
3000	All but exceptions below	Predefined refraction corrective functions. US Standard 76.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction (but the target is computed using refraction)
3001	All but exceptions below	Predefined refraction corrective functions. Tropical.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction (but the target is computed using refraction)
3002	All but exceptions below	Predefined refraction corrective functions. Middle latitudes in summer.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction (but the target is computed using refraction)

Table 8: raytracing modes.

iray	Parameters affected	Ray tracing model
3003	All but exceptions below	Predefined refraction corrective functions. Middle latitudes in winter.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction (but the target is computed using refraction)
3004	All but exceptions below	Predefined refraction corrective functions. Subartic latitudes in summer.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction (but the target is computed using refraction)
3005	All but exceptions below	Predefined refraction corrective functions. Subartic latitudes in winter.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction)
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction (but the target is computed using refraction)
4000	All but exceptions below	Predefined refraction corrective functions. Average User -defined Corrective function.
	earth-fixed target related parameters: res[37:44]	Simplified algorithm (but the target is computed using refraction of course)
	Sun and Moon related parameters: res[51:60] + res[64:71]	No refraction (but the target is computed using refraction of course)

Table 8: raytracing modes.

The following atmospheres are optimized in terms of acuracy for the latitude band they cover. A warning is raised in case the latitude of the tangent point falls outside the latitude band.

Moreover, a warning is also returned if the atmosphere is used in a period of the year for which the data is not optimized.

Description (atmosphere model)	Enumeration value (iray)	Latitude band[a]	Time interval[b]
US Standard 76	PP_US76_REF	[0, 90]	All year
Tropical	PP_TROPIC_REF	[0, 30]	All year
Middle latitudes in Summer	PP_MID_SUM_REF	[30, 66.55]	From March 21st to September 20th
Middle latitudes in Winter	PP_MID_WIN_REF	[30, 66.55]	From September 21st to March 20th
Subartic latitudes in Summer	PP_SUBAR_SUM	[66.55, 90]	From March 21st to September 20th
Subartic latitudes in Winter	PP_SUBAR_WIN	[66.55, 90]	From September 21st to March 20th

Table 9: Latitude Band (Predefined atmospheres)

a. For the predefined atmospheres, latitude bands are symmetric with respect to the equator (the northern latitude band is shown)
b. For the southern latitude band the time interval is the complementary one (except for the first two atmospheres, which are valid all the year).

7.1.3.6 User refraction initialization.

Before calling pp_target using *iray* = 2 or *iray* = 20 (user's refraction ray tracing model) it MUST be called first with *mode* = -1 to initialize the user's file containing the description of the atmosphere.

The name of the file is taken from the environment variable PP_USER_REF_ATM_FILE_NAME, which should be created by the user. If the variable is undefined the function attempts to read, by default, a file named n_z_table.dat, located in the working directory.

The file must contain the co-index of refraction at different geometric altitudes, starting from 0 Km. The altitude should be strict monotonic increasing.

The format of that file must be as follows:

1st column Geometric altitude [m]	**2nd column** Co-index of refraction N]
0.000	262.049
1000.000	238.630
2000.000	216.928
3000.000	195.392
...	...
90000.000	0.001
95000.000	0.000
100000.000	0.000

Table 10: User refraction file format

Note in this table that:

- the relative index of refraction m = 1 + N x 10^{-6}, where N is the co-index of refraction.
- the fields of each row must be separated by blanks

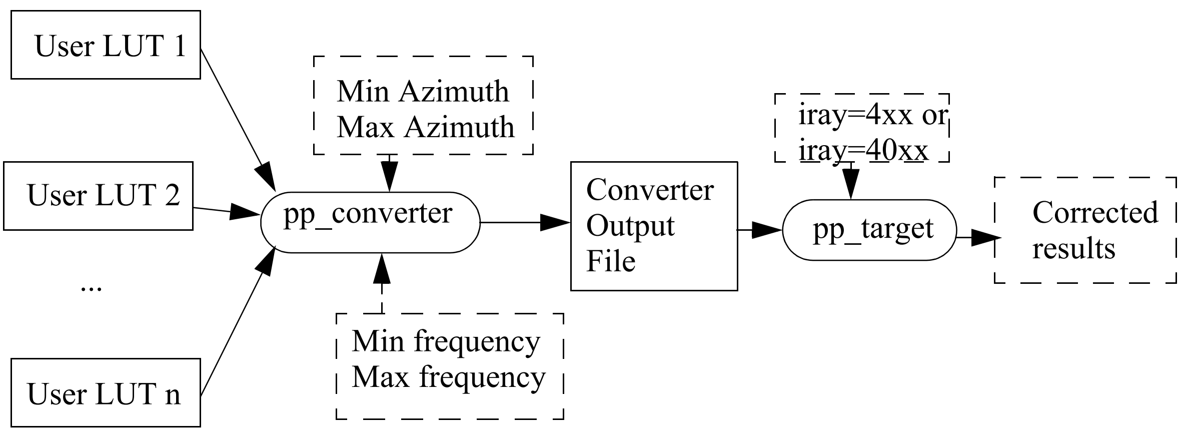
7.1.3.7 Predefined refraction initialization.

Before calling **pp_target** using *iray* = 400 or *iray* = 4000 (user's predefined refraction ray tracing model) it MUST be called first with *mode* = -2 to initialize the user's file containing the description of the atmosphere.

The name of the file is taken from the environment variable PP_USER_REF_CONV_FILE_NAME, which should be created by the user with **pp_converter**. If the variable is not defined the function returns an error.

pp_converter is an executable (from a UNIX shell) program that allows the user to select different atmosphere models to obtain different corrective functions (as for *idir*=30x, but the corrective functions are not predefined), each of them with a given latitude band and a given validity time. The output file name must be defined in the same emviromental variable, PP_USER_REF_CONV_FILE_NAME, before running **pp_converter**.

The overall data flow of **pp_converter** and **pp_target** (iray=4xx and iray=40xx) is as follows:



pp_converter shall be called from a UNIX shell as follows:

```
pp_converter -cif User_LUT_1
             [-cif User_LUT_2]
                 ...
             [-cif User_LUT_n]
             -min_az min_azimuth
             -max_az max_azimuth
             -min_freq min_frequency
             -max_freq max_frequency
             [-v]
             [-pl_v]
             [-po_v]
```

[-pp_v]

taking into account the following considerations:

- Order of parameters does not matter
- Bracketed parameters are not mandatory
- *pl_v* for PPF_LIB verbose mode.
- *po_v* for PPF_ORBIT verbose mode.
- *pp_v* for PPF_POINTING verbose mode.
- *v* for Verbose mode for the 3 libraries (default is silent).
- *h* or *help* option will print the above text on stderr (no execution).

Input parameters are described in the following table:

Keyword	Value after keyword	Type	Description	Unit	Range
cif	User_LUT_i	string	Name of the LUT file (input file, path included). The number of LUTs is arbitrary.	-	-
min_az	min_azimuth	double	Minimum azimuth value of the looking direction to be considered in computations.	deg	0<=min_az<360.0
max_az	max_azimuth	double	Maximum azimuth value of the looking direction to be considered in computations.	deg	0<=max_az<360.0
min_freq	min_frequency	double	Minimum frequency value to be considered in computations.	MHz	0<=min_freq
max_freq	max_frequency	double	Minimum azimuth value of the looking direction to be considered in computations.	MHz	0<=max_freq

Table 11: pp_convert input parameters.

The format of the input file is described in 10.1. In addition to the **pp_converter** output file, some intermediate output files are produced with data that can be plotted. They are named *interm_outp_file_xx.dat* (where *xx* is the number of the LUT file) and *interm_outp_file_av.dat* (for the average) and they can be plotted with **gnuplot** (in a UNIX shell). An example input file for **gnuplot** can be found in the directory *validation* with the name *interm_outp_file_av.gnu* (it plots the intermediate results *interm_outp_file_av.dat*).

For **pp_target**, the selection of the user atmosphere determines the correction to be applied to the parameters of the unrefracted tangent point (tangent altitude, etc.). The following table defines the relation between the iray input parameter and the selected corrective function:

IRAY	Mnemonic	Description
400	PP_LUT_REF	Average User-defined Corrective function
401	PP_LUT_REF+1	First User-defined Corrective function
...		
400+n	PP_LUT_REF+n	Last User-defined Corrective function (*n* being the number of LUT input files for **pp_converter**)
4000	PP_LUT_REF_N	Average User-defined Corrective function (No refraction in Sun and Moon related parameters).
4001	PP_LUT_REF_N+1	First User-defined Corrective function (No refraction in Sun and Moon related parameters).
...		
4000+n	PP_LUT_REF_N+n	Last User-defined Corrective function (No refraction in Sun and Moon related parameters).

Table 12: iray input vs corrective function.

7.1.3.8 Allowed combinations of idir and iray parameters.

The allowed combinations of *idir* and *iray* are:

Mode (idir)	Allowed ray tracing model (iray)
0	0 / 1 / 2 / 3 300 / 301 / 302 / 303 / 304 / 305 / 4xx
1 - 2	0 / 1 / 2 / 10 / 20
3 - 6	0
7 - 9	0 / 1 / 2 / 3 / 10 / 20 / 30 300 / 301 / 302 / 303 / 304 / 305 / 4xx / 3000 / 3001 / 3002 / 3003 / 3004 / 3005 / 40xx
10	0

Table 13: Allowed combinations of idir and iray parameters.

Important:

The standard atmosphere refraction mode (*iray* = 1) is very accurate but extremely slow.

The predefined refraction corrective functions mode (*iray* = 3, 30, 3xx, 4xx, 30xx, 40xx) is very fast (as fast as the no refraction mode *iray* = 0) while still continuous and achieving good accuracy.

7.1.4 Output parameters

The output parameters of the **pp_target** CFI function are:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
pp_target	long	-	Main status flag	-	-1, 0, +1
res [75]	double	all	Results vector	-	-
ierr[15]	long	all	Status vector	-	-

Table 14: pp_target output parameters.

7.1.5 Results vector

pp_target performs calculations for 11 categories of data. It is possible to select individually the calculation of any of those categories, or to make **ANY** combination of them. The different categories are shown below, together with the short mnemonic used in the results vector table and the enumeration value used in *ieres*:

- between satellite and target

Calculation of main satellite to target related parameters (no derivatives)	sat ↔ target (basic)	PP_TARG_RES_SAT2TARG
Calculation of auxiliary satellite to target related parameters (no derivatives)	sat ↔ target (aux)	PP_TARG_RES_SAT2TARG_AUX
Calculation of main satellite to target related parameters (first derivatives)	sat ↔ target (basic deriv)	PP_TARG_RES_SAT2TARG_D
Calculation of auxiliary satellite to target related parameters (first derivatives)	sat ↔ target (aux deriv)	PP_TARG_RES_SAT2TARG_AUX_D

- between satellite and earth-fixed target

Calculation of all satellite to Earth Fixed target related parameters.	sat↔EFtarget	PP_TARG_RES_SAT2EF_TARG

- between sun and target

Calculation of target to Sun related parameters (no derivatives)	sun ↔ target	PP_TARG_RES_TARG2SUN
Calculation of target to Sun related parameters (first derivatives)	sun ↔ target (deriv)	PP_TARG_RES_TARG2SUN_D

- between satellite and sun

Calculation of satellite to Sun related parameters (no derivatives)	sat ↔ sun	PP_TARG_RES_SAT2SUN
Calculation of satellite to Sun related parameters (first derivatives)	sat ↔ sun (deriv)	PP_TARG_RES_SAT2SUN_D

- between satellite and moon

Calculation of satellite to Moon related parameters (no derivatives)	sat ↔ moon	PP_TARG_RES_SAT2MOON
Calculation of satellite to Moon related parameters (first derivatives)	sat ↔ moon (deriv)	PP_TARG_RES_SAT2MOON_D

The user selects the appropriate combination assigning to *ieres* the corresponding enumeration value:

```
ieres = PP_TARG_RES_SAT2TARG;
ieres = PP_TARG_RES_SAT2TARG_AUX + PP_TARG_RES_SAT2TARG_D+
        PP_TARG_RES_SAT2MOON;
ieres = PP_TARG_RES_SAT2EF_TARG + PP_TARG_RES_TARG2SUN;
```

In order to calculate some elements it might be necessary to calculate elements which have not been explicitly requested (e.g. if the derivative of a parameter is requested, the parameter itself will also be calculated). The function identifies internally all the dependencies and those elements are also returned in the result vector.

The results vector table also indicates whether each parameter is part of the basic set of results (category in bold in the next table), *ieres* = 0; or the extended set of results, *ieres* = 1. These two values for *ieres* **CAN NOT** be combined with the previous values.

The results vector of the **pp_target** CFI function is the following (note that there is also an enumeration associated to the elements of the result vector, which **does not have anything to do** with the selection of the outputs, performed throught the *ieres* parameter):

parameter (res elem & enumeration value.)	Category	Description (Reference)	Unit (Format)	Allowed Range
[0:2] PP_TARG_RES_X_COORD PP_TARG_RES_Y_COORD PP_TARG_RES_Z_COORD	**sat ↔ target (basic)**	Cartesian position of the target (X,Y,Z) (Earth fixed CS)	m	-
[3] PP_TARG_RES_GEOC_LONG	**sat ↔ target (basic)**	Geocentric longitude of the target (Earth fixed CS)	deg	>= 0 < 360
[4] PP_TARG_RES_GEOC_LAT	**sat ↔ target (basic)**	Geocentric latitude of the target (Earth fixed CS)	deg	>= -90 <= +90
[5] PP_TARG_RES_GEOD_LAT	**sat ↔ target (basic)**	Geodetic latitude of the target. (Earth fixed CS)	deg	>= -90 <= +90
[6] PP_TARG_RES_GEOD_ALT	**sat ↔ target (basic)**	Geodetic altitude of the target. (Earth fixed CS)	m	-
[7] PP_TARG_RES_TARG2SAT_RANGE	**sat ↔ target (basic)**	Target to satellite range. (Earth fixed CS)	m	>= 0
[8] PP_TARG_RES_TARG2SAT_AZ_TOP	**sat ↔ target (basic)**	Target to satellite azimuth. (Topocentric CS)	deg	>= 0 < 360
[9] PP_TARG_RES_TARG2SAT_EL_TOP	**sat ↔ target (basic)**	Target to satellite elevation. (Topocentric CS)	deg	>= -90 <= +90
[10] PP_TARG_RES_SAT2TARG_AZ_TOP	**sat ↔ target (basic)**	Satellite to target azimuth. (Topocentric CS)	deg	>= 0 < 360
[11] PP_TARG_RES_SAT2TARG_EL_TOP	**sat ↔ target (basic)**	Satellite to target elevation. (Topocentric CS)	deg	>= -90 <= +90
[12] PP_TARG_RES_SAT2TARG_AZ_SRAR	**sat ↔ target (basic)**	Satellite to target azimuth. (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
[13] PP_TARG_RES_ SAT2TARG_EL_SRAR	**sat ↔ target (basic)**	Satellite to target elevation. (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
[14] PP_TARG_RES_DIST_NAD_TARG_GT	**sat ↔ target (aux)**	Minimum distance from the nadir of the target to the ground track. (Earth fixed CS) It is regarded as positive distance when the nadir of the target is located on the left hand side of the ground track.	m	-
[15] PP_TARG_RES_RAD_CUR	**sat ↔ target (aux)**	Radius of curvature in the look direction at the nadir of the target. (Earth fixed CS)	m	>= 0

Table 15: pp_target results vector.

parameter (res elem & enumeration value.)	Category	Description (Reference)	Unit (Format)	Allowed Range
[16] PP_TARG_RES_SIGN_ROUNDTRIP_TIME	**sat ↔ target (basic)**	Signal roundtrip time. (Earth fixed CS)	s	>= 0
[17:19] PP_TARG_RES_X_COORD PP_TARG_RES_Y_COORD PP_TARG_RES_Z_COORD	sat ↔ target (basic deriv)	Cartesian velocity of the target (X,Y,Z) (Earth fixed CS)	m/s	-
[20] PP_TARG_RES_ GEOC_LONG_D	sat ↔ target (basic deriv)	Geocentric longitude-rate of the target. (Earth fixed CS)	deg/s	-
[21] PP_TARG_RES_GEOC_LAT_D	sat ↔ target (basic deriv)	Geocentric latitude-rate of the target. (Earth fixed CS)	deg/s	-
[22] PP_TARG_RES_GEOD_LAT_D	sat ↔ target (basic deriv)	Geodetic latitude-rate of the target. (Earth fixed CS)	deg/s	-
[23] PP_TARG_RES_GEOD_ALT_D	sat ↔ target (basic deriv)	Geodetic altitude-rate of the target. (Earth fixed CS)	m/s	-
[24] PP_TARG_RES_TARG2SAT_RANGE_D	sat ↔ target (basic deriv)	Target to satellite range-rate. (Earth fixed CS)	m/s	-
[25] PP_TARG_RES_TARG2SAT_AZ_TOP_D	sat ↔ target (basic deriv)	Target to satellite azimuth-rate. (Topocentric CS)	deg/s	-
[26] PP_TARG_RES_TARG2SAT_EL_TOP_D	sat ↔ target (basic deriv)	Target to satellite elevation-rate. (Topocentric CS)	deg/s	-
[27] PP_TARG_RES_SAT2TARG_AZ_TOP_D	sat ↔ target (basic deriv)	Satellite to target azimuth-rate. (Topocentric CS)	deg/s	-
[28] PP_TARG_RES_SAT2TARG_EL_TOP_D	sat ↔ target (basic deriv)	Satellite to target elevation-rate. (Topocentric CS)	deg/s	-
[29] PP_TARG_RES_SAT2TARG_ AZ_SRAR_D	sat ↔ target (basic deriv)	Satellite to target azimuth-rate. (Satellite Relative Actual Reference CS)	deg/s	-
[30] PP_TARG_RES_SAT2TARG_ EL_SRAR_D	sat ↔ target (basic deriv)	Satellite to target elevation-rate. (Satellite Relative Actual Reference CS)	deg/s	-
[31] PP_TARG_RES_DIST_NAD_TARG_GT_D	sat ↔ target (aux deriv)	Distance-rate from the nadir of the target to the ground track. (Earth fixed CS)	m/s	-
[32] PP_TARG_RES_RAD_CUR_D	sat ↔ target (aux deriv)	Radius of curvature-rate in the look direction at the nadir of the target (Earth fixed CS)	m/s	-
[33] PP_TARG_RES_NORTH_VEL	sat ↔ target (aux)	Northward component of the velocity relative to the Earth of the nadir of the target. (Topocentric CS)	m/s	-

Table 15: pp_target results vector.

parameter (res elem & enumeration value.)	Category	Description (Reference)	Unit (Format)	Allowed Range
[34] PP_TARG_RES_EAST_VEL	sat ↔ target (aux)	Eastward component of the velocity relative to the Earth of the nadir of the target. (Topocentric CS)	m/s	-
[35] PP_TARG_RES_AZ_VEL	sat ↔ target (aux)	Azimuth of the velocity relative to the Earth of the nadir of the target. (Topocentric CS)	deg	>= 0 < 360
[36] PP_TARG_RES_MAG_VEL	sat ↔ target (aux)	Magnitude of the velocity relative to the Earth of the nadir of the target. (Topocentric CS)	m/s	>= 0
[37] PP_TARG_RES_EF_TARG2SAT_RANGE_D	sat↔EFtarget	Earthfixed target to satellite range-rate. (Earth fixed CS)	m/s	-
[38] PP_TARG_RES_EF_TARG2SAT_AZ_TOP_D	sat↔EFtarget	Earthfixed target to satellite azimuth-rate. (Topocentric CS)	deg/s	-
[39] PP_TARG_RES_EF_TARG2SAT_EL_TOP_D	sat↔EFtarget	Earthfixed target to satellite elevation-rate. (Topocentric CS)	deg/s	-
[40] PP_TARG_RES_SAT2EF_TARG_AZ_TOP_D	sat↔EFtarget	Satellite to earthfixed target azimuth-rate. (Topocentric CS)	deg/s	-
[41] PP_TARG_RES_SAT2EF_TARG_EL_TOP_D	sat↔EFtarget	Satellite to earthfixed target elevation-rate. (Topocentric CS)	deg/s	-
[42] PP_TARG_RES_SAT2EF_TARG_AZ_SRAR_D	sat↔EFtarget	Satellite to earthfixed target azimuth-rate. (Satellite Relative Actual Reference CS)	deg/s	-
[43] PP_TARG_RES_SAT2EF_TARG_EL_SRAR_D	sat↔EFtarget	Satellite to earthfixed target elevation-rate. (Satellite Relative Actual Reference CS)	deg/s	-
[44] PP_TARG_RES_DOPPLER_SHIFT	sat↔EFtarget	2-way Doppler shift of the signal (Earth fixed CS)	Hz	-
[45] PP_TARG_RES_PITCH	**sat ↔ target (basic)**	Satellite pitch angle. (Satellite Reference CS)	deg	>= -180 < +180
[46] PP_TARG_RES_ROLL	**sat ↔ target (basic)**	Satellite roll angle. (Satellite Reference CS)	deg	>= -180 < +180
[47] PP_TARG_RES_YAW	**sat ↔ target (basic)**	Satellite yaw angle. (Satellite Reference CS)	deg	>= -180 < +180
[48] PP_TARG_RES_PITCH_D	**sat ↔ target (basic deriv)**	Satellite pitch rate. (Satellite Reference CS)	deg/s	-

Table 15: pp_target results vector.

parameter (res elem & enumeration value.)	Category	Description (Reference)	Unit (Format)	Allowed Range
[49] PP_TARG_RES_ROLL_D	**sat ↔ target (basic deriv)**	Satellite roll rate. (Satellite Reference CS)	deg/s	-
[50] PP_TARG_RES_YAW_D	**sat ↔ target (basic deriv)**	Satellite yaw rate. (Satellite Reference CS)	deg/s	-
[51] PP_TARG_RES_ SAT2MOON_AZ_SRAR	**sat ↔ moon**	Satellite to Moon (centre) azimuth. (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
[52] PP_TARG_RES_ SAT2MOON_EL_SRAR	**sat ↔ moon**	Satellite to Moon (centre) elevation. (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
[53] PP_TARG_RES_ SAT2SUN_AZ_SRAR	**sat ↔ sun**	Satellite to Sun (centre) azimuth. (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
[54] PP_TARG_RES_ SAT2SUN_EL_SRAR	**sat ↔ sun**	Satellite to Sun (centre) elevation. (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
[55] PP_TARG_RES_SAT2MOON_ AZ_SRAR_D	sat ↔ moon (deriv)	Satellite to Moon (centre) azimuth-rate. (Satellite Relative Actual Reference CS)	deg/s	-
[56] PP_TARG_RES_SAT2MOON_ EL_SRAR_D	sat ↔ moon (deriv)	Satellite to Moon (centre) elevation-rate. (Satellite Relative Actual Reference CS)	deg/s	-
[57] PP_TARG_RES_SAT2SUN_ AZ_SRAR_D	sat ↔ sun (deriv)	Satellite to Sun (centre) azimuth-rate. (Satellite Relative Actual Reference CS)	deg/s	-
[58] PP_TARG_RES_SAT2SUN_ EL_SRAR_D	sat ↔ sun (deriv)	Satellite to Sun (centre) elevation-rate. (Satellite Relative Actual Reference CS)	deg/s	-
[59] PP_TARG_RES_SAT2MOON_ TANG_ALT	**sat ↔ moon**	Tangent altitude over the Earth in the satellite to Moon (centre) look direction. (Earth fixed CS) If looking up (i.e. no tangetn point) this is the altitude of the satellite.	m	-
[60] PP_TARG_RES_ SAT2SUN_TANG_ALT	**sat ↔ sun**	Tangent altitude over the Earth in the satellite to Sun (centre) look direction. (Earth fixed CS) If looking up (i.e. no tangent point) this is the altitude of the satellite.	m	-
[61] PP_TARG_RES_MLST	**sat ↔ target (aux)**	Mean Local Solar Time at target.	decimal hour	>= 0 < 24
[62] PP_TARG_RES_TLST	**sat ↔ target (aux)**	True Local Solar Time at target.	decimal hour	>= 0 < 24

Table 15: pp_target results vector.

parameter (res elem & enumeration value.)	Category	Description (Reference)	Unit (Format)	Allowed Range
[63] PP_TARG_RES_DIST_NAD_TARG_SAT_NAD	**sat ↔ target (aux)**	Distance from the nadir of the target to the satellite nadir. (Earth fixed CS)	m	>= 0
[64] PP_TARG_RES_SAT2MOON_VIS_FLAG	**sat ↔ moon**	Satellite to Moon visibility flag: - 1: Moon eclipsed by the Earth. +1: Moon in sight.		-1, +1
[65] PP_TARG_RES_ SAT2SUN_VIS_FLAG	**sat ↔ sun**	Satellite to Sun visibility flag: - 1: Sun eclipsed by the Earth. +1: Sun in sight.		-1, +1
[66] PP_TARG_RES_ TARG2SUN_AZ_TOP	sun ↔ target	Target to Sun (centre) azimuth. (Topocentric CS)	deg	>= 0 < 360
[67] PP_TARG_RES_ TARG2SUN_EL_TOP	sun ↔ target	Target to Sun (centre) elevation. (Topocentric CS)	deg	>= -90 <= +90
[68] PP_TARG_RES_TARG2SUN_ AZ_TOP_D	sun ↔ target (deriv)	Target to Sun (centre) azimuth-rate. (Topocentric CS)	deg/s	-
[69] PP_TARG_RES_TARG2SUN_ EL_TOP_D	sun ↔ target (deriv)	Target to Sun (centre) elevation-rate. (Topocentric CS)	deg/s	-
[70] PP_TARG_RES_TARG2SUN_TANG_ALT	sun ↔ target	Tangent altitude over the Earth in the target to Sun (centre) look direction (Earth fixed CS)	m	-
[71] PP_TARG_RES_ TARG2SUN_VIS_FLAG	sun ↔ target	Target to Sun visibility flag: - 1: Sun eclipsed by the Earth. +1: Sun in sight.		-1, +1
[72] PP_TARG_RES_ TARG2SUN_VIS_FLAG	**sat ↔ target (aux)**	Right ascension at which the look direction from the satellite to the target points after crossing the Earth atmosphere. (True of Date CS)	deg	>= 0 < 360
[73] PP_TARG_RES_ DEC_LOOK_DIR	**sat ↔ target (aux)**	Declination at which the look direction from the satellite to the target points after crossing the Earth atmosphere. (True of Date CS)	deg	>= -90 <= +90
[74] PP_TARG_RES_DIST_SSP_MIN_DIST_GT	**sat ↔ target (aux)**	Distance from the SSP to the point located on the ground track that is at a minimum distance from the nadir of the target (Earth fixed CS) It is regarded as positive distance when that point is located on the ground track ahead the SSP (in the direction of the motion of the SSP)	m	-

Table 15: pp_target results vector.

7.1.6 AATSR and MERIS Results vector

In order to improve the runtime performance of the **AATSR** and **MERIS** Level 1B processor (geolocation algorithms), special modes have been implemented in the **pp_target** CFI function.

These modes only calculate those parameters required by the corresponding algorithms, reducing the amount of internal calculations to the minimun and thus reaching the highest performance, compatible with a generic tool.

7.1.6.1 AATSR Mode

The *idir* value required in the AATSR geolocation algorithm is PP_AATSR (which is equivalent to PP_INTER_1ST).

The allowed values for *iray* are PP_NO_REF, PP_STD_REF or PP_USER_REF.

There are two possible values for *ieres*: PP_TARG_MODE_AATSR_0 and PP_TARG_MODE_AATSR_1, depending on the specific requirements of the geolocation algorithm. These values for *ieres* **CAN NOT** be combined with any of the other *ieres* values mentioned before.

The following tables show which elements are returned in each case:

PP_TARG_MODE_AATSR_0

parameter (res elem & enumeration value.)	Description (Reference)	Unit (Format)	Allowed Range
[0:2] PP_TARG_RES_X_COORD PP_TARG_RES_Y_COORD PP_TARG_RES_Z_COORD	Cartesian position of the target (X,Y,Z) (Earth fixed CS)	m	-
[3] PP_TARG_RES_GEOC_LONG	Geocentric longitude of the target (Earth fixed CS)	deg	>= 0 < 360
[4] PP_TARG_RES_GEOC_LAT	Geocentric latitude of the target (Earth fixed CS)	deg	>= -90 <= +90
[5] PP_TARG_RES_GEOD_LAT	Geodetic latitude of the target. (Earth fixed CS)	deg	>= -90 <= +90
[6] PP_TARG_RES_GEOD_ALT	Geodetic altitude of the target. (Earth fixed CS)	m	-
[7] PP_TARG_RES_TARG2SAT_RANGE	Target to satellite range. (Earth fixed CS)	m	>= 0
[16] PP_TARG_RES_SIGN_ROUNDTRIP_TIME	Signal roundtrip time. (Earth fixed CS)	s	>= 0
[45] PP_TARG_RES_PITCH	Satellite pitch angle. (Satellite Reference CS)	deg	>= -180 < +180
[46] PP_TARG_RES_ROLL	Satellite roll angle. (Satellite Reference CS)	deg	>= -180 < +180
[47] PP_TARG_RES_YAW	Satellite yaw angle. (Satellite Reference CS)	deg	>= -180 < +180

Table 16: AATSR mode 0 results vector.

(some elements might not be needed by the algorithm but are calculated by **pp_target** as subproducts)

PP_TARG_MODE_AATSR_1

parameter (res elem & enumeration value.)	Description (Reference)	Unit (Format)	Allowed Range
[0:2] PP_TARG_RES_X_COORD PP_TARG_RES_Y_COORD PP_TARG_RES_Z_COORD	Cartesian position of the target (X,Y,Z) (Earth fixed CS)	m	-
[3] PP_TARG_RES_GEOC_LONG	Geocentric longitude of the target (Earth fixed CS)	deg	>= 0 < 360
[4] PP_TARG_RES_GEOC_LAT	Geocentric latitude of the target (Earth fixed CS)	deg	>= -90 <= +90
[5] PP_TARG_RES_GEOD_LAT	Geodetic latitude of the target. (Earth fixed CS)	deg	>= -90 <= +90
[6] PP_TARG_RES_GEOD_ALT	Geodetic altitude of the target. (Earth fixed CS)	m	-
[7] PP_TARG_RES_TARG2SAT_RANGE	Target to satellite range. (Earth fixed CS)	m	>= 0
[8] PP_TARG_RES_TARG2SAT_AZ_TOP	Target to satellite azimuth. (Topocentric CS)	deg	>= 0 < 360
[9] PP_TARG_RES_TARG2SAT_EL_TOP	Target to satellite elevation. (Topocentric CS)	deg	>= -90 <= +90
[16] PP_TARG_RES_SIGN_ROUNDTRIP_TIME	Signal roundtrip time. (Earth fixed CS)	s	>= 0
[45] PP_TARG_RES_PITCH	Satellite pitch angle. (Satellite Reference CS)	deg	>= -180 < +180
[46] PP_TARG_RES_ROLL	Satellite roll angle. (Satellite Reference CS)	deg	>= -180 < +180
[47] PP_TARG_RES_YAW	Satellite yaw angle. (Satellite Reference CS)	deg	>= -180 < +180
[66] PP_TARG_RES_ TARG2SUN_AZ_TOP	Target to Sun (centre) azimuth. (Topocentric CS)	deg	>= 0 < 360
[67] PP_TARG_RES_ TARG2SUN_EL_TOP	Target to Sun (centre) elevation. (Topocentric CS)	deg	>= -90 <= +90

Table 17: AATSR mode 1 results vector.

(some elements might not be needed by the algorithm but are calculated by **pp_target** as subproducts)

7.1.6.2 MERIS Mode

The *idir* value required in the MERIS geolocation algorithm is PP_MERIS_GRID (which is equivalent to PP_GR_RAN).

In this case *iray* has to be set to PP_NO_REF

There are two possible values for *ieres*: PP_TARG_MODE_MERIS_0 and PP_TARG_MODE_MERIS_1, depending on the specific requirements of the geolocation algorithm. These values for *ieres* **CAN NOT** be combined with any of the other *ieres* values mentioned before.

The following tables show which elements are returned in each case:

PP_TARG_MODE_MERIS_0

parameter (res elem & enumeration value.)	Description (Reference)	Unit (Format)	Allowed Range
[0:2] PP_TARG_RES_X_COORD PP_TARG_RES_Y_COORD PP_TARG_RES_Z_COORD	Cartesian position of the target (X,Y,Z) (Earth fixed CS)	m	-
[3] PP_TARG_RES_GEOC_LONG	Geocentric longitude of the target (Earth fixed CS)	deg	>= 0 < 360
[4] PP_TARG_RES_GEOC_LAT	Geocentric latitude of the target (Earth fixed CS)	deg	>= -90 <= +90
[5] PP_TARG_RES_GEOD_LAT	Geodetic latitude of the target. (Earth fixed CS)	deg	>= -90 <= +90
[6] PP_TARG_RES_GEOD_ALT	Geodetic altitude of the target. (Earth fixed CS)	m	-
[12] PP_TARG_RES_SAT2TARG_AZ_SRAR	Satellite to target azimuth. (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
[13] PP_TARG_RES_ SAT2TARG_EL_SRAR	Satellite to target elevation. (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
[16] PP_TARG_RES_SIGN_ROUNDTRIP_TIME	Signal roundtrip time. (Earth fixed CS)	s	>= 0
[45] PP_TARG_RES_PITCH	Satellite pitch angle. (Satellite Reference CS)	deg	>= -180 < +180
[46] PP_TARG_RES_ROLL	Satellite roll angle. (Satellite Reference CS)	deg	>= -180 < +180
[47] PP_TARG_RES_YAW	Satellite yaw angle. (Satellite Reference CS)	deg	>= -180 < +180

Table 18: MERIS mode 0 results vector.

(some elements might not be needed by the algorithm but are calculated by **pp_target** as subproducts)

PP_TARG_MODE_MERIS_1

parameter (res elem & enumeration value.)	Description (Reference)	Unit (Format)	Allowed Range
[0:2] PP_TARG_RES_X_COORD PP_TARG_RES_Y_COORD PP_TARG_RES_Z_COORD	Cartesian position of the target (X,Y,Z) (Earth fixed CS)	m	-
[3] PP_TARG_RES_GEOC_LONG	Geocentric longitude of the target (Earth fixed CS)	deg	>= 0 < 360
[4] PP_TARG_RES_GEOC_LAT	Geocentric latitude of the target (Earth fixed CS)	deg	>= -90 <= +90
[5] PP_TARG_RES_GEOD_LAT	Geodetic latitude of the target. (Earth fixed CS)	deg	>= -90 <= +90
[6] PP_TARG_RES_GEOD_ALT	Geodetic altitude of the target. (Earth fixed CS)	m	-
[7] PP_TARG_RES_TARG2SAT_RANGE	Target to satellite range. (Earth fixed CS)	m	>= 0
[8] PP_TARG_RES_TARG2SAT_AZ_TOP	Target to satellite azimuth. (Topocentric CS)	deg	>= 0 < 360
[9] PP_TARG_RES_TARG2SAT_EL_TOP	Target to satellite elevation. (Topocentric CS)	deg	>= -90 <= +90
[12] PP_TARG_RES_SAT2TARG_AZ_SRAR	Satellite to target azimuth. (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
[13] PP_TARG_RES_ SAT2TARG_EL_SRAR	Satellite to target elevation. (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
[16] PP_TARG_RES_SIGN_ROUNDTRIP_TIME	Signal roundtrip time. (Earth fixed CS)	s	>= 0
[45] PP_TARG_RES_PITCH	Satellite pitch angle. (Satellite Reference CS)	deg	>= -180 < +180
[46] PP_TARG_RES_ROLL	Satellite roll angle. (Satellite Reference CS)	deg	>= -180 < +180
[47] PP_TARG_RES_YAW	Satellite yaw angle. (Satellite Reference CS)	deg	>= -180 < +180
[66] PP_TARG_RES_ TARG2SUN_AZ_TOP	Target to Sun (centre) azimuth. (Topocentric CS)	deg	>= 0 < 360
[67] PP_TARG_RES_ TARG2SUN_EL_TOP	Target to Sun (centre) elevation. (Topocentric CS)	deg	>= -90 <= +90

Table 19: MERIS mode 1 results vector.

(some elements might not be needed by the algorithm but are calculated by **pp_target** as subproducts)

7.1.7 Warnings and errors

Next table lists the possible error messages that can be returned by the **pp_target** CFI function after translating the returned status vector into the equivalent list of error messages by calling the function of the PPF_POINTING software library **pp_vector_msg** (see RD 3).

This table also indicates the type of message returned, i.e. either a warning (WARN) or an error (ERR), the cause of such a message and the impact on the performed calculation, mainly on the results vector.

The table is completed by the error code and value. These error codes can be obtained translating the status vector returned by the **pp_target** CFI function by calling the function of the PPF_POINTING software library **pp_vector_code** (see RD 3).

Note that each error message indicates in brackets to which of the five problems solved by **pp_target** that message relates to, that is:

- Calculation of the signal path from Envisat-1 to a (moving) target ⇒ Sat_tar
- Calculation of the signal path from Envisat-1 to the related (Earth-fixed) target ⇒ Sat_ef_tar
- Calculation of the light path from the Sun to that (moving) target ⇒ Tar_sun
- Calculation of the light path from the Sun to Envisat-1 ⇒ Sat_sun_moon
- Calculation of the light path from the Moon to Envisat-1 ⇒ Sat_sun_moon

Error type	Error message	Cause and impact	Error Code	Error No
ERR	No target was found (Sat_tar)	No target was found No calculation performed	PP_CFI_TARGET_T_TARGET_NOT_FOUND_ERR	0
ERR	Initialization of user's file failed (Sat_tar)	Initialization of user's file when idir = -1 failed	PP_CFI_TARGET_T_FAILED_INIT_ERR	1
ERR	Enviromental variable PP_USER_REF_CONV_FILE_NAME not defined (Sat_tar)	The enviroment variable with the name of the input file has not been defined. Initialization of user's file when idir = -2 failed	PP_CFI_TARGET_T_ENVIRON_VAR_NOT_DEF_ERR	2
ERR	Initialization of predefined refraction user's file failed (Sat_tar)	Initialization of user's file when idir = -2 failed	PP_CFI_TARGET_T_FAILED_INIT_PRED_ERR	3
ERR	Input state vector does not satisfy loose Envisat tolerance requirements (Sat_tar)	Input Envisat-1 state vector does not satisfy the loose tolerance requirements No calculation performed	PP_CFI_TARGET_T_INVALID_PPF_SV_ERR	4
WARN	Input state vector does not satisfy tight Envisat tolerance requirements (Sat_tar)	Input Envisat-1 state vector does not satisfy the tight tolerance requirements	PP_CFI_TARGET_T_INVALID_PPF_SV_WARN	5

Table 20: pp_target warnings and errors.

Error type	Error message	Cause and impact	Error Code	Error No
WARN	Path from Envisat to target occulted by the Earth (Sat_tar)	The path from the target to Envisat-1 is occulted by the Earth, i.e. the tangent altitude of that path is negative No calculation of the parameters associated with the Earth fixed target is performed	PP_CFI_TARGET_T_NEGATIVE_ALTITUDE_WARN	6
WARN	Tangent point latitude is outside the selected corrective function latitude band. (Sat_tar)	The latitude of the output tangent point is outside the latitude band for which the selected corrective function is considered to be valid. Computations performed.	PP_CFI_TARGET_T_PRED_WRONG_LAT_WARN	18
WARN	Input azimuth is outside the selected corrective function azimuth range (Sat_tar)	The azimuth of the line of sight is outside the azimuth band for which the selected corrective function is computed. Computations performed.	PP_CFI_TARGET_T_PRED_WRONG_AZ_WARN	19
ERR	Wrong direction mode swith on input (not allowed switch)	Not allowed idir value No calculation performed	PP_CFI_TARGET_SWITCH_ERROR_MODE_NOT_DEFINED_ERR	128
ERR	Wrong ray tracing switch on input (not allowed switch)	Not allowed iray value No calculation performed	PP_CFI_TARGET_SWITCH_ERROR_RAY_MODE_NOT_DEFINED_ERR	129
ERR	Wrong extended results vector switch on input (not allowed switch)	Not allowed ieres value No calculation performed	PP_CFI_TARGET_SWITCH_ERROR_RES_MODE_NOT_DEFINE_ERR	130
ERR	Wrong combination of switches on input (not allowed combination)	Not allowed combination of idir and iray No calculation performed	PP_CFI_TARGET_SWITCH_ERROR_NOT_ALLOWED_COMB_ERR	131
ERR	Wrong UTC on input (out of range)	Not allowed mjdp[0] value No calculation performed	PP_CFI_TARGET_WRONG_INITIAL_MJDP_DAYS	132
ERR	Wrong DUT1 on input (out of range)	Not allowed mjdp[1] value No calculation performed	PP_CFI_TARGET_WRONG_INITIAL_MJDP_DELTA_UT1	133
ERR	Target not initialized with Converter Output File	Computations not performed.	PP_CFI_TARGET_NO_CONVERTER_ERR	134
ERR	Number of LUT atmosphere greater than the maximum in Converter Output File	Computations not performed.	PP_CFI_TARGET_WRONG_LUT_NUMBER_ERR	135

Table 20: pp_target warnings and errors.

Error type	Error message	Cause and impact	Error Code	Error No
WARN	Input date is outside the selected corrective function time period	The input date is outside the time period for which the selected corrective function is considered to be valid. Computations performed.	PP_CFI_TARGET_PRED_WRONG_TIME_WARN	136
WARN	Input frequency is outside the selected corrective function frequency range	The input date is outside the frequency range for which the selected corrective function is computed. Computations performed.	PP_CFI_TARGET_PRED_WRONG_FREQ_WARN	137

Table 20: pp_target warnings and errors.

7.1.8 Runtime performances

The following runtime performances have been measured.

Due to the large amount of combinations allowed for the *ieres* parameter only some representative cases have been calculated. The runtime for the different predefined refraction corrective functions is the same for all of them, and thus the table only includes the runtimes for *iray* 3 and 30.

For each combination of *idir*, *iray* and *ieres*, two runtimes are provided, one with fixed inputs, i.e. the function has been called several times with the same UT1 time, position and velocity vetors, but modifying the other input parameters (e.g. the values within the *dir* parameter); and a second one with random inputs, i.e all the inputs have been modified from call to call and the average time has been taken.

Note that a **minimum** and a **maximum** execution time values are provided when a refraction raytracing model is used (iray = 1, 2, 10, 20) given that the execution time depends on the calculated ray path to the target, the Sun or the Moon (the closer the ray path to the Earth the bigger the time)

idir	iray	ieres	Ultra Sparc [ms] RANDOM inputs	Ultra Sparc [ms] FIXED inputs
-2	-	-	20.537	20.537
-1	-	-	3.052	3.052
0	0	0 1	2.550 3.006	0.996 1.277
	1	0 1	2.550 <= t <= 5200.0 3.006 <= t <= 12500.0	0.996 <= t <= 4800.0 1.277<= t <= 12400.0
	2	0 1	2.550 <= t <= 6300.0 3.006<= t <= 14500.0	0.996 <= t <= 5500.0 1.277<= t <= 14200.0
	3	0 1	5.126 6.837	3.787 5.087

Table 21: pp_target runtime performance.

idir	iray	ieres	Ultra Sparc [ms] RANDOM inputs	Ultra Sparc [ms] FIXED inputs
1/2	0	0	4.862	3.899
		1	7.105	5.845
		PP_TARG_RES_SAT2TARG	1.244	0.946
		PP_TARG_RES_SAT2TARG_AUX	3.057	2.016
		PP_TARG_RES_SAT2TARG_D	1.701	1.213
		PP_TARG_RES_SAT2TARG_AUX_D	1.744	1.273
		PP_TARG_RES_SAT2EF_TARG	1.747	1.270
		PP_TARG_RES_TARG2SUN	1.856	0.907
		PP_TARG_RES_TARG2SUN_D	2.226	0.940
		PP_TARG_RES_SAT2MOON	1.730	0.763
		PP_TARG_RES_SAT2MOON_D	2.198	0.937
		PP_TARG_RES_SAT2SUN	1.707	0.772
		PP_TARG_RES_SAT2SUN_D	2.183	0.939
		PP_TARG_MODE_AATSR_0	1.071	0.384
		PP_TARG_MODE_AATSR_1	1.747	0.615
	1	0	4.862 <= t <= 5500.0	3.899 <= t <= 5500.0
		1	7.105 <= t <= 11500.0	5.845 <= t <= 11500.0
	2	0	4.862 <= t <= 6000.0	3.899 <= t <= 6000.0
		1	7.105 <= t <= 19500.0	5.845 <= t <= 19500.0
	10	0	4.862 <= t <= 875.0	3.899 <= t <= 200.0
		1	7.105 <= t <= 1150.0	5.845 <= t <= 350.0
	20	0	4.862 <= t <= 600.0	3.899 <= t <= 300.0
		1	7.105 <= t <= 750.0	5.845 <= t <= 350.0
3	0	0	6.175	5.290
		1	8.772	7.827
		PP_TARG_MODE_MERIS_0	2.798	2.259
		PP_TARG_MODE_MERIS_1	3.460	2.448
4	0	0	8.280	7.339
		1	8.772	9.537
5	0	0	5.186	4.131
		1	7.644	6.268
6	0	0	5.460	4.514
		1	7.862	6.615

Table 21: pp_target runtime performance.

idir	iray	ieres	Ultra Sparc [ms] RANDOM inputs	Ultra Sparc [ms] FIXED inputs
7	0	0	4.993	4.113
		1	7.459	6.078
		PP_TARG_RES_SAT2TARG	1.739	1.357
		PP_TARG_RES_SAT2TARG_AUX	4.767	3.673
		PP_TARG_RES_SAT2TARG_D	2.181	1.604
		PP_TARG_RES_SAT2TARG_AUX_D	4.123	3.595
		PP_TARG_RES_SAT2EF_TARG	2.174	1.619
		PP_TARG_RES_TARG2SUN	2.272	1.307
		PP_TARG_RES_TARG2SUN_D	2.687	1.409
		PP_TARG_RES_SAT2MOON	2.033	1.072
		PP_TARG_RES_SAT2MOON_D	2.502	1.229
		PP_TARG_RES_SAT2SUN	1.982	1.042
		PP_TARG_RES_SAT2SUN_D	2.515	1.215
	1	0	4.993 <= t <= 7000.0	4.113 <= t <= 4500.0
		1	7.459 <= t <= 26500.0	6.078 <= t <= 26500.0
	2	0	4.993 <= t <= 8000.0	4.113 <= t <= 5500.0
		1	7.459 <= t <= 25300.0	6.078 <= t <= 24400.0
	3	0	8.022	6.856
		1	13.290	6.078
		PP_TARG_RES_SAT2TARG	3.391	2.939
		PP_TARG_RES_SAT2TARG_AUX	6.756	5.641
		PP_TARG_RES_SAT2TARG_D	4.079	3.500
		PP_TARG_RES_SAT2TARG_AUX_D	6.110	5.510
		PP_TARG_RES_SAT2EF_TARG	4.150	3.570
		PP_TARG_RES_TARG2SUN	4.430	3.472
		PP_TARG_RES_TARG2SUN_D	5.739	4.518
		PP_TARG_RES_SAT2MOON	4.125	3.040
		PP_TARG_RES_SAT2MOON_D	5.085	3.714
		PP_TARG_RES_SAT2SUN	4.056	3.035
		PP_TARG_RES_SAT2SUN_D	5.037	3.807
	10	0	4.993 <= t <= 600.0	4.113 <= t <= 50.0
		1	7.459 <= t <= 800.0	6.078 <= t <= 50.0
	20	0	4.993 <= t <= 450.0	4.113 <= t <= 50.0
		1	7.459 <= t <= 500.0	6.078 <= t <= 50.0
	30	0	7.094	5.790
		1	10.380	8.361

Table 21: pp_target runtime performance.

idir	iray	ieres	Ultra Sparc [ms] RANDOM inputs	Ultra Sparc [ms] FIXED inputs
8	0	0 1	4.929 7.576	3.996 6.930
	1	0 1	4.929 <= t <= 8000.0 7.576 <= t <= 25800.0	3.996 <= t <= 4500.0 6.930 <= t <= 19600.0
	2	0 1	4.929 <= t <= 9500.0 7.576 <= t <= 24900.0	3.996 <= t <= 5500.0 6.930 <= t <= 24900.0
	3	0 1	7.766 12.708	6.611 10.928
	10	0 1	4.929 <= t <= 3000.0 7.576 <= t <= 6900.0	3.996 <= t <= 50.0 6.930 <= t <= 50.0
	20	0 1	4.929 <= t <= 2300.0 7.576 <= t <= 4400.0	3.996 <= t <= 50.0 6.930 <= t <= 50.0
	30	0 1	6.720 9.687	5.616 8.077
9	0	0 1	5.323 7.554	4.101 6.043
	1	0 1	5.323 <= t <= 6500.0 7.554 <= t <= 27500.0	4.101 <= t <= 4500.0 6.043 <= t <= 27500.0
	2	0 1	5.323 <= t <= 8100.0 7.554 <= t <= 26600.0	4.101 <= t <= 5500.0 6.043 <= t <= 24700.0
	3	0 1	8.248 13.560	6.779 11.274
	10	0 1	5.323 <= t <= 600.0 7.554 <= t <= 1000.0	4.101 <= t <= 50.0 6.043 <= t <= 50.0
	20	0 1	5.323 <= t <= 400.0 7.554 <= t <= 650.0	4.101 <= t <= 50.0 6.043 <= t <= 50.0
	30	0 1	7.226 10.034	5.862 8.465
10	0	0 1	4.823 7.061	3.824 5.693

Table 21: pp_target runtime performance.

The following runtime performance has been measured for **pp_converter** with 1 LUT input file (1 user atmosphere). With more input files, the runtime must be multiplied by the number of input atmospheres.

Function	Ultra Sparc [min]
pp_converter (1 atmosphere)	120

Table 22: pp_converter runtime performance.

7.2 pp_stavis

7.2.1 Overview

The **pp_stavis** CFI function calculates the most relevant observation parameters of the link between a Ground station and Envisat-1

7.2.2 Calling interface

The calling interface of the **pp_stavis** CFI function is the following (input parameters are <u>underlined</u>):

```
#include <ppf_pointing.h>
{
       double mjdp[2], pos[3], vel[3], acc[3];
       double sta[4], aocs[3], att[3], datt[3];
       double res[31];
       long status, ierr[4];

       status = pp_stavis (mjdp, pos, vel, acc,
                              aocs, att, datt, sta,
                              res, ierr);
}
```

For Fortran programs the declaration and calling procedure is as follows (input parameters are <u>underlined</u>, note that the C preprocessor must be used because of the presence of the `#include` statement):

```
#include <ppf_pointing.inc>

       REAL*8 MJDP(2), POS(3), VEL(3), ACC(3)
       REAL*8 AOCS(3), ATT(3), DATT(3)
       REAL*8 STA(4), RES(31)
       INTEGER*4 STATUS, IERR(4)

        STATUS = PP_STAVIS(MJDP, POS, VEL, ACC,
      &                     AOCS, ATT, DATT, STA,
      &                     RES, IERR)
```

7.2.3 Input parameters

The **pp_stavis** CFI function has the following input parameters:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
mjdp[2]	double	[0]	Specified UTC (UT1 time)	decimal days (Processing format)	>= -18262.0 < +36525.0
		[1]	Specified ΔUT1 = UT1 - UTC (UT1 time)	s (Processing format)	>= -1.0 <= +1.0
pos[3]	double	all	Satellite position vector at specified UT1 (Earth fixed CS)	m	-
vel[3]	double	all	Satellite velocity vector at specified UT1 (Earth fixed CS)	m/s	-
acc[3]	double	all	Satellite acceleration vector at specified UT1 (Earth fixed CS)	m/s^2	-
aocs[3]	double	[0]	AOCS Cx parameter [pitch] (Satellite Reference CS)	deg	if no better value, assume -0.167074
		[1]	AOCS Cy parameter [roll] (Satellite Reference CS)	deg	if no better value, assume +0.050233
		[2]	AOCS Cz parameter [yaw] (Satellite Reference CS)	deg	if no better value, assume +3.912987
att[3]	double	[0]	Pitch mispointing angle (Satellite Relative Reference CS)	deg	if no better value, assume 0.0
		[1]	Roll mispointing angle (Satellite Relative Reference CS)	deg	if no better value, assume 0.0
		[2]	Yaw mispointing angle (Satellite Relative Reference CS)	deg	if no better value, assume 0.0
datt[3]	double	[0]	Pitch mispointing rate (Satellite Relative Reference CS)	deg/s	if no better value, assume 0.0
		[1]	Roll mispointing rate (Satellite Relative Reference CS)	deg/s	if no better value, assume 0.0
		[2]	Yaw mispointing rate (Satellite Relative Reference CS)	deg/s	if no better value, assume 0.0

Table 23: pp_stavis input parameters.

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
sta[4]	double	[0]	GS geocentric longitude (Earth fixed CS)	deg	>= 0 < 360
		[1]	GS geodetic latitude (Earth fixed CS)	deg	>= -90 < = 90
		[2]	GS geodetic altitude (Earth fixed CS)	m	>= $-b_{WGS}$
		[3]	GS minimum link elevation (Topocentric CS)	deg	>= -90 <= 90

Table 23: pp_stavis input parameters.

7.2.4 Output parameters

The output parameters of the **pp_stavis** CFI function are:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
pp_stavis	long	-	Main status flag	-	-1, 0, +1
res [31]	double	all	Results vector	-	-
ierr[4]	long	all	Status vector	-	-

Table 24: pp_stavis output parameters.

7.2.5 Results vector

The results vector is (note that there is an enumeration associated to the elements of the result vector):

Result parameter (res element)	Description (Reference)	Unit (Format)	Allowed Range
[0] PP_STA_RES_RANGE	Station to satellite range (Earth fixed CS)	m	>= 0
[1] PP_STA_RES_STA2SAT_AZ_TOP	Station to satellite azimuth (Topocentric CS)	deg	>= 0 < 360
[2] PP_STA_RES_ STA2SAT_EL_TOP	Station to satellite elevation (Topocentric CS)	deg	>= -90 <= +90
[3] PP_STA_RES_ SAT2STA_AZ_SRAR	Satellite to station azimuth (Satellite Relative Actual Reference CS)	deg	>= 0 < 360

Table 25: pp_stavis results vector

Result parameter (res element)	Description (Reference)	Unit (Format)	Allowed Range
[4] PP_STA_RES_ SAT2STA_EL_SRAR	Satellite to station elevation (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
[5] PP_STA_RES_RANGE_D	Station to satellite range-rate (Earth fixed CS)	m/s	-
[6] PP_STA_RES_ STA2SAT_AZ_TOP_D	Station to satellite azimuth-rate (Topocentric CS)	deg/s	-
[7] PP_STA_RES_ STA2SAT_EL_TOP_D	Station to satellite elevation-rate (Topocentric CS)	deg/s	-
[8] PP_STA_RES_ STA2SAT_EL_TOP_D	Satellite to station azimuth-rate (Satellite Relative Actual Reference CS)	deg/s	-
[9] PP_STA_RES_ SAT2STA_EL_SRAR_D	Satellite to station elevation-rate (Satellite Relative Actual Reference CS)	deg/s	-
[10] PP_STA_RES_RANGE_2D	Station to satellite range-rate-rate (Earth fixed CS)	m/s^2	-
[11] PP_STA_RES_ STA2SAT_AZ_TOP_2D	Station to satellite azimuth rate-rate (Topocentric CS)	deg/s^2	-
[12] PP_STA_RES_ STA2SAT_EL_TOP_2D	Station to satellite elevation rate-rate (Topocentric CS)	deg/s^2	-
[13] PP_STA_RES_SAT2STA_AZ_SRAR_2D	Satellite to station azimuth rate-rate (Satellite Relative Actual Reference CS)	deg/s^2	-
[14] PP_STA_RES_ SAT2STA_EL_SRAR_2D	Satellite to station elevation rate-rate (Satellite Relative Actual Reference CS)	deg/s^2	-
[15] PP_STA_RES_PITCH	Satellite pitch angle (Satellite Reference CS)	deg	>= -180 < +180
[16] PP_STA_RES_ROLL	Satellite roll angle (Satellite Reference CS)	deg	>= -180 < +180
[17] PP_STA_RES_YAW	Satellite yaw angle (Satellite Reference CS)	deg	>= -180 < +180
[18] PP_STA_RES_PITCH_D	Satellite pitch rate (Satellite Reference CS)	deg/s	-
[19] PP_STA_RES_ROLL_D	Satellite roll rate (Satellite Reference CS)	deg/s	-
[20] PP_STA_RES_YAW_D	Satellite yaw rate (Satellite Reference CS)	deg/s	-
[21] PP_STA_RES_PITCH_2D	Satellite pitch rate-rate (Satellite Reference CS)	deg/s^2	-
[22] PP_STA_RES_ROLL_2D	Satellite roll rate-rate (Satellite Reference CS)	deg/s^2	-

Table 25: pp_stavis results vector

Result parameter (res element)	Description (Reference)	Unit (Format)	Allowed Range
[23] PP_STA_RES_YAW_2D	Satellite yaw rate-rate (Satellite Reference CS)	deg/s^2	-
[24] PP_STA_RES_DISTANCE	Geocentric distance of the station (Earth fixed CS)	m	>= 0
[25:27] PP_STA_RES_X_COORD PP_STA_RES_Y_COORD PP_STA_RES_Z_COORD	Cartesian coordinates of the station (X,Y,Z) (Earth fixed CS)	m	-
[28] PP_STA_RES_OSC_TRUE_LAT	Satellite osculating true latitude (True of Date CS)	deg	>= 0 < 360
[29] PP_STA_RES_OSC_TRUE_LAT_D	Satellite osculating true latitude rate (True of Date CS)	deg/s	-
[30] PP_STA_RES_ OSC_TRUE_LAT_2D	Satellite osculating true latitude rate-rate (True of Date CS)	deg/s^2	-

Table 25: pp_stavis results vector

7.2.6 Warnings and errors

Next table lists the possible error messages that can be returned by the **pp_stavis** CFI function after translating the returned status vector into the equivalent list of error messages by calling the function of the PPF_POINTING software library **pp_vector_msg** (see RD 3).

This table also indicates the type of message returned, i.e. either a warning (WARN) or an error (ERR), the cause of such a message and the impact on the performed calculation, mainly on the results vector.

The table is completed by the error code and value. These error codes can be obtained translating the status vector returned by the **pp_stavis** CFI function by calling the function of the PPF_POINTING software library **pp_vector_code** (see RD 3).

Error type	Error message	Cause and impact	Error code	Error No
ERR	Input state vector does not satisfy loose Envisat tolerance requirements	Input Envisat-1 state vector does not satisfy the loose tolerance requirements No calculation performed	PP_CFI_STAVIS_INVALID_PPF_SV_ERR	0
WARN	Input state vector does not satisfy tight Envisat tolerance requirements	Input Envisat-1 state vector does not satisfy the tight tolerance requirements	PP_CFI_STAVIS_INVALID_PPF_SV_WARN	1
WARN	Ground station elevation below minimum elevation link	The path from the ground station to Envisat-1 does not satisfy the minimum elevation link constraint	PP_CFI_STAVIS_GS_MIN_ELEV_LINK_WARN	2

Table 26: pp_stavis warnings and errors.

Error type	Error message	Cause and impact	Error code	Error No
ERR	Wrong UTC on input (out of range)	Not allowed mjdp[0] value No calculation performed	PP_CFI_STAVIS_NDAYS_ERR	32
ERR	Wrong DUT1 on input (out of range)	Not allowed mjdp[1] value No calculation performed	PP_CFI_STAVIS_DUT1_ERR	33
ERR	Wrong ground station geodetic latitude on input (out of range)	Not allowed sta[1] value No calculation performed	PP_CFI_STAVIS_GS_GLAT	34
ERR	Wrong ground station minimum elevation link (out of range)	Not allowed sta[3] value No calculation performed	PP_CFI_STAVIS_GS_ELEV	35

Table 26: pp_stavis warnings and errors.

The ground station altitude range is not checked, so in case it does not satisfy its allowed range it may result in raising an internal error (see 8).

7.2.7 Runtime performances

The following runtime performances have been measured.

Two runtimes are provided, one with fixed inputs, i.e. the function has been called several times with the same UT1 time, position, velocity and acceleration vectors, but modifying the ground station position (furthermore, this runtime gain can be obtained if a different function, such as **pp_target** or **po_ppforb** is called before with the same time, position, etc. such that the following call to **pp_stavis** benefits from the internal calculations already performed within those functions); and a second one with random inputs, i.e all the inputs have been modified from call to call and the average time has been taken.

Ultra Sparc [ms] RANDOM inputs	Ultra Sparc [ms] FIXED inputs
1.345	0.615

Table 27: pp_stavis runtime performance.

7.3 pp_genstate_drs

7.3.1 Overview

The **pp_genstate_drs** CFI function calculates the cartesian state vector (i.e. position and velocity vectors expressed in the Earth fixed coordinate system) of the DRS at a specified UT1 time

7.3.2 Calling interface

The calling interface of the **pp_genstate_drs** CFI function is the following (input parameters are underlined):

```
#include <ppf_pointing.h>
{
       double mjddrs[2], longdrs, posdrs[3], veldrs[3];
       long status, ierr[4];

       status = pp_genstate_drs (mjddrs, &longdrs,
                                  posdrs, veldrs, ierr);
}
```

For Fortran programs the declaration and calling procedure is as follows (input parameters are underlined, note that the C preprocessor must be used because of the presence of the `#include` statement):

```
#include <ppf_pointing.inc>

       REAL*8 MJDDRS(2), LONGDRS
       REAL*8 POSDRS(3), VELDRS(3)
       INTEGER*4 STATUS, IERR(4)

       STATUS = PP_GENSTATE_DRS(MJDDRS, LONGDRS,
                                POSDRS, VELDRS, IERR)
```

7.3.3 Input parameters

The **pp_genstate_drs** CFI function has the following input parameters:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
mjd[2]	double	[0]	DUMMY	-	-
		[1]	DUMMY	-	-
longdrs	long *	-	DRS reference longitude (Earth fixed CS)	deg	>= 0 < 360

Table 28: pp_genstate_drs input parameters.

7.3.4 Output parameters

The output parameters of the **pp_genstate_drs** CFI function are:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
pp_genstate_drs	long	-	Main status flag	-	-1, 0, +1
posdrs[3]	double	all	DRS position vector (Earth fixed CS)	m	-
veldrs[3]	double	all	DRS velocity vector (Earth fixed CS)	m/s	-
ierr[4]	long	all	Status vector	-	-

Table 29: pp_genstate_drs output parameters.

7.3.5 Warnings and errors

Next table lists the possible error messages that can be returned by the **pp_genstate_drs** CFI function after translating the returned status vector into the equivalent list of error messages by calling the function of the PPF_POINTING software library **pp_vector_msg** (see RD 3).

This table also indicates the type of message returned, i.e. either a warning (WARN) or an error (ERR), the cause of such a message and the impact on the performed calculation, mainly on the results vector.

The table is completed by the error code and value. These error codes can be obtained translating the status vector returned by the **pp_genstate_drs** CFI function by calling the function of the PPF_POINTING software library **pp_vector_code** (see RD 3).

Error type	Error message	Cause and impact	Error Code	Error No
	(no errors)			

Table 30: pp_genstate_drs warnings and errors.

7.3.6 Runtime performances

The following runtime performances have been measured.

Two runtimes are provided, one with fixed inputs, i.e. the functions has been called several times with the same UT1 time, but modifying the other input parameters; and a second one with random inputs, i.e all the inputs have been modified from call to call and the average time has been taken.

Ultra Sparc [ms] RANDOM inputs	Ultra Sparc [ms] FIXED inputs
0.01	0.005

Table 31: pp_genstate_drs runtime performance.

7.4 pp_drsvis

7.4.1 Overview

The **pp_drsvis** CFI function calculates the most relevant observation parameters of the link between the DRS and Envisat-1

7.4.2 Calling interface

The calling interface of the **pp_drsvis** CFI function is the following (input parameters are underlined):

```
#include <ppf_pointing.h>
{
       double mjdp[2], pos[3], vel[3], acc[3];
       double aocs[3], att[3], datt[3];
       double mjddrs[2], posdrs[3], veldrs[3];
       double res[33];
       long status, ierr[4];

       status = pp_drsvis(mjdp, pos, vel, acc,
                          aocs, att, datt,
                          mjddrs, posdrs, veldrs,
                          res, ierr);
}
```

For Fortran programs the declaration and calling procedure is as follows (input parameters are underlined, note that the C preprocessor must be used because of the presence of the `#include` statement):

```
#include <ppf_pointing.inc>

       REAL*8 MJDP(2), POS(3), VEL(3), ACC(3)
       REAL*8 AOCS(3), ATT(3), DATT(3)
       REAL*8 MJDDRS(2), POSDRS(3), VELDRS(3)
       REAL*8 RES(33)
       INTEGER*4 STATUS, IERR(4)

       STATUS = PP_DRSVIS(MJDP, POS, VEL, ACC,
      &                   AOCS, ATT, DATT,
      &                   MJDDRS, POSDRS, VELDRS,
      &                   RES, IERR)
```

7.4.3 Input parameters

The **pp_drsvis** CFI function has the following input parameters:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
mjdp[2]	double	[0]	Satellite UTC (UT1 time)	decimal days (Processing format)	>= -18262 < +36525
		[1]	Satellite ∆UT1 (UT1 time)	s (Processing format)	>= -1.0 <= +1.0
pos[3]	double	all	Satellite position vector at specified UT1 (Earth fixed CS)	m	-
vel[3]	double	all	Satellite velocity vector at specified UT1 (Earth fixed CS)	m/s	-
acc[3]	double	all	Satellite acceleration vector at specified UT1 (Earth fixed CS)	m/s^2	-
aocs[3]	double	[0]	AOCS Cx parameter [pitch] (Satellite Reference CS)	deg	if no better value, assume -0.167074
		[1]	AOCS Cy parameter [roll] (Satellite Reference CS)	deg	if no better value, assume +0.050233
		[2]	AOCS Cz parameter [yaw] (Satellite Reference CS)	deg	if no better value, assume +3.912987
att[3]	double	[0]	Pitch mispointing angle (Satellite Relative Reference CS)	deg	if no better value, assume 0.0
		[1]	Roll mispointing angle (Satellite Relative Reference CS)	deg	if no better value, assume 0.0
		[2]	Yaw mispointing angle (Satellite Relative Reference CS)	deg	if no better value, assume 0.0
datt[3]	double	[0]	Pitch mispointing rate (Satellite Relative Reference CS)	deg/s	if no better value, assume 0.0
		[1]	Roll mispointing rate (Satellite Relative Reference CS)	deg/s	if no better value, assume 0.0
		[2]	Yaw mispointing rate (Satellite Relative Reference CS)	deg/s	if no better value, assume 0.0

Table 32: pp_drsvis input parameters.

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
mjddrs[2][a]	double	[0]	DRS reference UTC (UT1 time)	decimal days (Processing format)	= mjdp[0]
		[1]	DRS reference ΔUT1 (UT1 time)	s (Processing format)	= mjdp[1]
posdrs[3]	double	all	DRS position vector at reference UT1 (Earth fixed CS)	m	-
veldrs[3]	double	all	DRS velocity vector at reference UT1 (Earth fixed CS)	m/s	-

Table 32: pp_drsvis input parameters.

a. Note that mjddrs must have the same value as mjdp, as **pp_drsvis** is computing visibility between the positions of Envisat-1 and the DRS satellite at the same time.
2 separate parameters have to be passed anyway, for historical reasons.

7.4.4 Output parameters

The output parameters of the **pp_drsvis** CFI function are:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
pp_drsvis	long	-	Main status flag	-	-1, 0, +1
res [33]	double	all	Results vector	-	-
ierr[4]	long	all	Status vector	-	-

Table 33: pp_drsvis output parameters

7.4.5 Results vector

The results vector is (note that there is an enumeration associated to the elements of the result vector):

Result parameter (res element)	Description (Reference)	Unit (Format)	Allowed Range
[0] PP_DRS_RES_RANGE	DRS to satellite range (Earth fixed CS)	m	>= 0
[1] PP_DRS_RES_DRS2SAT_AZ_TOP	DRS to satellite azimuth (Topocentric CS)	deg	>= 0 < 360
[2] PP_DRS_RES_ DRS2SAT_EL_TOP	DRS to satellite elevation (Topocentric CS)	deg	>= -90 <= +90
[3] PP_DRS_RES_ SAT2DRS_AZ_SRAR	Satellite to DRS azimuth (Satellite Relative Actual Reference CS)	deg	>= 0 < 360
[4] PP_DRS_RES_ SAT2DRS_EL_SRAR	Satellite to DRS elevation (Satellite Relative Actual Reference CS)	deg	>= -90 <= +90
[5] PP_DRS_RES_RANGE_D	DRS to satellite range-rate (Earth fixed CS)	m/s	-
[6] PP_DRS_RES_ DRS2SAT_AZ_TOP_D	DRS to satellite azimuth-rate (Topocentric CS)	deg/s	-
[7] PP_DRS_RES_ DRS2SAT_EL_TOP_D	DRS to satellite elevation-rate (Topocentric CS)	deg/s	-
[8] PP_DRS_RES_ DRS2SAT_EL_TOP_D	Satellite to DRS azimuth-rate (Satellite Relative Actual Reference CS)	deg/s	-
[9] PP_DRS_RES_ SAT2DRS_EL_SRAR_D	Satellite to DRS elevation-rate (Satellite Relative Actual Reference CS)	deg/s	-
[10] PP_DRS_RES_RANGE_2D	DRS to satellite range-rate-rate (Earth fixed CS)	m/s^2	-
[11] PP_DRS_RES_ DRS2SAT_AZ_TOP_2D	DRS to satellite azimuth rate-rate (Topocentric CS)	deg/s^2	-
[12] PP_DRS_RES_ DRS2SAT_EL_TOP_2D	DRS to satellite elevation rate-rate (Topocentric CS)	deg/s^2	-
[13] PP_DRS_RES_SAT2DRS_AZ_SRAR_2D	Satellite to DRS azimuth rate-rate (Satellite Relative Actual Reference CS)	deg/s^2	-
[14] PP_DRS_RES_ SAT2DRS_EL_SRAR_2D	Satellite to DRS elevation rate-rate (Satellite Relative Actual Reference CS)	deg/s^2	-
[15] PP_DRS_RES_PITCH	Satellite pitch angle (Satellite Reference CS)	deg	>= -180 < +180
[16] PP_DRS_RES_ROLL	Satellite roll angle (Satellite Reference CS)	deg	>= -180 < +180

Table 34: pp_drsvis results vector.

Result parameter (res element)	Description (Reference)	Unit (Format)	Allowed Range
[17] PP_DRS_RES_YAW	Satellite yaw angle (Satellite Reference CS)	deg	>= -180 < +180
[18] PP_DRS_RES_PITCH_D	Satellite pitch rate (Satellite Reference CS)	deg/s	-
[19] PP_DRS_RES_ROLL_D	Satellite roll rate (Satellite Reference CS)	deg/s	-
[20] PP_DRS_RES_YAW_D	Satellite yaw rate (Satellite Reference CS)	deg/s	-
[21] PP_DRS_RES_PITCH_2D	Satellite pitch rate-rate (Satellite Reference CS)	deg/s^2	-
[22] PP_DRS_RES_ROLL_2D	Satellite roll rate-rate (Satellite Reference CS)	deg/s^2	-
[23] PP_DRS_RES_YAW_2D	Satellite yaw rate-rate (Satellite Reference CS)	deg/s^2	-
[24] PP_DRS_RES_DISTANCE	Geocentric distance of the DRS (Earth fixed CS)	m	>= 0
[25:27] PP_DRS_RES_X_COORD PP_DRS_RES_Y_COORD PP_DRS_RES_Z_COORD	Cartesian coordinates of the DRS (X, Y, Z) (Earth fixed CS)	m	-
[28] PP_DRS_RES_OSC_TRUE_LAT	Satellite osculating true latitude (True of Date CS)	deg	>= 0 < 360
[29] PP_DRS_RES_OSC_TRUE_LAT_D	Satellite osculating true latitude rate (True of Date CS)	deg/s	-
[30] PP_DRS_RES_ OSC_TRUE_LAT_2D	Satellite osculating true latitude rate-rate (True of Date CS)	deg/s^2	-
[31] PP_DRS_RES_TANG_ALTITUDE	Tangent altitude over the Earth in the satellite to DRS look direction. (Earth fixed CS)	m	-
[32] PP_DRS_RES_TANG_ALTITUDE_D	Rate of the tangent altitude over the Earth in the satellite to DRS look direction (Earth fixed CS)	m/s	-

Table 34: pp_drsvis results vector.

7.4.6 Warnings and errors

Next table lists the possible error messages that can be returned by the **pp_drsvis** CFI function after translating the returned status vector into the equivalent list of error messages by calling the function of the PPF_POINTING software library **pp_vector_msg** (see RD 3).

This table also indicates the type of message returned, i.e. either a warning (WARN) or an error (ERR), the cause of such a message and the impact on the performed calculation, mainly on the results vector.

The table is completed by the error code and value. These error codes can be obtained translating the status vector returned by the **pp_drsvis** CFI function by calling the function of the PPF_POINTING software library **pp_vector_code** (see RD 3).

Error type	Error message	Cause and impact	Error Code	Error No
ERR	Input state vector does not satisfy loose Envisat tolerance requirements	Input Envisat-1 state vector does not satisfy the loose tolerance requirements No calculation performed	PP_CFI_DRSVIS_INVALID_PPF_SV_ERR	0
WARN	Input state vector does not satisfy tight Envisat tolerance requirements	Input Envisat-1 state vector does not satisfy the tight tolerance requirements	PP_CFI_DRSVIS_INVALID_PPF_SV_WARN	1
ERR	Input state vector does not satisfy loose DRS tolerance requirements	Input DRS state vector does not satisfy the loose tolerance requirements No calculation performed	PP_CFI_DRSVIS_INVALID_DRS_SV_ERR	2
WARN	Input state vector does not satisfy tight DRS tolerance requirements	Input DRS state vector does not satisfy the tight tolerance requirements	PP_CFI_DRSVIS_INVALID_DRS_SV_ERR	3
ERR	Envisat and DRS UT1 time are not the same	The mjdp[] and mjddrs[] are not the same No calculation performed	PP_CFI_DRSVIS_DRS_PPF_TIME_ERR	4
WARN	Path from Envisat to DRS occulted by the Earth	The path from the DRS to Envisat-1 is occulted by the Earth, i.e. the tangent altitude of the path is negative	PP_CFI_DRSVIS_NEGATIVE_TANG_DRS_ALT_WARN	10
ERR	Wrong UTC on input (out of range)	Not allowed mjdp[0] value No calculation performed	PP_CFI_DRSVIS_NDAYS_ERR	32
ERR	Wrong DUT1 on input (out of range)	Not allowed mjdp[1] value No calculation performed	PP_CFI_DRSVIS_DUT1_ERR	33

Table 35: pp_drsvis warnings and errors.

7.4.7 Runtime performances

The following runtime performances have been measured.

Two runtimes are provided, one with fixed inputs, i.e. the function has been called several times with the same UT1 time, position, velocity and acceleration vectors, but modifying the DRS position (furthermore, this runtime gain can be obtained if a different function, such as **pp_target** or **po_ppforb** is called before with the same time, position, etc. such that the following call to **pp_drsvis** benefits from the internal calculations already performed within those functions); and a second one with random inputs, i.e all the inputs have been modified from call to call and the average time has been taken.

Ultra Sparc [ms] RANDOM inputs	Ultra Sparc [ms] FIXED inputs
1.921	1.270

Table 36: pp_drsvis runtime performances.

7.5 pp_init_attitude_file

7.5.1 Overview

The **pp_init_attitude_file** CFI function reads in memory an attitude file, as defined in RD 7, and calculates (optionally) some statistics and performance parameters.

The calculation of statistics depends on the selected model at initialisation (see table 42 for enumerations).

Enumeration value	Performance
PP_NO_STATISTIC	No statistics calculated - bias_flag = rms_flag = bias_good = rms_good = [0,0,0] stat_flags = [0,0]
PP_STATISTIC	Statistics calculated. See table 38 for further selection of calculated data.

Table 37: Statistics calculation.

If statistics are selected, the performance parameter calculation depends now on the selected model at initialisation (see table 41 for enumerations).

Enumeration value	Performance
PP_NO_PERFO	Do not estimate performance - bias_good and rms_good is calculated for all given data in user specified interval - bias_flag = [0,0,0] - rms_flag = [0,0,0] - flag_stats[0] = 0
PP_ESTIMATOR_PERFO	If one or more of the difference between estimator and harmonic values for a time is abs.larger than the abs. threshold (perfo_params[0]-[2]) then data is flagged from t minus time_constant till t plus time_constant (time_constant = perfo_params[3]) - bias_good and rms_good is calculated for all non-flagged data in user specified interval - bias_flag and rms_flag is calculated for all flagged data in user specified interval

Table 38: Performance calculation.

The values selected at initialisation affect the way outputs are calculated in **pp_get_attitude_ aocs**.

7.5.2 Calling interface

The calling interface of the **pp_init_attitude_file** CFI function is the following (input parameters are underlined):

```
#include <ppf_pointing.h>
{
   char   *filename;
   long   mode_out, mode_perfo;
   double perfo_params[6];
   long   mode_statistic;
   double bias_good[3], rms_good[3];
   double bias_flag[3], rms_flag[3];
   double flag_stats[2];
   long status, ierr[1];
   double start, stop;

   status = pp_init_attitude_file(filename, &mode_out,
                                  &start, &stop,
                                  &mode_perfo,
                                  perfo_params, &mode_statistic,
                                  bias_good, rms_good,
                                  bias_flag, rms_flag,
                                  flag_stats, ierr);
}
```

For Fortran programs the declaration and calling procedure is as follows (input parameters are underlined, note that the C preprocessor must be used because of the presence of the `#include` statement):

```
TBD
```

7.5.3 Input parameters

The **pp_init_attitude_file** CFI function has the following input parameters:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
filename	char*	all	Attitude filename (only one file) See RD 7 for file format.	-	-
mode_out	long *	-	Flag to select the model to be used	-	See table 40
utc_start	double *	-	Start of time segment for processing (UTC)	decimal days (Processing format)	>= -18262 < +36525
utc_stop	double *	-	Stop of time segment for processing (UTC)	decimal days (Processing format)	>= -18262 < +36525
mode_perfo	long *	-	Flag to select performance calculations	-	See table 41
perfo_params[6]	double	[0]	pitch attitude level	deg	>= 0 < +180
		[1]	roll attitude level	deg	>= 0 < +180
		[2]	yaw attitude level	deg	>= 0 < +180
		[3]	time constant	sec	>0
		[4]	unused		
		[5]	unused		
mode_statistic	long *	-	Flag to select statistics calculations	-	See table 42

Table 39: pp_init_attitude_file input parameters.

7.5.3.1 mode_out enumeration values

The following table shows the enumeration values that can be assigned to the *mode_out* parameter.
It is strongly recommended to use enumeration values rather than integer values.

Input	Description	Enumeration value	long
mode_out	Attitude Model	PP_ATT_HARMONIC	0
		PP_ATT_ESTIMATOR	1

Table 40: mode_out enumeration values.

7.5.3.2 mode_perfo enumeration values

The following table shows the enumeration values that can be assigned to the *mode_perfo* parameter.
It is strongly recommended to use enumeration values rather than integer values.

Input	Description	Enumeration value	long
mode_perfo	Flag to select the performance calculation	PP_NO_PERFO	0
		PP_ESTIMATOR_PERFO	1

Table 41: mode_perfo enumeration values.

7.5.3.3 mode_statistic enumeration values

The following table shows the enumeration values that can be assigned to the *mode_statistic* parameter.
It is strongly recommended to use enumeration values rather than integer values.

Input	Description	Enumeration value	long
mode_statistic	Flag to select the statistics calculation	PP_NO_STATISTIC	0
		PP_STATISTIC	1

Table 42: mode_statistic enumeration values.

7.5.4 Output parameters

The output parameters of the **pp_init_attitude_file** CFI function are:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
pp_init_attitude_file	long	-	Main status flag	-	-1, 0, +1
bias_good[3]	double	[0]	Average of pitch over time interval for data not flagged	deg	>= -180 < +180
		[1]	Average of roll over time interval for data not flagged	deg	>= -180 < +180
		[2]	Average of yaw over time interval for data not flagged	deg	>= -180 < +180
rms_good[3]	double	[0]	Std dev of pitch over time interval for data not flagged	deg	>0
		[1]	Std dev of roll over time interval for data not flagged	deg	>0
		[2]	Std dev of yaw over time interval for data not flagge	deg	>0
bias_flag[3]	double	[0]	Average of pitch over time interval for data flagged	deg	>= -180 < +180
		[1]	Average of roll over time interval for data flagged	deg	>= -180 < +180
		[2]	Average of yaw over time interval for data flagged	deg	>= -180 < +180
rms_flag[3]	double	[0]	Std dev of pitch over time interval for data flagged	deg	>0
		[1]	Std dev of roll over time interval for data flagged	deg	>0
		[2]	Std dev of yaw over time interval for data flagged	deg	>0
flag_stats[2]	double	[0]	Percentage of data flagged over the time interval	-	0-100
		[1]	Percentage of time gap over the time interval	-	0-100
ierr[1]	long	-	Status vector	-	-

Table 43: pp_init_attitude_file output parameters.

7.5.5 Warnings and errors

Next table lists the possible error messages that can be returned by the **pp_init_attitude_file** CFI function after translating the returned status vector into the equivalent list of error messages by calling the function of the PPF_POINTING software library **pp_vector_msg** (see RD 3).

This table also indicates the type of message returned, i.e. either a warning (WARN) or an error (ERR), the cause of such a message and the impact on the performed calculation, mainly on the results vector.

The table is completed by the error code and value. These error codes can be obtained translating the status vector returned by the **pp_init_attitude_file** CFI function by calling the function of the PPF_POINTING software library **pp_vector_code** (see RD 3).

Error type	Error message	Cause and impact	Error Code	Error No
ERR	ERROR in pp_init_attitude_file: EOF reached	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_EOF_ERR	0
ERR	Wrong inputs: "Start Time" is after "Stop Time"	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_WRONG_INPUT_TIMES_ERR	1
ERR	ncorrect input for "mode_statistics" parameter	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_STATISTIC_PARAM_ERR	2
ERR	Incorrect input for "mode_out" parameter	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_MODE_OUT_PARAM_ERR	3
ERR	Incorrect input for "mode_perfo" parameter	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_MODE_PERFO_PARAM_ERR	4
ERR	"perfo_params" are not within the allowed ranges"	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_PERFO_PARAM_ERR	5
ERR	Cannot open the Attitude file	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_OPEN_FILE_ERR	6
ERR	Time transformation error	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_TIME_TRANSFORMATION_ERR	7
ERR	Input interval out of the limits of the file	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_OUT_OF_FILE_ERR	8
ERR	Memory allocation error	No calculation performed	PP_CFI_INIT_ATTITUDE_FILE_MEMORY_ERR	9
WARN	Input file does not cover the requiered time interval	Time from file used to initialise.	PP_CFI_INIT_ATTITUDE_FILE_OUT_FILE_WARN	10
WARN	Not enough estimated data to compute statistics and performance flags	Statistic not computed. Performance no computed.	PP_CFI_INIT_ATTITUDE_FILE_NO_ESTIMATORS_WARN	11

Table 44: pp_init_attitude_file warnings and errors.

7.5.6 Runtime performances

The following runtime performances have been measured.

Ultra Sparc [ms] FIXED inputs
TBD

Table 45: pp_init_attitude_file runtime performances.

7.6 pp_get_attitude_aocs

7.6.1 Overview

The **pp_get_attitude_aocs** CFI function calculates the Pitch, Roll and Yaw mispointing angles, from the input file initialised in **pp_init_attitude_file,** for a given time. If selected, it also returns some performance data.

The algorithm used to calculate the output pitch, roll and yaw misponting depends on the selected model at initialisation (see table 40).

Enumeration value	Algorithm
PP_ATT_HARMONIC	Calculate att and datt using linear, orbital and daily harmonics only
PP_ATT_ESTIMATOR	Calculate att and datt using a linear interpolation (only below 1.9 sec) between the estimator data.

Table 46: Algorithm.

The performance is only returned if the performance flag was set at initialisation time (see table 38).

Enumeration value at initialisation	Performance
PP_NO_PERFO	perfo_flag = PP_NO_PERFORMANCE, PP_NO_PERFORMANCE_GAP
PP_ESTIMATOR_PERFO	perfo_flag = PP_ATTITUDE_NOMINAL, PP_ATTITUDE_FLAGGED, PP_ATTITUDE_FLAGGED_GAP OR PP_ESTIMATOR_GAP

Table 47: Performance calculation.

See section section 7.6.4.1 for more details on how performance is calculated.

7.6.2 Calling interface

The calling interface of the **pp_get_attitude_aocs** CFI function is the following (input parameters are underlined):

#include <ppf_pointing.h>

```
{
double utc_time;
   double aocs[3];
   double att[3];
   double datt[3];
   long perfo_flag;
   char mode[5];
   long status, ierr[1];

   status = pp_get_attitude_aocs(&utc_time, aocs,
                                att, datt, &perfo_flag,
                                mode, ierr);
}
```

For Fortran programs the declaration and calling procedure is as follows (input parameters are underlined, note that the C preprocessor must be used because of the presence of the `#include` statement):

```
TBD
```

7.6.3 Input parameters

The **pp_get_attitude_aocs** CFI function has the following input parameters:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
utc_time	double *	-	Time for aocs calculation (UTC)	decimal days (Processing format)	>= -18262 < +36525

Table 48: pp_get_attitude_aocs input parameters.

7.6.4 Output parameters

The output parameters of the **pp_get_attitude_aocs** CFI function are:

C name	C type	Array Element	Description (Reference)	Unit (Format)	Allowed Range
pp_get_attitude_aocs	long	-	Main status flag	-	-1, 0, +1
aocs[3]	double	[0]	aocs harmonics, (copied from header)	deg	-
		[1]	aocs harmonics, (copied from header)	deg	-
		[2]	aocs harmonics, (copied from header)	deg	-
att[3]	double	[0]	Pitch mispointing angle (Satellite Relative Refer ence CS)	deg	>= -180 < +180
		[1]	Roll mispointing angle (Satellite Relative Reference CS)	deg	>= -180 < +180
		[2]	Yaw mispointing angle (Satellite RelativeReference CS)	deg	>= -180 < +180
datt[3]	double	[0]	Pitch mispointing rate (Satellite Relative Reference CS)	deg/s	>0
		[1]	Roll mispointing rate (Satellite Relative Reference CS)	deg/s	>0
		[2]	Yaw mispointing rate (Satellite RelativeReference CS)	deg/s	>0
perfo_flag	long	-	Flag to indicate the performance for the input time	-	See table 50
mode[5]	char	-	Attitude Mode	-	("SYSM", etc TBD)
ierr[1]	long	-	Status vector	-	-

Table 49: pp_get_attitude_aocs output parameters.

7.6.4.1 perfo_flag enumeration values

The following table shows the enumeration values that returned by the *perfo_flag* parameter.

It is strongly recommended to use enumeration values rather than integer values.

Output	Description	Enumeration value	long
perfo_flag	Performance flag	PP_NO_PERFORMANCE	0
		PP_NO_PERFORMANCE_GAP	1
		PP_ATTITUDE_NOMINAL	2
		PP_ATTITUDE_FLAGGED	3
		PP_ATTITUDE_FLAGGED_GAP	4
		PP_ESTIMATOR_GAP	5

Table 50: perfo_flag enumeration values.

When initialising with PP_NO_PERFO the performace flag is not calculated. In case the requested time is within a gap (estimator data separation is greater than 1.9 seconds), *perfo_flag* returns PP_NO_PERFORMANCE_GAP , otherwise, perfo_flag will be PP_NO_PERFORMANCE.

In the other way, when initialising with PP_ESTIMATOR_PERFO the performace flag is calculated. If the requested time is within a flagged window *perfo_flag* returns PP_ATTITUDE_FLAGGED. In case the requested time is within a gap (estimator data separation is greater than 1.9 seconds), *perfo_flag* returns PP_ESTIMATOR_GAP. When the two conditions are met, *perfo_flag* is equal to PP_ATTITUDE_FLAGGED_GAP. Otherwise, *perfo_flag* is PP_ATTITUDE_NOMINAL.

The following table shows the expected output flag “perfo_flag” and how the mispointing angles are computed as a function of the modes selected during initialisation (“mode_out” and “mode_perfo”):

mode_out	mode_perfo	Output perfo_flag	Output mispointing angles
PP_ATT_HARMONIC	PP_NO_PERFO	PP_NO_PERFORMANCE	Computed from harmonic angles
		PP_NO_PERFORMANCE_GAP if requested time is within a time gap.	Computed from harmonic angles
	PP_ESTIMATOR_PERFO	PP_ATTITUDE_NOMINAL	Computed from harmonic angles
		PP_ESTIMATOR_GAP if requested time is within a time gap.	Computed from harmonic angles
		PP_ATTITUDE_FLAGGED if requested time is within a flagged window.	Computed from harmonic angles
		PP_ATTITUDE_FLAGGED_GAP if requested time is within a time gap and within a flagged window.	Computed from harmonic angles
PP_ATT_ESTIMATOR	PP_NO_PERFO	PP_NO_PERFORMANCE	Computed from estimated angles
		PP_NO_PERFORMANCE_GAP if requested time is within a time gap.	Computed from estimated angles
	PP_ESTIMATOR_PERFO	PP_ATTITUDE_NOMINAL	Computed from estimated angles
		PP_ESTIMATOR_GAP if requested time is within a time gap.	Computed from estimated angles
		PP_ATTITUDE_FLAGGED if requested time is within a flagged window.	Computed from estimated angles
		PP_ATTITUDE_FLAGGED_GAP if requested time is within a time gap and within a flagged window.	Computed from estimated angles

Table 51: ***Input-Ouput dependency for pp_get_attitude_aocs***

7.6.5 Warnings and errors

Next table lists the possible error messages that can be returned by the **pp_get_attitude_aocs** CFI function after translating the returned status vector into the equivalent list of error messages by calling the function of the PPF_POINTING software library **pp_vector_msg** (see RD 3).

This table also indicates the type of message returned, i.e. either a warning (WARN) or an error (ERR), the cause of such a message and the impact on the performed calculation, mainly on the results vector.

The table is completed by the error code and value. These error codes can be obtained translating the status vector returned by the **pp_get_attitude_aocs** CFI function by calling the function of the PPF_POINTING software library **pp_vector_code** (see RD 3).

Error type	Error message	Cause and impact	Error Code	Error No
ERR	No orbit update for the input time	No calculation performed	PP_CFI_GET_ATTITUDE_AOCS_NO_ORBIT_UP_ERR	0
WARN	Time gap detected in the attitude file. Mispointing angles will not be calculated	Calculation performed	PP_CFI_GET_ATTITUDE_AOCS_GAP_WARN	1

Table 52: pp_get_attitude_aocs warnings and errors.

7.6.6 Runtime performances

The following runtime performances have been measured.

Ultra Sparc [ms] FIXED inputs
TBD

Table 53: pp_get_attitude_aocs runtime performances.

7.7 pp_free_attitude

7.7.1 Overview

The **pp_free_attitude** CFI function cleans any memory reserved by the initialisation function.

7.7.2 Calling interface

The calling interface of the **pp_free_attitude** CFI function is the following (input parameters are underlined):

```
#include <ppf_pointing.h>
{
   pp_free_attitude();
}
```

For Fortran programs the declaration and calling procedure is as follows (input parameters are underlined, note that the C preprocessor must be used because of the presence of the `#include` statement):

```
TBD
```

7.7.3 Input parameters

The **pp_free_attitude** CFI function has no input parameters.

7.7.4 Output parameters

The **pp_free_attitude** CFI function has no output parameters.

7.7.5 Warnings and errors

No errors are returned by **pp_free_attitude**.

7.7.6 Runtime performances

The following runtime performances have been measured.

Ultra Sparc [ms] FIXED inputs
TBD

Table 54: pp_free_attitude runtime performances.

8 LIBRARY PRECAUTIONS

The following precautions shall be taken into account when using PPF_POINTING software library:

- When a message like

  PPF_POINTING >>> ERROR in *pp_function*: Internal computation error # *n*

  or

  PPF_POINTING >>> WARNING in *pp_function*: Internal computation warning # *n*

  appears, run the program in *verbose* mode for a complete description of warnings and errors, and call for maintenance if necessary.

9 KNOWN PROBLEMS

The following precautions shall be taken into account when using the CFI software libraries:

CFI library	Problem	Work around solution
pp_target	The time derivatives of some of the output parameters, e.g. the altitude rate of the target, of pp_target when the ray tracing model is either iray = PP_STD_REF or PP_USER_REF are sometimes noisy.	Use iray PP_PRED_REF or equivalent predefined refraction mode.
pp_target	The accuracy in the calculation of along and across track distances from the SSP to the nadir of the target (result elements 14 and 74) depends greatly on the distance from the SSP to the nadir of the target. The error becomes greater than 1 km at a distance around 800 km, while at distances around 400 km the error is only of few meters.	Use propagation (e.g. po_ppforb belonging to PPF_ORBIT library) to calculate the ground track more precisely and then use pp_stavis to calculate the minimum distance to the ground track (iterative algorithm).

Table 55: Known problems.

10 APPENDIX A

10.1 pp_converter input file format

The format of each **pp_converter** input file follows the directives described in RD 6.

10.1.1 FHR

Follows the format described in RD 6. The fields DESTINATION, PHASE_START, CYCLE_START, REL_START_ORBIT and ABS_START_ORBIT have no meaning and could be fixed to any value.

10.1.2 VHR

Table 56: details the pp_converter input file VHR format.

N	Description	units	Byte Length	Data Type	C Format
1	**RECORD**	keyword	6	string	%6s
	blank space		1	string	%1s
	cif_vhr	keyword	7	string	%7s
	blank space		1	string	%1s
	; Variable Header	comment		string	%s
	newline character	terminator	1	string	\n
2	newline character	empty line	1	string	\n
3	**LUT_DESCRIPTOR=**	keyword	15	string	%15s
	quotation mark	-	1	string	\”
	Atmosphere Descriptor		10	string	%10s
	quotation mark	-	1	string	\”
	newline character	terminator	1	string	\n
4	**START_DAY_OF_YEAR=**	keyword	18	string	%18s
	First day in the year for which the atmosphere is considered to be valid		4	+xxx	%+04d
	newline character	terminator	1	string	\n
5	**STOP_DAY_OF_YEAR=**	keyword	17	string	%17s
	Last day in the year for which the atmosphere is considered to be valid		4	+xxx	%+04d
	newline character	terminator	1	string	\n

Table 56: pp_converter input file VHR format

N	Description	units	Byte Length	Data Type	C Format
6	**MIN_LAT=**	keyword	8	string	%8s
	Minimum latitude for the latitude band		8	+xxx.xxx	%+08.3f
	<deg>	unit	5	string	%5s
	newline character	terminator	1	string	\n
7	**MAX_LAT=**	keyword	8	string	%8s
	Maximum latitude for the latitude band		8	+xxx.xxx	%+08.3f
	<deg>	unit	5	string	%5s
	newline character	terminator	1	string	\n
8	**NUM_LUT=**	keyword	8	string	%8s
	Number of pairs of coindex of refraction vs. altitude		4	+xxx	%+04d
	newline character	terminator	1	string	\n
9	newline character	empty line	1	string	\n
10	**ENDRECORD**	keyword	9	string	%9s
	blank space		1	string	%1s
	cif_vhr	keyword	7	string	%7s
	newline character	terminator	1	string	\n

Table 56: pp_converter input file VHR format

10.1.3 Data block

The following table describes the data block format.

N	Description	units	Byte Length	Data Type	C Format
1	**LIST**	keyword	4	string	%4s
	blank space		1	string	%1s
	num_alt=	keyword	8	string	%8s
	Number of pairs of coindex of refraction vs. altitude		3	xxx	%03d
	blank space		1	string	%1s
	newline character	terminator	1	string	\n
2	newline character	empty line	1	string	\n

Table 57: pp_converter input file data block format

N	Description	units	Byte Length	Data Type	C Format
3	list of **num_alt** alt Records (see format below)				
4	newline character	empty line	1	string	\n
5	**ENDLIST**	keyword	7	string	%7s
	blank space		1	string	%1s
	num_alt	keyword	7	string	%7s
	newline character	terminator	1	string	\n

Table 57: pp_converter input file data block format

N	Description	units	Byte Length	Data Type	C Format
1	2 blank spaces		2	string	%2s
2	**RECORD**	keyword	6	string	%6s
	blank space		1	string	%1s
	alt:	keyword	4	string	%4s
	blank space	indentation	1	string	%1s
3	**ALT=**	keyword	4	string	%4s
	Altitude	-	8	+xxx.xxx	%+08.3f
	<km>	unit	4	string	%4s
	blank space	indentation	1	string	%1s
4	**REFR=**	keyword	5	string	%5s
	Coindex of refraction	-	8	+xxx.xxx	%+08.3f
	blank space	indentation	1	string	%1s
5	**ENDRECORD**	keyword	9	string	%9s
	blank space		1	string	%1s
	newline character	terminator	1	string	\n

Table 58: pp_converter alt record format

10.1.4 Example

An example **pp_converter** input file is shown below.

```
FILE; Converter Input File
;----------------------------------------------------
RECORD fhr ; Fixed Header
```

```
FILENAME="REFRACT_1976.DAT_conv"
DESTINATION="PDS,FOS  "
PHASE_START=+000
CYCLE_START=+000
REL_START_ORBIT=+00000
ABS_START_ORBIT=+00000
ENDRECORD fhr
;---------------------------------------------------
RECORD cif_vhr ; Variable Header
LUT_DESCRIPTOR="STD_1976  "
START_DAY_OF_YEAR=+001
STOP_DAY_OF_YEAR=+365
MIN_LAT=-090.000<deg>
MAX_LAT=+090.000<deg>
NUM_ALT=+046
ENDRECORD cif_vhr
;---------------------------------------------------
LIST num_alt=046
  RECORD alt: ALT=+000.000<km> REFR=+271.950 ENDRECORD
  RECORD alt: ALT=+001.000<km> REFR=+246.930 ENDRECORD
  RECORD alt: ALT=+002.000<km> REFR=+223.620 ENDRECORD
  RECORD alt: ALT=+003.000<km> REFR=+202.060 ENDRECORD
                        ...
  RECORD alt: ALT=+090.000<km> REFR=+010.000 ENDRECORD
  RECORD alt: ALT=+095.000<km> REFR=+000.000 ENDRECORD
  RECORD alt: ALT=+100.000<km> REFR=+000.000 ENDRECORD
ENDLIST num_alt
;---------------------------------------------------
ENDFILE
```

11 APPENDIX B

11.1 Predefined refraction accuracy.

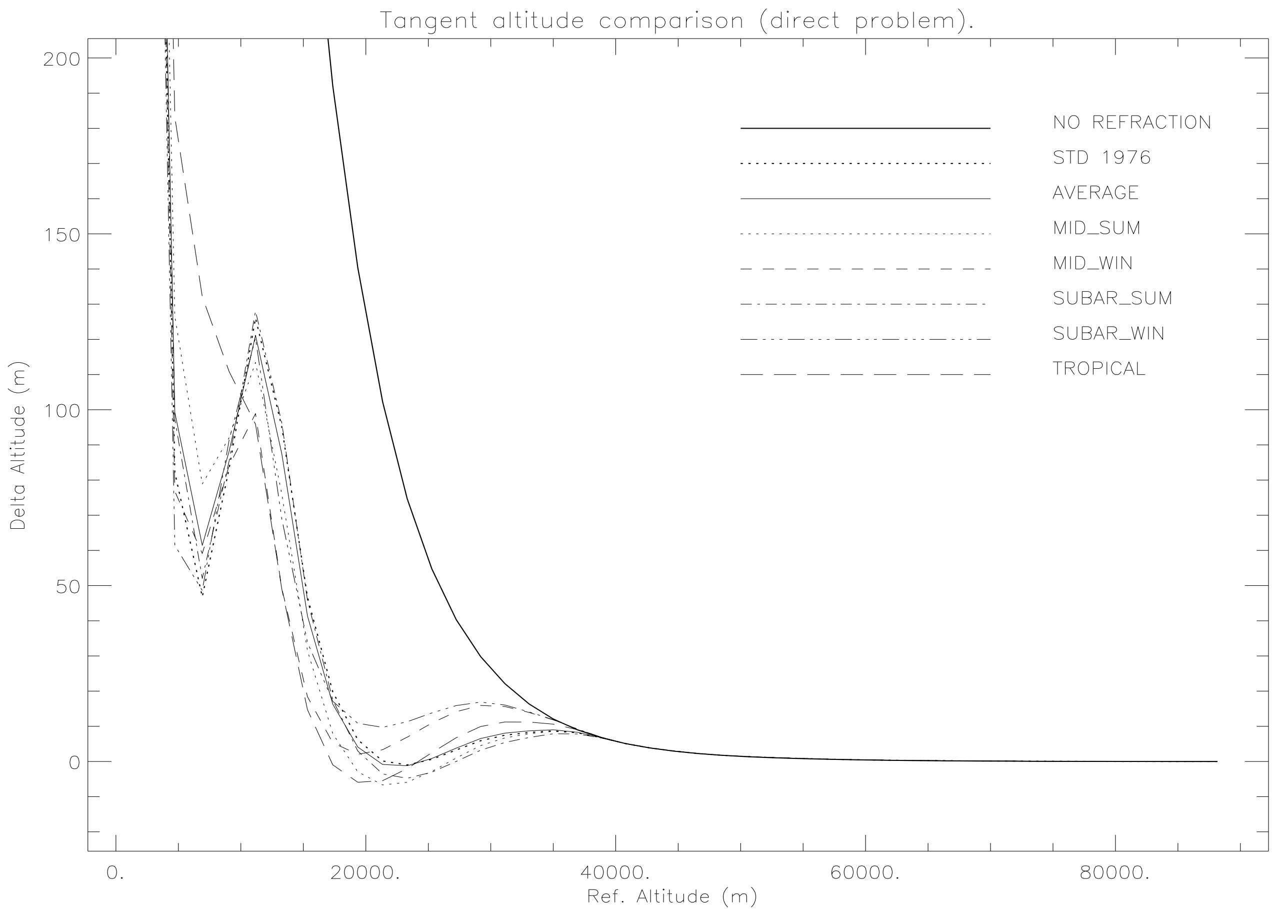
The following plots show the accuracy of the predefined refraction models.

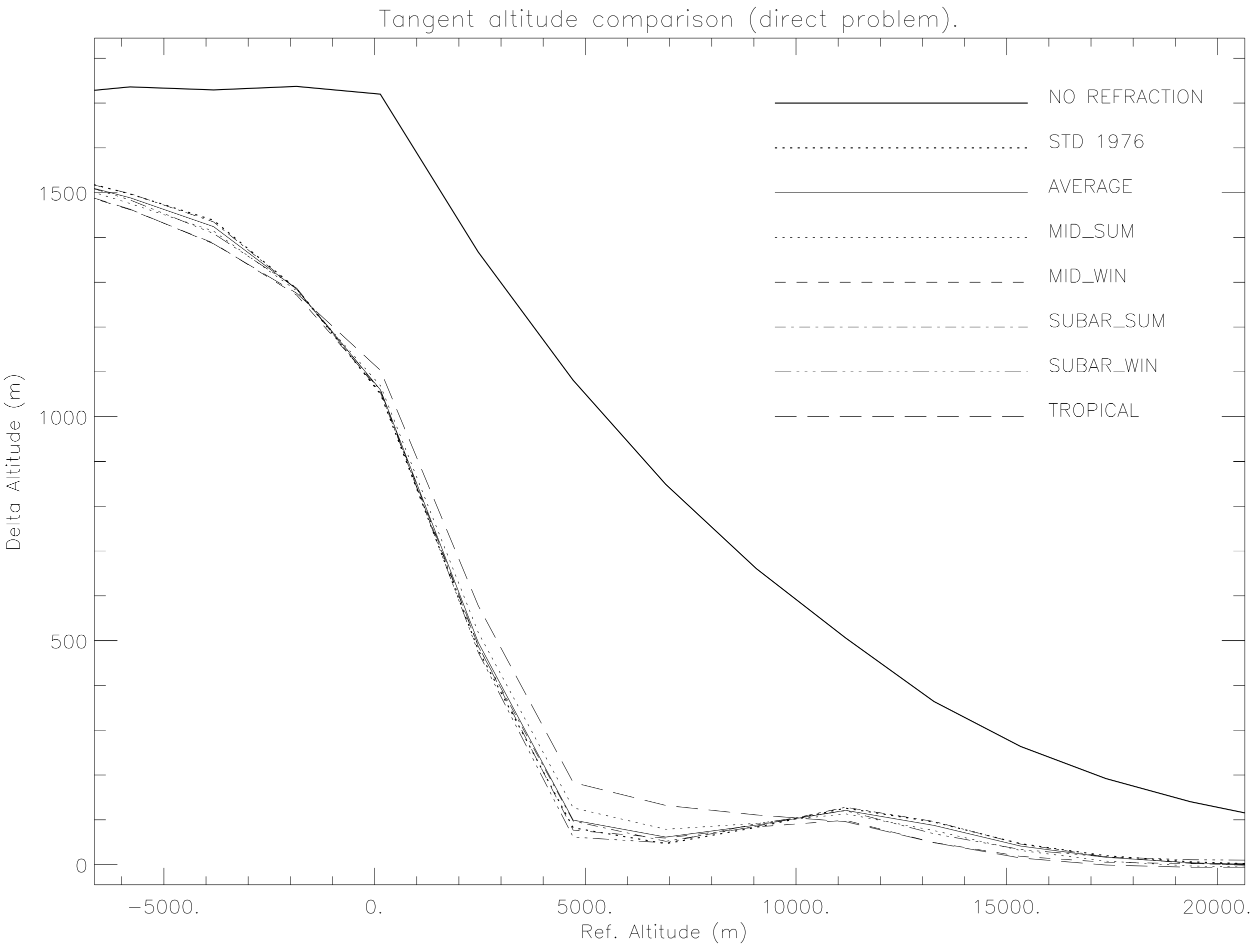
- the direct problem has been calculated taking *idir* = 7 and a frequency of 5500 nm (MIPAS).
- the inverse problem is calculated using *idir* = 9 and 575 nm (GOMOS)

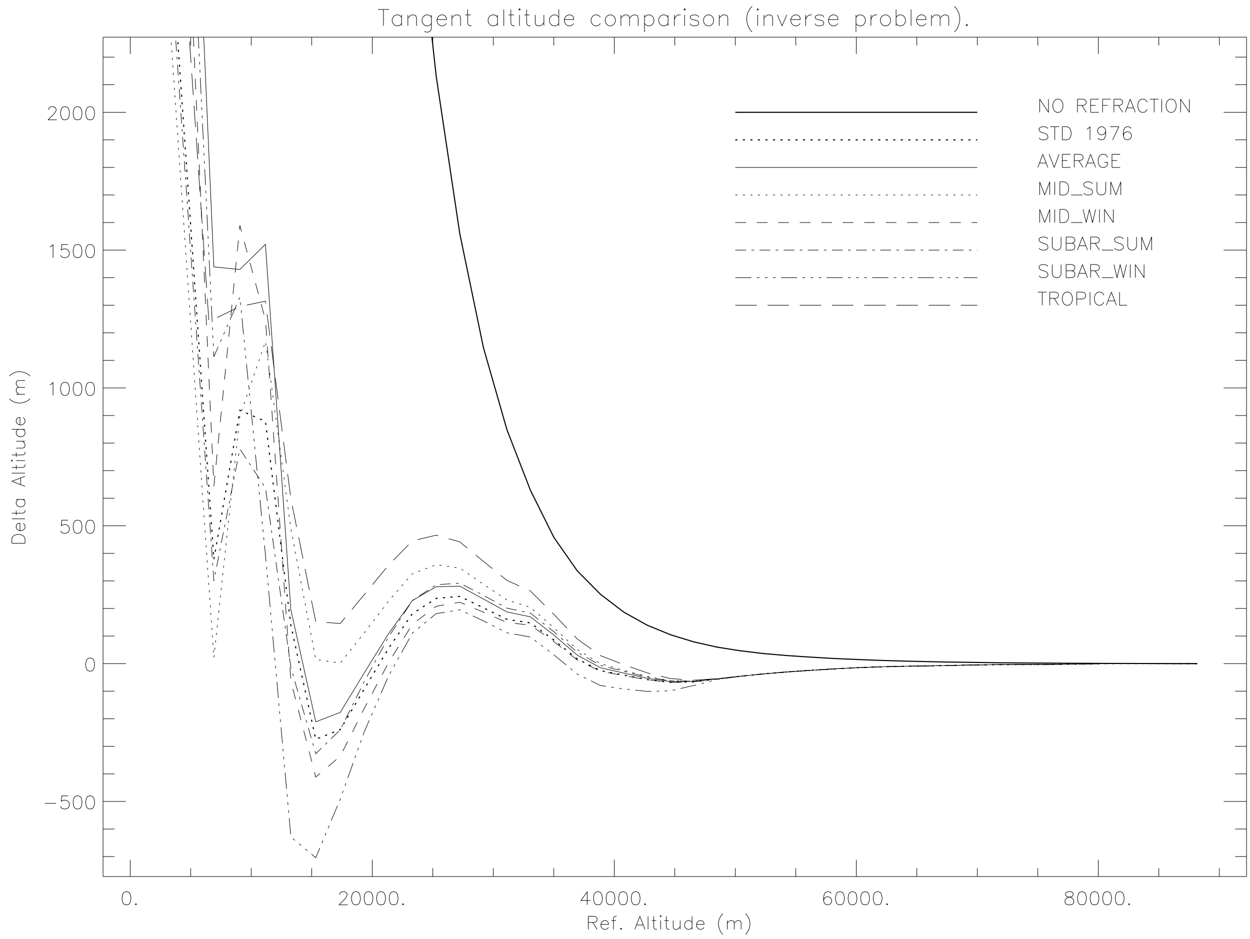
For each atmosphere, the tangent altitude of the predefined refraction mode (*iray* = 3xx) is compared with the tangent altitude of the corresponding refraction model (*iray* = 2) to determine the difference between both models ("delta altitude"), and the results are plotted againts the refracted tangent altitude, whereas the no refraction mode (*iray* = 0) is compared with the standard atmosphere (*iray* = 1).

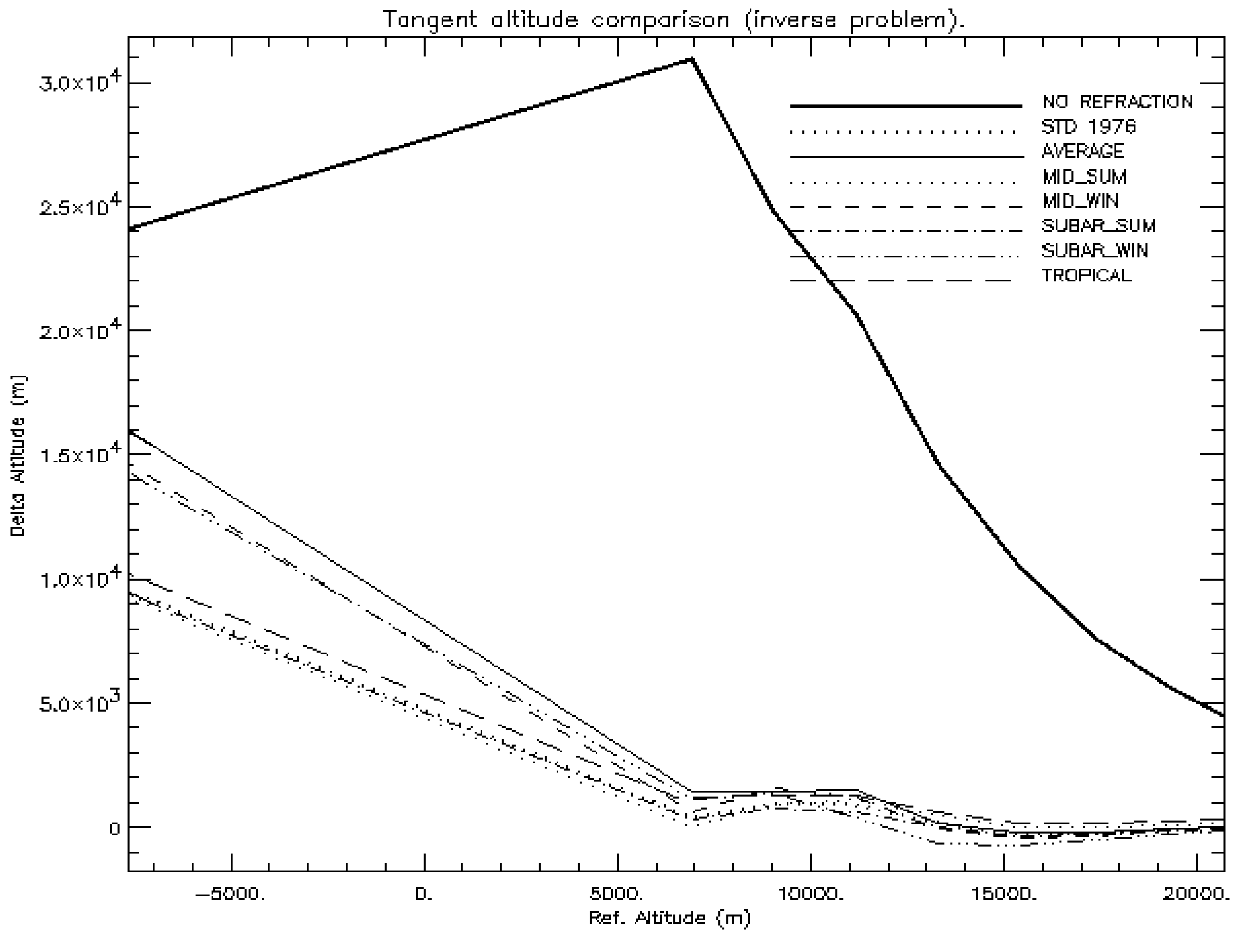
In other words, these pictures show the accuracy of the tangent altitude calculation if no refraction model is used (NO REFRACTION curve), and the same accuracy if a predefined refraction model corresponding to the appropriate atmosphere is used (e.g. STD 1976 curve).

Note that the accuracy of the predefined refraction model is about 10 time better than the no refraction one (e.g. the error goes from 10 Km to 1 Km in the direct problem when the tangent altitude is 5 Km, and from 30 Km to 3 Km in the inverse problem), and that the accuracy starts to degrade below a tangent altitude of about 5 Km.




Tangent altitude comparison (direct problem).
NO REFRACTION
STD 1976
AVERAGE
MID_SUM
MID_WIN
SUBAR_SUM
SUBAR_WIN
TROPICAL
Delta Altitude (m)
1500
1000
500
0
-5000.
0.
5000.
10000.
15000.
20000.
Ref. Altitude (m)


Tangent altitude comparison (inverse problem).
NO REFRACTION
STD 1976
AVERAGE
MID_SUM
MID_WIN
SUBAR_SUM
SUBAR_WIN
TROPICAL
2000
1500
1000
500
0
−500
Delta Altitude (m)
0.
20000.
40000.
60000.
80000.
Ref. Altitude (m)


Tangent altitude comparison (inverse problem).
NO REFRACTION
STD 1976
AVERAGE
MID_SUM
MID_WIN
SUBAR_SUM
SUBAR_WIN
TROPICAL
Delta Altitude (m)
Ref. Altitude (m)
3.0×10^4
2.5×10^4
2.0×10^4
1.5×10^4
1.0×10^4
5.0×10^3
0
-5000.
0.
5000.
10000.
15000.
20000.