Title : **MERIS Level 1 Detailed Processing Model, Parameters Data List**

Doc. no : PO-TN-MEL-GS-0002

Issue : 8

Revision : 0

Date : 10 May 2011

	Function	Name	Company	Signature	Date
Prepared:		MERIS Team	ACRI		10/05/2011
Approved:	Project Manager	L. BOURG	ACRI		10/05/2011
Released:	Project Manager		ESA		10/05/2011

External Distribution

Name	Quantity
P. GORYL (ESA /ESRIN)	1
J-P. HUOT (ESA /ESTEC)	1

Internal Distribution

Name	Quantity
All ESL laboratories	1
L. BOURG	1
MERIS DPQC Team	1

Change Record

Issue	Revision	Date	Description	Approval
Preliminary	18/9/95		No	
1	0	17 Oct 1995	Final report	Yes
2	Draft	31 Jan 1996	Reorganisation (include relevant ATBD sections), algorithm changes	No
2	1	25 Mar 1996	Review by ESA (NWP/SD/3017)	Yes
2	2	21 Jun 1996		Yes
3	Draft	08 Nov 1996	Review & new inputs from ESA	
3	0	2 Dec 1996	Prototyping phase final report	Yes
3	1	6 Dec 1996	Prototyping phase final report	Yes
3	2	19 Dec 1996	Revised final report (change pages : pp 3-6, 6-7 to 6-14, 8-1, 8-5, 9-5, 9-7, 9-8, 9-11, 9-12, 9-15, A-2 to A-11)	Yes
3	3	6 June 1997	Revised final report Section 10 : ECMWF files change, Section 11 : Applicable documents update.	Yes
3	4	15 Oct. 1997	Revised final report Section 7, step1.4.2 : updated description Section 8 : product limits algorithm,revised orbit propagator selection.	Yes
3	5	15 Dec. 1997	Typos: 4-7, 4-8, 4-10, 4-11, 4-14; 7-3, 7-13, 8-18 to 8-25 : evolution of product limits algorithm.	Yes
4	0	23 Dec. 1998	Revised final report	Yes
4	1	17 Dec. 1999	Section 7: SPxAC stray light correction uses per module SRDFs, AL stray light correction deleted. Section 8: revised Product Limits Algorithm, new exception processing in attitude perturbation computation. Section 10: input pressure data changed from "surface" to "mean sea level", relative humidity field selected at 1000 hPa level instead of 850.	
4	2	17 Dec. 1999	Revised after ESA comments. Change bars are kept relative to 4.0. Changed pages (relative to v4.1): 2-1, 2-6, 3-2, 3-3, 4-2, 4-4, 4-12, 6-1, 6-2, 6-5, 8-8, 10-5, 10-11.	Yes
4	3	25 Feb. 2000	Revised Smear Dynamic Correction (§ 6). Change bars are kept relative to 4.0. Changed pages (relative to v4.2): 3.6, 6-5, 6-13	Yes
4	4	7 Sep. 2001	typos (§ 4, 6 & 8, annex A)	Yes
5	0	14 Sep. 2001	handling of Level 0 products not starting	Yes

			at the beginning of a frame (§ 4, pp. 4-2 & 4-11), improved handling of saturated samples within Stray Light Correction (§ 7, pp. 7-6 to 7-11, Annex A pp. A-5 & A-6), conversion of ECMWF total ozone field, now in kg/m2, into DU (§ 10, pp 10-5 & 10-8)	
5	1	26 Jul. 2002	handling of OBT disruption due to PAUSE mode (§4, p 4-13), modification of the Suspect flag setting (§11, p 11-14)	Yes
6	0	8 Nov. 2002	Spectral Shift Index of Level 1b product Flags MDS (MDS 16) replaced by Detector Index (§8, pp 8-20, 8-28, 8-29, 8-30; §11 pp 11-6, 11-15)	Yes
6	1	28 Mar 2003	simplification of packet format tests (step 1.1.1.1-14, §4 p 4-12)	Yes
			Addition of an Instrument Response Degradation Model to apply on radiometric gains (§6, new step 1.3.0.2, pp 6-9 & 6-12)	Yes
6	1a	16 May 2003	explicit radians to degree conversion introduced in equations 1.5.4.3-2 & -3 (§8 p 8-31))	Yes
7	0	30 June 2005	handling of unappropriate OSV data in geolocation processing (§8, steps 1.5.1.2 pp 8-22 to 8-24, 1.5.1.8 pp 8-26, 1.5.2.3 p 8-27, 1.5.2.4 p 8-27)	Yes
			Correction of equation 1.5.4.3-3 (§8, p 8-29)	Yes
7	1	30 Oct. 2006	Correction of equations 1.5.1.2-4 & 1.5.1.8-3 (§8, pp 8-22 & 8-26, linked to CR 137) ; Addition of exception processing blocks after steps 1.5.1.3-3, -6 & -8 ($8, pp 8-24 & 8-25) ; modification of step 1.8.7 (p 11-14). ***Change bars are kept relative to 6.1a, all sections but 8 and 11 kept as 7.0.***	Yes
8	0	10 May 2011	Clarification of steps 1.4.1.1-4 & -5 in section 7.3.3, p. 7-10 Addition of step 1.5.0 "Pointing Vectors pre-processing" in section 8.4, p 8-22. Addition of variables $U_x^{FR/RR}(k,m)$, $U_y^{FR/RR}(k,m)$ $\Delta\phi^{FR/RR}$ (implied in the above change) to the PDL section (Annex 1, p. A-7), removal of obsolete variable $\psi^{FR/RR}(k,m)$, $\delta\phi^{FR/RR}(k,m)$.	

Table of Contents

1. - Introduction

1.1 - General

This document is the Detailed Processing Model and Parameters Data List document for the MERIS data processing. It covers the MERIS Level 1 processing, as defined in "MERIS System Architecture Theoretical Basis Document", PO-TN-MEL-GS-0001 (RD9).

1.2 - Purpose and Scope

This document provides a detailed specification of the MERIS Level 1B processing algorithms in terms of algorithms and data. The interfaces to MERIS Level 1B Processing are specified in AD1; the initial input and final output parameters and their correspondence to AD1 are summarised in the section "Parameters Data List". This document is intended to serve as a functional requirements specification for the MERIS data processing entities within the ENVISAT-1 ground segment.

This document describes in detail and fully specifies the data processing to be applied to the MERIS Full Resolution or Reduced Resolution Level 0 Products, in order to derive the **MERIS Level 1b Products** as specified in AD1. An overview of the MERIS processing architecture is described in the MERIS System Architecture Theoretical Basis Document, PO.TN.MEL.GS.0001 (RD9).

1.3 - Guide to This Specification

This specification includes,

- in chapter 3, the overview of the MERIS Level 1B processing; this overview provides a top level break-down into processing steps;
- in chapters 4 to 11, the detailed description of each processing step;
- in Appendix A the correspondence between processing input parameters and input data products as specified in AD1.

Chapter 3 includes

- descriptive sections :
  - ⇒ introduction (3.1)
  - ⇒ overview (3.2)
  - ⇒ algorithm description (3.3.1)
- a top level functional breakdown diagram, which shall be considered a requirement;
- a top level control flow diagram, which shall be considered a requirement;
- requirements sections :
  - ⇒ list of breakpoints (3.3.2)

Each chapter 4 to 11 includes

- descriptive sections :
  - ⇒ introduction (x.1)
  - ⇒ overview (x.2)
  - ⇒ algorithm description (x.3.1)
- a set of functional breakdown diagrams, each of which shall be considered a requirement;

- requirements sections :
  ⇒ list of variables (x.3.2)
  ⇒ equations (x.3.3)
  ⇒ accuracy (x.3.4)
  ⇒ summary list of Product Confidence Data (x.3.5)
  ⇒ exception handling (x.3.6) when applicable

Descriptive sections shall not contain any requirement.
In the requirements sections each individual requirement is numbered.
Numbering shall be unique throughout the MERIS processing.

2. - References, abbreviations and definitions

2.1 - Applicable Documents

AD1. MERIS I/O Data Definition, PO-TN-MEL-GS-0003
AD2. *deleted*
AD3. PPF_Orbit Software User Manual, PO-IS-GMV-GS-0058 Issue 4.5.
AD4. ENVISAT-1 Ground Segment Time Handling and Processing, PPF-TN-ESA-GS-0248
AD5. PPF_Pointing Software User Manual, PO-IS-GMV-GS-0059 Issue 4.5.
AD6. Tailoring of the PSS-05-0 ESA Software engineering standards for the ENVISAT G/S Software development, PO-TN-ESA-GS-0530.
AD7. ENVISAT-1 Product Specifications, PO-RS-MDA-GS-2009
AD8. Measurement Data Definition and Format Description for MERIS, PO-ID-DOR-SY-0032, Vol. 4, 7
AD9. ENVISAT Meteo Products, PO-TN-ESA-GS-00462 Issue 1
AD10. ECMWF - PDS Interface, PO-RP-ES-GS-00622 Issue 2
AD11. PPF Software User Manual, PO-IS-GMV-GS-0057 Issue 4.5.

2.2 - Reference Documents

RD1. ENVISAT-1 Product Definition, PO-TN-ESA-GS-0231
RD2. MERIS Specification, PO-RS-ESA-PM-0023, Iss.2 rev. 1
RD3. MERIS Assumptions on the Ground Segment, PO-RS-DOR-SY-0029, Iss. 1, Vol. 6
RD4. Mission Conventions Document, PO-IS-ESA-GS-0561, Issue 2.0.
RD5. *deleted*
RD6. *deleted*
RD7. *deleted*
RD8. MERIS Level 2 Algorithms Theoretical Basis Document, PO-TN-MEL-GS-0005, Iss. 2
RD9. System Architecture Theoretical Basis Document, PO-TN-MEL-GS-0001,Iss. 3.2
RD10. MERIS Radiometric Image Quality error items estimates, PO-TN-AER-ME-0008
RD11. ENVISAT-1 Reference Definitions Document For Mission Related Software, PO-TN-ESA-GS-0361, Iss. 1.0
RD12. MERIS Resampling Matrix, PO-TN-MEL-GS-0007, Issue 1
RD13. MERIS Viewing Model, PO-TN-ACR-SIM-0001, Draft
RD14. MERIS Image quality budgets, PO-TN-AER-ME-0001, Iss. 3
RD15. ECMWF Meteorological Bulletin M1.9/3 - Encoding and decoding GRIB and BUFR data (GRIBEX)

2.3 - Abbreviations

A/D	Analogic to Digital
AC	across-track
AD	Applicable Document
ADS	Annotation Data Set
ADSR	Annotation Data Set Record
ADC	Analogic to Digital Converter
AL	along-track
AOCS	Attitude and Orbit Control System
APID	Application Process IDentifier
CCD	Charge Coupled Device
CD-ROM	Compact Disc, Read Only Memory (trade mark)
CFI	Customer Furnished Item
DEM	Digital Elevation Model
ECMWF	European Centre for Medium-term Weather Forecast
FOV	Field Of View
FR	Full Resolution
GADS	Global Annotation Data Set
ICU	Intelligent Control Unit
IR	Infra Red
JD	Julian Day
LSB	Least Significant Bit
MDS	Measurement Data Set
MDSR	Measurement Data Set Record
MERIS	Medium Resolution Imaging Spectrometer
MJD2000	Modified Julian Day 2000
MPH	Main Product Header
MTF	Modulation Transfer Function
NIR	Near Infra Red
PCD	Product Confidence Data
PD-HF	Payload Data Handling Facility
PDS	Payload Data Segment
PSF	Point Spread Function
RD	Reference Document
RR	Reduced Resolution
SATBD	System Architecture Theoretical Basis Document
SP	spectral (dimension of the sensor)
SPH	Specific Product Header
sqq.	and the following ones
SSP	Sub-Satellite Point
TBC	To Be Confirmed
TBD	To Be Defined
TOA	Top Of Atmosphere
UTC	Universal Time Coordinate
VEU	Video Electronics Unit
WGS	World Geodetic Standard

2.4 - Notations and Conventions

2.4.1 - Indexing

The subscripts of the array data structures shall be

- f frame ($f \in \{1..NF\}$);
- b band ($b \in \{1..B+1$; (B+1 for smear band)$\}$);
- m module ($m \in \{1..M\}$);
- k MERIS column ($k \in \{1..K\}$);
- k blank pixel column($k \in \{1..KB\}$);
- j Level 1b product column ($j \in \{1..NC\}$);

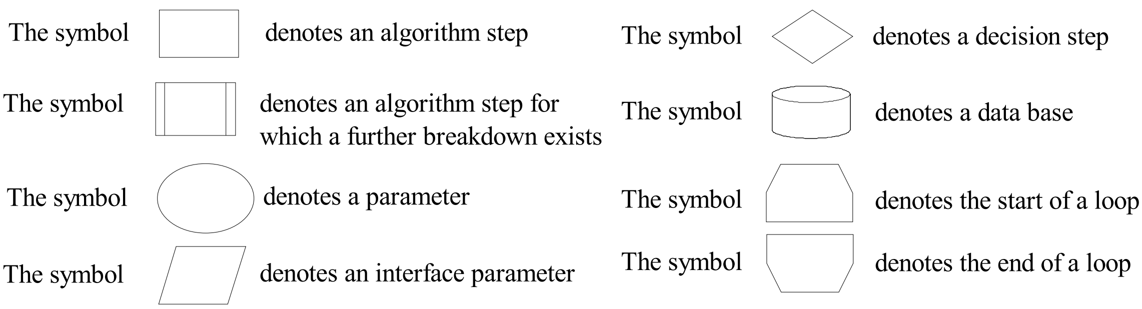
unless otherwise specified.

Note: module and pixel indexing throughout this document adopts the same variation direction: refering to Earth imaging on the descending part of the ENVISAT-1 orbit, module index and pixels index both increase from East to West.

It should be noted here that M and NF shall vary according to processing parameters : if the Reduced Resolution Level 1b Product uses all the valid data from the Level 0 Product, the Full Resolution Level 1b Product is limited to a pre-defined ground scene size (650 km along-track by 582 km across-track corresponding to 2241 by 2241 full resolution level 1b product pixels for the Full Resolution Scene and 325 km by 281 km or 1121 by 1121 pixels for the Full Resolution Imagette). To avoid useless processing, packets and MERIS modules within packets are selected within the Level 0 Product at the packet extraction stage (see chapter 4 below) using the Product Limits Parameters derived from the requested Full Resolution Product centre location and size (see chapter 8 below). Product Limits Parameters are time of first and last frames, first (wrt to instrument numbering rules) and total number of modules to process. The first selected frame will then be numbered 1 as well as the first module, M and NF designating respectively the total number of modules and frames actually processed.

Indices of arrays in equations may indifferently appear as subscripts or enclosed in square brackets : $X_{b,k,m,f}$ is equivalent to X[b,k,m,f]. Moreover, a mix of the two styles may be used to enhance a specific dependency, e.g. $PSF_{b,i}[f]$.
The character * is used as a shorthand for all the values in an index range.

2.4.2 - Block diagrams symbols



Arrows in the block diagrams indicate precedence : data input /output to a step or logical succession of steps.

2.4.3 - Variables

The column labelled "T" (for Type) in the lists of variables below describes the type of the variable :

- i input to the algorithm step
- s input to the algorithm step (from a data base described in the IODD, AD1)
- c intermediate result
- o output of the algorithm step

The following table describes the units, or symbols used to derive units, used in this document, shown in column "U" in the lists of variables :

Unit symbol	Name
ct	counter tick
dl	dimensionless
EU or $W.m^{-2}.\mu m^{-1}$	spectral irradiance
LU or $W.m^{-2}.sr^{-1}.\mu m^{-1}$	spectral radiance
jd	julian date
nc	numerical count (*)
e^-	(photo-)electrons
m	metre
s	seconds
%	percentage
K	degree Kelvin (temperature)
° or deg	degree (angle)
rad	radian
sr	steradian
hPa	hectoPascal
DU	Dobson Unit (10^{-3} atm.cm)

(*) : For the computations done at numerical count level, when the samples are read from the packets, the numerical counts are equivalent to Least Significant Bits (LSB); due to floating-point mode computations, numerical counts are understood as floating-point numbers.

2.4.4 - Algorithms

The pseudo-code used to specify the algorithms (when applicable) uses `Courrier type` and uses control structures close to those of the C language.

2.4.5 - Requirements

In section 3, each requirement is labelled "(R<sequence number>")"

In the "Equations" sections of chapters 4 to 11 below (sections x.3.3) :

- each requirement is followed by a unique number with the following syntax :
  "("<step number>"-"<sequence number>")"

- the sequencing of operations within a step follows the order of the statements in the document.

2.4.6 - Algorithm steps numbering

The numbers in all functional breakdown diagrams are those of a hierachical algorithms step numbering scheme as follows:

<process level>.<top level function number>.{<lower level function number>}
1 (Level 1B) 1 to 8
defined in chapters 4 to 11

Numbering within one numbering level does not reflect precedence of steps.

2.4.7 - MERIS Bands

The following specification assumes the following set of bands to be measured by MERIS instrument, and uses the corresponding band indexing conventions :

Band no.	Wavelength (nm)	Index notations
1	412.5	b1, b412
2	442.5	b2, b442
3	490.0	b3, b490
4	510.0	b4, b510
5	560.0	b5, b560
6	620.0	b6, b620
7	665.0	b7, b665
8	681.25	b8, b681
9	705.0	b9, b705
10	753.75	b10, b753
11	760.625	b11, b760
12	775.0	b12, b775
13	865.0	b13, b865
14	885.0	b14, b890
15	900.0	b15, b900

Table 2.4.7-1 : MERIS Bands

2.5 - Definitions

Auxiliary data	Data other than the instrument measurements which are necessary to the product generation algorithm.
Column	Product value of data acquired at a single pixel during the segment (or scene).
Detection Elements	CCD Elements (AC & SP elementary detection elements) providing the signal for one spectral sample.
Elementary Detection Element	Rectangular element of the CCD matrix.
Flag	Boolean element of information associated to a pixel
Frame	The set of product lines containing all data acquired at the same time.
Granule	The set of 16 x 16 product pixels in RR (or 64 x 64 product pixels in FR) children of the same tie point.
int	Rounding to nearest lower integer
Line	The set of MERIS pixels data making up the MDSR (without header). This corresponds to the instrument source packet measurement data for level 0, and to a resampled product line image for levels 1b & 2.
Near Real Time	Product processed within a few hours to a few days from the time of acquisition (synonym of *unconsolidated*).
nint	Rounding to nearest integer
Off-line	Product processed without any specific constraint on delivery delay, typically a few days to a few weeks (synonym of *consolidated*).
Pixel	Picture element: the set of measurements taken for a given location at a given time.
Record	The set of samples making up an image line.
Resolution	The smallest spatial, radiometric or spectral feature detectable; this is always higher than the *sampling* (spatial, spectral) or quantification (radiometric) interval.
Sample	Product value at a given pixel of the product grid (or associated instrument spectral sample).
Sampling	The spatial or spectral step at which data are measured.
Scene Product	User product consisting in a square image.
Segment	A segment corresponds to a continuous operation of MERIS over one orbit in a specific mode (e.g. 43.5 mn in the nominal RR mode).
Spectral Sample	Signal generated by one detection element.
SRDF	Spectral Region Distribution Function
Stabilisation Mode	Refer to RD3.
Tie frame	Set of tie points corresponding to a given satellite position
Tie point	The set of product pixels where location (w/ other auxiliary data) is provided.

3. - MERIS Level 1B Processing Overview

3.1. - Introduction

This chapter describes the overall logic of the data processing to be applied to the MERIS Full Resolution or Reduced Resolution Level 0 Products, in order to derive the MERIS Level 1b products.

3.2. - Algorithm Overview

The **MERIS Level 1B processing** is in charge of reading the MERIS Level 0 product; checking the packets; extracting measurement data, ancillary data from the packets; correcting, calibrating and geo-locating the Earth imaging data into spectral radiance values at the top of the atmosphere; ingesting ancillary data; creating level 1 products which include radiances, geo-location and other annotations. On-line quality checks are performed at each processing stage.

3.3. - Algorithm Description

3.3.1. - Physics of The Problem

3.3.1.1 - Source data packet extraction

MERIS Level 0 processing is assumed to sort packets in the data stream which correspond to the Observation modes of MERIS, from those corresponding to on-board characterisation modes.

At the initial stage of L1B processing, information in the packet header and data field header is used to detect such anomalies in the FR or RR stream of packets as :

- transmission error;
- format error;
- sequence error.

The on-board time code needs to be converted to Universal Time (UT) for datation of the packets acquisition.

3.3.1.2 - Saturated pixels

MERIS samples may be affected by phenomena outside the range of the useful measurements, i.e. a spectral radiance between 0 and L_{sat}. Such samples are totally invalid, the corresponding cells being affected temporarily or permanently. When possible, invalid pixels should be replaced by a good estimate.

Such phenomena are :

1. saturation by radiance level above L_{sat} (caused by e.g. Sun glint, cloud, bright land or snow /ice), which affects cells temporarily (typically several columns in several bands over several frames);
2. recovery from saturation : after saturation, components of the acquisition chain need some time (a few pixel columns) to recover; in the meantime the measurement is affected;
3. blooming : samples in bands and columns close to a saturated one may be temporarily affected by photon or photo-electrons diffusing from the saturated pixel;
4. glitches, high intensity impacts (e.g. laser) : will generate isolated high value samples;
5. dead pixel : due to manufacturing defects or to ageing in space, the response of some CCD cells to light will "die", i.e. permanently deviate too much (to the extent that gain correction is not usable) from the useful measurement range. Such dead pixels need to be known.

Samples affected by saturation /recovery /blooming (1, 2, 3) are flagged.
Samples corresponding to dead pixels (5) are replaced with a cosmetically interpolated value after radiometric calibration within the radiometric processing step.
Glitches are neither detected nor corrected due to unavailability of a simple model for detection.

3.3.1.3 - Radiometric processing

The valid MERIS samples are digital counts resulting from the acquisition by MERIS of passive optical spectral radiance remote sensing. The objective of the radiometric processing is to estimate the spectral radiance which caused these counts. An inverse model of the MERIS acquisition is used for that purpose, using parameters stored in the Characterisation data base and the MERIS samples themselves. The MERIS acquisition model is described as :

$$X_{b,k,m,f} = NonLin_{b,m}\left[g(T_f^{VEU}).\left[A_{b,k,m}.\left(L_{b,k,m,f} + G_{b,k,m}(L_{*,*,*,f})\right) + Sm_{b,k,m,f}(L_{b,k,m,*})\right] + g_c(T_f^{CCD}).C^0_{b,k,m}\right] + \varepsilon$$

where

- $X_{b,k,m,f}$ is the MERIS raw sample (not corrected on board);
- $NonLin_{b,m}$ is a non-linear function;
- T_f^{VEU} is the amplification unit temperature;
- T_f^{CCD} is the sensor temperature;
- $g(T)$ and $g_c(T)$ are temperature dependent gain terms (close to 1);
- $A_{b,k,m}$ the "absolute radiometric gain";
- $L_{b,k,m,f}$ the spectral radiance distribution in front of MERIS;
- $Sm_{b,k,m,f}$ the smear signal, due to continuous sensing of light by MERIS;
- $G_{b,k,m}$ a linear process representing the stray light contribution to the signal. For a given sample, some stray light is expected from all the other simultaneous samples in the module, spread into the sample by specular (ghost image) or scattering processes.
- $C^0_{b,k,m}$ the dark signal (corrected on board for temperature effects by the Offset Control Loop);
- ε is a random process representative of the instrument errors and parasitic processes not accounted for in the other terms of the model.

All terms not indexed by f (frame) do evolve in time due to ageing, but with a much slower rate which allows to represent them, for a given Level 1B product, as fixed quantities retrieved from data bases.

The radiance sensed by MERIS $L_{b,k,m,f}$ is, for a given set of target physical parameters and illumination and observation angles, proportional to the extra-terrestrial Sun spectral flux. Because there is no absolute spectral measurement of the Sun irradiance simultaneous to MERIS acquisition, all results are produced with reference to a Sun spectral flux model which must be included in the product header.

The term $A_{b,k,m}$ reflects all the amplification gains inside the instrument, which depend on :

- instrument programming (band settings, amplification programmable gains);
- components ageing;
- components temperature;
- power supply voltage.

In order to provide for limitation or failure of the on-board temperature regulation, there shall be a residual correction for $g(T)$, $g_c(T)$. In normal operation, T depends on the time elapsed since the Sun zenith angle has decreased below a threshold (80°) and can be predicted.

3.3.1.4 - Stray light correction

The stray light term $G_{b,k,m}(L_{*,*,*,f})$ in the MERIS acquisition model above may be strong enough to affect the Least Significant Bits of the raw data. This may happen in particular when MERIS is observing a scene with some high radiance areas (Sun glint patch, partly cloudy ...). As the linear transform $G_{b,k,m}$ is assumed to be known well enough from instrument characterisation, it is possible to compute an estimate of the stray light, and correct for it.

Stray light correction is handled separately from radiometric processing due to the specific nature of the processing in that stage : de-convolution; and to the fact that it can be switched on /off.

3.3.1.5 - Geo-location

The geo-location problem encompasses all processing which is directly related to the location on Earth of the MERIS measurement data.

The points where the MERIS radiance samples have been measured are determined by the projection on Earth of the line of sight of every pixel. That projection depends on
- the shape of the Earth;
- the altitude of the sample;
- the position of the ENVISAT satellite at the time of acquisition;
- the orientation of the MERIS modules;
- the optics of each MERIS module.

In order to simplify product handling, the MERIS radiance samples are re-located by nearest neighbour interpolation to the MERIS product grid, which has the following characteristics (FR grid) :
- central column : sub-satellite point track on Earth;
- line orientation : perpendicular to spacecraft velocity, projected on Earth;
- columns spacing : fixed for one product, 260 m (with very small variations);
- number of columns : 4481;
- line spacing : variable with time and orbit altitude, fixed by the MERIS frame time of 0.044s (mean ≈ 292 m).

The RR-grid is a 4x4 sub-sampled version of that grid.

The surface of altitude 0 on Earth is approximated by a geoid model. The model WGS-84 used by the ENVISAT-1 orbit propagator shall be used.

Knowledge of the ENVISAT platform and attitude relies on :
- prediction or estimation of the satellite position and attitude; the ESA CFI software is used :
  - po_ppforb or po_interpol for orbit propagation,
  - pp_target for attitude modelling
- accurate datation of the MERIS samples, to the MJD2000 time reference used by the orbit and attitude prediction /estimation.

The interpolation algorithm for re-sampling MERIS data to the grid may use characterisation data defining the MERIS pixels de-pointing. Neglecting the surface elevation causes an error in pixel location, proportional to altitude and to the tangent of the observer zenith angle. That error is estimated at the tie points.

Sun zenith and azimuth angle[1], observer zenith and azimuth angle, may be computed for any pixel knowing pixel location and Sun direction in a common frame but are stored only at the product tie points.

[1] with reference to the topocentric coordinates system, as defined in RD4

Sun glint, because of the high radiance values measured there, has an impact on both the direct usage of L1B data and on L2 processing. A first estimate of the affected pixels is performed. The location of the potential Sun glint can be predicted for each pixel, from the illumination and observation geometry.

Geo-location processing is broken down into 5 main algorithm steps :

- Product limits
- Tie points Earth location
- Altitude retrieval
- Re-sampling
- Sun glint

3.3.1.6 - Pixel Classification

In order to make easier the exploitation of TOA radiances by further processing (e.g. Level 2, Browse), the level 1 product contains appended information about the nature of each MERIS pixel. The classification process uses the *a priori* knowledge of a land /ocean map indexed by longitude and latitude, and the information in the TOA radiance bands to classify each valid pixel into :

- clear sky / ocean;
- clear sky / land;
- bright pixel / ocean,
- bright pixel / land;
  bright pixels include clouds, bright sand or soil, ice, snow, Sun glint...; the *a priori* known nature of the underlying surface is kept;

Clear sky is to be understood as clear enough to pursue atmosphere corrections.

3.3.1.7 - External Data Assimilation

In order to make easier the exploitation of TOA radiances by further processing (e.g. Level 2), the level 1 product contains appended information about the environmental conditions prevailing at the time and place of the MERIS acquisition. The parameters of interest are :

- atmospheric pressure at surface level for prediction of the Rayleigh reflectance, optical thickness;
- surface wind speed and direction for prediction of Sun glint and whitecaps;
- relative humidity at 850 hPa for verification of the aerosol correction;
- total ozone column contents for atmosphere absorption correction;

These parameters are acquired from external source (ECMWF data) and are interpolated, space-wise, to the tie points.

3.3.1.8 - Formatting

All the data and flags derived in the above algorithms steps are formatted into a file compliant with the Level 1B product description found in AD1.

3.3.2. - Functional Breakdown and Control Flow

NOTE : Requirements in this section are labelled *(R-xx)*.

The logic of the Level 1B Processing algorithm follows the functional breakdown diagram shown in figure 3.3.2-1 below. The same logic applies to RR and to FR processing.

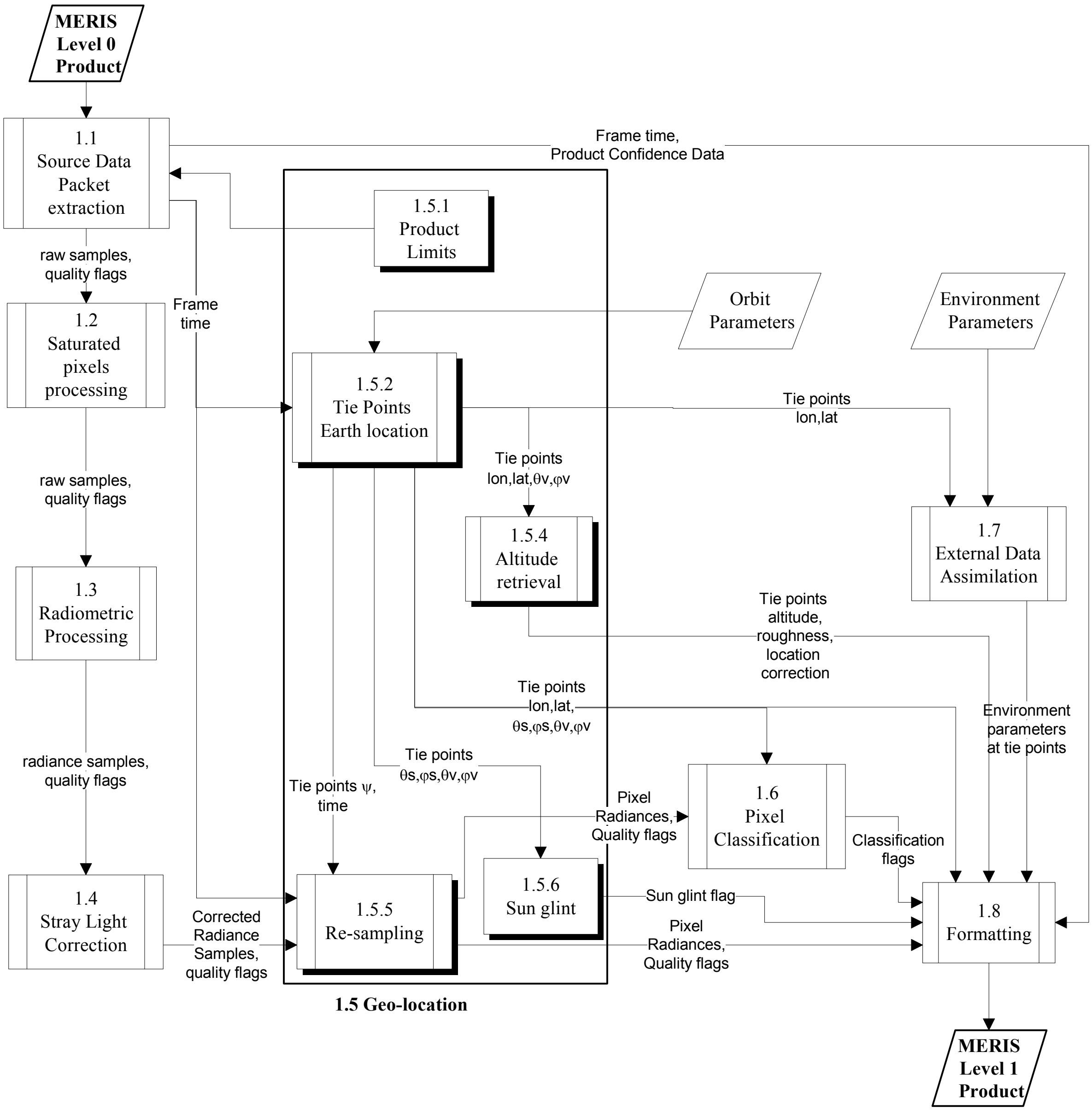


Figure 3.3.2-1 : Functional Breakdown for Level 1B processing algorithm

Note : for clarity this block diagram omits the other data products which are input to L1B processing. These products are identified in lower level breakdowns.

The Control Flow of Level 1B Processing algorithm is shown in the flow chart in figure 3.3.2-2 below. The same flow chart applies to RR and to FR processing. The arrows in the diagram show the sequence of operations, with the **exceptions** that :

- steps 1.5.4 and 1.7 may be performed in any order;
- steps 1.5.6 and 1.6 may be performed in any order;

The implementation of the FIFO buffers in this diagram is out of the scope of this document. We will summarise the requirements of the algorithm steps in terms of capacity :

1. DELETED.. *(R1)*
2. the resampling algorithm step (1.5.5) needs access to 33 MERIS frames in FR or 9 in RR, 2 tie frames simultaneously (16 FR or 4 RR frames before and 16 FR or 4 RR frames after the time of the product frame it is processing, 1 tie frame before or at the current time and one after). .... *(R2)*

3.3.3 - Breakpoints

The following data shall be used as breakpoints in the testing of the Level 1B process :

1. Radiance samples at the output of step 1.3; ....*(R3)*
2. Quality flags at the output of step 1.3 : invalid, saturated, dubious, cosmetic;............................ *(R4)*
3. Corrected Radiance samples at the output of step 1.4; .... *(R5)*
4. Quality flags at the output of step 1.4 : invalid, saturated, dubious, cosmetic, stray light risk; .... *(R6)*
5. Tie points annotations at the output of step 1.7 : longitude, latitude, Sun zenith and azimut angles, observer zenith and azimut angles, pointing angle, altitude, roughness, altitude correction for longitude and latitude, surface pressure, wind zonal and meridional components, ozone, relative humidity; .... *(R7)*

As these breakpoints correspond to the FIFO buffers illustrated in diagram 3.3.2-2 below, implementation should consider the use of intermediate files.

3.4 - Directory of Algorithm Steps

The following chapters describe in detail each of the Level 1B algorithm steps :

Chapter	Algorithm step(s)
4	1.1 Source data packet extraction
5	1.2 Saturated pixel processing
6	1.3 Radiometric processing
7	1.4 Stray light correction
8	1.5 Geo-location processing
9	1.6 Classification
10	1.7 External Data Assimilation
11	1.8 Formatting

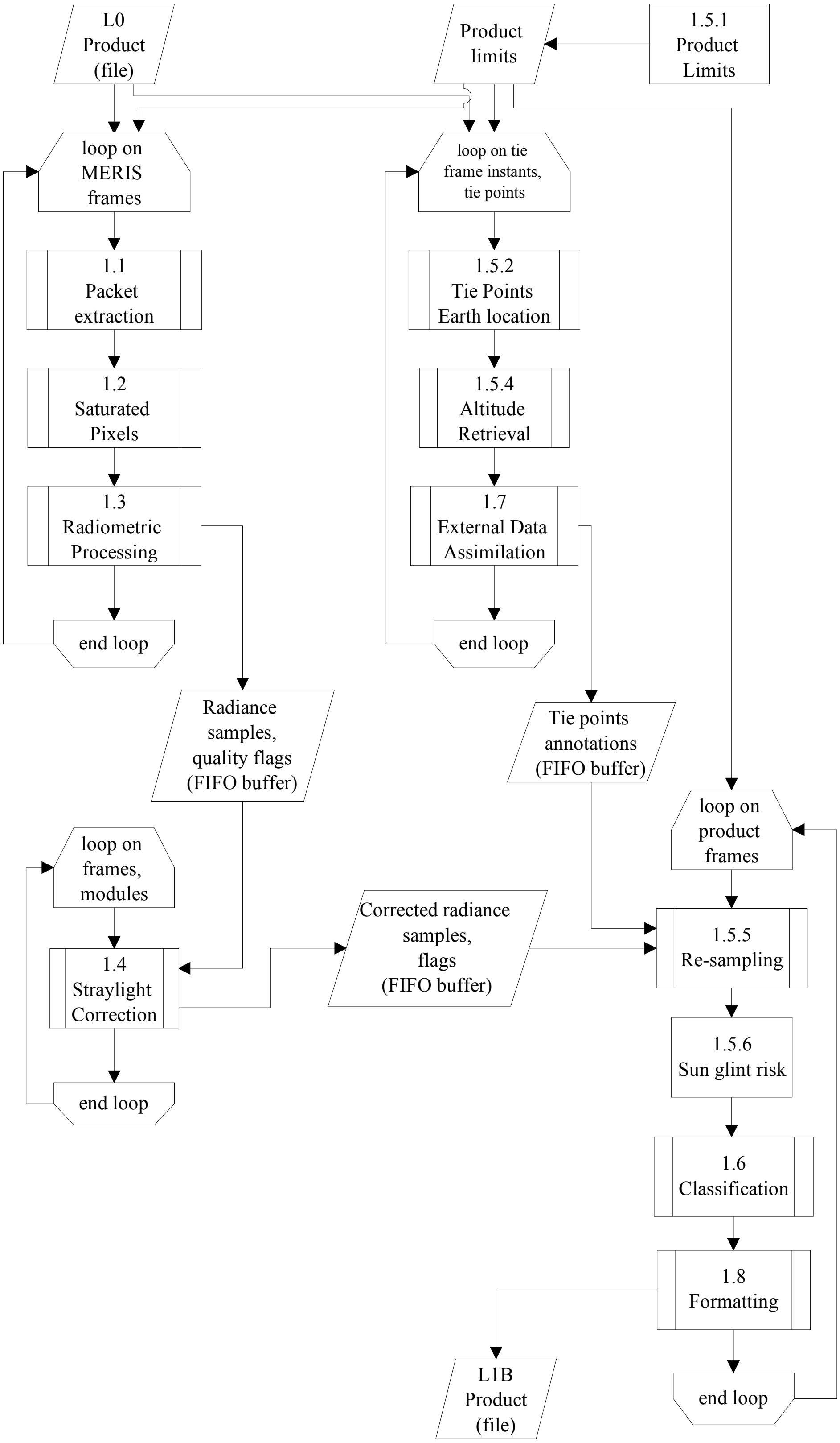


Figure 3.3.2-2 : Overall control flow chart for Level 1B processing

4. - MERIS Source Data Packet Extraction Algorithm

4.1. - Introduction

This chapter describes the data processing to be applied to the MERIS Full Resolution or Reduced Resolution Observation Mode packets, in order to derive **the input parameters of MERIS processing**. Packet extraction is part of the MERIS Level 1b processing.

4.2. - Algorithm Overview

The source data packet processing checks that packets are to be processed by the Level 1, i.e. observation mode ones, through their APID. The sequence and validity of the observation mode packets is checked. Data sets representing one frame are built from the packet contents and submitted to further processing.

Using time limits provided by the relevant geo-location function (algorithm section 1.5.1), only those packets corresponding to the desired output product are processed. In the same way, in the Full Resolution processing, only useful MERIS modules (but always contiguous and complete modules) are extracted from the packets radiances and submitted to further processing. Across-track limits are provided by the same geo-location function (algorithm section 1.5.1). In order to allow the same processing strategy for a Reduced Resolution product, these limits are also provided and set to values such that all modules are processed. Same limits are applied, here and in the following sections, to all the auxiliary data sized with any of these dimensions. For instance the gain coefficients $AL^{-1}_{b,k,m}$ (see §4.5 below) will be selected for the relevant modules only.

4.3. - Algorithm Description

4.3.1. - Theoretical Description

4.3.1.1. - "Physics" of The Problem

The MERIS measurement data are ordered and packaged with additional information about the instrument status, into a sequence of strings of bits compliant with the ESA "Standard Packet". The MERIS packets are described in detail in AD8.

Information in the packet header allows to identify :

- measurements from MERIS operational modes other than Full Resolution or Reduced Resolution Observation Mode (Reduced Field Of View Observation Mode, Calibration modes, as defined in AD8);
- events and exceptions in the operation of MERIS : disruptions in the clock or counter sequence, instrument configuration changes.

No error correction code is applied at the packet level; thus undetected invalid data may be present in the incoming packets.

The packets are input to the processing in the form of Level 0 Product. It is assumed, according to AD7, that :

- the level 0 products contain packets whose Application Process Identifier field (in the packet header) corresponds to Full Resolution or Reduced Resolution mode data, following table 4.1-1 below;
- the level 0 product may contain gaps (missing packets) of any size;
- overlaps are assumed to have been removed by pre-processing at PF-HS.

	APID values (hexadecimal)
FR Mode	0A0, 0A1, 0A4, 0A5
RR Mode	0C0, 0C1, 0C2, 0C3, 0C4, 0C5, 0C6, 0C7

Table 4.1-1 : Applicable MERIS packets APIDs

An instrument configuration change occurs whenever one of the gains is changed for any band, or the position or length of a band is changed, or the on-board processing is switched on /off, or the Offset control loop is switched on /off. It is assumed that

- no configuration change occurs within a level 0 product;
- no configuration change occurs without updating the auxiliary parameters data bases prior to data processing;

The following operation time line is assumed for MERIS (characterisation sequences excluded):

Time	**Event**
T0 :	ascending crossing node
T1 (fixed duration after T0, depending on day of year)	MERIS is turned on and goes into stabilisation mode
T3 (before T2)	MERIS exits stabilisation mode
T2 (fixed duration after T1)	MERIS downlinks the contents of its on-board memory in a calibration mode sequence
T2+16x176ms	MERIS starts operation in averaging mode or in direct and averaging mode

It is assumed that the consolidated product starts at the beginning of a frame (band counter=0), however near-real-time products may start within a frame.

4.3.1.2. - Mathematical Description of Algorithm

The packet extraction algorithm follows the flow chart shown in figure 4.3.1.2-1 below. The same flow chart applies to FR processing.

The notations used for indexing are : B: number of spectral bands (15); b: band number (in 0..B;); band B is the "smear band"; K^X: number of columns (740 for FR, 185 for RR); k: CCD column index (in 1..K^X); Mt : number of MERIS modules; M: number of modules to process (depends on processing parameters, 3 to 5); m: module index (in 1..M or in 1...Mt); f : frame index (reset to 1 for the first processed frame, total number depends either on input product or on processing parameters); L: number of micro-bands in a band; l: micro-band index (in 1..L).

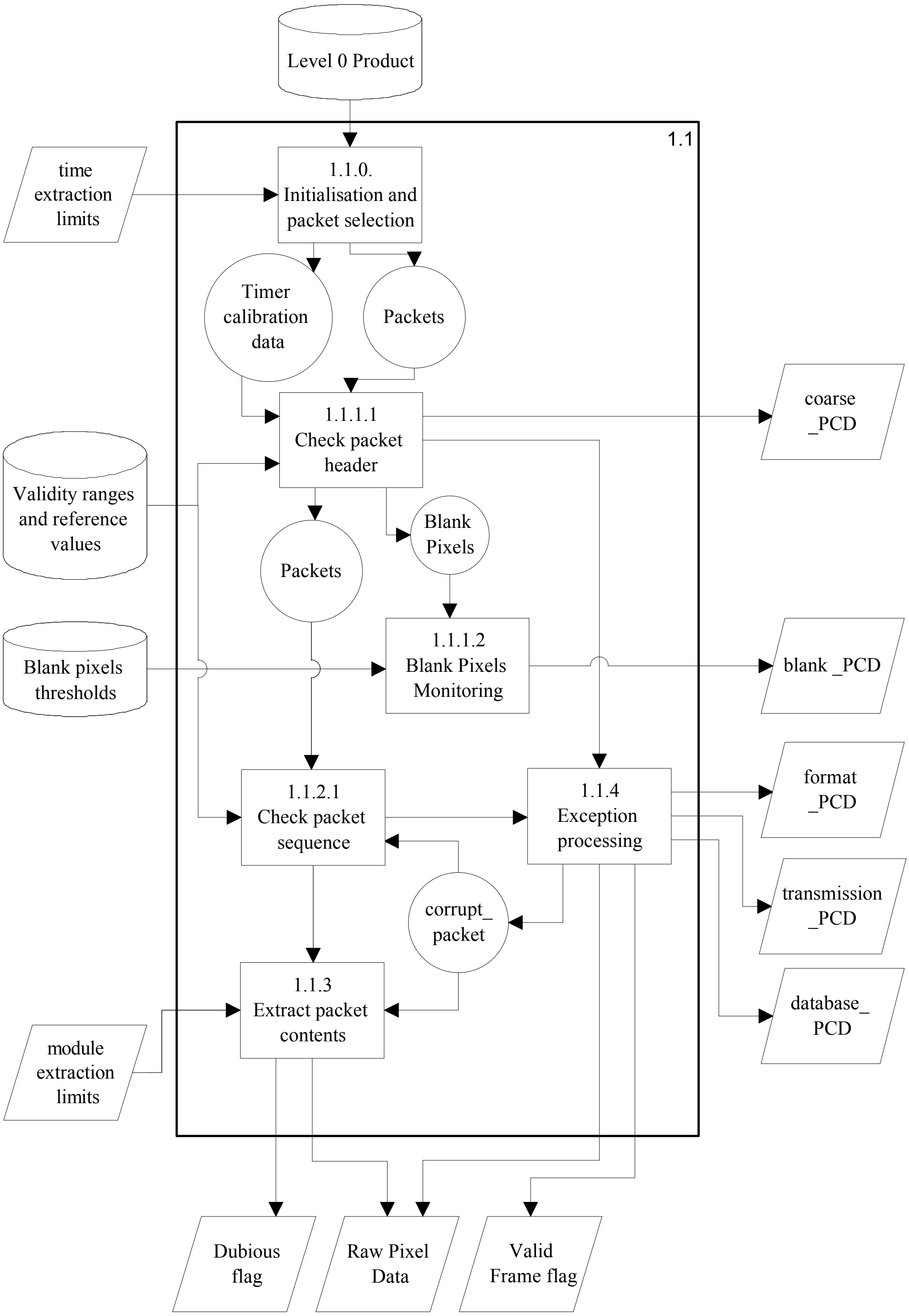


Figure 4.3.1.2-1 : Functional breakdown block diagram for the packets extraction algorithm

4.3.1.3. - Packet header checking

All the fields of a packet, which represent values which do not change with time, are checked against reference values representative of the instrument programming following table 4.3.1.3.-1 below. These reference values are assumed to be fixed at least for the duration of a product; they are stored in the "MERIS Instrument" and the "Radiometric calibration" data bases (see AD1) :

Secondary Header Field	Reference value
Data field header length	yes (286)
Instrument mode, format	bits 0 to 4, 8 depend on the APID field reference value for bits 5, 6, 7, 9 to 15
ICU on board time code	no
Redundancy definition vector	yes
Band characteristics	yes
Format Definition	Check deleted
Blank pixel data	blank pixels are monitored according to 4.3.1.4 below
Calibration data	yes
Spare words	no
Coarse Offsets	yes

Table 4.3.1.3.-1 : Secondary header fields / reference

Note : in table 4.3.1.3.-1 above, bit 0 is the most significant, as in AD8.

Whenever a check is negative, the "format error" PCD is incremented. Each sample in the packet data field is flagged as "dubious".

If the values read from the packet headers are the same from the first frame of the L0 product to the second one, but different from the reference values, an inconsistency between the processing data bases and the current instrument settings is detected. The "database " PCD is set.

4.3.1.4 - Blank pixel monitoring

In each packet the blank pixels are read, they are checked against a maximum value, average values are computed for two subsets and their difference is checked against a maximum value. A counter is incremented each time a tested values is above the specified threshold. That counter will be used for elaborating a set of Product Confidence Data (PCD) at product level (see § 11. below).

4.3.1.5. - Packet sequence checking

The following constraints define a valid packet sequence :

- the packet sequence counter (PC) may start at any value;
- the PC may be reset during the sequence, without disruption in the data flow;
- the PC should be incremented by 1 every packet, with reset to 0 every 16 384 packets;
- the ICU on-board time counter should be incremented every B+1 packet, when the band number is reset;
- the band number should be incremented by 1 every packet, with reset to 0 every B+1 packets.

If the PC is incremented by more than 1 (modulo the PC wraparound value),

- either the PC has been reset to 0; this is considered a normal event and no further check is done;
- or a small gap has occurred (at most 16 packets). Then
  - the "transmission error" PCD is incremented by the number of missing packets;
  - affected frames are flagged invalid ("valid_frame_f" is set to FALSE) and radiances are reset to null for all pixels and all bands;
  - dummy frames with null radiances and the "valid_frame_f" flag set to FALSE are inserted in the data if needed;
  - a flag is set to true in order to allow cosmetic filling of the one or two frames containing the missing packets (otherwise, this flag is always set to false);
- or a larger gap has occurred. Then
  - the "transmission error" PCD is incremented by the number of missing packets;
  - affected frames are flagged invalid ("valid_frame_f" is set to FALSE) and radiances are reset to null for all pixels and all bands;
  - dummy frames with null radiances and the "valid_frame_f" flag set to FALSE are inserted in the data;

If the band number is not incremented by the same amount as the packet counter, modulo B+1 (taking resets into account), a format error exception is raised : the "format error" PCD is incremented;

If the on-board time counter is not incremented by 11 or 12 (FR mode), 45 or 46 (RR mode) (it should be noted that the frame time of 44 ms does not correspond to an integer number of ticks) between two resets of the band number, an instrument problem is likely. The "format error" PCD is incremented.

The on-board time counter is calibrated in order to yield a UTC time for each frame : T_JD[f]. This may be done using ESA Time conversion library CFI (see AD4).

After sequence checking, packets are grouped by frame : a set of B+1 packets with numbers in sequence, with the same time code and with band number from 0 to B.

4.3.1.6. - Packet contents extraction

For each frame :

- The frame time T_JD_f obtained from the ICU time code field of the first packet of the frame is stored in MJD transport format (see AD7) and provided to the radiometric processing and the geo-location algorithms;
- Useful modules are extracted from the "Measurement data" field of the B+1 packets, formatted in one array $X_{b,k,m,f}$ and submitted to the saturated pixels detection and radiometric processing algorithms.

4.3.2 - List of Variables

Variable	Descriptive Name	T	U	Range - References
UTC_REF_FOR_OBT	UTC reference time for OBT conversion	s	jd	from L0 product header
OBT_REF	OBT counter value corresponding to the reference UTC	s	ct	from L0 product header
OBT_TICK	Duration of one tick of the OBT counter	s	ms	from L0 product header
B	Number of MERIS bands	s	dl	15
KB	Number of blank pixels in one module	s	dl	14
K^{RR}	Number of columns in one RR module	s	dl	185
K^{FR}	Number of columns in one FR module	s	dl	740
Mt	Number of MERIS modules	s	dl	5
DFH_LENGTH_R	Ref. value for data field header length	s	nc	286
MODE_MASK	Binary mask for the APID dependent bits in the instrument mode field	s	nc	
MODE_BITS_R^{RR}	dictionary of ref. values for APID dependent bits in instrument mode field	s	nc	indexed by APID values
MODE_BITS_R^{FR}	dictionary of ref. values for APID dependent bits in instrument mode field	s	nc	indexed by APID values
OCL_MASK	Binary mask for the OCL dependent bits in the instrument mode field	s	nc	
OCL_R	OCL switch reference	s	nc	
OB_MASK	Binary mask for the on-board correction switch dependent bits in the instrument mode field	s	nc	
OB_R	on-board correction switch reference	s	nc	
OTHER_MASK	Binary mask for the other bits in the instrument mode field	s	nc	
OTHER_BITS_R	Ref. value for other bits in instrument mode field	s	nc	
REDUND_VECTOR_R	Ref. value for redundancy vector	s	nc	
BAND_POS_R[b]	Ref. values for band position	s	nc	b: 0..B*
BAND_LEN_R[b]	Ref. values for band length	s	nc	b: 0..B*
BAND_GAIN_R[m,b]	Ref. values for band gain settings	s	nc	b:0..B*; m:1..Mt
BAND_MB_R[b]	Ref. values for no. of micro-bands	s	nc	b:0..B*
COARSE_THR[l]	Upper threshold for coarse offsets of each µband	s	nc	l:1..45*
RELAX_COF_R[b]	Weights for on-board Spatial and Temporal Relaxation (per band)	s	nc	b: 0..B
BLANK_THR[b]	Upper threshold for blank pixels	s	nc	b:0..B-1*
BLANK_DIF_THR[b]	Difference threshold for blank pixels	s	nc	b:0..B-1*
MS_TO_JD	Expression of 1 ms in MJD2000	s	jd/ms	1/86 400 000
PC_WRAPAROUND	Wraparound value for PC	s	dl	
MAX_GAP_P	Maximum gap between two packets allowing cosmetic filling	s	dl	
DT^{RR}	Delay between two RR frames	s	ms	176
DT^{FR}	Delay between two FR frames	s	ms	44
PK_LEN^{RR}	Packet length field for RR	s	dl	2135 (AD8)
PK_LEN^{FR}	Packet length field for FR	s	dl	7685 (AD8)
PK_SCALE	scaling factor for packet header float data coding	s	dl	16384 (AD8)

Table 4.3.2-1 : List of Variables

Variable	Descriptive Name	T	U	Range - References
M	Number of MERIS modules to process	i	dl	from 1.5.1
first_module	index of the first module to extract	i	dl	from 1.5.1
begin_JD	time of first frame to extract	i	jd	from 1.5.1
end_JD	time of last frame to extract	i	jd	from 1.5.1
DT_TICKS	Delay in OBT ticks between two frames (nearest lower integer)	c	ct	45 in RR, 11 in FR
dsrn	level0 product Data Set Record index	c	dl	
NP	total number of packets (or DSR)	c	dl	
current_f	frame counter	c	dl	
current_p	packet counter	c	dl	0..PC_WRAPAROUND-1
current_b	band counter	c	dl	0..B*
current_OBT	OBT counter	c	ct	
new_p	packet counter value read in current packet	c	dl	
new_OBT	OBT value read in current packet	c	ct	
packet	current packet data structure	c	dl	
new_JDT	MJD2000 time computed from packet on board timer	c	jd	
wide_gap	flag indicating that a sequence disruption larger than MAX_GAP_P has occurred	c	dl	Boolean
first_frame_hdr[b]	structure containing copies of the headers of the first frame	c	dl	
coarse_of_r[l,b]	Ref. values for coarse offsets	c	nc	b:0..B*;l:1..BAND_MB_R[b]
blank[b,k,m,f]	blank pixel data for frame f	c	nc	k: 1..KB
corrupt_packet	flag indicating that packet is corrupted	c	dl	Boolean
T_JD[f]	MJD2000 time for frame f	o	jd	to 1.3, 1.5.2, 1.5.5
X^{RR}[b,k,m,f]	pixel data for RR frame f	o	nc	to 1.2, 1.3; k:1..K^{RR}
dubious_f^{RR}[b,k,m,f]	dubious sample flag for frame f	o	dl	to 1.2; k:1..K^{RR}
X^{FR}[b,k,m,f]	pixel data for RR frame f	o	nc	to 1.2, 1.3; k:1..K^{FR}
dubious_f^{FR}[b,k,m,f]	dubious sample flag for frame f	o	dl	to 1.2; k:1..K^{FR}
blank_PCD[b,m,f]	counter of out-of-range blank pixels	o	dl	b : 0..B ; to 1.8
do_cosmetic_f[f]	flag enabling cosmetic filling of frame f	o	dl	to 1.3; Boolean
valid_frame_f[f]	valid frame flag	o	dl	to 1.2, 1.3, 1.4; 1.5.5, Boolean
database_PCD	flag set when auxiliary parameters read from a data base are found inconsistent with instrument packets	o	dl	to 1.8; Boolean
transmission_PCD	counter of transmission errors in the segment	o	dl	to 1.8
format_PCD	counter of format errors in the segment	o	dl	to 1.8
coarse_PCD	flag set when the coarse offsets are above a threshold	o	dl	to 1.8; Boolean

Table 4.3.2-1 : List of Variables (cont.)

NOTES :
* band numbering in pseudo-code of next section follows the packets internal coding of band numbers : bands 1 to 15 are numbered 0 to 14 and smear band is numbered 15 (see AD8).

The header of the MERIS packet is described by the data structures packet_header_type and data_field_header_type; table 4.3.2-2 below shows its correspondence with the packet structure description in AD8.

```
typedef packet_header_type struct {
  version : 0..7;                                    PCK_VERSION
  type : 0..1;                                       PCK_TYPE
  data_fld_hd_f : 0..1;                              DATA_FLD_HD_FLAG
  APID : unsigned short;                             APP_ID
  sgflag : 0..3;                                     SEG_FLAG
  counter : unsigned short;                          SEQ_COUNT
  length : unsigned short;                           PACKET_LENGTH
}
typedef data_field_header_type struct {
  data_fld_hd_len : unsigned short;                  DATA_FLD_HD_LEN
  mode : unsigned short;                             MODE_FORMAT
  obt : unsigned long;                               ICU_OBT
  redund_vector : unsigned short;                    REDUND_VECTOR
  band_char : band_char_type;                        BD_CHARACTER
  format_defn : byte;                                FORMAT_DEFN
  blank_pixel : unsigned short[14][5];               BLANK_PIXEL
  coarse_offsets : unsigned short[35];               CAL_DATA.COARSE_OFFSETS
  Aij_coeff : unsigned short[16];                    CAL_DATA.AIJ_COEF
  Kbm : unsigned short[5];                           CAL_DATA.KBM_COEFF
  FOV_parameter : unsigned short;                    CAL_DATA.FOV_PARAMETER
  cal_frames : unsigned short;                       CAL_DATA.NB_FRAMES
  Abm : unsigned short[5];                           CAL_DATA.ABM_COEF
  spare : unsigned short;                            SPARE
}
typedef band_char_type struct {
  BD_POS : unsigned short;                           BD_POS
  BD_LEN : unsigned char;                            BD_LEN
  GN_FACT : unsigned char[Mt];                       GN_FACT
  BD_NUM : unsigned char;                            BD_NUM;
  MBD_LEN : unsigned char;                           MBD_LEN
}
```

Table 4.3.2-2 : Description of the packet data structures (left : DPM identifiers, right : AD8 identifiers)

```
typedef packet_type struct {
  header: packet_header_type;
  sec_header: data_field_header_type;
  data_field: unsigned short[Mt,K^RR/FR];
}
```

Table 4.3.2-3 : Description of the MERIS packet structure

Important note: AD8 uses a pixel indexing convention linked to the instrument electronics, opposite to the one adopted for this document. This is taken into account in the Equations section below (step 1.1.3-2).

4.3.3 - Equations

The numbers between parentheses at the right of each comment or pseudo-code line are unique numbers for individual processing steps.

All equations are written here for RR processing. FR processing is the same as RR processing except that variables with FR superscript should replace those with RR superscript, as appropriate.

Structure `packet` used in equations below is of type packet_type (see table 4.3.2-3 above).

Note on exception processing: the statement `raise(<exception_identifier>)` corresponds to the activation of the corresponding routine in the "exception handling" section.

int is the truncation to lower integer function, nint is the truncation to nearest integer function, % is the modulo function.

Step 1.1.0 Initialisations and packet selection

Initialisations

```
current_f=0;                                                          (1.1.0-1)
deleted                                                               (1.1.0-2)
current_b=B;                                                          (1.1.0-3)
extract total number of packets, NP, from level0 product SPH;        (1.1.0-4)
extract UTC reference time for OBT conversion, UTC_REF_FOR_OBT, from level0
product MPH;                                                          (1.1.0-5)
extract OBT value corresponding to UTC reference time, OBT_REF, from level0
product MPH;                                                          (1.1.0-6)
extract duration of the OBT counter tick, OBT_TICK, from level0 product
MPH;                                                                  (1.1.0-7)
```

compute frame sampling step duration in OBT ticks (nearest lower integer) *(1.1.0-11))*

```
DT_TICKS = int(DT^RR / OBT_TICK);
```

convert tick duration in mjd2000 *(1.1.0-12))*

```
JD_TICK = OBT_TICK*MS_TO_JD;
```

Main loop

```
for (dsrn=0; dsrn < NP; dsrn++) {
```

extract packet from product *(1.1.0-8)*

```
   read one MERIS packet from Level 0 product MDS at dsr number: dsrn, store
   it in structure packet;
```

if first extracted packet, initialise current_p *(1.1.0-13)*

```
   if (dsrn ==0) {
     if(packet.header.counter==0)
      current_p = PC_WRAPAROUND-1;
     else
```

```
    current_p = packet.header.counter - 1;
}
```

calibrate timer *(1.1.0-9)*

```
call ESA CFI SBTUTC
  inputs : UTC_REF_FOR_OBT, OBT_REF, OBT_TICK, packet.sec_hdr.obt
  outputs : new_JDT
```

select packets within product limits *(1.1.0-10)*

```
if ((new_JDT - begin_JD) < JD_TICK)
  current_p ++;
else if ( ((new_JDT-begin_JD)≥JD_TICK) && ((new_JDT-end_JD)≤JD_TICK) ) {
```

skip incomplete frames at start of selection *(1.1.0-11)*

```
    if ( (current_f == 0) && (current_b == B) &&
    ((packet.sec_hdr.band_char.BD_NUM ) != 0 )
      raise (transmit_error_x)
```

Step 1.1.1.1 Check Packet Header

packet header length check *(1.1.1.1-1)*

```
  if (packet.header.length != PK_LEN^RR)
    raise(transmit_error_x);
```

data field header length check *(1.1.1.1-2)*

```
  if (packet.sec_hdr.data_fld_hd_len != DFH_LEN_R)
    raise(format_error_x);
```

instrument mode field check

Check APID dependent bits

```
  if((packet.sec_hdr.mode & MODE_MASK)!= MODE_BITS_R^RR[packet.hdr.APID])
    raise(format_error_x);
```

(1.1.1.1-3)

Check OCL dependent bits

```
  if (OCL_R && ((packet.sec_hdr.mode & OCL_MASK) != OCL_MASK)))
    raise(format_error_x);                                      (1.1.1.1-4)
  if ((! OCL_R) && ((packet.sec_hdr.mode & OCL_MASK) != 0)))
    raise(format_error_x);                                      (1.1.1.1-6)
```

Check on-board correction switch dependent bits

```
  if (OB_R && ((packet.sec_hdr.mode & OB_MASK) == 0)))
    raise(format_error_x);                                      (1.1.1.1-7)
  if ((! OB_R) && ((packet.sec_hdr.mode & OB_MASK) == OB_MASK)))
    raise(format_error_x);                                      (1.1.1.1-8)
```

Check other bits

```
  if ((packet.sec_hdr.mode & OTHER_MASK)!= OTHER_BITS_R)
    raise(format_error_x);                                      (1.1.1.1-9)
```

redundancy vector field check *(1.1.1.1-10)*

```
  if (packet.sec_hdr.redund_vector != REDUND_VECTOR_R) )
    raise(format_error_x);
```

extract and check band index *(1.1.1.1-11)*

```
current_b = (current_b+1) % (B+1);
if (current_b == 0) {
  current_f++;
  do_cosmetic_f[f] = FALSE;
  valid_frame_f[current_f] = TRUE;
}
if ( (packet.sec_hdr.band_char.BD_NUM ) != current_b )
  raise (format_error_x);
```

check band characteristics *(1.1.1.1-12)*

```
if ((packet.sec_hdr.band_char.BD_POS != BAND_POS_R[current_b]) ||
  (packet.sec_hdr.band_char.BD_LEN != BAND_LEN_R[current_b]) ||
  (packet.sec_hdr.band_char.GN_FACT[*] != BAND_GAIN_R[current_b,*]) ||
  (packet.sec_hdr.band_char.MBD_LEN != BAND_MB_R[current_b]))
    raise(format_error_x);
```

deleted *(1.1.1.1-13)*

check calibration data *(1.1.1.1-14)*

check the relaxation coefficients

```
if
(packet.sec_hdr.Aij_coeff[*]!=nint(RELAX_COF_R[current_b]*PK_SCALE))
  raise(format_error_x);
```

check coarse offsets

```
if (current_f == 1) {
  coarse_of_r[current_b,*] = packet.sec_hdr.coarse_offsets[*];    (1.1.1.1-15)
  if (coarse_of_r[current_b,*] > COARSE_THR[current_b])
    coarse_PCD = TRUE;                                              (1.1.1.1-16)
}
else
  if (packet.sec_hdr.coarse_offsets[*] != coarse_of_r[current_b,*])
    raise(format_error_x);                                          (1.1.1.1-17)
```

detect inconsistency with auxiliary parameters data base *(1.1.1.1-18)*

```
if ((current_f == 1) && corrupt_packet) {
  copy packet.sec_hdr into first_frame_hdr [current_b];
}
if ((current_f == 2) && corrupt_packet) {
  if (packet.sec_hdr == first_frame_hdr [current_b])
    raise (auxiliary_parameters_x);
  }
}
```

Step 1.1.1.2 Blank Pixels Monitoring

store blank pixel data in working array *(1.1.1.2-1)*

```
for (m=1; m <= Mt; m++) {
  for (k=1; k <= KB; k++) {
    blank[current_b,k,m,current_f] = packet.sec_hdr.blank_pixel[k,m];
```

check absolute value *(1.1.1.2-2)*

```
    if (blank[current_b,k,m,current_f] > BLANK_THR[current_b])
      blank_PCD[current_b,m,current_f]++;
  }
```

check difference *(1.1.1.2-3)*

if ($\left(\frac{1}{5}\sum_{k=6}^{10}\mathrm{blank[current\_b,k,m,current\_f]}\right)-\left(\frac{1}{4}\sum_{k=11}^{14}\mathrm{blank[current\_b,k,m,current\_f]}\right)$ >

```
      BLANK_DIF_THR[current_b])
    blank_PCD[current_b,m,current_f]++;
}
```

Step 1.1.2 Check Packet Sequence

initialise current_OBT if first selected packet *(1.1.2-0)*

```
if (current_f == 1 && current_b == 0)
  current_OBT = packet.sec_hdr.obt - DT_TICKS;
```

sequence errors check *(1.1.2-1)*

```
current_p= (current_p+1)%PC_WRAPAROUND;
new_p = packet.header.counter;
new_OBT= packet.sec_hdr.obt;
new_b = packet.sec_hdr.band_char.BD_NUM;
```

detect disruption in packet counter *(1.1.2-2)*

```
if (new_p < current_p) {
```

new_p=0 : normal packet counter reset, update reference value *(1.1.2-6)*

```
  if (new_p == 0) {
```

Check OBT disruption due to instrument PAUSE mode, if detected pad with packets

```
    if (!((new_OBT - (current_OBT+DT_TICKS)) in [0..1]) ) {
      n_miss_frames = int((new_OBT - current_OBT)*OBT_TICK/DT^RR) - 1
      pk_gap= n_miss_frames*(B+1) + (B+1)-current_b
      current_p = new_p - pk_gap
      raise (missing_packets_x);
    }
    else current_p = 0
  }
  else {
```

OBT lower than before : transmission error (packet overlap) *(1.1.2-3)*

```
    if (new_OBT <= current_OBT)
      decrement current_b, current_p taking care of limits and of
      current_f (see 1.1.1.1-11)
      raise (transmit_error_x);
```

OBT too high = gap in sequence : assume transmission error, pad with packets *(1.1.2-4)*

```
    else if ((new_OBT - (current_OBT+DT_TICKS)) > 1)
      raise (missing_packets_x);
```

```
        }
      }
      else if (new_p > current_p) {
```
packet counter too high = gap in sequence : assume transmission error, pad with packets (1.1.2-7)
```
          raise (missing_packets_x);
      }
```
new frame, on-board timer should increase (1.1.2-5)
```
      if (current_b == 0) {
        if (!((new_OBT - (current_OBT+DT_TICKS)) in [0..1]) ) {
          raise (format_error_x) ;
          current_OBT += DT_TICKS;
        }
        else
          current_OBT= new_OBT;
      }
```

Step 1.1.3 Extract Packet Content

set frame time (1.1.3-1)
```
      if (current_b == 0) T_JD^RR[current_f] = new_JDT;
```
check sampling time regularity (1.1.3-4)
```
      if (current_f > 1)
```
$$\text{if}\left(\left|\text{new\_JDT} - \left(\text{T\_JD}^{RR}[1] + (\text{current\_f} - 1)*\text{DT}^{RR}*\text{MS\_TO\_JD}\right)\right| > \text{OBT\_TICK}\right);$$
```
          raise (format_error_x) ;
        end if
      end if
```
extract data from useful modules, revert pixel numbering (1.1.3-2)
```
      for m = 1,M
       for k = 1,K^RR
         X^RR[current_b,k,m,current_f] =
            packet.data_field[K^RR+1-k,m+first_module-1];
```
flag all samples as "dubious" if a format error has been detected (1.1.3-3)
```
      if (corrupt_packet)
        dubious_f^RR[current_b,*,*,current_f]= TRUE;
      else
        dubious_f^RR[current_b,*,*,current_f]= FALSE;
```

} *end of selection on time limits*

} *end on input product*

Step 1.1.4 Exception processing

transmission error exception (1.1.4-1)
```
transmit_error_x() {
  transmission_PCD++;
  do not process packet further, process next packet;
}
```

format error exception *(1.1.4-2)*

```
format_error_x() {
  format_PCD++;
  corrupt_packet = TRUE;
  continue packet processing;
}
```

missing packets exception : add empty packets to fill gap, set default values for whole affected frame(s), set do_cosmetic flag if gap small enough *(1.1.4-3)*

```
missing_packets_x() {
  set current frame to invalid, reset radiance
  valid_frame_f[current_f] = FALSE;
  X^RR[*,*,*,current_f]=0;
  if a new frame is to be created, update current_OBT for next extraction
  if(new_b≤current_b)current_OBT = packet_sec_hdr.obt - DT_TICKS;
  check gap length, update cosmetic flag of current frame accordingly
  wide_gap = (new_p > (current_p + MAX_GAP_P)%PC_WRAPAROUND );
  if (wide_gap)
    do_cosmetic_f[current_f] = FALSE;
  else
    do_cosmetic_f[current_f] = TRUE;
  if gap begins with a new frame, set frame time
  if (current_b ==0) {
    if (current_f>1)
      T_JD^RR[current_f] = T_JD^RR[current_f-1] + DT^RR*MS_TO_JD;
    else
      T_JD^RR[current_f] = begin_JD;
  }
  pad with dummy packets
  do {
    transmission_PCD++;
    current_p=(current_p+1)%PC_WRAPAROUND;
    current_b=(current_b+1)%(B+1);
    if (current_b==0) {
    update frame index and time
     current_f++;
     T_JD^RR[current_f]=T_JD^RR[current_f-1]+DT^RR*MS_TO_JD;
     set new frame to invalid except if frame change is for loaded (valid) packet
     if(current_p<new_p) {
      X^RR[*,*,*,current_f]=0;
      if (wide_gap)
       do_cosmetic_f[current_f] = FALSE;
      else
       do_cosmetic_f[current_f] = TRUE;
     }
     else {
```

```
      valid_frame_f[current_f] = TRUE;
      do_cosmetic_f[current_f] = FALSE;
     }
    }
  } while(current_p != new_p)
  continue packet processing
}
```

auxiliary parameters exception : set database_PCD *(1.1.4-4)*

```
auxiliary_parameters_x() {
  database_PCD = TRUE;
  continue packet processing
}
```

4.3.4. - Accuracy Requirements

All comparisons and data extraction, as done on integers, must be exact.
Julian days computations and comparisons must be exact to the ninth significant digit.

4.3.5. - Product Confidence Data summary.

Most of the processing described in 4.1.2 above is control of the validity of the incoming data. The following PCD are generated in the process :

"valid_frame_f"	Boolean frame flag set to False for each frame for which at least one packet is missing in Level0 product
"dubious_f"	Boolean sample flag set for any sample extracted from a corrupted packet

These intermediate PCD are used by the following steps and reduced at the formatting step (see §11 below).

"blank_PCD" : counter of out-of-range blind pixels for each band, module.

This PCD is reduced at the formatting step (see §11 below).

transmission_PCD :	number of transmission errors which occurred in the product
format_PCD :	number of format errors which occurred in the product
database_PCD :	Boolean flag set when the processing parameters data base contents does not match the packet header contents
coarse_PCD :	Boolean flag set when the coarse offsets are above a threshold

These product level PCD are reflected in the Level 1B product header (see §11 below).

5. - MERIS Saturated Pixels detection Algorithm

5.1. - Introduction

This chapter describes the processing to be applied to the MERIS raw or on-board processed samples, in order to identify the saturated samples.

5.2. - Algorithm Overview

The algorithms scans the MERIS measurements to detect saturated samples, and flags these pixels as saturated and their neighbours as dubious, within an extent depending of the saturation characteristics.

5.3. - Algorithm Description

5.3.1. - Theoretical Description

5.3.1.1. - Physics of The Problem

MERIS samples may be affected by phenomena outside the range of the useful measurements, i.e. a spectral radiance between 0 and L_{sat} (as defined in RD2). Such samples are totally or partly invalid and must be identified before any further processing.

Such phenomena are :

1. **saturation** by radiance level above L_{sat} (caused by e.g. Sun glint, cloud, snow or ice), which affects samples temporarily. Typically several columns in several bands over several frames are saturated. Not all the components of the acquisition chain have the same saturation level, one may distinguish in ascending order :
   - the analogue-to-digital converters;
   - the video amplification chain;
   - the CCD shift register cells;
   - the CCD cells;
2. **recovery from saturation** : after saturation, components of the acquisition chain need some time to recover;
3. **blooming** : when an area of the CCD sensor is saturated, samples in bands and columns close to that area are temporarily affected by photons or photo-electrons diffusing from the saturated pixel;

Definitions : The radiance levels L_{sat}, L4, L_{sg} are defined in RD2.

5.3.1.2. - Mathematical Description of Algorithm

The saturated pixels processing follows the logic shown in the block diagram in fig 3.1.2-1 below.

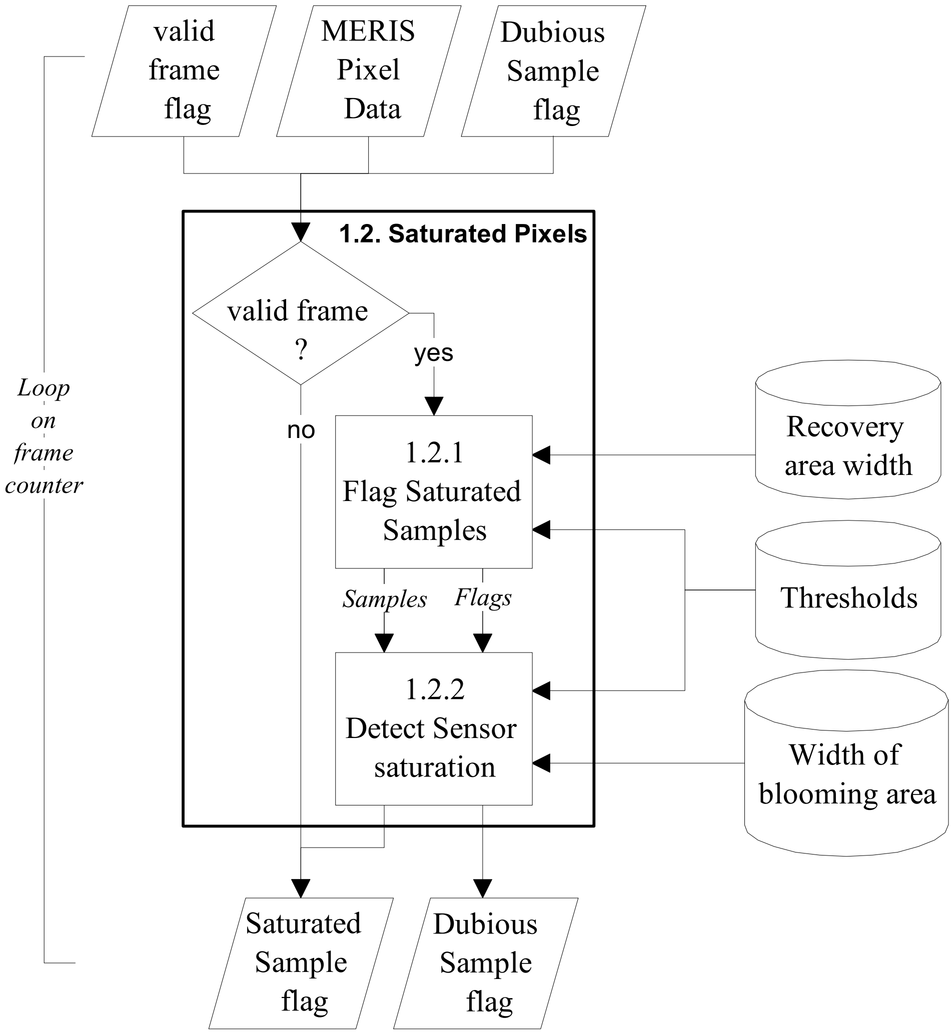


Figure 3.2.1-2 : Saturated pixels processing block diagram
(Note: the FR chain and RR chain architecture are identical)

5.3.1.2.1. - Saturation detection and flagging

Whenever a sample from (sum of) CCD cells has the saturation value (resolution and band dependant, due to the spatial relaxation coefficients and to the variable number of micro-bands), MERIS is assumed to be saturated. The "saturated sample" flag is raised for that sample. The samples from the same module and band processed by MERIS immediately after that one are affected by VEU recovery from saturation. For the **Sat_rec_k** following columns, the "dubious sample" flag is raised. Saturation may occur in the smear band so that the smear band samples shall be processed similarly to useful pixels.

5.3.1.2.2. - Sensor saturation detection and flagging

Upon saturation of the sensor by Sun glint, blooming is to be expected. When a pixel k,m,f is saturated in **all** bands CCD sensor saturation is assumed to have occurred.

Then :

a) all the bands of the pixel k,m,f is flagged as "saturated";

b) all the valid bands in the neighbouring **glint_bloom_k** measurement data pixels are flagged as "dubious" samples;

Otherwise no flag is raised nor modified.

5.3.2 - List of Variables

Variable	Descriptive Name	T	U	Range - References
B	Number of MERIS bands	s	dl	15
K^{RR}	number of columns in a RR module	s	dl	185
K^{FR}	number of columns in a FR module	s	dl	740
$SAT\_REC\_K^{FR}$	Number of following samples affected by an FR pixel saturation during read-out	s	dl	
$GLINT\_BLOOM\_K^{FR}$	Number of neighbour pixels affected by saturation in a pixel	s	dl	
$SAT\_SAMPLE^{FR}[b]$	Saturation value for a MERIS FR sample	s	nc	
RELAX_COF_R[b]	Weights for on-board Spatial and Temporal Relaxation (per band)	s	nc	b: 1..B+1
$SAT\_REC\_K^{RR}$	Number of following samples affected by an RR pixel saturation during read-out	s	dl	
$GLINT\_BLOOM\_K^{RR}$	Number of neighbour pixels affected by saturation in a pixel	s	dl	
NF	number of frames in Level1b product	i	dl	from 1.5.1
M	Number of MERIS modules to process	i	dl	from 1.5.1
$X^{RR}[b,k,m,f]$	Pixel data for RR frame f	i	nc	from 1.1
$dubious\_f^{RR}[b,k,m,f]$	dubious sample flag for RR frame f	i/o	dl	from 1.1
$X^{FR}[b,k,m,f]$	Pixel data for FR frame f	i	nc	from 1.1
$dubious\_f^{FR}[b,k,m,f]$	dubious sample flag for FR frame f	i/o	dl	from 1.1
valid_frame_f[f]	valid frame flag	i	dl	from 1.1
$SAT\_SAMPLE^{RR}[b]$	Saturation value for a MERIS RR sample	c	nc	
saturated	number of saturated samples per pixel	c	dl	
$saturated\_f^{RR}[b,k,m,f]$	saturated sample flag for RR frame f	o	dl	to 1.3
$saturated\_f^{FR}[b,k,m,f]$	saturated sample flag for FR frame f	o	dl	to 1.3
$dubious\_f^{RR}[b,k,m,f]$	dubious sample flag for RR frame f	i/o	dl	to 1.3
$dubious\_f^{FR}[b,k,m,f]$	dubious sample flag for FR frame f	i/o	dl	to 1.3

5.3.3. - Equations

5.3.3.1. - RR Processing

1.2.0 initialisation :

compute band saturation levels for RR samples from FR values (1.2.0-1)

```
for each band b ∈{1..B,s}
  SAT_SAMPLE^RR[b] = SAT_SAMPLE^FR[b] * RELAX_COF_R[b] * 16
end for

for each frame f {1..NF}
  if (valid_frame_f[f])
    for each module m ∈{1..M}
```

1.2.1 flag saturated samples :

```
      for each pixel k ∈{1..K^RR}
```

reset saturated samples counter (1.2.1-1)

```
        saturated=0
        for each band b ∈{1..B,s}
          if (X^RR[b,k,m,f] ≥ SAT_SAMPLE^RR[b]) then
```

saturated sample : set its "saturated" flag to TRUE (1.2.1-2)

```
            saturated_f^RR[b,k,m,f] = True
```

saturated sample : increment saturated samples counter (1.2.1-3)

```
            saturated = saturated + 1
            for eack sample k' ∈{k+1, k+SAT_REC_K^RR }
```

saturated sample : flag "dubious" the SAT_REC_K^RR next read samples (1.2.1-4)

```
              dubious_f^RR[b,k+k',m,f] = True
            end for
          end if
        end for
```

1.2.2 blooming detection :

```
        if (saturated ≥ B) then
          for each dk ∈{1..GLINT_BLOOM_K^RR}
            for each b ∈{1..B,s}
```

blooming detected : flag "dubious" the GLINT_BLOOM_K^RR next pixels (1.2.1-5)

```
              if(k+dk ≤ K^RR) dubious_f^RR[b,k+dk,m,f] = True
```

blooming detected : flag "dubious" the GLINT_BLOOM_K^RR previous pixels (1.2.1-6)

```
              if(k-dk ≥ 1) dubious_f^RR[b,k-dk,m,f] = True
            end for
          end for
        end if
      end for
    end for
  end if
end for
```

5.3.3.2 - FR Processing

1.2.0 initialisation :

```
no initialisation is required for FR Processing

for each frame f
   if (valid_frame_f[f])
      for each module m ∈{1..M}
```

1.2.1 flag saturated samples :

```
         same processing as in RR mode, replacing the variables indexed RR
         with FR as appropriate
```

1.2.2 blooming detection :

```
         same processing as in RR mode, replacing the variables indexed RR
         with FR as appropriate

      end for

   end if

end for
```

5.3.4. - Accuracy Requirements

All comparisons between samples and saturation values, as done on integers, must be exact.

5.3.5. - Product Confidence Data summary

Sample level PCD : "saturated_f" flag;
"dubious_f" flag.

6. - MERIS Radiometric Processing Algorithm

6.1. - Introduction

This chapter describes the radiometric processing to be applied to the MERIS raw or on-board pre-processed samples, in order to derive **corrected top of atmosphere radiance values**. Radiometric processing is part of the MERIS Level 1b processing.

6.2. - Algorithm Overview

Depending on whether samples are Full or Reduced Resolution samples, have been processed on board or not, the incoming MERIS samples are processed one by one into radiance at TOA. Radiometric processing includes :

- non-linearity correction (if corrections not done on-board and corresponding switch set to "enabled")
- dark signal correction (if not on-board)
- smear correction (if not on-board)
- absolute gain calibration (different on-board and on-ground)
- temperature corrections of dark signal, smear, gain (if corrections not done on-board)

At the end of the correction steps some missing samples are filled with cosmetic radiance values and flagged "cosmetic" :

- radiances of pixels listed in the "dead pixels" map are replaced by an interpolation of their valid neighbours,
- Empty frames generated during extraction because of missing packets are filled, if the packet gap is small enough, by values from the previous valid frame.

6.3. -Algorithm Description

6.3.1 - Theoretical Description

6.3.1.1. - Physics of The Problem

The valid MERIS samples are digital counts resulting from the detection and acquisition by MERIS of a bi-dimensional field of spectral radiance in front of the instrument. The objective of the radiometric processing, together with the stray light correction (see chapter 5 below), is to estimate that spectral radiance. An inverse model of the MERIS processing is used for that purpose, using parameters stored in the Characterisation and Radiometric Calibration data bases and the MERIS samples themselves. The MERIS acquisition model may be described as :

$$X_{b,k,m,f} = NonLin_{b,m}\left[g(T_f^{VEU}).\left[A_{b,k,m}.\left(L_{b,k,m,f} + G_{b,k,m}(L_{*,*,*,f}) \right) + Sm_{b,k,m,f}(L_{b,k,m,*}) \right] + g_c(T_f^{CCD}).C^0_{b,k,m} \right] + \varepsilon$$

where

- $X_{b,k,m,f}$ is the MERIS raw sample (not yet corrected on board);
- $NonLin_{b,m}$ is a non-linear function, representing the non-linear transformations which take place in the CCD, amplifier and A/D converter; NonLin depends on band and gain settings;

- T_f^{VEU} is the temperature of the MERIS amplifiers (VEUs) at the time of frame f;
- T_f^{CCD} is the temperature of the MERIS detectors (CCDs) at the time of frame f;
- g and g_c are (dimensionless) temperature correction functions;
- $AL_{b,k,m}$ the "absolute radiometric gain" in counts/radiance unit; AL depends on band & gain settings;
- $L_{b,k,m,f}$ the spectral radiance distribution in front of MERIS;
- $Sm_{b,k,m,f}$ the smear signal, due to continuous sensing of light by MERIS;
- $C^0_{b,k,m}$ the calibrated dark signal (possibly including an on-board compensation), dependent on band and gain settings;
- $G_{b,k,m}$ a linear operator (weighted sum) representing the stray light contribution to the signal. For a given sample, some stray light is expected from all the other samples in the module, spread into the sample by specular (ghost image) or scattering processes (see chapter 7).
- ε is a random process representative of the noise and measurement errors.

Note : all the above quantities, if they are subscripted k and/or f, are sampled at either full or reduced resolution, referred to as FR or RR hereafter.

Assuming that ε can be estimated and accounted for in the error budget, the purpose of radiometric processing is to retrieve $[L_{b,k,m,f} + G_{b,k,m}(L_{*,*,*,*})]$ from $X_{b,k,m,f}$ using knowledge of $NonLin_{b,m}$, $C^0_{b,k,m}$, $AL_{b,k,m}$, T_f, g and g_c.

The MERIS instrument itself provides a number of characterisation measurements supporting the radiometric processing :

- a **smear band** $X_{s,k,m,f}$ includes an integrated measure of $S_{*,k,m,f}$, $C_{s,k,m,f}$, $G_{*,k,m}(L_{*,k,m,f})$, and noise.

6.3.1.2. - Mathematical Description of Algorithm

The algorithm processes input data pixel by pixel following the flow chart in figures 6.3.1.2-1 and 6.3.1.2-2 below. The processing of FR and RR data is highly similar.

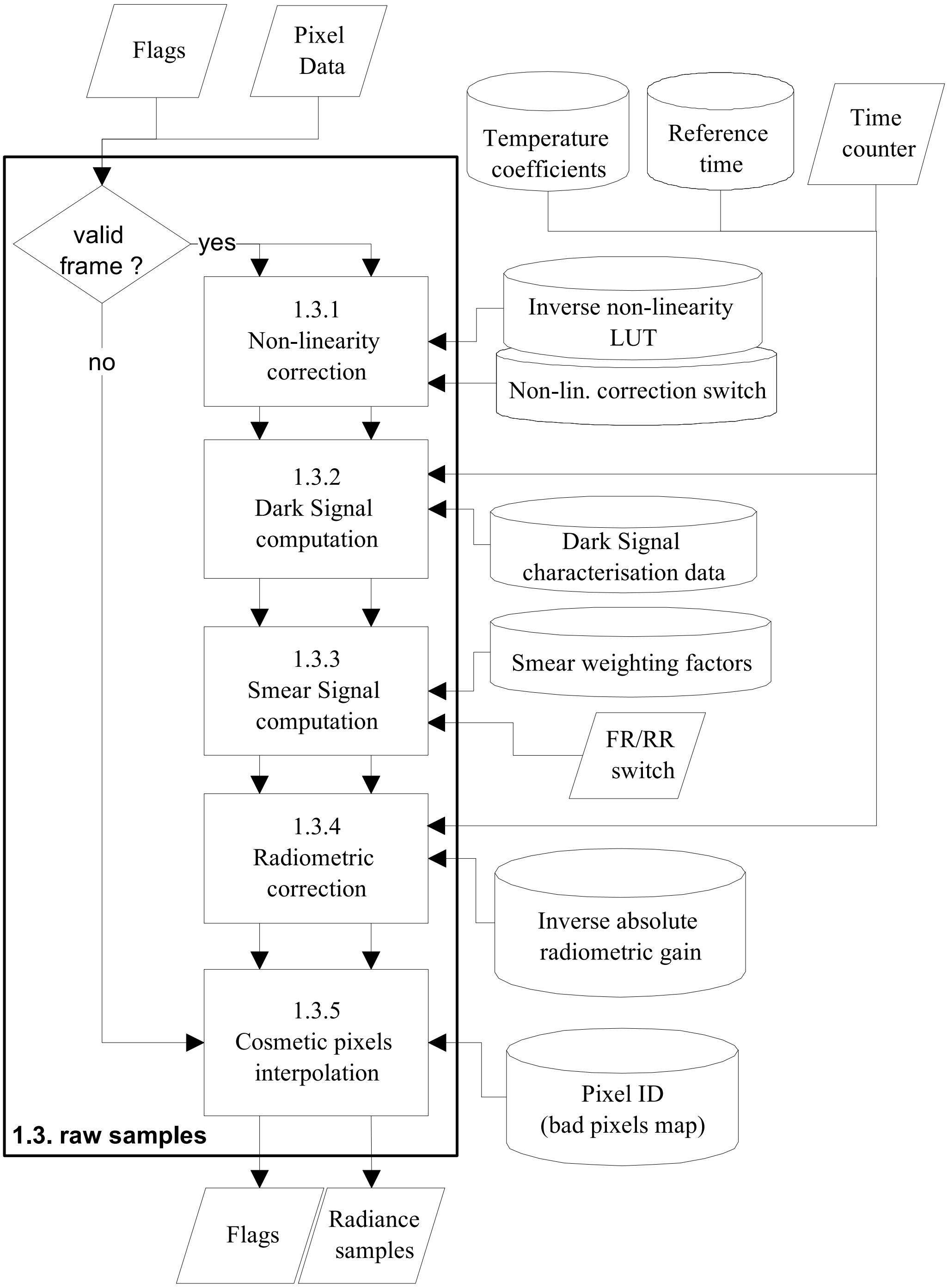


Figure 3.1.2-1 : Radiometric processing block diagram, RR and FR Raw samples

6.3.1.2.1. - RR Raw samples processing branch

6.3.1.2.1.1 - Non-linearity correction

The non-linearity correction applies to all valid samples of all bands, including the smear band.

Correction for non-linearity is provided by replacing each raw data quantised value by the corresponding corrected value. Corrected values are read from a look-up table implementing an approximation of the reciprocal of the function NonLin for each possible numerical level of a micro-band of each band of each module. The ADC converting micro-band signal to counts having 12 digis, the number of entry of the table associated to any micro-band is 4096. Befor any correction can take place, correction tables at band level must be derived from tables at micro-band level using the parameters of the band samples building processes: spatial relaxation (micro-bands accumulation into band) and spatial relaxation (FR samples weighted sum to build RR samples) and the assumption of local invariance of the signal over the relaxation domains.

6.3.1.2.1.2 - Dark signal correction coefficient

The dark signal correction applies to all valid samples of all bands, including the smear band.

Nominal processing is with the on-board Offset Control Loop enabled.

A correction of the uncompensated dark signal is applied based on

- the dark signal characterisation measurements $C^{0RR}_{b,k,m}$,corrected for temperature dependancies and representative of signal for reference temperatures T^{CCD}_{ref} and T^{VEU}_{ref} .
- a temperature dependent correction expressed as a polynomial. As that correction depends only on CCD temperature T^{CCD}_{f}, and that temperature T^{CCD}_{f} depends only on the time elapsed since instrument switch-on, the correction may be simply expressed as a function of time.

$$C_{b,k,m,f} = C^{0RR}_{b,k,m}\cdot\left[g_{c0} + g_{c1}(t_f - t_{ref}) + g_{c2}(t_f - t_{ref})^2\right]$$

The reference time is intentionally left without CCD superscript because it corresponds to the temperature of CCD and VEU for the same calibration measurements. It is in fact not absolute time but relative to the ascending crossing nodal time (CNT). As it depends on solar elevation angle, it varies with time and therefore it is read from the Reference Time Calendar in the Level1b Processing parameters data base. Obviously, t_f in the above equation must be relative to crossing nodal time as well.

Processing with OCL disabled : it must be ensured that a valid set of $C^0{}_{b,k,m}$ with OCL disabled is available (see § 4 above). The algorithm is the same as above.

6.3.1.2.1.3 - Smear correction coefficient

The smear correction applies to all valid samples of all bands, except the smear band.

The smear correction coefficient is estimated from the offset-corrected smear band in the current frame :

$$S_{b,k,m,f} = Ksm_b^{RR}.(X'_{s,k,m,f} - C_{s,k,m,f})$$

Note: $S_{b,k,m,f}$ is an estimate of $g(T_f^{VEU}) \cdot Sm_{b,k,m,f}(L_{b,k,m,*})$

If a smear sample is "saturated" all other bands for the same pixels are flagged saturated and processed as such.

If a smear sample happens to be "dead", it is assumed that all MERIS bands for the same pixel are listed in the "dead pixels" map.

6.3.1.2.1.4 - Radiometric correction

The inverse of the absolute instrument gain $AL_{b,k,m}$ is applied to the valid samples of all bands after dark and smear signal subtraction, with a compensation for the estimated temperature which, as before (see 3.1.2.1.2), is expressed as a function of time :

$$R_{b,k,m,f} = \left(AL^{RR}_{b,k,m}\right)^{-1}.\left\{\left(X'_{b,k,m,f} - S_{b,k,m,f}\right).\left[g_0 + g_1(t_f - t_{ref}) + g_2(t_f - t_{ref})^2\right] - C_{b,k,m,f}\right\}$$

If a sample is flagged "saturated", correction is by-passed and a default value is assigned to it.

6.3.1.2.1.5. - Cosmetic pixels interpolation

The radiances $R_{b,k,m,f}$ of any sample listed in the "dead pixels" map is replaced by a linear interpolation of the neighbour columns (in the same band). In the along track direction, where no more than two consecutive samples (in the same band) are to be cosmetically filled (frames flagged "do_cosmetic" during the packet extraction, see chapter 4), interpolation is constant and each partially invalid frame is replaced as a whole, to avoid spectral signatures mixing.

6.3.1.2.2. - FR Raw samples processing branch

The non-linearity correction is the same as for RR processing described in 3.1.2.1.1 above. The look-up table at micro-band level is the same as for RR processing, the band level tables are build taking account of spectral relaxation only.

The dark signal correction coefficient computation is the same as for RR processing described in 6.3.1.2.1.2. above. The characterisation data $C^{0FR}_{b,k,m}$ are specific of FR processing. The g_{c0}, g_{c1}, g_{c2} coefficients are the same as in RR processing.

The smear correction coefficient is estimated from the offset-corrected smear band in the next frame and the current frame :

$$S_{b,k,m,f} = \left[Ksm^{FR}_{b,1}.\left(X'_{s,k,m,f} - C_{s,k,m,f+1}\right) + Ksm^{FR}_{b,2}.\left(X'_{s,k,m,f-1} - C_{s,k,m,f}\right)\right].$$

Note: $S_{b,k,m,f}$ is an estimate of $g(T_f^{VEU}) \cdot Sm_{b,k,m,f}(L_{b,k,m,*})$

The radiometric correction is the same as for RR processing described in 3.1.2.1.4 above. The characterisation data $AL^{FR}{}_{b,k,m}$ are specific of FR processing. The g_0, g_1, g_2 coefficients are the same as in RR processing.

The Cosmetic pixels interpolation is the same than in the RR on-ground processed samples processing branch (see § 6.3.1.2.1.5) except that the "dead pixels" map is specific to FR.

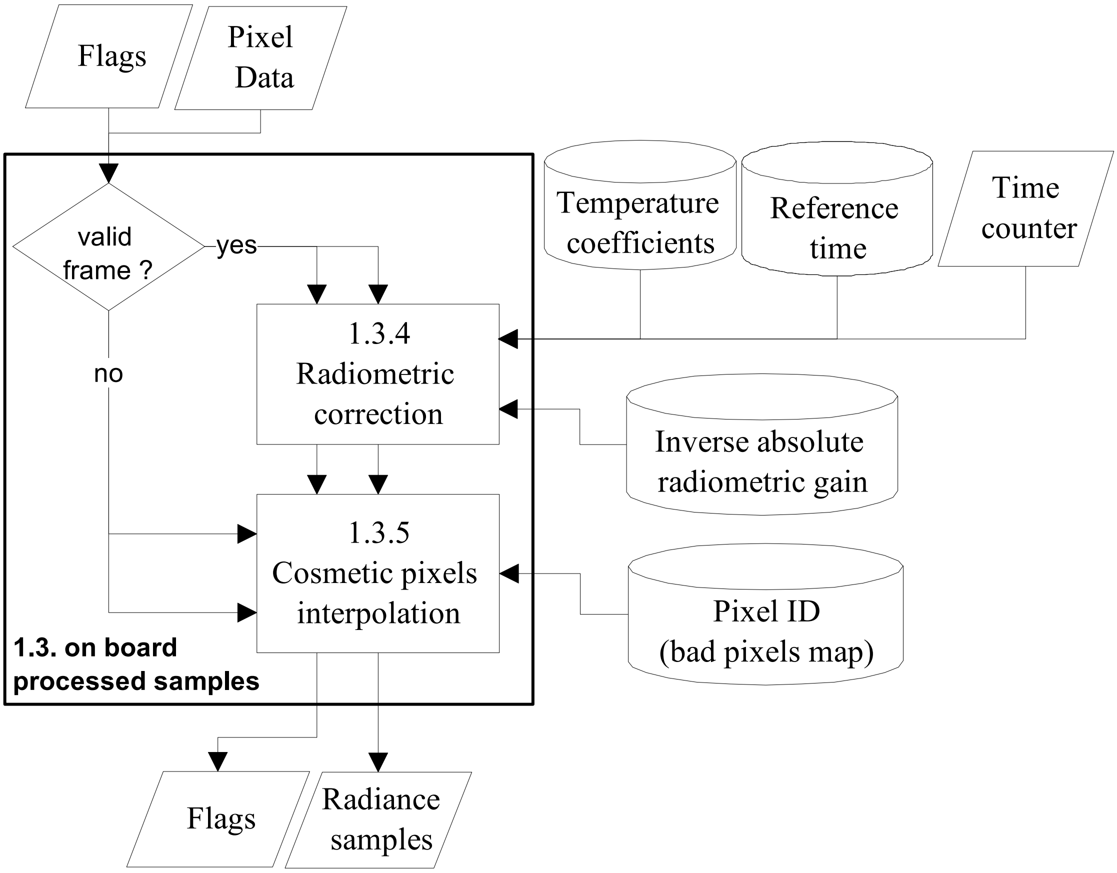


Figure 3.1.2-2 : Radiometric processing block diagram, RR and FR On-board processed samples

6.3.1.2.3. - On-board processed samples processing branch

On-board processing provides the following (see RD3):

$$X_{b,k,m,f} = N^{-1}_{b,k,m} \cdot \left(X''_{b,k,m,f} - C^0_{b,k,m} - S_{b,k,m,f} \right)$$

Absolute radiance is derived directly for all valid pixels of all bands following:

$$R_{b,k,m,f} = ALB^{-1}_{b,m} \cdot X_{b,k,m,f} \cdot \left[g_0 + g_1 \cdot \left(t_f - t_{ref} \right) + g_1 \cdot \left(t_f - t_{ref} \right)^2 \right]$$

where $ALB_{b,m} = \frac{1}{K} . \sum_{k=1}^{k=K} AL_{b,k,m}$

(it should be noted that ALB is the same for RR and FR processing).

The Cosmetic pixels interpolation is exactly the same than in the on-ground processed samples processing branch (see § 6.3.1.2.1.5.), assuming that the "dead pixels" map used is selected according to the product resolution.

6.3.2. - List of Variables

Variable	Descriptive Name	T	U	Range - References
K^{RR}	number of columns in a RR module	s	dl	185
K^{FR}	number of columns in a FR module	s	dl	740
B	number of bands	s	dl	15
Mt	number of MERIS modules	s	dl	5
$T\_JD_{ref}[d]$	Reference time for temperature models	s	jd	relative to CNT; d:0..365
RR_NONLIN_F	Switch enabling RR data non-linearity correction	s	dl	
FR_NONLIN_F	Switch enabling FR data non-linearity correction	s	dl	
$NonLinLUT_{b,m}[x]$	Inverse non-linearity LUT at micro-band level	s	dl	x in [0,4095], b:1..B+1, m:1..Mt
Aij_b	Weights for on-board Spatial and Temporal Relaxation (per band)	s	dl	b:1..B+1
MB_b	number of micro-bands in each band	s	dl	b:1..B+1
$C^{0FR}{}_{b,k,m}$	FR Dark signal characterisation data	s	nc	b:1..B+1;k:1..K^{FR};m:1..Mt
$AL^{0FR-1}{}_{b,k,m}$	FR Inverse Absolute gain coefficients	s	LU/nc	b:1..B+1;k:1..K^{FR};m:1..Mt
$C^{0RR}{}_{b,k,m}$	RR Dark signal characterisation data	s	nc	b:1..B+1;k:1..K^{RR};m:1..Mt
$AL^{0RR-1}{}_{b,k,m}$	RR Inverse Absolute gain coefficients	s	LU/nc	b:1..B+1;k:1..K^{RR};m:1..Mt
g_{c0}	0-order coeff. of dark temp. correction	s	dl	
g_{c1}	1st order coeff. of dark temperature correction	s	jd^{-1}	
g_{c2}	2nd order coeff. of dark temperature correction	s	jd^{-2}	
g_0	0-order coeff. of gain temp. correction	s	dl	
g_1	1st order coeff. of gain temperature correction	s	jd^{-1}	
g_2	2nd order coeff. of gain temperature correction	s	jd^{-2}	
$Ksm^{RR}{}_b$	Smear weighting factor for RR	s	dl	b:1..B
$Ksm^{FR}{}_{b,j}$	Smear weighting factor for FR	s	dl	b:1..B; j:1,2
Sat_rad_b	Saturation radiance values	s	LU	b=1,...,B
Def_rad_b	Default radiance value for saturated samples	s	LU	b=1,...,B
$Def\_rad\_O_b$	Default radiances for samples above range limits	s	LU	b=1,...,B
$dead\_pix^{RR}[b,k,m]$	dead pixels map for RR	s		
$dead\_pix^{FR}[b,k,m]$	dead pixels map for FR	s		
$ALB^{-1}{}_{b,m}$	Inverse mean absolute gain	s	LU/nc	b:1..B; m:1..Mt
$A\_JD^{FR}{}_{ref}$	Reference time for FR Instrument response degradation model	s	jd	
$\beta^{FR}{}_{b,k,m}$	Degradation Model amplitude for FR	s	dl	b:1..B+1;k:1..K^{FR};m:1..Mt
$\gamma^{FR}{}_{b,k,m}$	Degradation model time shift for FR	s	dl	b:1..B+1;k:1..K^{FR};m:1..Mt
$\delta^{FR}{}_{b,k,m}$	Degradation model time scale for FR	s	jd^{-1}	b:1..B+1;k:1..K^{FR};m:1..Mt
$A\_JD^{RR}{}_{ref}$	Reference time for RR Instrument response degradation model	s	jd	
$\beta^{RR}{}_{b,k,m}$	Degradation Model amplitude for RR	s	dl	b:1..B+1;k:1..K^{RR};m:1..Mt
$\gamma^{RR}{}_{b,k,m}$	Degradation model time shift for RR	s	dl	b:1..B+1;k:1..K^{RR};m:1..Mt
$\delta^{RR}{}_{b,k,m}$	Degradation model time scale for RR	s	jd^{-1}	b:1..B+1;k:1..K^{RR};m:1..Mt

Variable	Descriptive Name	T	U	Range - References
M	number of modules to process	i	dl	from 1.5.1
first_module	index of the first module to process	i	dl	from 1.5.1
CNT_JD	JD time at ascending node	i	jd	from 1.5.1
valid_frame_f[f]	valid frame flag for frame f	i/o	dl	from 1.1
do_cosmetic_f[f]	flag enabling cosmetic filling of empty frame f (small packet gap case)	i	dl	from 1.1
X^{RR}[b,k,m,f]	Pixel data for RR frame f	i	nc	from 1.1; b:1..B,s;k:1..K^{RR};m:1..M
X^{FR}[b,k,m,f]	Pixel data for FR frame f	i	nc	from 1.1; b:1..B,s;k:1..K^{FR};m:1..M
saturated_f^{RR}[b,k,m,f]	saturated sample flag for RR	i/o	dl	from 1.2
saturated_f^{FR}[b,k,m,f]	saturated sample flag for FR	i/o	dl	from 1.2
T_JD[f]	MJD2000 Time for frame f	i	jd	from 1.1
m'	index of module in characterisation data bases including offset due to product limits	c	dl	
nl_fact	global micro-band to band amplification factor	c	dl	
$InvNonLin^{FR}_{b,m}$[x]	Inverse non-linearity LUT at band level for FR samples	c	dl	x in [0,n*4095] with n number of µband in the band, b:1..B, s; m:1..M
$InvNonLin^{RR}_{b,m}$[x]	Inverse non-linearity LUT at band level for RR samples	c	dl	x in [0,n*4095] with n # of µband in the band *16*Aij_b, b:1..B, s; m:1..M
$C_{b,k,m,f}$	Dark signal correction coefficients	c	nc	b:1..B, s; k:1..$K^{RR/FR}$; m:1..M
$S_{b,k,m,f}$	Smear correction coefficients	c	nc	b:1..B, s; k:1..$K^{RR/FR}$; m:1..M
$X'_{b,k,m,f}$	pixel data after non linearity correction	c	nc	b:1..B, s; k:1..$K^{RR/FR}$; m:1..M
dt	difference between current time and temperature correction reference time	c	jd	
w1, w2	weights for cosmetic linear interpolation	c	LU/col.	
valid_frame_f[f]	valid frame flag for frame f	i/o	dl	to 1.4
R^{RR}[b,k,m,f]	RR Radiance	o	LU	to 1.4
saturated_f^{RR}[b,k,m,f]	saturated sample flag for RR	i/o	dl	to 1.4
cosmetic_f^{RR}[b,k,m,f]	cosmetic sample flag for RR	o	dl	to 1.5.5
R^{FR}[b,k,m,f]	FR Radiance	o	LU	to 1.4
saturated_f^{FR}[b,k,m,f]	saturated sample flag for FR	i/o	dl	to 1.4
cosmetic_f^{FR}[b,k,m,f]	cosmetic sample flag for FR	o	dl	to 1.5.5
out_r_PCD[b,m,f]	counter of out-of-range image samples	o	dl	to 1.8

6.3.3. - Equations

6.3.3.1. - RR Raw Samples Processing

step 1.3.0 initialisations

step 1.3.0.1: non-linearity tables building (if applicable)

```
if((RR_NONLIN_F)
   for each band b ∈{1..B,s}
      for each module m ∈{1..M}
         m' = m + first_module - 1
```

compute global micro-band to band gain factor (1.3.0.1-1)

```
         nl_fact = MB_b * 16 * Aij_b
```

at zero level, expanded table fits micro-band one (1.3.0.1-2)

```
         InvNonLin^RR_b,m[0] = NonLinLUT_b,m[0]
         for each level x in [1,4095]
```

extract value for next node (1.3.0.1-3)

```
            InvNonLin^RR_b,m[nl_fact*x] = NonLinLUT_b,m[x] * nl_fact
```

interpolate in between (1.3.0.1-4)

```
            for each intermediate level y in [1,nl_fact-1]
```

$$p = \frac{nl\_fact - y}{nl\_fact}$$

```
               InvNonLin^RR_b,m[nl_fact*(x-1)+y] = p*NonLinLUT_b,m[x-1] + (1-p)* NonLinLUT_b,m[x]
            end for
         end for
      end for
   end for
end if
```

step 1.3.0.2: correction of AL^{-1} Coefficients for Instrument Degradation

for each band b ∈{1..B,s}, module m ∈{1..M} and each pixel k ∈{1..K^{RR}}

$$AL^{RR-1}_{b,k,m} = \frac{AL^{0RR-1}_{b,k,m}}{1-\beta^{RR}_{b,k,m}\cdot\left(1-\gamma^{RR}_{b,k,m}\cdot e^{\left(-\delta^{RR}_{b,k,m}\cdot\left(CNT\_JD - A\_JD^{RR}_{ref}\right)\right)}\right)} \quad (1.3.0.2\text{-}1)$$

end for

```
for each frame f
   if (valid_frame_f_f = True)
      for each module m ∈{1..M}
         m' = m + first_module - 1
```

step 1.3.1 non-linearity correction :

```
         for each band b ∈{1..B,s}
            for each pixel k ∈{1..K^RR}
               if (RR_NONLIN_F AND NOT saturated_f^RR_b,k,m,f)
```

if applicable, proceed to non-linearity correction (1.3.1-1)

$$X'_{b,k,m,f} = InvNonLin_{b,m'}\left[X^{RR}_{b,k,m,f}\right]$$

```
else
```
else,copy input data (1.3.1-2)

$$X'_{b,k,m,f} = X^{RR}_{b,k,m,f}$$

```
end if
end for
```

step 1.3.2 dark signal correction coefficient :

```
for each pixel k ∈{1..K^RR}
```
compute dark signal corrected for temperature variation (1.3.2-1)

$$dt = T\_JD^{RR}_{f} - T\_JD_{ref}[mod(T\_JD^{RR}_{1}, 365.25)] - CNT\_JD$$

$$C_{b,k,m,f} = C^{0RR}_{b,k,m'} \cdot \left[g_{c0} + g_{c1} \cdot dt + g_{c2} \cdot dt^2\right]$$

```
    end for
end for
```

step 1.3.3 smear signal correction coefficient :

```
for each pixel k ∈{1..K^RR}
  if(saturated_ f^RR s,k,m,f) then
    for each band b ∈{1..B}
```
if smear sample saturated, smear signal set to default null value (1.3.3-1)

```
      Sb,k,m,f=0
```
if smear sample saturated, flag all bands of same pixel as saturated (1.3.3-2)

```
      saturated_f^RR b,k,m,f=TRUE
    end for
  else
    for each band b ∈{1..B}
```
if smear sample not saturated, compute smear signal (1.3.3-3)

$$S_{b,k,m,f} = Ksm^{RR}_{b} . (X'_{s,k,m,f} - C_{s,k,m,f})$$

```
    end for
  end if
```

step 1.3.4 radiometric correction :

```
for each band b ∈{1..B}
  if (saturated_f^RR b,k,m,f) then
```
if sample saturated, set to default value (1.3.4-1)

$$R^{RR}_{b,k,m,f} = Def\_rad_{b}$$

```
  else
```
else, proceed to radiometric corrections (1.3.4-2)

$$dt = T\_JD^{RR}_{f} - T\_JD_{ref}[mod(T\_JD^{RR}_{1}, 365.25)] - CNT\_JD$$

$$R^{RR}_{b,k,m,f} = \left(AL^{RR}_{b,k,m'}\right)^{-1} . \left\{\left(X'_{b,k,m,f} - S_{b,k,m,f}\right) . \left[g_0 + g_1 \cdot dt + g_2 \cdot dt^2\right] - C_{b,k,m,f}\right\}$$

```
    if (R^RR b,k,m,f <0 OR R^RR b,k,m,f > Sat_Rad_b) then
```
if result out of range, increment corresponding PCD (1.3.4-3)

```
      out_r_PCD[b,m,f] = out_r_PCD[b,m,f]+1
```
... and clip output radiance (1.3.4-4)

```
            if (R^RR b,k,m,f <0)
```
$$R^{RR}_{b,k,m,f} = 0$$
```
            else
```
$$R^{RR}_{b,k,m,f} = Def\_rad\_O_b$$
```
            end if
          end if
        end if
      end for
    end for
  end for
 end if
end for
```

6.3.3.2. - RR On-board processed Samples Processing

step 1.3.4 radiometric correction :

```
for each frame f
 if (valid_frame_ff = True)
   for each module m ∈{1..M}
     m' = m + first_module - 1
     for each pixel k ∈{1..K^RR}
       for each band b ∈{1..B}
         if (saturated_f^RR b,k,m,f) then
```
if sample saturated, set to default value (1.3.4-5)

$$R^{RR}_{b,k,m,f} = Def\_rad_b$$
```
         else
```
else, proceed to radiometric corrections (1.3.4-6)

$$dt = T\_JD^{RR}_f - T\_JD_{ref}[mod(T\_JD^{RR}_1, 365.25)] - CNT\_JD$$

$$R^{RR}_{b,k,m,f} = ALB^{-1}_{b,m'} . X^{RR}_{b,k,m,f} \cdot [g_0 + g_1 \cdot dt + g_2 \cdot dt^2]$$
```
          if (R^RR b,k,m,f <0 OR R^RR b,k,m,f > Sat_Rad_b) then
```
if result out of range, increment corresponding PCD (1.3.4-7)
```
            out_r_PCD[b,m,f] = out_r_PCD[b,m,f]+1
```
... and clip output radiance (1.3.4-8)
```
            if (R^RR b,k,m,f <0)
```
$$R^{RR}_{b,k,m,f} = 0$$
```
            else
```
$$R^{RR}_{b,k,m,f} = Def\_rad\_O_b$$
```
            end if
          end if
        end if
      end for
    end for
  end for
 end if
end for
```

6.3.3.3. - FR Raw Samples Processing

step 1.3.0 initialisation

step 1.3.0.1: non-linearity tables building (if applicable)

```
if((FR_NONLIN_F)
  for each band b ∈{1..B,s}
    for each module m ∈{1..M}
      m' = m + first_module - 1
```

compute global micro-band to band gain factor (1.3.0.1-1)

```
      nl_fact = MB_b
```

at zero level, expanded table fits micro-band one (1.3.0.1-2)

```
      InvNonLin^FR_b,m[0] = NonLinLUT_b,m[0]
      for each level x in [1,4095]
```

extract value for next node (1.3.0.1-3)

```
        InvNonLin^FR_b,m[nl_fact*x] = NonLinLUT_b,m[x] * nl_fact
```

interpolate in between (1.3.0.1-4)

```
        for each intermediate level y in [1,nl_fact-1]
```

$$p = \frac{nl\_fact - y}{nl\_fact}$$

```
          InvNonLin^FR_b,m[nl_fact*(x-1)+y] = p*NonLinLUT_b,m[x-1] + (1-p)* NonLinLUT_b,m[x]
        end for
      end for
    end for
  end for
end if
```

step 1.3.0.2: correction of AL^{-1} Coefficients for Instrument Degradation

for each band b ∈{1..B,s}, module m ∈{1..M} and each pixel k ∈{1..K^{FR}}

$$AL^{FR-1}_{b,k,m} = \frac{AL^{0FR-1}_{b,k,m}}{1-\beta^{FR}_{b,k,m}\cdot\left(1-\gamma^{FR}_{b,k,m}\cdot e^{\left(-\delta^{FR}_{b,k,m}\cdot\left(CNT\_JD - A\_JD^{FR}_{ref}\right)\right)}\right)} \quad (1.3.0.2\text{-}1)$$

end for

```
For each frame f
  if (valid_frame_f_f = True)
    for each module m ∈{1..M}
      m' = m + first_module - 1
```

step 1.3.1 non-linearity correction :

```
      for each band b ∈{1..B,s}
        for each pixel k ∈{1..K^FR}
          if (FR_NONLIN_F AND NOT saturated_f^FR_b,k,m,f)
```

if applicable, proceed to non-linearity correction (1.3.1-3)

$$X'_{b,k,m,f} = InvNonLin_{b,m'}\left[X^{FR}_{b,k,m,f}\right]$$

```
          else
```

else,copy input data (1.3.1-4)

$$X'_{b,k,m,f} = X^{FR}_{b,k,m,f}$$

```
        end if
      end for
```

step 1.3.2 dark signal correction coefficient :

for each pixel k ∈{1..K^{FR}}

compute dark signal corrected for temperature variation (1.3.2-2)

$$dt = T\_JD_f^{FR} - T\_JD_{ref}[mod(T\_JD_1^{FR}, 365.25)] - CNT\_JD$$

$$C_{b,k,m,f} = C^{0FR}_{b,k,m'} \cdot \left[g_{c0} + g_{c1} \cdot dt + g_{c2} \cdot dt^2\right]$$

```
      end for
    end for
```

step 1.3.3 smear signal correction coefficient :

for each pixel k ∈{1..K^{FR}}

if (saturated_$f^{FR}_{s,k,m,f}$) then

for each band b ∈{1..B}

if smear sample saturated, smear signal set to default null value (1.3.3-4)

$S_{b,k,m,f}=0$

if smear sample saturated, flag all bands of same pixel as saturated (1.3.3-5)

saturated_$f^{FR}_{b,k,m,f}$=TRUE

```
        end for
      else
        for each band b ∈{1..B}
```

if smear sample not saturated, compute smear signal (1.3.3-6)

$$S_{b,k,m,f} = \left[Ksm^{FR}_{b,1}.(X'_{s,k,m,f+1} - C_{s,k,m,f+1}) + Ksm^{FR}_{b,2}.(X'_{s,k,m,f} - C_{s,k,m,f})\right]$$

```
        end for
      end if
```

step 1.3.4 radiometric correction :

for each band b ∈{1..B}

if (saturated_$f^{FR}_{b,k,m,f}$) then

if sample is saturated, set to default value (1.3.4-9)

$$R^{FR}_{b,k,m,f} = Def\_rad_b$$

else

else, proceed to radiometric corrections (1.3.4-10)

$$dt = T\_JD_f^{FR} - T\_JD_{ref}[mod(T\_JD_1^{FR}, 365.25)] - CNT\_JD$$

$$R^{FR}_{b,k,m,f} = \left(AL^{FR}_{b,k,m'}\right)^{-1} \cdot \left\{\left(X'_{b,k,m,f} - S_{b,k,m,f}\right) . \left[g_0 + g_1 \cdot dt + g_2 \cdot dt^2\right] - C_{b,k,m,f}\right\}$$

if ($R^{FR}_{b,k,m,f}$<0 OR $R^{FR}_{b,k,m,f}$>Sat_rad_b) then

if result out of range, increment corresponding PCD (1.3.4-11)

out_r_PCD[b,m,f] = out_r_PCD[b,m,f]+1

... and clip output radiance (1.3.4-12)

if ($R^{FR}_{b,k,m,f}$ <0)

$$R^{FR}_{b,k,m,f} = 0$$

else

$$R^{FR}_{b,k,m,f} = Def\_rad\_O_b$$

```
            end if
          end if
        end if
      end for
    end for
  end for
 end if
end for
```

6.3.3.4. - FR On-board processed Samples Processing

step 1.3.4 radiometric correction :

```
For each frame f
  if (valid_frame_ff = True)
    for each module m ∈{1..M}
      m' = m + first_module - 1
      for each pixel k ∈{1..K^RR}
        for each band b ∈{1..B}
          if (saturated_f^FR_b,k,m,f) then
```

if sample is saturated, set to default value *(1.3.4-13)*

$$R^{FR}_{b,k,m,f} = Def\_rad_b$$

```
          else
```

else, proceed to radiometric corrections *(1.3.4-14)*

$$dt = T\_JD^{FR}_f - T\_JD_{ref}[mod(T\_JD^{FR}_1, 365.25)] - CNT\_JD$$

$$R^{FR}_{b,k,m,f} = ALB^{-1}_{b,m'} \cdot X^{FR}_{b,k,m,f} \cdot \left[g_0 + g_1 \cdot dt + g_1 \cdot dt^2\right]$$

```
            if (R^FR_b,k,m,f<0 OR R^FR_b,k,m,f>Sat_radb) then
```

if result out of range, increment corresponding PCD *(1.3.4-15)*

out_r_PCD[b,m,f] = out_r_PCD[b,m,f]+1

... and clip output radiance *(1.3.4-16)*

```
              if (R^FR_b,k,m,f <0)
```

$$R^{FR}_{b,k,m,f} = 0$$

```
              else
```

$$R^{FR}_{b,k,m,f} = Def\_rad\_O_b$$

```
              end if
            end if
          end if
        end for
      end for
    end for
  end if
end for
```

6.3.3.5. - Cosmetic pixels processing

Note : subscripts RR / FR have been intentionnaly omitted as processing is identical for RR raw or on board processed samples and identical for FR raw or on board processed samples.

step 1.3.5 cosmetic pixels interpolation

```
for each frame f
  if (valid_frame_ff = True) then
```

proceed to across-track interpolation if needed

```
    for each module m ∈{1..M}
      for each band b ∈ {1..B}
```

reset column index to module start (1.3.5-1)

```
        k=1
        while(k ≤ K)
          if (dead_pix[b,k,m] = True) then
```

hole found : reset upper limit to lower one value (1.3.5-2)

```
            k2 = k
            while (dead_pix[b,k2,m] = True)
```

hole continues : increment upper limit value (1.3.5-3)

```
              k2=k2+1
            end while
            if (k>1 AND k2<K) then
```

compute coefficients of linear interpolation between $R_{b,k-1,m,f}$ and $R_{b,k2+1,m,f}$

case two samples available : compute coefficients for linear interpolation (1.3.5-4)

$$w1 = \frac{R_{b,k-1,m,f}}{k2-k+2}$$

$$w2 = \frac{R_{b,k2+1,m,f}}{k2-k+2}$$

proceed to linear interpolation for each k' ∈{k..k2}

```
              for k' = k, k2
```

case two samples available : proceed to linear interpolation (1.3.5-5)

$$R_{b,k',m,f} = w2\cdot(k'-k+1) + w1\cdot(k2+1-k')$$

case two samples available : flag sample cosmetic (1.3.5-6)

```
                cosmetic_f[b,k,m,f] = True
              end for
            elseif(k2==K)
```

last pixel of module within hole : fill hole with last valid one

```
              for k' = k, k2
```

case no sample available at the end: fill hole with last valid sample (1.3.5-7)

$$R_{b,k',m,f} = R_{b,k-1,m,f}$$

case no sample available at the end: flag sample cosmetic (1.3.5-8)

```
                cosmetic_f[b,k,m,f] = True
              end for
            elseif(k==1)
```

first pixel of module within hole : fill hole with next valid one

```
              for k' = k, k2
```

case no sample available at the beginning: fill hole with next valid sample *(1.3.5-9)*

$$R_{b,k',m,f} = R_{b,k2+1,m,f}$$

case no sample available at the beginning: flag sample cosmetic *(1.3.5-10)*

```
                cosmetic_f[b,k,m,f] = True
              end for
            endif
```

set column index to index of first valid sample after hole *(1.3.5-11)*

```
            k=k2+1
          else
```

current sample not in "dead" pixels list : increment column index *(1.3.5-12)*

```
            k++
          end if
        end while
      end for     end of loop over bands
    end for     end of loop over modules

  elseif (do_cosmetic_ff AND f>1) then
```

small gap : whole frame replaced by previous frame values
check if previous frame is valid

```
    if (valid_frame_ff-1) then
```

small gap and previous frame valid : set valid_frame_f to TRUE *(1.3.5-13)*

```
      valid_frame_ff = True
    end if
```

previous frame valid

```
    for each module m ∈{1..M}
      for each band b ∈ {1..B}
        for each column k ∈ {1..K}
```

small gap : fill sample with value of corresponding one in last frame *(1.3.5-14)*

$$R_{b,k,m,f} = R_{b,k,m,f-1}$$

small gap and previous frame valid : set cosmetic_f to TRUE *(1.3.5-15)*

```
          if (valid_frame_ff-1) then
            cosmetic_f[b,k,m,f] = True
          end if
        end for
      end for
    end for
  else
```

wide gap : fill frame samples with zeros *(1.3.5-16)*

```
      for each module, and and column
```

$$R_{b,k,m,f} = 0$$

```
      end for
  end if
end for
```

small gap AND previous frame does exist

6.3.4. - Accuracy Requirements

All comparisons with reference test values must be exact to the fifth significant digit.

6.3.5. - Product Confidence Data Summary

Any out of range radiance $R_{b,k,m,f}$ (< 0 or > Sat_rad_b) is taken into account by an "out of range" PCD counter per module, band and frame. This will be used to set flags in the Product Formatting step (see section 11).

If a smear sample is "saturated" the flag "saturated" is set for all other bands for the same pixels and the pixel is processed as such.

7. - MERIS Stray Light Correction Algorithm

7.1. - Introduction

The signal of a given sample is polluted by stray light coming within the instrument from other samples by means of either specular reflections (ghost images) or scatter. Stray light may be an important contributor to the measured signal, particularly in the infrared for ocean pixels close to clouds or land covered by vegetation. This chapter describes an algorithm using characterisation of the stray light contamination to estimate the degradation and correct it.

7.2. - Algorithm Overview

Stray light contribution to signal is evaluated and corrected. It can be described as the weighted sum of neighbouring samples. The correction algorithm uses knowledge of the system response to evaluate the signal degradation. Once it is known, it can be subtracted from the measured signal.

7.3. - Algorithm Description

7.3.1. - Theoretical Description

7.3.1.1. - Physics of The Problem

Stray light contribution to signal is a two-dimensional process with a spectral component, hereafter referred to as SP, and a spatial component referred to as AC (for across-track).

Instrument characterisation has shown that a few per cent of the energy lies in the stray light. A direct consequence is that the fundamental structure of the signal is preserved (even if it is masked) either on spatial and spectral point of view. This allows to use a very robust and fast correction method based on the following hypothesis : a second degradation of the signal by the system would have the same impact on the (already) degraded signal as the first one had on the original signal. As the system response is known, it is possible to degrade a second time the measured signal and, by means of a simple subtraction, to estimate the degradation itself. It is then straightforward to subtract it and get a good estimate of the original radiances. This method is based on the same approximation principle as the well known formula :

$$(1+\varepsilon)^2 \approx 1+2\varepsilon \quad \text{if} \quad \varepsilon << 1$$

It can be expressed mathematically as follows :

The degraded version of a signal x can be written as the sum of the original signal and the degradation itself :

$$\hat{x} = x + \tilde{x}$$

The second degradation on the result gives :

$$\hat{\hat{x}} = (x+\tilde{x}) + \widetilde{\left(x+\tilde{x}\right)} = x + 2\tilde{x} + \tilde{\tilde{x}}$$

If the degradation operator can be considered as a perturbation (in the physics sense) - that is verifying : energy($\tilde{x}$) << energy(x) - it follows :

$$\hat{\hat{x}} \cong x + 2\tilde{x} \qquad \text{as } \tilde{\tilde{x}} \text{ can be neglected.}$$

And x may be retrieved by : $x = 2 \cdot \hat{x} - \hat{\hat{x}}$

This method will be refered to as the "second degradation method" hereafter.

The next point is to define a mathematical representation of the system degradation, manageable by numeric tools. The degradation step has been characterised using a ray tracing model : the ASAP software.

The ACxSP degradation has been characterised as an additive process : for a monochromatic point source input to the instrument, part of the beam energy lost during its path through the optical components is re-distributed over the whole CCD sensor surface. For a given input beam, characterisation data is output as weighting factors expressing the amount of energy, relative to the direct beam, received by each CCD cell, building a matrix called the Diffuse Light Distribution Function (DLDF). These DLDF have been chracterised for a set of 25 (AC,SP) locations of the input beam, regularly sampling the AC and SP domains and defining 25 regions on the sensor (5 AC by 5 SP) within which the DLDF are considered constant.
It should be noted here that the light spread inside the spectrometer has gone through all the major optics components and hence can be considered as scaled by the overall transmission factor of the optics. Moreover, as it is spread over the whole CCD and thus across the spectral dimension of the sensor it must be scaled by the mean spectral response of the detector prior to any addition. Thus the DLDF apply on an equivalent photo-electron flux field instead of the radiance field.

Study of the 25 DLDF has shown that for a given spectral region, the variation of the DLDF across the 5 AC regions lies mainly in the relative importance of the diffuse part with respect to the direct beam. This allow the use of only one DLDF per spectral region providing that the input radiance has been properly scaled according to its AC region prior to the stray light computation.

For correction purposes, only the MERIS bands are available instead of the whole CCD surface and some assumptions have to be made on the radiance distribution between bands. Considering the complex structure of a top-of-atmosphere spectrum, its variability over natural targets and the relatively low level of the diffuse light a simple linear model has been found satisfactory enough for the stray light estimation. This assumption allows to use a spectrally resampled version of the photo-electron field as input to the stray light evaluation process and hence to allow faster computations. Spectral resampling is done on the spectral region grid basis, i.e. yields only 5 electron flux values per ground pixel. This imply the use of resampled versions of the DLDF, the Spectral Region Distribution Functions (SRDF), expressing the contribution of each spectral zone to the stray light of each band.

7.3.1.2. - Mathematical Description of Algorithm

The MERIS retrieved radiances will be corrected for the AcxSP stray light. The functional breakdown and logic of the whole correction process is shown on figure 7.3.1.2.-1 below.

It is assumed that the PSF and DLDF vary slowly with the sample AC and/or SP location and are accurately known, at the instrument discretisation. It it is assumed that each DLDF element is several orders of magnitude below the direct illumination beam level. This assumption ensures that correction by the "second degradation" method is appropriate.

It must be noted here that the design of the spectrometer stray light correction algorithm described below assume fixed values for many parameters wich may appear as free otherwise, as in AD1 for instance. Among those are :

- the number of bands in MERIS, assumed equal to 15;
- the number of spectral regions, assumed equal to 5;
- the bands wavelength, assumed equal to those listed in table 3.2.1 of AD1;
- some of the instrument gain characteristics ensuring that no bands but bands 9, 12 and 13 (see AD1, table 3.2.1) will saturate over any cloud (conversely, if the gains are higher, the domain of applicability of the algorithm is restricted).

If any of those assumptions is not verified, part or all of the algorithm may have to be revised to ensure expected performances.

7.3.1.2.1. - Algorithm Functional Breakdown

Corrections will take place after the Radiometric Processing and will act on radiances. However, spectral weighting factors including optics transmission factors and detector quantum efficiency will be used for radiance in the correction step, as the degradation takes place inside the spectrometer, just before the detection process.

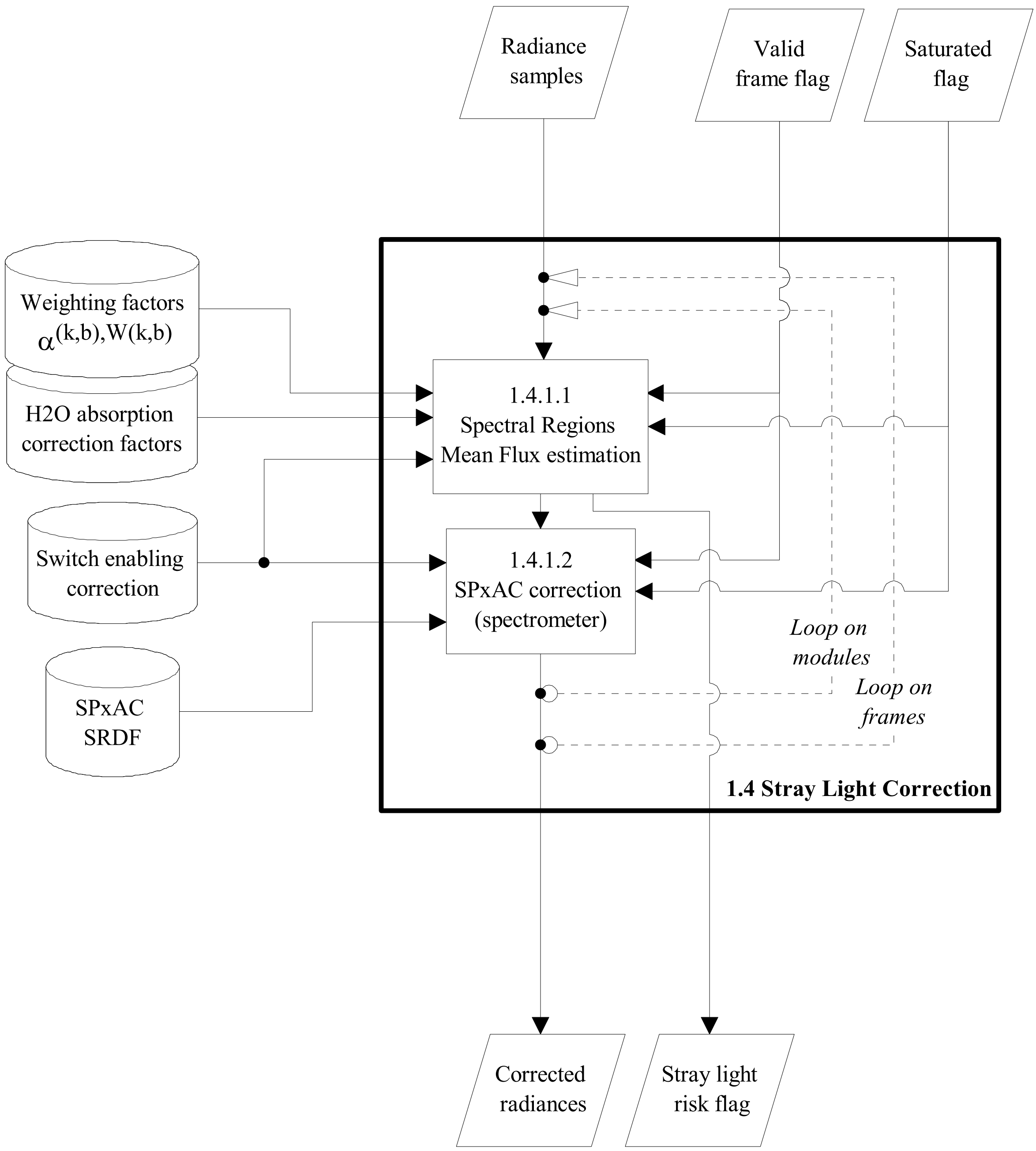


Figure 7.3.1.2-1 : Stray light correction algorithm block diagram

7.3.1.2.2. - Spectral by Across-Track "Spectrometer Term" Deconvolution (step 1.4.1)

7.3.1.2.2.1 - Principle of the Correction (step 1.4.1.2)

The output of the radiometric correction is the degraded radiance array $R_{b,k,m,f}$. It may be described as a sum of two terms (index f will be ommitted as we are restricted to one frame here) :

$$R_{b,k,m} = L'_{b,k,m} + G_{b,k,m}(L'_{*,*,m}), \quad (1)$$

where $G_{b,k,m}$, a function of $L'_{*,*,m}$, is the spectrometer stray light contribution to the signal (referred to as step 2 in 7.3.1.1.) and L' is the radiance entering the spectrometer (which is the sum of the target radiance and step 1 (ground imager) stray light contribution, see 7.3.1.2.3 below).

If one can derive values for $G_{b,k,m}$, the correction becomes straightforward :

$$L'_{b,k,m} = R_{b,k,m} - G_{b,k,m} \quad (2)$$

$G_{b,k,m}$ can be expressed as :

$$G_{b,k,m} = \frac{1}{\alpha_{b,k,m}} \cdot \sum_{\lambda} \sum_{k'} \alpha_{\lambda,k',m} \cdot L'_{\lambda,k',m} \cdot \mathrm{DLDF}(\lambda, k', b, k)$$

and, with the second degradation method assumption :

$$G_{b,k,m} \approx \frac{1}{\alpha_{b,k,m}} \cdot \sum_{\lambda} \sum_{k'} \alpha_{\lambda,k',m} \cdot R_{\lambda,k',m} \cdot \mathrm{DLDF}(\lambda, k', b, k) \quad (3)$$

where α is the product of the optics transmission $\tau_{\lambda,k,m}$ by the sensor's spectral response $QE_{\lambda,k,m}$.

Considering the fact that the DLDF shape is fairly constant along the across-track dimension of the sensor (that is for a fixed wavelength), the DLDF, if acting on properly weighted radiances, can be considered as shift-invariant with respect to k and equation (7.-3) becomes :

$$G_{b,k,m} = \frac{1}{\alpha_{b,k,m}} \cdot \sum_{\lambda} \sum_{k'} \alpha_{\lambda,k',m} \cdot W_{\lambda,k'} \cdot R_{\lambda,k',m} \cdot \mathrm{DLDF}(\lambda, b, k - k') \quad (4)$$

where $W_{\lambda,k}$ is the radiance across-track weighting function, representing the variation of the relative weight of the diffuse light to the direct beam.

An estimate of $G_{b,k,m}$ could be achieved using equation (4), providing that models are available for the radiance L' and for the calibration factor α between the available samples, i.e. the MERIS bands. This solution implies heavy computations and a simplified model of stray light flux estimation is used without significant loss in radiometric performances.

The simplified model defines :

- 5 spectral regions of constant width and regularly spaced along the spectral dimension of the CCD : region sr is defined by the interval $[\lambda_{sr},\lambda_{sr+1}]$, where

$\lambda_{sr} = \lambda_0 + (sr-1).(\lambda_0-\lambda_1)/5$, λ_0 and λ_1 being the limits of the spectrum imaged on the CCD.

- the Spectral Region Distribution Function SRDF :

$$SRDF(sr, b, k-k') = \sum_{\lambda=\lambda_{sr}}^{\lambda_{sr+1}} DLDF(\lambda, b, k-k') \quad (5)$$

- the weighted equivalent photo-electron flux :

$$\Phi_{sr,k,m} = \frac{1}{\lambda_{sr+1}-\lambda_{sr}} \cdot \sum_{\lambda=\lambda_{sr}}^{\lambda_{sr+1}} \alpha_{\lambda,k,m} \cdot W_{\lambda,k} \cdot R_{\lambda,k,m} \quad (6)$$

Then the estimate of the stray light degradation can be written :

$$dG_{b,k,m}^{[\lambda_{sr},\lambda_{sr+1}]} = \frac{1}{\alpha_{b,k,m}} \cdot \sum_{sr}\sum_{k'} \Phi_{sr,k,m} \cdot SRDF(sr, b, k-k') \quad (7)$$

$$G_{b,k,m} = \sum_{sr} dG_{b,k,m}^{[\lambda_{sr},\lambda_{sr+1}]} \quad (8)$$

The operational correction algorithm takes the SRDF set as an input and implements equations (6) to (8).

7.3.1.2.2.1 - Spectral flux extimate (step 1.4.1.1)

As already noted earlier, the only available radiance samples are the MERIS bands and the equivalent photo-electron flux evaluation over the spectral regions must rely on them. Its computation still needs some assumptions on the flux behaviour between measured bands. Simulations have shown that a linear model for the flux variation is accurate enough for the straylight evaluation for the main contributors which are the clouds and, to a lesser.extent, vegetation.

However, those targets are likely to cause saturation of bands for which programmable gain has been tuned for dark targets, and then those samples must be discarded from the computations. The flux evaluation algorithm includes an interpolation scheme, linear in reflectance between bracketing valid samples, that gives a good estimate of the radiance of saturated samples providing that few samples of a given pixel are.saturated and under the assuption that the albedo of the main stray light contributors is spectrally flat around the potentially saturated bands.

In order to cover the whole CCD bandwith, wich extents beyond the extreme bands, extrapolation of the flux is necessary, especially in the infrared. Spectral region 5 for instance contains no band. The linear model is extended from the two extreme bands to some spectral limits where the flux is assumed to vanish, partly because of the solar irradiance decrease and partly because of the sensor response bandwith.

Using a kernel $P_{sr,b}$ to account for the linear model and considering that $W_{\lambda,k}$ has been characterised for each spectral region sr, flux estimation becomes:

$$\Phi_{sr,k,m} = W_{sr,k} \sum_{b} P_{sr,b} . \phi_{b,k,m} \quad (9)$$

where $\phi_{b,k,m}$ stands for $\alpha_{b,k,m} \cdot L'_{b,k,m}$.

This resampling scheme takes into account that the 900 nm band is dedicated to the measurement of H_2O absorption by means of comparison with the 890 nm band. The MERIS band at 760 nm, at the maximum O_2 absorption, is also used in the flux calculation for region 3. It seems that the linear model fits rather well with the line shape.

The logic of the processing is as follows :

- loop on frames
    - loop on modules
        - loop on bands
            - loop on regions
                - loop on columns
                    - compute absorption correction factor
                    - compute weighted mean flux using (9)
                    - compute contribution of region sr to stray light of band b (for all columns k) using (7)

                    (Note : this is a convolutive process and may be implemented via Fourier transform in which case it will be out of the column loop)
            - add region's contribution to total stray light of band b : $G_{b,k,m} = \sum_{sr} dG^{sr}_{b,k,m}$ (in case of convolution via Fourier transform, this should be done prior to inverse transform, i.e. accumulating transforms)
        - unweight total stray light of band b and subtract it from degraded radiance (after inverse transform if needed)

7.3.2. - List of Variables

Variable	Descriptive Name	T	U	Range - References
K^{RR}	Number of columns in a MERIS RR module	s	dl	185
K^{FR}	Number of columns in a MERIS FR module	s	dl	740
B	Number of MERIS bands	s	dl	15
Mt	Number of MERIS modules	s	dl	5
SR	Number of spectral regions for spectrometer stray light evaluation	s	dl	5
λ[b]	band central wavelengths	s	nm	
Bs	index of bands that can be used for radiance estimation of saturated samples	s	dl	
R_{ref}	Default radiance for pixels with all bands saturated	s	LU	
b_{ref}	Band index for default radiance R_{ref}	s	dl	
F_{0b}	Extra-terrestrial Sun irradiance at reference date	s	IU	
Stray_corr_AC_s	Switch to enable ACxSP stray light correction	s	dl	
$SAT\_STRAY\_THR^{RR}$	Threshold on saturated RR samples count to flag for stray light risk	s	dl	
$SAT\_STRAY\_THR^{FR}$	Threshold on saturated FR samples count to flag for stray light risk	s	dl	
$SRDF^{RR}_{m,sr,b}[k]$	RR Spectral Region Distribution Function for region sr contribution to stray light of band b	s	nc	m=1,..,Mt ; sr=1,..,SR ; b=1,...,B
$SRDF^{FR}_{m,sr,b}[k]$	FR Spectral Region Distribution Function for region sr contribution to stray light of band b	s	nc	m=1,..,Mt ; sr=1,..,SR ; b=1,...,B
$Nright^{RR}$, $Nleft^{RR}$	half-extent in forward and backward directions respectively of RR SRDF (total extent is Nleft+1+Nright)	s	dl	Nleft+1+Nright < 2 K see note 2 at end of § 7.3.3
$Nright^{FR}$, $Nleft^{FR}$	half-extent in forward and backward directions respectively of FR SRDF (total extent is Nleft+1+Nright)	s	dl	Nleft+1+Nright < 2 K see note 2 at end of § 7.3.3
α^{RR}[b,k,m]	product of optics transmission by CCD spectral response	s	dl	b=1,...,B ; k=1,...,K ; m=1,...,M
α^{FR}[b,k,m]	product of optics transmission by CCD spectral response	s	dl	b=1,...,B ; k=1,...,K ; m=1,...,M
W^{RR}[sr,k]	radiance across-track weighting factors for RR	s	dl	sr=1,..,SR ; k=1,..,K
W^{FR}[sr,k]	radiance across-track weighting factors for FR	s	dl	sr=1,..,SR ; k=1,..,K
P[b,sr]	interpolation coeff for spectral region flux estimation	s	dl	sr=1,...,SR, b=1..B
$Def\_rad\_O_b$	Default radiances for samples above range limits	s	LU	b=1,...,B
M	Number of MERIS used modules	i	dl	from 1.5.1
first_module	index of first extracted MERIS module	i	dl	from 1.5.1
NF	number of frames to process	i	dl	from 1.5.1
R^{RR}[b,k,m,f]	RR radiance samples	i	LU	from 1.3
R^{FR}[b,k,m,f]	FR radiance samples	i	LU	from 1.3
$saturated\_f^{RR}$[b,k,m,f]	RR saturated sample flag	i	dl	from 1.3
$saturated\_f^{FR}$[b,k,m,f]	FR saturated sample flag	i	dl	from 1.3
valid_frame_f[f]	Valid frame flag	i	dl	from 1.3

Table 7.3.2-1 : Stray light correction parameters

Variable	Descriptive Name	T	U	Range - References
m'	index of module in characterisation data bases including offset due to product limits	c	dl	
sat_count	number of saturated samples in the frame	c	dl	
ϕ[b,k,m]	photo-electron flux per band	c	$e^- m^{-2} sr^{-1} \mu m^{-1}$	
p	interpolation weight for saturated samples	c	dl	
b1, b2	next lower and next upper band indices for interpolation of saturated samples	c	dl	
Φ[sr,k,m]	Mean weighted photo-electron flux	c	$e^- m^{-2} sr^{-1} \mu m^{-1}$	k = 1..K
dG[k]	spectral region contribution to stray light	c	LU	k = 1..K
G[b,k]	spectrometer stray light term	c	LU	
L^{RR}[b,k,m,f]	RR straylight corrected radiance	o	LU	to 1.5.5
L^{FR}[b,k,m,f]	FR straylight corrected radiance	o	LU	to 1.5.5
stray_f^{RR}[k,m,f]	RR straylight risk flag	o	dl	to 1.5.5
stray_f^{FR}[k,m,f]	FR straylight risk flag	o	dl	to 1.5.5

Table 7.3.2-1 (cont) : Stray light correction parameters

7.3.3. - Equations

NOTES :

- superscript RR or FR will be omitted in equations below as processing is exactly the same;
- symbol ⊗ stands for the convolution operator;

loop on frames :

```
for each frame f in 1,NF
  if (Stray_corr_AC_s) then
    if (valid_frame_ff) then
```

loop on modules :

```
      for m in 1 to M
```

compute data bases index corresponding to current module (1.4.1.1-0)

```
        m' = m + first_module - 1
```

Step 1.4.1.1 Spectral Regions Mean Flux Estimation :

check incoming samples

$$sat\_count = \sum_{k=1}^{k=K} \sum_{b=1}^{b=B} saturated\_f_{b,k,m,f} \quad (1.4.1.1\text{-}1)$$

(note : 1.4.1.1-1 above assumes that the Boolean quantity TRUE is equivalent to the integer 1)

```
        if (sat_count ≥ SAT_STRAY_THR) then
          for k in 1 to K
            stray_fk,m,f = TRUE                          (1.4.1.1-2)
          end for
        else
          for k in 1 to K
            stray_fk,m,f = FALSE                         (1.4.1.1-3)
          end for
        end if
```

compute mean weighted flux over regions

```
for b in 1 to B; for k in 1 to K
  if(saturated_f_{b,k,m,f}) then
    find the greatest b1 such as
      b1∉Bs and b1<b and !saturated_f_{b1,k,m,f}                (1.4.1.1-4)
    find the smallest b2 such as
      b2∉Bs and b2>b and !saturated_f_{b2,k,m,f}                (1.4.1.1-5)
    if(b1 and b2 could be found) then
```

$$p = \frac{\lambda_{b2} - \lambda_b}{\lambda_{b2} - \lambda_{b1}}$$

$$\phi_{b,k,m} = F_{0b}\left(p \cdot \frac{R_{b1,k,m}}{F_{0b1}} + (1-p)\cdot\frac{R_{b2,k,m}}{F_{0b2}}\right)\cdot \alpha_{b,k,m'} \qquad (1.4.1.1\text{-}6a)$$

```
    elseif(only b1 could be found) then
```

$$\phi_{b,k,m} = R_{b1,k,m}\cdot\frac{F_{0b}}{F_{0b1}}\cdot\alpha_{b,k,m'} \qquad (1.4.1.1\text{-}6b)$$

```
    elseif(only b2 could be found) then
```

$$\phi_{b,k,m} = R_{b2,k,m}\cdot\frac{F_{0b}}{F_{0b2}}\cdot\alpha_{b,k,m'} \qquad (1.4.1.1\text{-}6c)$$

```
    elseif(none could be found) then
```

$$\phi_{b,k,m} = R_{ref}\cdot\frac{F_{0b}}{F_{0b_{ref}}}\cdot\alpha_{b,k,m'} \qquad (1.4.1.1\text{-}6d)$$

```
    endif
  else
```

$$\phi_{b,k,m} = R_{b,k,m}\cdot\alpha_{b,k,m'} \qquad (1.4.1.1\text{-}6e)$$

```
  endif
end for; end for
for sr in 1 to SR ; for k in 1 to K
```

$$\Phi_{sr,k,m} = W_{sr,k}\cdot\sum_{b=1}^{b=B} P_{b,sr}\cdot\phi_{b,k,m} \qquad (1.4.1.1\text{-}9)$$

```
end for ; end for
```

Step 1.4.1.2 ACxSP correction (spectrometer) :

initialise array $G_{b,k}$ (1.4.1.2-0)

```
for b in 1 to B ; for k in 1 to K
  G_{b,k} = 0
end for ; end for
```

loops on bands :

```
for b in 1 to B
```

loops on spectral regions :

```
  for sr in 1 to SR
```

convolute weighted flux with SRDF in the AC direction (k index) (see notes 1&2 below) :

$$dG_k = \Phi_{sr,k,m} \otimes SRDF_{m',sr,b,k} \qquad (1.4.1.2\text{-}1)$$

accumulate result in array G_b :

```
    for k in 1 to K
      G_{b,k} = G_{b,k} + dG_k                                  (1.4.1.2-2)
    end for                                         End of column loop
  end for                                           End of region loops
```

subtract stray light estimate for samples which are neither saturated nor out of range:

```
        for k in 1 to K
          if (saturated_f_{b,k,m,f} OR R_{b,k,m,f} = Def_rad_O_b) then
            L_{b,k,m,f} = R_{b,k,m,f}                              (1.4.1.2-3a)
          else
            L_{b,k,m,f}= R_{b,k,m,f} - G_{b,k}/α_{b,k,m'}          (1.4.1.2-3b)
          end if
        end for                                         End of column loop
      end for                                           End of band loop
    end for                                             End of module loop
```

invalid frame, no need to correct :

```
   else
     for m in 1 to M; for k in 1 to K; for b in 1 to B
       L_{b,k,m,f} = R_{b,k,m,f}                                   (1.4.1.2-4)
     end for; end for; end for
   end if                                               End of invalid frame branch
```

corrections are disabled:

```
  else
    for m in 1 to M; for k in 1 to K
      for b in 1 to B
        L_{b,k,m,f} = R_{b,k,m,f}                                  (1.4.1.2-5)
      end for
    end for; end for
  end if                                   End of disabled AC correction branch
end for                                                 End of product
```

Notes :

1. Convolutions have intentionally not been described: they may be implemented through Fast Fourier transform but this choice is considered as a matter of implementation as results are strictly identical. Obviously Fourier convolution will save computing time despite the fact that arrays to convolute must be extended by zero padding to the next power of 2. In fact, to convolute an array of N samples with a PSF of [Nleft+1+Nright] length, one must use arrays zero-padded to the power of 2 next to (N+max(Nleft,Nright)) as input to the Fast Fourier Transform.
2. Parameters nright and nleft have been included in list of variables because they have been identified as key parameters for convolutions (whatever the choosen implementation) ; however they may not appear explicitely in the above equations because convolutions are not described.

7.3.4. - Accuracy Requirements

Stray light corrected radiances shall be computed with a relative accuracy better than 10^{-5}.

7.3.5. - Product Confidence Data Summary

$stray\_f_{k,m,f}$	Straylight risk flag for each pixel. The flag is set for each column of a given frame and module when an excessive number of saturated samples is present in the input frame.

8 - MERIS Geo-location Processing Algorithms

8.1. - Introduction

This chapter describes the geo-location processing performed in the MERIS Level 1b processing.

8.2. - Overview

8.2.1 - Objectives

Geo-location processing has three purposes :

1. To define the product limits for data extraction from level 0 product and data storage in Level 1B product. In reduced resolution, this process is straightforward as across-track extraction limits are those of the Level 0 product and along track limits - i.e. time limits - are specified in the Work Order to comply with the product splitting needed by the processing. In Full resolution however extraction limits are computed on the basis of the the requested scene size and centre location, all parameters extracted from the Work Order.

2. To establish the elements, in the MERIS Level 1B product, which provide the capability to identify for any *product pixel* :

- its location on the Earth geoid : longitude, latitude;
- the observation and illumination geometry when the pixel was measured : Sun zenith and azimuth angle, observer zenith and azimuth angle;
- relevant information related to the pixel location and observation and illumination geometry : altitude (bathymetry for ocean pixels), surface roughness, location correction term due to altitude, Sun glint risk flag.

3. To perform, based on their relative locations, the resampling of the *MERIS pixels* to the *product pixels*.

8.2.2 - Definitions and conventions

geoid the WGS84 Earth geoid model, referred to as "reference ellipsoid" in RD11.

frame a set of product pixels corresponding to a given satellite position

location coordinates (geodetic latitude, longitude) of a point on the geoid, expressed in the Earth fixed coordinates system

satellite fixed coordinate system
Z_S points in the direction of the Earth outward local normal.
X_S is perpendicular to the satellite orbit plane.

Y_S completes the right-handed system and is the direction of the opposite of the satellite velocity (see figure 8.2.2-1 below)

This coordinate system is defined and referred to as the "Satellite Relative Actual Reference" system in RD4.

pointing direction angle between a look direction lying in the (Y_S=0) plane in the satellite fixed coordinates system, and the $-Z_S$ axis of that system. Positive around Y_S. Notation: ψ (see figure 8.2.2-2 below)

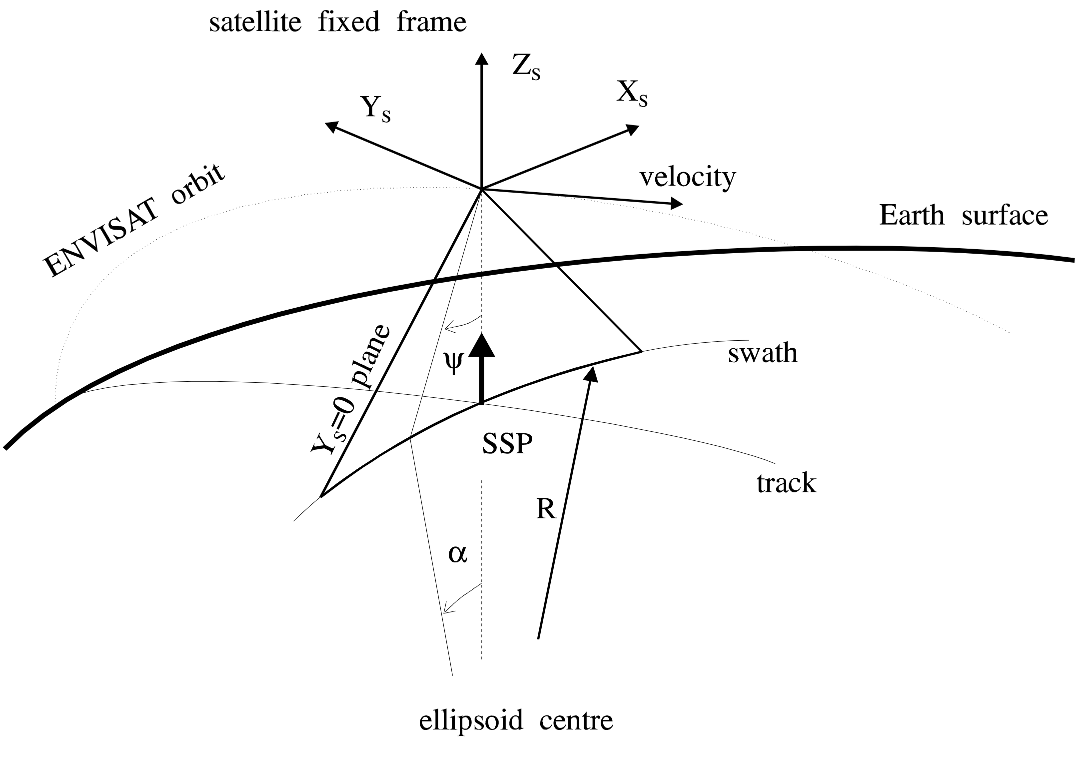


Figure 8.2.2-1 : satellite fixed coordinate system.

swath angle angle sub-tending the arc between swath centre and a point on the swath. Notation is α (see figure 8.2.2-1 above)

product swath arc on the geoid between the two extreme product pixels at a given time. The product swath is wider than the widest possible *MERIS swath*.

tie point Tie points for a given product are a matrix of Earth points, where

1) lines (*tie frames*) correspond to regularly spaced (time-wise) instants t_f, origin at the first *frame* of the product. Tie points are located at successive projections at instants t_f of the (Y_S=0) plane in the satellite fixed frame (X_S, Y_S, Z_S);

2) the central tie point is at the swath centre, i.e. the projection on the geoid of the axis Z_S;
3) tie points at a given instant are spaced at even distance (the same for all tie frames) along the *swath*.

(see figure 8.2.2-2 below)

tie frame a set of *tie points* corresponding to a given time and location of the satellite. (see figure 8.2.2-2 below)

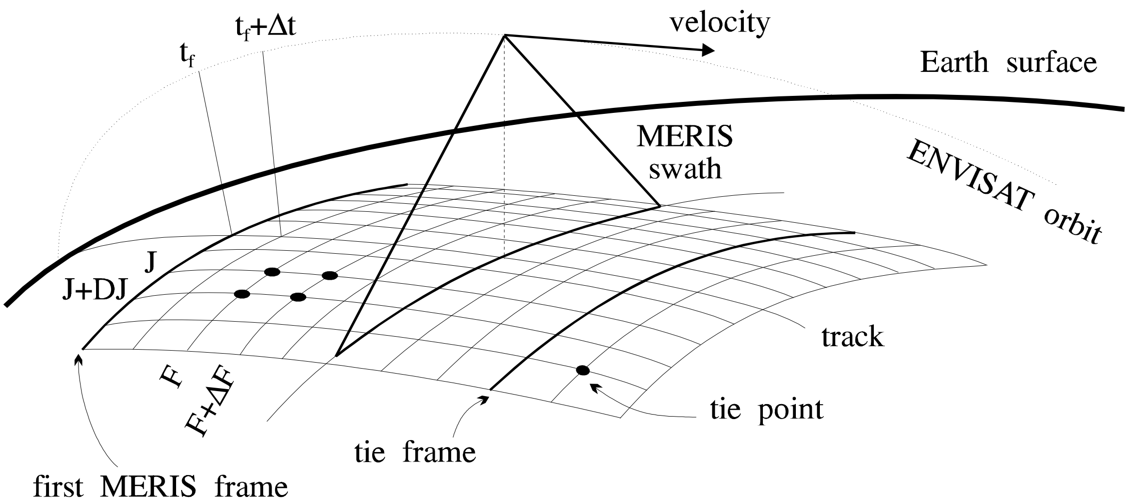


Figure 8.2.2-2 : tie points.

MERIS frame a set of simultaneously acquired MERIS measurements; by extension the time when that set is acquired. The actual MERIS pixels are located at the known lines of sight of the MERIS pixel centres at the MERIS sampling instants. These are characterised by a pointing angle $\psi_{k,m}$ and an along-track offset from the (Y_S=0) plane, noted $\delta\theta_{k,m}$. Considering the small variability of the along-track sampling distance along the orbit that offset is taken to be directly expressed in frames. As MERIS sensor elements have a nearly even angular spacing, the distance between their projections on Earth increases from centre to end of frame.

MERIS swath projection on the geoid of the sector between the extreme look directions of MERIS in the (Y_S=0) plane, at a given time. (see figure 8.2.2-2 above)

product pixel Product pixels are a matrix of points where

1) lines (frames) correspond to the MERIS sampling instants and cope with the swath at those instants;
2) columns correspond to regular subdivisions of the interval between two adjacent columns of the tie points matrix, i.e. product columns are sampling the swath at constant distance.

product swath arc on the geoid between the two extreme product pixels at a given time. The product swath is wider than the widest possible *MERIS swath*.

zenith angle angle between a look direction in the topocentric coordinates system, and the Zenith axis of that system (zenith angle + elevation angle =90°). Notation: θ_S for Sun, θ_V for viewing. (see figure 8.2.2-3 below)

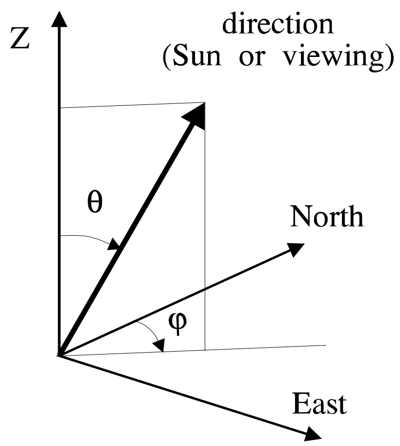


Figure 8.2.2-3 : topocentric system,zenith & azimuth angles .

Other definitions found in RD4.

latitude (geodetic) shall be noted ϕ.

longitude shall be noted λ.

azimuth shall be noted φ.

8.2.3 - Principle

The tie points are the key elements of the geo-location process.

- The initial step is, at the instants selected to include tie points, to compute the satellite location and attitude; then to compute the tie points pointing direction so that these points will be evenly spaced (in distance) along the swath; then to compute their Earth location (λ, ϕ) and the observation and illumination geometry : θ_s, φ_s, θ_v, φ_v. This is illustrated below.

1. Compute satellite motion (**po_ppforb** or **po_interpol**)
2. Compute Earth location, pointing angle and observation and illumination geometry for tie points (**pp_target**) using nominal satellite attitude (AOCS parameters) and a perturbation term.

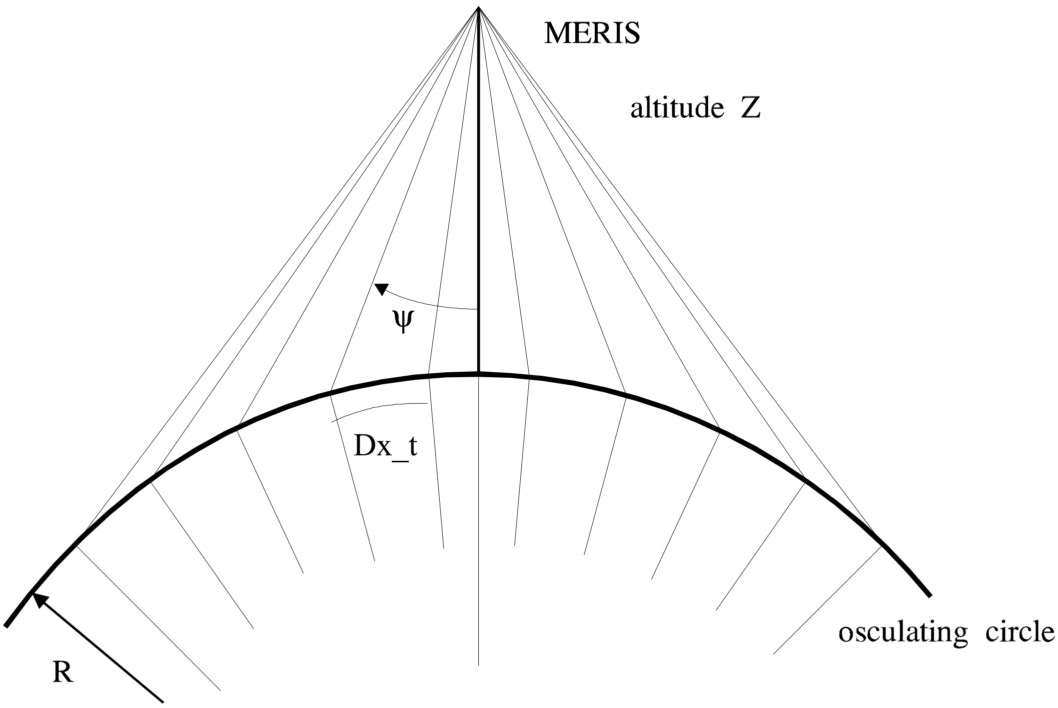


Figure 8.2.3-1 : tie points pointing direction.

- For any product pixel, its location, Sun zenith and azimuth angle, view zenith and azimuth angle can be interpolated from the location , Sun zenith and azimuth angle, view zenith and azimuth angle of the tie points which surround it. This is illustrated in fig. 8.2.3.2 below :

$$X(J+j, F+f) = \left(\frac{DJ-j}{DJ}\right)\left[\left(\frac{DF-f}{DF}\right)X(J,F) + \left(\frac{f}{DF}\right)X(J,F+DF)\right] + \left(\frac{j}{DJ}\right)\left[\left(\frac{DF-f}{DF}\right)X(J+DJ,F) + \left(\frac{f}{DF}\right)X(J+DJ,F+DF)\right]$$

where X is : longitude, latitude, zenith angle, pointing angle, swath angle
DF is tie frame spacing
DJ is tie points column spacing

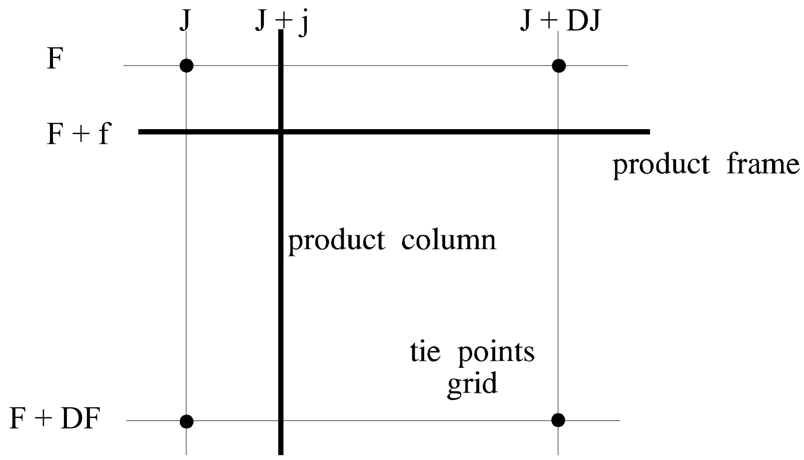


Figure 8.2.3-2 : product pixel location interpolation.

- From the location of a tie point, the Earth surface altitude and roughness at that point are read from a digital elevation data base. For land tie points, a location correction (illustrated in figure 8.2.3-3 below) is computed, stored in the product but not applied to the tie point coordinates.

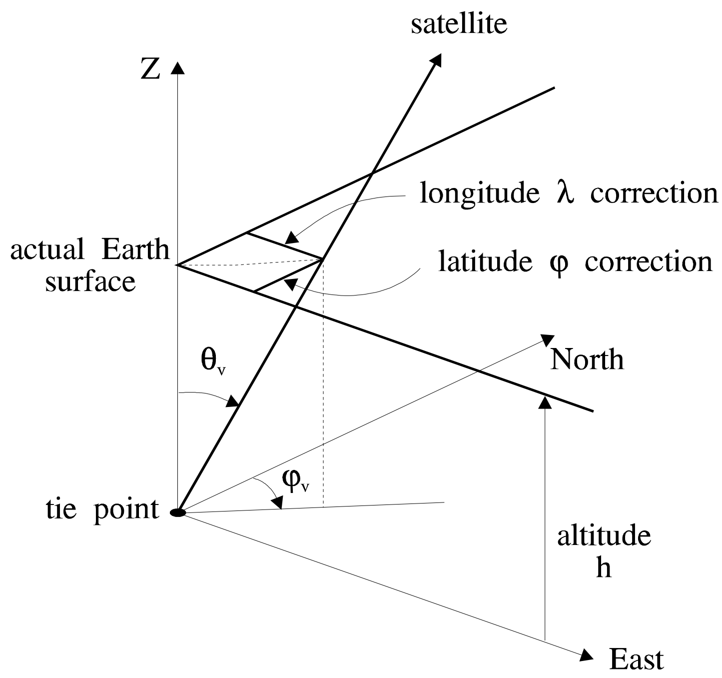


Figure 8.2.3-3 : tie point location correction with altitude .

Note that a spherical Earth assumption is considered sufficient to convert the distance correction term h.tan(θv) into a latitude and a longitude correction terms.

- From the pointing directions of the MERIS pixels and of the tie points, and the tie points locations, the relative location of any MERIS pixel and product pixel can be computed (figure 8.2.3-4 below). This provides the basis for resampling the MERIS radiances and associated flags to the product grid.

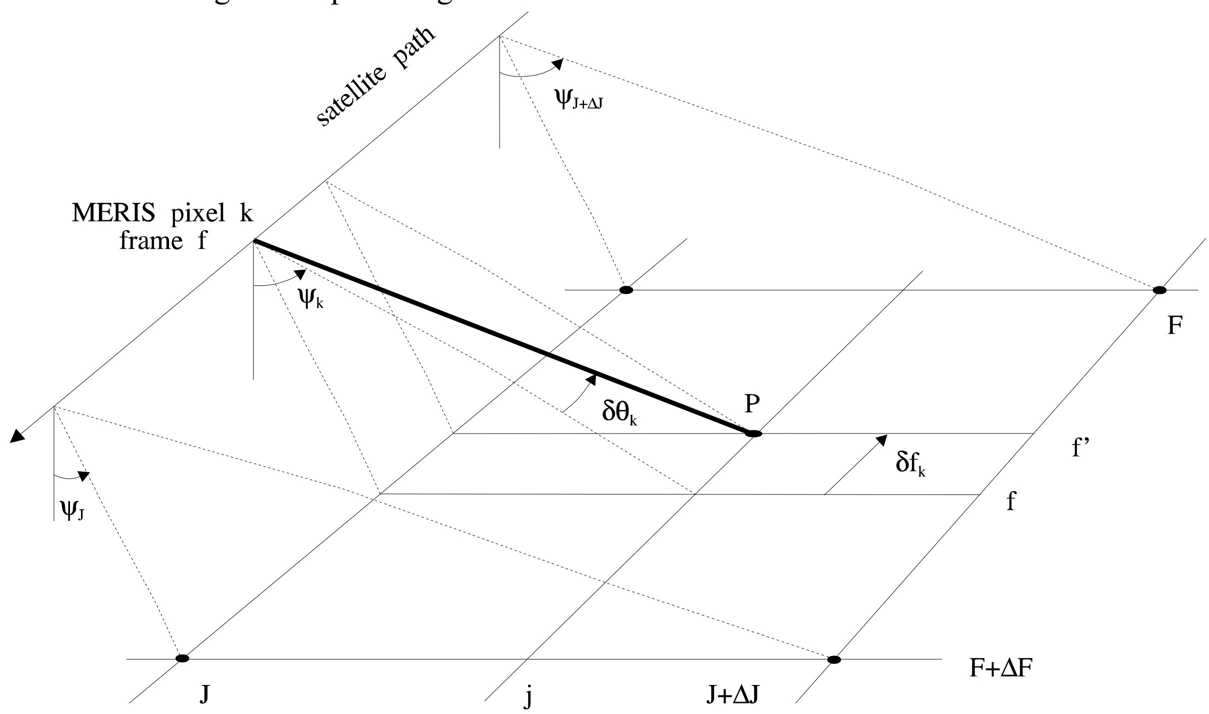


Figure 8.2.3-4 :MERIS pixel location .

For commodity the along-track depointing of the MERIS pixel $\delta\theta_k$ is expressed in terms of an integer frame offset, δf_k. This may be computed off-line using the relationship $\delta f_k = \mathrm{nint}\left(\frac{Z.\delta\theta_k}{Dx\_al}\right)$ where Z is the mean orbit altitude, Dx_al the mean along-track sampling step, nint the "nearest integer" function.

From the pointing direction of the MERIS pixels and of the tie points, the nearest MERIS pixel to any product pixel can be found. The radiances at that MERIS pixel are copied to the product pixel. When two product pixels are resampled from the same MERIS pixel both are marked as duplicate. This flag allows partial reversibility of the resampling process. This is illustrated in figure 8.2.3-5 below.

1) Bi-linear interpolation of product pixel $\psi_{i,j}$ from surrounding tie points ψ
2) Find, among those pixels allowed for resampling by the Resampling Selection Map, the MERIS column (k,m) which minimises $|\psi_{k,m} - \psi_{i,j}|$. Raise "Duplicate" flag if (k,m) already used
3) $f' = f + \delta f_{k,m}$
4) Resample $TOAR_{b,i,j} = L''_{b,f',k,m}$ for all b, etc.

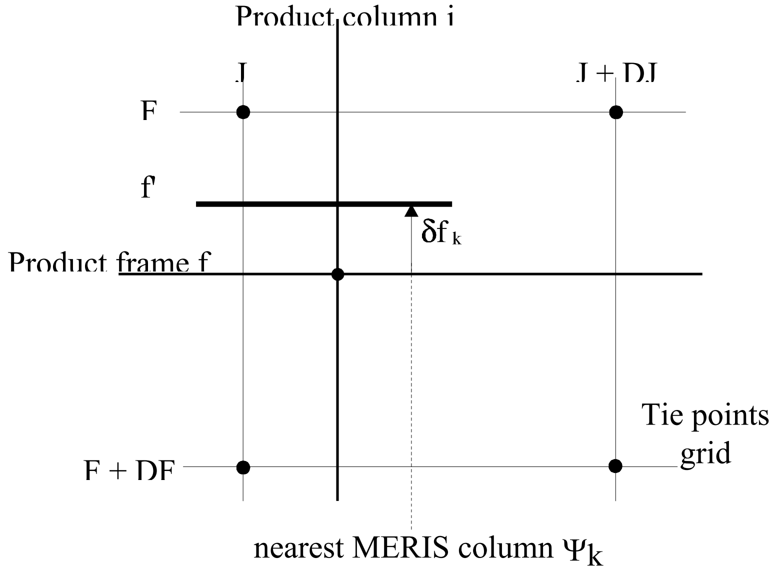


Figure 8.2.3-5 - MERIS pixel to product pixel radiances resampling (index k has been used instead of (k,m) for clarity)

8.3 - Algorithm Description

8.3.1 - Theoretical Description

8.3.1.1 - Physics of The Problem

The MERIS geo-location process makes use mostly of simple geometry, taking advantage of established models :

1. the orbital motion of the satellite around the Earth is modelled by the Orbit Propagator CFI, (described in AD3);
2. for location purposes, the shape of the Earth is represented by the WGS84 geoid described in RD11; the rotation of the Earth is represented by the Earth fixed frame, defined in RD11. Both are modelled within the Orbit propagator CFI and Target CFI (described in AD5);
3. the nominal attitude of the satellite is described by the AOCS parameters, and modelled by the Target CFI;
4. projection from the satellite to the Earth surface is modelled by the Target CFI;
5. the direction of the Sun in the topocentric coordinates system at any point on Earth is modelled by the Target CFI, neglecting surface declivity
6. the altitude and roughness at any point are taken to be those of the nearest cell in the DEM and DRM data bases, which are two matrices regularly sampled in latitude and longitude.

In addition :

1. a known rotation perturbation is applied to the nominal satellite attitude in order to derive the satellite fixed frame. That perturbation term is assumed to depend only on the time elapsed since ascending crossing node, and read from a data base;
2. the look directions of the MERIS pixels are modelled as fixed directions in the satellite fixed frame F_S: thermo-mechanical distortions and vibrations are ignored; for a given sensor element the look direction is the same for all bands, i.e. spatial registration is ignored;
3. when applying the Target CFI, the altitude of the target is 0.

In order to reduce computation and storage requirements for the product, the latitude and longitude, illumination and viewing angles, are stored at tie points only.

The illumination and viewing angles : Sun and observer zenith and azimuth angles, are of prime importance for further processing of the MERIS signal. They are computed for each tie point using knowledge of the Sun direction and of the projection geometry, and neglecting the declivity so that the local normal is the same as the normal to the Earth geoid.

The observation and illumination geometry can be used to derive a condition for sun glint risk (i. e. specular reflection of the Sun light at the product pixel), **assuming a flat surface**. That condition is satisfied when

- the observation and Sun zenith angles are equal within a tolerance;
- and the observation and Sun azimut angles are opposite within a tolerance.

8.3.1.2 - Mathematical Description

The Geo-location processing includes five algorithms (step 1.5.3 does not exist) :

Hierarchical number	Identification	Remarks
1.5.1	Product limits	
1.5.2	Tie Points Location	
1.5.4	Altitude annotation and correction	
1.5.5	Radiance re-sampling	
1.5.6	Glint risk flag	

The overall control flow must ensure, when processing a MERIS frame, that the tie points location for the following tie points frame has already been performed; and when processing a product frame, that the following MERIS frames have been processed to ensure compensation of the along-track depointing of MERIS pixels with respect to the swath. This is described in chapter 3 above, where one process produces the tie points informations based solely on elapsed time (encompassing algorithms 1.5.2, 1.5.4), and one process uses these informations to resample the corrected radiances and flags (encompassing algorithm 1.5.5).

Step 1.5.1 - Product limits

Product limits determination have two distinct goals according to the product resolution:

- For a Reduced resolution product, the only limits to determine correspond to the along-track splitting of the Level 0 product needed by the processing. They are directly extracted from the Work Order as times of first and last frames to process.
- In Full Resolution, as a scene with pre-defined dimensions in both directions have to be extracted from the Level 0 product following a user specific request, inputs are differents and limits must be computed in both directions (along- and across-track) ; in addition different across-track limits have to be determined for data extraction and for the Level 1B product. Its inputs, extracted from the Work Order, are the location of the desired scene centre (lat., lon.) and the scene type (scene / imagette of known sizes). The corresponding product limits are derived so that the actual scene centre location is as close as possible to the requested one with the following restrictions:
  - The first frame of the Level 1b product will allways match the Tie Point grid defined with respect to the Level0 product limits;
  - The first column of the Level 1b product will allways match the Tie Point grid defined with respect to the Level0 product limits;

However, a common list of outputs have been defined to simplify the interface with either the data extraction and the geo-location and spatial resampling algorithms : it consist in all parameters needed to specify data extraction limits to step 1.1 - Source Data packets Extraction - in both along-track (or time) and across-track directions and the corresponding number of Tie Points allowing the geolocation of all Level 1B product pixels without extrapolation. Parameters are listed below.

1. time of first and last frames to extract from the Level 0 product and corresponding number of frames,
2. first and last MERIS modules to extract from each packet,

3. first and last tie points columns needed for the above across-track extraction limits (relative to the fixed numbering corresponding to the whole swath width, see Tie Points definition in section 8.2.2 above) and corresponding number of product pixels columns.

In Full Resolution, the Product Limits algorithm makes use of the ESA CFI **po_ppforb** (see AD3)**, pp_stavis** (see AD5), **pl_geo_distance** (see AD11) and **pp_target** (see AD5), in order to compute the parameters listed above. It first computes the time at which the Scene Centre is imaged by MERIS and, from scene size, deduce the along track limits of the Level 1B product. In a second step, the tie point columns bracketting the center are identified and the across-track limits of the Level 1B product are derived. Finally, the across-track limits of extraction - identification of the MERIS modules necessary to cover the desired scene - are derived by comparisons of the Pointing angles of the extreme *product* pixels of the scene (which would be tie points if the central frame was a tie frame) with those of the MERIS modules edges. Deriving limits from the central frame geometry have been found sufficiently accurate even if it may induce, in some cases, lack of data over small zones at image edges.

Step 1.5.2 - Tie Points Location Algorithm

That algorithm is performed at each tie frame of the product, except for

- the orbit propagator initialisation (1.5.2.3)

which is done once at processing initialisation. The data and control flow within the algorithm are shown in fig. 8.3.1.2-2 below.

The Tie Points Location algorithm makes use of the ESA CFI **po_ppforb** or **po_interpol** (see AD3), **pp_target** (see AD5), in order to compute the latitude, longitude, view zenith and azimuth angles, Sun zenith and azimuth angles, pointing angle of all tie points.

Step 1.5.2.1 - deleted

Step 1.5.2.2 - Tie points frame instants

The first tie frame is defined at the time of the first MERIS frame of the product. Then every DFth frame (DF is 16 in RR, 64 in FR processing) is a tie points frame. The first tie frame time is corrected for the delay inside MERIS : as the tie points grid is defined at the top of the corresponding frame (see AD1), a correction is performed to take into account the delay between start of exposure for frame f, and the read-out by the instrument of the on-board time for copying into the product header. That bias has a different value for FR and RR processing.

Step 1.5.2.3 - Initialise Orbit Propagator

Depending on processing type, consolidated or not, two different orbit propagators are used. In consolidated processing, **po_interpol** is choosen as it can manage the DORIS and ESOC orbit files. It is initialised with the orbit files names and the Level 1B product time limits. In non-consolidated processing, **po_ppforb** is used. One state vector near ascending crossing node, assumed to be extracted from the Level 0 product Main Product Header, is used to initialise the orbit propagator for the whole orbit.

Step 1.5.2.4 - Propagate orbit

The instants where the propagator computes the orbital motion of the ENVISAT satellite are those of the tie frames. These instants are provided by step 1.5.2.2.
The satellite state vector, acceleration at tie frame time are computed by the CFI routines :

- **po_interpol** for consolidated processing,
- **po_ppforb** for non-consolidated processing.

Both routines are described in AD3.

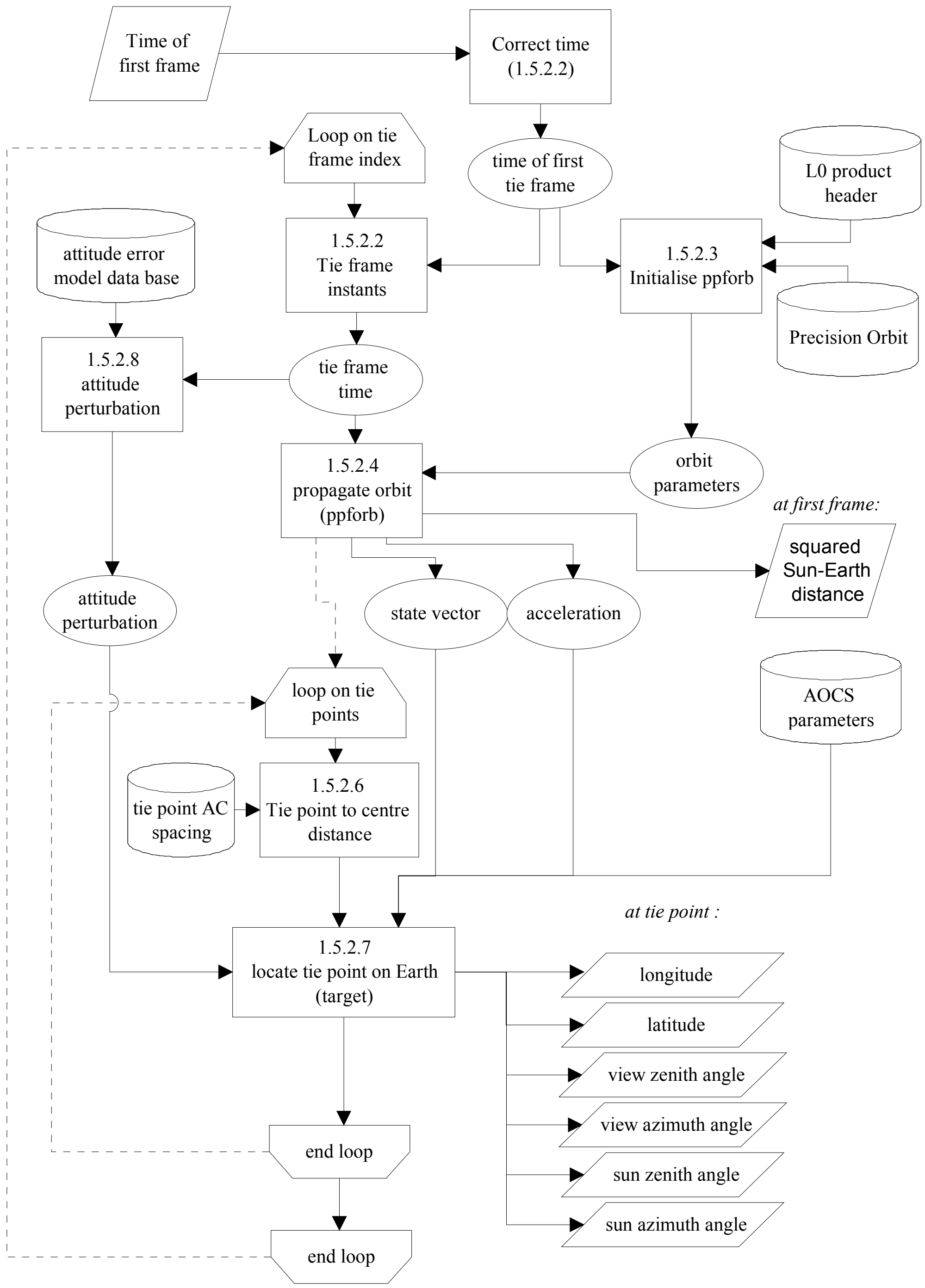


Figure 8.3.1.2-2 : Tie Points Location block diagram

Step 1.5.2.6 - Tie point distance from swath centre

The tie points for the considered frame are constructed by even spatial spacing along the swath, with the central tie point at swath centre (elevation from satellite to target = 90°)

$dist_{J,F} = Dx\_t * (J - J\_centre)$

Step 1.5.2.7 - Locate tie point on Earth

For each tie point J,F the look direction from the satellite is given as a function of $dist_{J,F}$ by :

if $dist_{J,F} < 0$ then azimuth = -90°
else azimuth = 90°
elevation = 90°

For each tie point, the **pp_target** routine is called (with parameter idir = 3) using

1. the state vector, acceleration, computed by **po_ppforb** or **po_interpol** (see 1.5.2.4 above);
2. the tie point distance to swath centre $dist_{J,F}$;
3. the AOCS parameters;
4. the attitude perturbation (see 1.5.2.8 below).

pp_target returns, at the tie point :

1. the latitude and longitude;
2. the satellite elevation and azimuth angles (then zenith angle = 90° - elevation angle);
3. the Sun elevation and azimuth angles;

Step 1.5.2.8 - Attitude perturbation

The attitude perturbation expressed as roll, pitch, yaw rotation terms, is interpolated between its value at sampled intervals along the orbit, read from the "ENVISAT-1 Platform Attitude product" (see AD1). That product is assumed to be always available.

Step 1.5.4 - Altitude Retrieval, Correction Algorithm

That algorithm is applied to all the tie points of the product after they have been located on Earth (see 1.5.2 above).

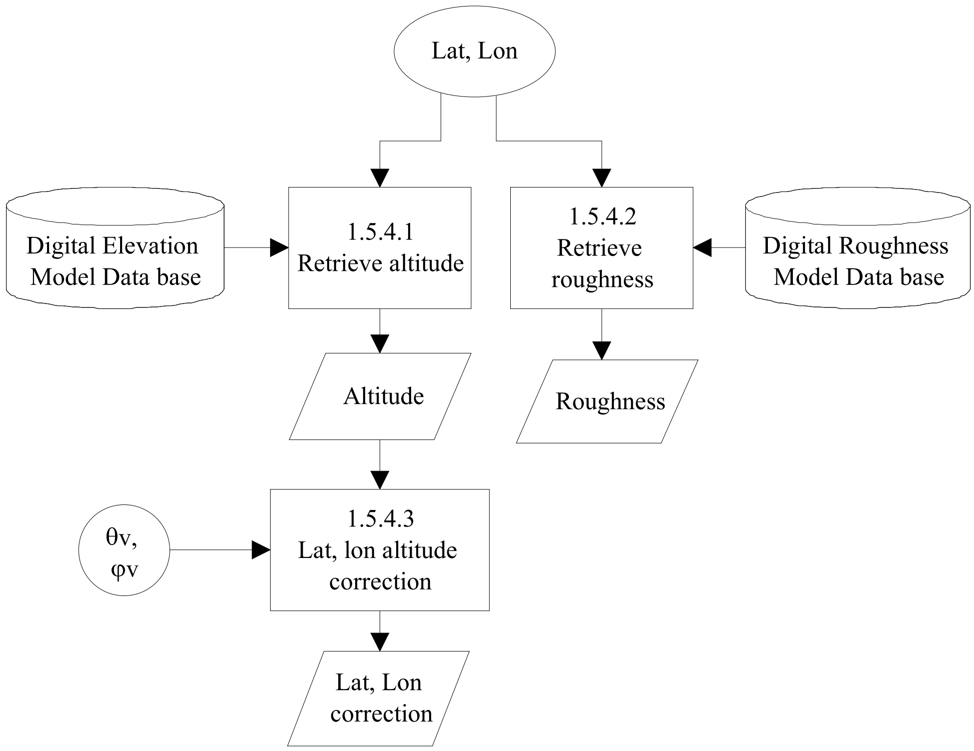


Figure 8.3.1.2-3 : Altitude annotation and correction block diagram

Step 1.5.4.1 - Retrieve altitude

The latitude and longitude of the tie point are scaled to line and column index in the Digital Elevation Model data base (see AD1), using the data base grid step, latitude and longitude origins. The value of altitude is read from the altitude matrix at those indices.

Step 1.5.4.2 - Retrieve roughness

The value of terrain roughness is read from the Digital Rooughness Model data base (see AD1) at the line and column indices of the tie point. It is assumed that elevation and roughness model use the same grid. The roughness provides a confidence element for the altitude and altitude correction terms : the higher the roughness, the more likely that a pixel near the tie point has a different altitude than the tie point.

Step 1.5.4.3 - Latitude, longitude correction for altitude

In case of a land product pixel, a correction is brought to the tie point longitude, latitude, in order to account for the displacement of the actual satellite point of view location when the target altitude in not 0. In order to preserve reversibility that correction is not applied to the tie point coordinates but stored with the product. The correction term is computed at every tie point to keep the control flow simple; the product formatting (see §11 below) will replace it with 0 when the tie point (more accurately, the product pixel co-located with the tie point) is classified as "ocean" (see chapter 9).

For a tie point altitude z, and assuming that altitude is uniform in the area surrounding the tie point, the correction in distance along the swath is $dx=z.\tan\theta$ (see figure 8.2.3-3 above). Using φ to project on the East and North axes of the local topocentric coordinates system, and using a spherical Earth approximation, dx is then converted to latitude and longitude correction terms.

Step 1.5.5 - Radiance Resampling Algorithm

That algorithm is applied to all product pixels within product limits in order to re-sample to the product grid, the quantities which have been computed for the MERIS pixels :

1. corrected radiance samples (from Stray light correction algorithm, see chapter 6 above);
2. quality flags (from Stray light correction algorithm : "valid" at frame level, "dubious", "saturated", "cosmetic" and "straylight risk" at pixel level, see chapter 6 above);

The data flow in the algorithm is shown in fig. 8.3.1.2-5 below.

That algorithm is enabled by a dedicated switch (nominal processing is resampling enabled). In case it is disabled, steps 1.5.5.1 to 1.5.5.4 are by-passed and replaced by step 1.5.5.5, where MERIS pixels are copied into product ones regardless of the Product across-track limits but taking account of the extraction limits.

Step 1.5.5.1 - Interpolate product pixel pointing

Product pixels shall be processed based on the neighbouring tie points J,F such that

$J \le j \le J + DJ$; $F \le f \le F + DF$

For all product pixels in frame f between these tie points columns, the pointing angle $\psi_{j,f}$ is linearly interpolated from $\psi_{J,f}$ and $\psi_{J+DJ,f}$; this preserves equidistance on the swath with an accuracy of ± 3 %.

Step 1.5.5.2 - Find nearest

The MERIS pixels AC pointing data base is searched to determine the pixel index (k,m) for which the value $\psi_{k,m}$ is nearest to the product pixel pointing $\psi_{j,f}$ and not listed as "unwanted" in the Resampling Map. If $| \psi_{j,f} - \psi_{k,m} |$ is too large (higher than 2*IDEFOV) then the product pixel is considered to be outside of the MERIS swath. Otherwise, the index (k,m) of that value is the MERIS column to be resampled. If the selected MERIS pixel (k,m) has already been used to fill another product pixel (j',f), then the flag "duplicate" is set to TRUE for the current product pixel (j,f). The Resampling Map is extracted from the Pixel ID field of the MERIS Instrument Product (most significant bit of the byte corresponding to a given column, see AD1).

Practically, as both the $\psi_{j,f}$ and the $\psi_{k,m}$ are monotone increasing values, an exhaustive search through the AC pointing data base is almost never needed ($\psi_{k,m}$ increases monotonously with column k except at module limits where they overlap).

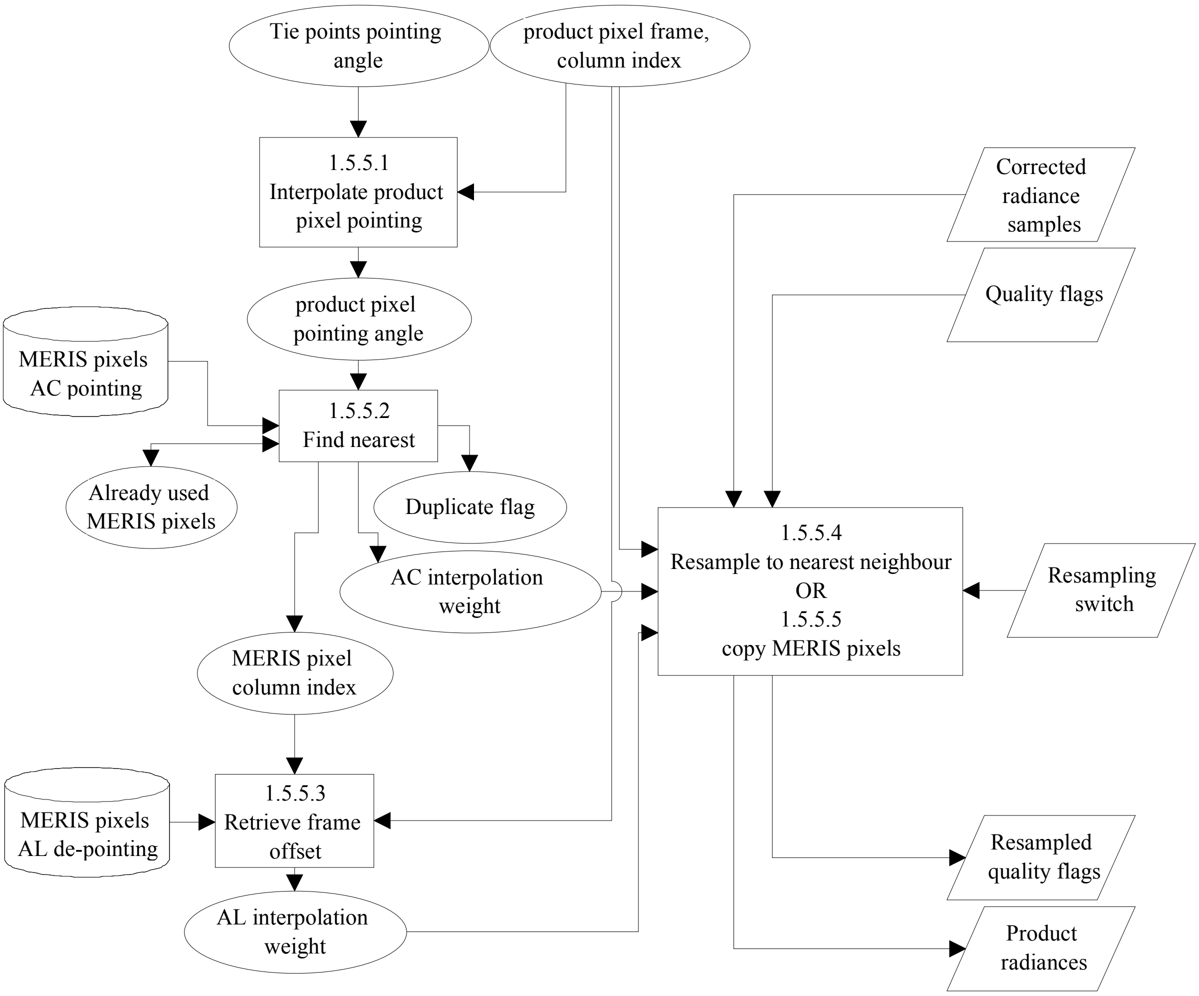


Figure 8.3.1.2-5 : Radiance re-sampling block diagram

Step 1.5.5.3 - Retrieve frame offset

From the nearest MERIS pixel column (k,m), the along-track depointing $\delta f_{k,m}$ is retrieved from the AL depointing data base. That depointing is an integer number of frames, the nearest value to the known along-track depointing of the pixel (k,m) with reference to the plane (Y_S=0). The MERIS frame index to be resampled is $f'=f+\delta f_{k,m}$. If that index is larger than the last available MERIS frame then the product pixel is outside of the MERIS image.

Step 1.5.5.4 - Resample to nearest neighbour

If the product pixel is outside of the MERIS extracted data, it is flagged as "invalid" in all band and its radiance is set to a default value. Otherwise, the quantities computed at MERIS pixel are resampled. $\delta f_{k,m}$ may be positive, i.e. the MERIS pixel corresponding to product pixel (j,f) be found in the input stream at a later time. The resampling to product frame f shall be performed when these quantities have been computed for MERIS frame $f + \max(\delta f_{k,m})$.

The resampling relationship
$X'_{j,f} = X_{k,m,f'}$
is applied to :

1. corrected radiances (for all b)
2. "dubious sample" flag (for all b)
3. "saturated sample" flag (for all b)
4. "cosmetic sample" flag (for all b)
5. "stray light risk" flag

Step 1.5.6 - Sun glint risk flag

The Sun glint risk flag is computed from the zenith and azimuth angles differences at each tie point; the result is then propagated to all pixels in the corresponding cell.

8.3.2 - List of Variables

Variable	Descriptive Name	T	U	Range - References
VECTOR_SOURCE	code for type of Orbit State Vector File	s	dl	from Level 0 MPH
VECTOR_FILE	Orbit State Vector File name	s	dl	from Level 0 SPH
Mt	number of MERIS modules	s	dl	5
K^{FR}	number of FR columns in a MERIS module	s	dl	740
K^{RR}	number of RR columns in a MERIS module	s	dl	740
JD0, JD1	JD of first and last frames in Level0 product	s	jd	
ϕ_{SSP0}, ϕ_{SSP1}	latitude of SSP for first and last frames of the Level 0 product	s	deg	
ϕ_{centre}, λ_{centre}	latitude, longitude of FR scene centre	s	deg	FR only
image_type	FR image type : imagette or scene	s	dl	FR only
begin_time, end_time	time of first and last frame to process	s	dl	RR only
Consolidated_procesing	Switch enabling Consolidated Processing options	s	dl	
NC^{IM}	Image AC size for FR imagette	s	dl	
NC^{FR}	Image AC size for FR scene	s	dl	
NC^{RR}	Image AC size for RR product	s	dl	
DT_frame^{FR}	Bias for FR frame time correction	s	jd	
DT_frame^{RR}	Bias for RR frame time correction	s	jd	
Re	Mean Earth radius	s	m	
resampling_switch	switch enabling re-sampling process	s	dl	
NJ	Number of tie points for full swath	s	dl	71
Dx_t	Across-track tie points pitch	s	m	16640 m
DJ^{FR}	Across-track pixel to tie point subsampling factor in FR	s	dl	64
DJ^{RR}	Across-track pixel to tie point subsampling factor in RR	s	dl	16
DF^{FR}	Along-track frame to tie frame subsampling factor in FR	s	dl	64
DF^{RR}	Along-track frame to tie frame subsampling factor in RR	s	dl	16
DT^{FR}	Delay between two FR frames	s	ms	44
DT^{RR}	Delay between two RR frames	s	ms	176
$max\_d\psi^{FR}$	Maximum across track angular distance allowing pixel selection in FR	s	deg	
$max\_d\psi^{RR}$	Maximum across track angular distance allowing pixel selection in RR	s	deg	
$resamp\_pix^{FR}_{k,m}$	FR pixels resampling selection map	s	dl	
$resamp\_pix^{RR}_{k,m}$	RR pixels resampling selection map	s	dl	
$U_x{}^{FR}{}_{k,m}$, $U_y{}^{FR}{}_{k,m}$	x and y components of MERIS FR pixels pointing unit vectors	s	deg	
$\Delta\phi^{FR}$	Along-track depointing angle corresponding to one FR frame	s	deg	
$U_x{}^{RR}{}_{k,m}$, $U_y{}^{RR}{}_{k,m}$	x and y components of MERIS RR pixels pointing unit vectors	s	deg	
$\Delta\phi^{RR}$	Along-track depointing angle corresponding to one RR frame	s	deg	
AOCS[3]	Pitch, roll, yaw amplitude	s	deg	
Att_error_model[]	Attitude error model data base	s	deg	see note 2
DEM[lon,lat]	Digital elevation model	s	m	see note 3
DRM[lon,lat]	Digital roughness model for land pixels	s	m	see note 3
glint_thr_zen	threshold on zenith angle difference for glint mask	s	deg	
glint_thr_azi	threshold on azimuth angle difference for glint mask	s	deg	
T_JD[f]	UTC time of extracted frames	i	jd	from 1.1
valid_frame_f[f]	valid frame flag	i	dl	from 1.3
L^{FR}[b,k,m,f]	Radiance at MERIS FR/RR pixels	i	LU	from 1.4
L^{RR}[b,k,m,f]	Radiance at MERIS FR/RR pixels	i	LU	from 1.4
$dubious\_f^{FR}$[b,k,m,f]	dubious sample flag for FR/RR	i	dl	from 1.2
$saturated\_f^{FR}$[b,k,m,f]	saturated sample flag for FR/RR	i	dl	from 1.3
$cosmetic\_f^{FR}$[b,k,m,f]	cosmetic sample flag for FR/RR	i	dl	from 1.3
$stray\_f^{FR}$[k,m,f]	stray light risk flag for FR/RR	i	dl	from 1.4
$dubious\_f^{RR}$[b,k,m,f]	dubious sample flag for FR/RR	i	dl	from 1.2
$saturated\_f^{RR}$[b,k,m,f]	saturated sample flag for FR/RR	i	dl	from 1.3
$cosmetic\_f^{RR}$[b,k,m,f]	cosmetic sample flag for FR/RR	i	dl	from 1.3
$stray\_f^{RR}$[k,m,f]	stray light risk flag for FR/RR	i	dl	from 1.4

Table 8.3.2-1 : Parameters used in the geo-location algorithm

Variable	Descriptive Name	T	U	Range - References
Applicable_vector	Applicable state vector	c	-	
mjdr, mjdp, mjdr0, mjdr1	UTC time structures for interface with orbit propagators	c	jd	see AD3
xm[6]	Mean Kepler state at true ascending node	c	-	see AD3
$\psi^{FR}_{k,m}$	Across-track pointing of MERIS pixel	c	deg	
$\delta f^{FR}_{k,m}$	Along-track depointing of MERIS pixel	c	dl	
$\psi^{RR}_{k,m}$	Across-track pointing of MERIS pixel	c	deg	
$\delta f^{RR}_{k,m}$	Along-track depointing of MERIS pixel	c	dl	
pos[3]	Predicted osculating cartesian position vector at frame time	c	m	see AD3
vel[3]	Predicted osculating cartesian velocity vector at frame time	c	$m.s^{-1}$	see AD3
acc[3]	Predicted osculating cartesian acceleration vector at frame time	c	$m.s^{-2}$	see AD3
F1,F2	first and last Level0 frames to process	c	dl	
J1,J2	Across-track limits of Level1b product (indices of first and last tie points)	c	dl	FR only
t1, t2	first and second estimations of scene centre imaging time	c	jd	FR only
λ_{SSP}, ϕ_{SSP}	SSP longitude and latitude	c	deg	FR only
sta	structure for ground station definition	c	-	see AD11, FR only
φ	satellite to scene center azimuth	c	deg	FR only
γ	Topocentric azimuth of y axis of Satellite frame	c	deg	FR only
f_{centre}	index of frame closest to Scene Centre	c	dl	FR only
d	SSP to Scene Center distance	c	m	FR only
az1	azimuth of Scene Centre from SSP		deg	FR only
d'	AC distance from swath eastern edge to Scene Centre	c	m	FR only
J_centre	index of central tie point (wrt full swath)	c	dl	(NJ+1)/2
k1, k2	indices of tie points bracketting Scene Centre	c	dl	FR only
ψ_1, ψ_2	Pointing angles of central frame extreme columns	c	deg	FR only
first_tie_k, last_tie_k	product columns index corresponding to extreme tie points (wrt full swath)	c	dl	
m, k	indices for MERIS modules and columns, respectively	c	dl	
rel_time	Relative time from ascending node	c	jd	
att_error[3]	Attitude error	c	deg	pitch, roll, yaw rotations
$\psi_{J,F}$	Pointing angle at tie point J,F	c	deg	
p	Interpolation weighting factor	c	dl	
dx	across-track location error	c	m	
$dist_{J,F}$	tie point J,F distance to swath centre	c	m	
$\psi_{j,f}$	Pointing angle at product pixel j,f	c	deg	
used[k,m]	flag set if MERIS pixel k,m already used in resampling	c	dl	
outputs of 1.5.1:				
first_module	index of first MERIS module to process	o	dl	to 1.1
M	number of MERIS modules to process	o	dl	to 1.1, 1.2, 1.3, 1.4
N_{TP}	number of tie pointsper frame in Level 1B product	o	dl	to 1.6, 1.7, 1.8
NF	number of frames in Level1b product	o	dl	to 1.4, 1.8
NC	number of columns in Level1b product	o	dl	to 1.8
begin_JD	lower time limit for packet extraction	o	jd	to 1.1
end_JD	upper time limit for packet extraction	o	jd	to 1.1
CNT_JD	JD time at ascending node	o	jd	to 1.3

Table 8.3.2-1 : Parameters used in the geo-location algorithm (cont)

Variable	Descriptive Name	T	U	Range - References
outputs of 1.5.2, 1.5.4, 1.5.5 and 1.5.6:				
T_JD'[f]	Corrected UTC time of frame f	o	jd	to 1.8
ϕ[J,F]	Geodetic latitude of tie point J,F	o	deg	to 1.6, 1.7, 1.8
λ[J,F]	Longitude of tie point J,F	o	deg	to 1.6, 1.7, 1.8
θ_s[J,F]	Sun zenith angle at tie point J,F	o	deg	to 1.6, 1.8
φ_s[J,F]	Sun azimuth angle at tie point J;F	o	deg	to 1.6, 1.8
θ_v[J,F]	Observer zenith angle at tie point J;F	o	deg	to 1.6, 1.8
φ_v[J,F]	Observer azimuth angle at tie point J;F	o	deg	to 1.6, 1.8
z[J,F]	Altitude at tie point J,F	o	m	to 1.8
σz[J,F]	Altitude standard deviation at tie point J,F	o	m	to 1.8
dlon[J,F]	Altitude correction term for latitude	o	deg	to 1.8
dlat[J,F]	Altitude correction term for longitude	o	deg	to 1.8
$TOAR^{FR}$[b,j,f]	FR resampled TOA radiance at pixel j,f	o	LU	to 1.6, 1.8
$TOAR^{RR}$[b,j,f]	RR resampled TOA radiance at pixel j,f	o	LU	to 1.6, 1.8
Invalid_f^{FR}[j,f]	FR "invalid pixel" flag	o	dl	to 1.6, 1.8
Invalid_f^{RR}[j,f]	RR "invalid pixel" flag	o	dl	to 1.6, 1.8
Dubious_f^{FR}[b,j,f]	FR resampled "dubious sample" flag	o	dl	to 1.8
Dubious_f^{RR}[b,j,f]	RR resampled "dubious sample" flag	o	dl	to 1.8
Saturated_f^{FR}[b,j,f]	FR resampled "saturated sample" flag	o	dl	to 1.6, 1.8
Saturated_f^{RR}[b,j,f]	RR resampled "saturated sample" flag	o	dl	to 1.6, 1.8
Cosmetic_f^{FR}[b,j,f]	FR resampled "cosmetic sample" flag	o	dl	to 1.8
Cosmetic_f^{RR}[b,j,f]	RR resampled "cosmetic sample" flag	o	dl	to 1.8
Glint_f^{FR}[j,f]	FR sun glint risk flag	o	dl	to 1.8
Glint_f^{RR}[j,f]	RR sun glint risk flag	o	dl	to 1.8
Duplicated_f^{FR}[j,f]	FR duplicated pixel flag	o	dl	to 1.8
Duplicated_f^{RR}[j,f]	RR duplicated pixel flag	o	dl	to 1.8
Stray_f^{FR}[j,f]	RR straylight risk flag for frame f	o	dl	to 1.8
Stray_f^{RR}[j,f]	RR straylight risk flag for frame f	o	dl	to 1.8
$Detector^{FR}$[j,f]	FR Detector index resampled at pixel j,f	o	dl	to 1.8
$Detector^{RR}$[j,f]	RR Detector index resampled at pixel j,f	o	dl	to 1.8

Table 8.3.2-1 : Parameters used in the geo-location algorithm (cont)

NOTES :

1. a state vector has the following structure :

Field no	Symbol	Description	Unit	Type	Remark
1	PTIME	Epoch	MJD2000	long[3]	
2	RR	Satellite Cartesian coordinates in F_g	m	double[3]	
3	RRD	Satellite Cartesian velocity in F_g	$m.s^{-1}$	double[3]	

2. the attitude error model data base contains a time-ordered array of elements with the following structure:

Field no	Symbol	Description	Unit	Type	Remark
1	time	Relative time since ascending node	MJD2000	double	
2	rot	Attitude error	deg	double[3]	

3. the digital elevation map provides for any lat /lon an altitude with reference to the geoid; the digital roughness map a local value of standard deviation of altitude.

8.3.3 - Equations

Note that in the following equations, section numbers correpond to the hierarchical numbering used in algorithm breakdown above. For the sake of clarity, the superscript FR or RR for those parameters which depend on resolution is omitted in sections 1.5.2 to 1.5.6, as processing is identical for both resolutions.

Tie point indexing is noted F (tie frame), J (tie column) where F is in the range {1, 1+DF,..., NF}, J is in the range {1, 1+DJ,...,1+(NJ-1)*DJ). Thus the tie frame number and the corresponding product frame number are the same.
Level 1B Product frame index is noted f as well as MERIS frame index (i.e. Level0 Product frame) but the latter, for sake of clarity is related to extraction limits instead of Level0 Product limits. Thus MERIS frame f and Level1b frame f correspond to the same sampling instant. Frame index f' is used to identify a MERIS frame, taking account of the along-track depointing.
Due to across-track product limit, a double indexing is sometimes used for pixel columns : j refers to column index in output product and is in the range [1,NC] while j' refers to column index with respect to full swath and is in the range [1, 1+(NJ-1)*DJ]. Column j=1 corresponds to j'=first_tie_k (this relation appears in equations each time double indexing is used).

It is important to note that despite tie points column numbering refers to full swath, calculations are always restricted to the useful range [first_tie_k, last_tie_k]. Numbering with respect to full swath has been chosen because it is easily related to symmetry around Nadir.

mod is the "modulo" function (mod(a,b) = remainder of the Euclidian division of a by b).
int is the truncation to *lower* integer function, nint is the truncation to *nearest* integer function.

IMPORTANT NOTE : algorithm step 1.5.1 and steps 1.5.2 to 1.5.6 are grouped together in the current section as they are closely related and share common resources and databases parameters. However, for implementation purposes, special attention must be paid to the internal i/o interfaces, not described here as they greatly rely on architectural choices, and to the CFI routines initialisation requirements.

step 1.5.0 – Pointing Vectors pre-processing

if Resolution is Full then

Full Resolution case

$$U_{z\ k,m}^{FR} = -\sqrt{1-\left(U_{x\ k,m}^{FR}\right)^2-\left(U_{y\ k,m}^{FR}\right)^2} \quad (1.5.0\text{-}1)$$

$$\psi_{k,m}^{FR} = \mathrm{atan}\left(\frac{U_{x\ k,m}^{FR}}{U_{z\ k,m}^{FR}}\right) \quad (1.5.0\text{-}2)$$

$$\delta f_{k,m}^{FR} = \mathrm{nint}\left(\frac{\mathrm{atan}\left(\frac{-U_{y\ k,m}^{FR}}{U_{z\ k,m}^{FR}}\right)}{\Delta\phi^{FR}}\right) \quad (1.5.0\text{-}3)$$

else

Reduced Resolution case

$$U_{z\ k,m}^{RR} = -\sqrt{1-\left(U_{x\ k,m}^{RR}\right)^2-\left(U_{y\ k,m}^{RR}\right)^2} \quad (1.5.0\text{-}4)$$

$$\psi_{k,m}^{RR} = \mathrm{atan}\left(\frac{U_{x\ k,m}^{RR}}{U_{z\ k,m}^{RR}}\right) \quad (1.5.0\text{-}5)$$

$$\delta f_{k,m}^{RR} = \mathrm{nint}\left(\frac{\mathrm{atan}\left(\frac{-U_{y\ k,m}^{RR}}{U_{z\ k,m}^{RR}}\right)}{\Delta\phi^{RR}}\right) \quad (1.5.0\text{-}6)$$

endif

step 1.5.1 - Product Limits

Full Resolution case

if Resolution is Full then

step 1.5.1.1 - Get FR Image definition parameters

retrieve centre location and image type from Work Order (1.5.1.1-1)

extract ϕ_{centre}, λ_{centre} and image_type

define the number of columns and frames of the Level 1B product accordingly (1.5.1.1-2)

if (image_type = IMAGETTE) then
  $NC = NC^{IM}$
else
  $NC = NC^{FR}$
endif
NF = NC

compute number of tie points in Level 1B product (1.5.1.1-3)

$N_{TP} = 1 + int((NC-1) / DJ^{FR})$

step 1.5.1.2 - Determine FR Along-Track Level 1B product limits

deleted (1.5.1.2-1)

extract state vector from Level 0 product (1.5.1.2-2)

extract Applicable_vector from Level 0 product *call CFI orbit propagator routine in init mode, determine time at ascending node and orbit period (in days)* *(1.5.1.2-3)*

call **po_ppforb**
inputs: mode=PO_INIT, Applicable_Vector
outputs: CNT_JD=mjdr, xm, orbit_period=res[52]/86400

Exception processing:
In case of failure of po_ppforb call, i.e. if the returned status is not 0, then
Apply steps 1.5.2.3-0

call CFI Precision Orbit interpolation/propagation routine in init mode, determine time at ascending node and orbit period (in days) *(1.5.1.2-4)*

call **po_interpol**
inputs: mode=PO_INIT_FILE, choice, ndc, ndp, ner,
doris_precise_file, doris_prelim_file, esoc_rest_file,
mjdr0=JD0, mjdr1=JD1
outputs: orbit_period=res[52]/86400, CNT_JD=res[53]-orbit_period
Set flag USE_INTERPOL to TRUE

End exception processing

check scene centre visibilty at product ends *(1.5.1.2-0)*
call CFI orbit propagator in propagation mode for beginning of product *(1.5.1.2-0.1)*

If USE_INTERPOL == FALSE then
call **po_ppforb**
inputs: mode=PO_PROPAG, mjdr, xm, mjdp=JD0
outputs: pos, vel, acc
Else
call **po_interpol**
inputs : mode=PO_INTERPOLATE, mjdr0= JD0
outputs : pos, vel, acc
Endif

call CFI satellite to ground station visibility *(1.5.1.2-0.2)*

compute attitude perturbation att_error as per step 1.5.2.8
call **pp_stavis**
inputs: mjdp=JD0, pos, vel, acc, AOCS, att_error, datt = 0,
sta = {λ_{centre}, ϕ_{centre}, 0., -90.}
outputs: φ = res[3],

check satellite to scene centre azimuth between "ahead" and "back", raise exception processing if "back" *(1.5.1.2-0.3)*

if(φ>180) φ=res[3]-360
if(|φ|>90)
exception: send error message, stop processing

call CFI orbit propagator in propagation mode for end of product *(1.5.1.2-0.4)*

If USE_INTERPOL == FALSE then
call **po_ppforb**
inputs: mode=PO_PROPAG, mjdr, xm, mjdp=JD1
outputs: pos, vel, acc
Else

call **po_interpol**
inputs : mode=PO_INTERPOLATE, mjdr0= JD1
outputs : pos, vel, acc
Endif

call CFI satellite to ground station visibility *(1.5.1.2-0.5)*
compute attitude perturbation att_error as per step 1.5.2.8
call **pp_stavis**
inputs: mjdp=JD1, pos, vel, acc, AOCS, att_error, datt = 0,
sta = {λ_{centre}, ϕ_{centre}, 0., -90.}
outputs: φ = res[3].

check satellite to scene centre azimuth between "ahead" and "back", raise exception processing if "ahead" *(1.5.1.2-0.6)*
if(φ>180) φ=res[3]-360
if(|φ|<90)
exception: send error message, stop processing

scene centre may be within MERIS swath, initialise search parameters *(1.5.1.2-4)*
t1 = JD0
t2 = JD1

begin recurrence to reach Scene Centre imaging time
(when satellite to target azimuth changes from "ahead" to "back")
do

call CFI orbit propagator in propagation mode for mid-time *(1.5.1.2-5)*
If USE_INTERPOL == FALSE then
call **po_ppforb**
inputs: mode=PO_PROPAG, mjdr, xm, mjdp=(t1+t2)/2
outputs: pos, vel, acc, λ_{SSP}=res[7], ϕ_{SSP}=res[8], γ=180-res[39]
Else
call **po_interpol**
inputs : mode=PO_INTERPOLATE, mjdr0=(t1+t2)/2
outputs : pos, vel, acc, λ_{SSP}=res[7], ϕ_{SSP}=res[8], γ=180-res[39]
Endif

call CFI satellite to ground station visibility *(1.5.1.2-6)*
compute attitude perturbation att_error as per step 1.5.2.8
call **pp_stavis**
inputs: mjdp=(t1+t2)/2, pos, vel, acc, AOCS, att_error, datt = 0,
sta = {λ_{centre}, ϕ_{centre}, 0., -90.}
outputs: φ = res[3]

check satellite to scene centre azimuth between "ahead" and "back", update bracketting times *(1.5.1.2-7)*
if(φ>180) φ=res[3]-360
if(|φ|<90)
t1 = (t1+t2)/2 *"ahead" case*
else
t2 = (t1+t2)/2 *"back" case*
while (|t2-t1|*86400000 ≥ DT^{FR})

determine central frame index within Level 0 product *(1.5.1.2-9)*

$f_{centre} = 1 + nint((t2 - (JD0 - DT\_frame^{FR}))*86400000/DT^{FR})$

determine first and last frame index within Level 0 product according to scene size, the first frame matching the Level 0-related tie point grid (1.5.1.2-10)

F1 = f_{centre} - (NF-1)/2

F1 = 1 + DF^{FR}*nint((F1-1)/DF^{FR})

F2 = F1 + NF - 1

compute corresponding times (1.5.1.2-11)

begin_JD = JD0 + (F1-1) * DT^{FR}/86400000

end_JD = begin_JD + (NF-1) * DT^{FR}/86400000

step 1.5.1.3 - Determine FR Across-Track Level 1B product limits

compute ground distance between SSP of Scene Centre frame and Scene Centre (1.5.1.3-1)

call **pl_geo_distance**

inputs: λ_{SSP}, ϕ_{SSP}, λ_{centre}, ϕ_{centre}, h=0

outputs: d, az1

compute distance between eastern tie point (full swath) and Scene Centre projection projection onto central frame taking account of azimuth (1.5.1.3-2)

J_centre = (NJ+1)/2

d' = (J_centre-1)*Dx_t - d*sin(az1+γ)

derive index of tie points bracketting Scene Centre (1.5.1.3-3)

k1 = 1 + int(d'/Dx_t)

k2 = k1 + 1

Exception processing: *requested scene centre is out of across-track swath*

If (k1 < 1) or (k2 > NJ)

stop processing, issue error message

Endif

End exception processing

check parity of number of tie points in Level 1B width (1.5.1.3-4)

if (mod(N_{TP},2) =1) then

central tie point exist and must be the closest among the 2 bracketting tie points, find it and derive index of first tie points in level 0 product (1.5.1.3-5)

if ((d' - (k1-1)*Dx_t) ≤ Dx_t/2) then

J1 = k1 - (N_{TP}-1)/2

else

J1 = k2 - (N_{TP}-1)/2

endif

else

derive index of first tie points in level 0 product (1.5.1.3-6)

J1 = k2 - N_{TP}/2

endif

Exception processing:

If J1 < 1

J1=1

Endif

End exception processing

compute corresponding pixel index within full swath (1.5.1.3-7)

first_tie_k = (J1-1)*DJ^{FR} + 1

derive index of last tie points in level 0 product (1.5.1.3-8)

$J2 = J1 + N_{TP} - 1$

Exception processing:

If J2 > NJ

J2 = NJ

J1 = J2 - NTP + 1

Endif

End exception processing

compute corresponding pixel index within full swath *(1.5.1.3-9)*

$last\_tie\_k = (J2-1)*DJ^{FR} + 1$

step 1.5.1.4 - Determine FR Across-Track extraction limits

compute pointing angle of tie point J1 *(1.5.1.4-1)*

call **pp_target**

inputs: idir=PP_GR_RAN, mjdp=t1, pos, vel, acc, AOCS, att_error, datt=0, azimuth=sign(J1-J_centre)*90°, elevation=90°,distance = |J1-J_centre|*Dx_t

output: ψ_1

initialise loop on pointings *(1.5.1.4-2)*

m = 2 ; k = 1;

compare tie point pointing with those of first pixel of MERIS modules *(1.5.1.4-3)*

while ($m \le Mt$ AND $\psi_1 \ge \psi^{FR}_{k,m}$)

m = m+1

end while

derive index of first module to extract *(1.5.1.4-4)*

first_module = m - 1

compute pointing angle of tie point #J2 *(1.5.1.4-5)*

call **pp_target**

inputs: idir=PP_GR_RAN, mjdp=t1, pos, vel, acc, AOCS, att_error, datt=0, azimuth=sign(J2-J_centre)*90°, elevation=90°, distance = |J2-J_centre|*Dx_t

output: ψ_2

initialise loop on pointings *(1.5.1.4-6)*

m = Mt-1 ; k = K^{FR};

compare tie point pointing with those of last pixel of MERIS modules *(1.5.1.4-7)*

while ($m \ge 1$ AND $\psi_2 \le \psi^{FR}_{k,m}$)

m = m-1

end while

derive number of modules to extract *(1.5.1.4-8)*

M = m + 2 - first_module

Reduced Resolution case

else

step 1.5.1.5 - Determine RR Along-Track Level 1B product limits

retrieve times of first and last frames from Work Order *(1.5.1.5-1)*

begin_JD = begin_time

end_JD = end_time

derive number of frames *(1.5.1.5-2)*

$NF = 1 + (end\_JD - begin\_JD) / DT^{RR}/86400000$

step 1.5.1.6 - Determine RR Across-Track Level 1B product limits

number of tie points and index of first one *(1.5.1.6-1)*

N_{TP} = NJ

corresponding column extreme indices and number *(1.5.1.6-2)*

first_tie_k = 1

last_tie_k = 1 + NJ*DJ^{RR}

NC = NC^{RR}

step 1.5.1.7 - Determine RR across track extraction limits

number of modules and index of first one *(1.5.1.7-1)*

M = Mt

first_module = 1

step 1.5.1.8 - Initialise Orbit Propagator, determine time at ascending crossing node and orbit period (in days)

extract state vector from Level 0 product *(1.5.1.8-1)*

extract Applicable_vector from Level 0 product

call CFI orbit propagator routine in init mode *(1.5.1.8-2)*

call **po_ppforb**

inputs: mode=PO_INIT, Applicable_Vector

outputs: CNT_JD=mjdr, xm, orbit_period=res[52]/86400

Exception processing:

In case of failure of po_ppforb call, i.e. if the returned status is not 0, then

Apply steps 1.5.2.3-0

call CFI Precision Orbit interpolation/propagation routine in init mode, determine time at ascending node and orbit period (in days) *(1.5.1.8-3)*

call **po_interpol**

inputs: mode=PO_INIT_FILE, choice, ndc, ndp, ner, doris_precise_file, doris_prelim_file, esoc_rest_file, mjdr0=JD0, mjdr1=JD1

outputs: orbit_period=res[52]/86400, CNT_JD=res[53]-orbit_period

Set flag USE_INTERPOL to TRUE

End exception processing

endif *end of product limits computation*

step 1.5.2 - Tie Points Location

At processing initialisation,

step 1.5.2.1- DELETED

step 1.5.2.2- DELETED

step 1.5.2.3- initialise orbit propagator for Consolidated Processing

if (Consolidated_processing AND NOT USE_INTERPOL) then

set po_interpol inputs according to State Vector File type and name *(1.5.2.3-0)*

doris_precise_file="" ; doris_prelim_file="" ; esoc_rest_file=""

switch VECTOR_SOURCE

case “DP”

ndc=1 ; ndp=0 ; ner=0

choice=PO_ONLY_DORIS_PRECISE

doris_precise_file=VECTOR_FILE ;
case “DI”
ndc=0 ; ndp=1 ; ner=0
choice=PO_ONLY_DORIS_PRELIMINARY
doris_prelim_file =VECTOR_FILE ;
case “FR”
ndc=0 ; ndp=0 ; ner=1
choice=PO_ONLY_ESOC_RESTITUTED
esoc_rest_file =VECTOR_FILE ;
end switch
call CFI Precision Orbit interpolation/propagation routine in init mode (1.5.2.3-1)
call **po_interpol**
inputs: mode=PO_INIT_FILE, choice, ndc, ndp, ner,
doris_precise_file, doris_prelim_file, esoc_rest_file,
mjdr0=begin_JD, mjdr1=end_JD
outputs: none
end if
DELETED *(replaced by 1.5.1.2-2 or 1.5.1.8-1, see the IMPORTANT NOTE above) (1.5.2.3-2)*
DELETED *(replaced by 1.5.1.2-3 or 1.5.1.8-2, see the IMPORTANT NOTE above) (1.5.2.3-3)*

step 1.5.2.2- tie frame selection; main loop of geo-location process
for (F = 1; F ≤ NF; F += DF)
compute time of current tie frame and apply time correction (1.5.2.2-1)
$T\_JD'_F = T\_JD_1 + (F-1)*DT/86400000 - DT\_frame$

step 1.5.2.4 - propagate orbit
propagate orbit using propagator selected according to the Consolidated switch (1.5.2.4-1)
if (Consolidated_processing OR USE_INTERPOL) then
call **po_interpol**
inputs : mode=PO_INTERPOLATE, mjdr0= $T\_JD'_F$;
outputs : pos, vel, acc
else
call **po_ppforb**
inputs : mode=PO_PROPAG, mjdr, xm, mjdp=$T\_JD'_F$;
outputs : pos, vel, acc
endif

step 1.5.2.8 - attitude perturbation
compute fraction of orbit period elapsed since ascending node (1.5.2.8-1)
rel_time = $T\_JD'_F$ - CNT_JD
retrieve corresponding bracketting data from attitude model data base (1.5.2.8-2)
scan the attitude error model data base to find i such that
Att_error_model[i].time ≤ rel_time < Att_err_model[i+1].time
Exception Processing:
If the number of elements of the attitude error model is equal to 1 or if there is no sample satisfying: rel_time < Att_err_model[i+1].time, **process exception as should be specified in AD7**.
End of Exception Processing
compute coefficient for linear interpolation with respect to time (1.5.2.8-3)

$$p = \frac{\text{rel\_time - Att\_error\_model[i].time}}{\text{Att\_error\_model[i+1].time - Att\_error\_model[i].time}}$$

proceed to linear interpolation at current time *(1.5.2.8-4)*

att_error = (1-p)*Att_error_model[i].rot + p *Att_error_model[i+1].rot

tie points location

for (J=first_tie_k; J ≤ last_tie_k; J += DJ)

step 1.5.2.6 - tie point distance to swath centre

compute tie point distance from swath centre *(1.5.2.6-1)*

J_centre = (NJ+1)/2

$$\text{dist}_{J,F} = \text{Dx\_t.}\left(1 + \frac{J-1}{DJ} - \text{J\_centre}\right)$$

step 1.5.2.7 - locate tie point on Earth

call CFI satellite-to-target pointing routine *(1.5.2.7-1)*

call **pp_target**

inputs: idir=PP_GR_RAN, mjdp=$T\_JD'_F$,

pos, vel, acc, AOCS, att_error, datt=0,

azimuth=sign($dist_{J,F}$)*90°, elevation=90°,

distance= $|dist_{J,F}|$

outputs : $\lambda_{J,F}$, $\phi_{J,F}$, $\theta_{vJ,F}$, $\varphi_{vJ,F}$, $\theta_{sJ,F}$, $\varphi_{sJ,F}$, $\psi_{J,F}$

end for *end of loop on tie points columns*

end for *end of loop on tie frames*

step 1.5.4 - Altitude retrieval and correction

for each tie point J,F(J in [first_tie_k,last_tie_k] step DJ, F in [1,NF] step DF)

step 1.5.4.1 - Altitude retrieval

retrieve altitude at tie point location from DEM *(1.5.4.1-1)*

$z_{J,F} = DEM(\lambda_{J,F}, \phi_{J,F})$

step 1.5.4.2 - Roughness retrieval

retrieve surface roughness at tie point location from DRM *(1.5.4.2-1)*

$\sigma z_{J,F} = DRM\ (\lambda_{J,F}, \phi_{J,F})$

step 1.5.4.3 - Altitude correction

compute across-track distance error due to non-zero altitude *(1.5.4.3-1)*

$dx = z_{J,F}.\ \tan\theta_{vJ,F}$

compute corresponding latitude correction *(1.5.4.3-2)*

$$dlat_{J,F} = \frac{dx\cos\varphi_{J,F}}{R_e}\cdot\frac{180}{\pi}$$

compute corresponding longitude correction *(1.5.4.3-3)*

$$dlon_{J,F} = \frac{dx.\sin\varphi_{J,F}}{R_e.\cos(\phi_{J,F})}\cdot\frac{180}{\pi}$$

end for ; end for

step 1.5.5 - Radiance Re-sampling

if(resampling_switch) then
for each product frame f
let F and F+DF be the previous and following tie frames ($F \le f \le F+DF$)

compute frame time and apply time correction (1.5.5.0-1)

$T\_JD'_f = T\_JD_1 + (f-1)*DT/86400000 - DT\_frame$

for each product pixel j', f ($j' \in$ [first_tie_k,last_tie_k])

compute column index relative to Level1b product limits (1.5.5.0-2)

j = j' - first_tie_k + 1

step 1.5.5.1 - Interpolate product pixel pointing

let J and J+DJ be the previous and following tie points columns ($J \le j' \le J+DJ$)

compute product pixel pointing with bi-linear interpolation (1.5.5.1-1)

$$\psi_{j,f} = \left(\frac{J+DJ-j'}{DJ}\right)\left(\frac{F+DF-f}{DF}\right)\psi_{J,F} + \left(\frac{J+DJ-j'}{DJ}\right)\left(\frac{f-F}{DF}\right)\psi_{J,F+DF} + \left(\frac{j'-J}{DJ}\right)\left(\frac{F+DF-f}{DF}\right)\psi_{J+DJ,F} + \left(\frac{j'-J}{DJ}\right)\left(\frac{f-F}{DF}\right)\psi_{J+DJ,F+DF}$$

step 1.5.5.2 - Find nearest

find nearest pointing angle within those of MERIS pixels (1.5.5.2-1)

find (k,m), MERIS pixel index <u>within extraction limits</u> such that:
$resamp\_pix_{k,m} = 1$ and $|\psi_{j,f} - \psi_{k,m}|$ is minimum

set product pixel Detector index accordingly (1.5.5.2-7)

$Detector_{j,f} = k+(m-1)*K$

if $|\psi_{j,f} - \psi_{k,m}| > max\_d\psi$ then

if out-of-swath, set flag "invalid" to TRUE (1.5.5.2-2)

Invalid_f = TRUE

if out-of-swath, set radiances to default values (all bands), reset Detector index (1.5.5.2-3)

$TOAR_{b,j,f} = 0$ for all b
$Detector_{j,f} = -1$

break the pixel loop (1.5.5.2-4)

process next pixel
else
if used[k,m] ≠ 0 then

if within swath but MERIS pixel already used, set "duplicate" flag to TRUE (1.5.5.2-5)

$Duplicated\_f_{j,f} = TRUE$
else

if within swath and MERIS pixel never used before, update "used" array (1.5.5.2-6)

used[k,m] = 1
end if
end if

step 1.5.5.3 - Retrieve frame offset

compute MERIS frame taking depointing into account (1.5.5.3-1)

$f' = f + \delta f_{k,m}$
if f' < 1 or f' > NF then

if out-of-swath, set "invalid" flag to TRUE (1.5.5.3-2)

$Invalid\_f_{j,f} = TRUE$ (outside imaged area) *if out-of-swath, set radiances to default values (all bands), reset Detector index* *(1.5.5.3-3)*

$TOAR_{b,j,f} = 0$ for all b

$Detector_{j,f} = -1$

break the pixel loop *(1.5.5.3-4)*

process next pixel

else

if (valid_frame_f[f']) then

step 1.5.5.4 - Resample to nearest neighbour

for all b in 1..B

within swath, resample radiance *(1.5.5.4-1)*

TOAR[b,j,f] = L[b,k,m,f']

within swath, resample dubious flag *(1.5.5.4-2)*

Dubious_f[b,j,f] = dubious_f[b,k,m,f']

within swath, resample saturated flag *(1.5.5.4-3)*

Saturated_f[b,j,f] = saturated_f[b,k,m,f']

within swath, resample cosmetic flag *(1.5.5.4-4)*

Cosmetic_f[b,j,f] = cosmetic_f[b,k,m,f']

end for

within swath, resample stray light risk flag *(1.5.5.4-5)*

Stray_f[j,f] = stray_f[k,m,f']

pixel is valid *(1.5.5.4-6)*

Invalid_f[j,f] = FALSE

else

MERIS frame f' corresponding to current pixel is invalid, resample radiance (set to default value by previous steps) *(1.5.5.4-7)*

for all b in 1..B

TOAR[b,j,f] = L[b,k,m,f']

end for

set Invalid flag for this pixel *(1.5.5.4-8)*

Invalid_f[j,f] = TRUE

end if

end if

end for *end of loop on product pixels columns*

end for *end of loop on product frames*

else

step 1.5.5.5 - Re-sampling disabled: copy MERIS frame into Product one

for each product frame f

compute frame time and apply time correction *(1.5.5.0-1)*

$T\_JD'_f = T\_JD_1 + (f-1)*DT/86400000 - DT\_frame$

if (valid_frame_f[f]) then

if (NC ≤ M*K) then

copy first NC pixels of MERIS frame, ignore product limits *(1.5.5.5-1)*

for all j in 1,NC

k = 1 + j%K

m = 1 + int(j/K)

for all b in 1..B

TOAR[b,j,f] = L[b,k,m,f]

Dubious_f[b,j,f] = dubious_f[b,k,m,f]

```
            Saturated_f[b,j,f] = saturated_f[b,k,m,f]
            Cosmetic_f[b,j,f] = cosmetic_f[b,k,m,f]
        end for
        Stray_f[j,f] = stray_f[k,m,f]
        Invalid_f[j,f] = FALSE
        Detector_{j,f} = k+(m-1)*K
    end for
else
```
copy all available pixels of MERIS frame into first pixels of product frame (1.5.5.5-2)
```
    for all j in 1,M*K
        k = 1 + j%K
        m = 1 + int(j/K)
        for all b in 1..B
            TOAR[b,j,f] = L[b,k,m,f]
            Dubious_f[b,j,f] = dubious_f[b,k,m,f]
            Saturated_f[b,j,f] = saturated_f[b,k,m,f]
            Cosmetic_f[b,j,f] = cosmetic_f[b,k,m,f]
        end for
        Stray_f[j,f] = stray_f[k,m,f]
        Invalid_f[j,f] = FALSE
        Detector_{j,f} = k+(m-1)*K
    end for
```
complete product frame with invalid pixels (1.5.5.5-3)
```
    for all j in M*K+1,NC
        TOAR_{b,j,f} = 0 for all b
        Invalid_f[j,f] = TRUE
        Detector_{j,f} = -1
    end for
  end if
else
```
corresponding MERIS frame is invalid, set whole frame to default (1.5.5.5-4)
```
    TOAR_{b,j,f} = 0 for all b
    Invalid_f[j,f] = TRUE
    Detector_{j,f} = -1
  endif
 end for
end if
```

step 1.5.6 - Sun glint risk flag

for each tie point J,F(J in [first_tie_k,last_tie_k] step DJ, F in [1,NF] step DF)

check Sun Glint condition for current tie point (1.5.6-1)

if (| $\theta_{sJ,F}$ - $\theta_{vJ,F}$ | ≤ glint_thr_zen) and (| 180 - |$\varphi_{sJ,F}$ - $\varphi_{vJ,F}$| | ≤ glint_thr_azi) then

for each product pixel j' in J..J+DJ-1, f in F..F+DF-1

if Sun Glint condition fulfilled set corresponding pixels "glint" flag to TRUE (1.5.6-2)

```
    j = j' -first_tie_k + 1      column index within product
    Glint_f[j,f] = TRUE
  end for
```

```
    end if
end for
```

8.3.4 - Accuracy requirements

All longitude and latitude comparisons with reference test values must be exact to the sixth significant digit.
Radiance and Sun and viewing angles comparisons with reference test values must be exact to the fifth significant digit.
All julian day comparisons with reference test values must be exact to the ninth significant digit.
All flags comparisons with reference test values must be exact.

8.3.5 - Product Confidence Data Summary

The following Product Confidence Data are included in the product :

- the type of orbit : precision orbit, or state vector extracted from L0 product, is reflected by the field "Vector Source" of the product MPH;
- at pixel level, the "duplicate" flag is set for all pixels which are duplicate of a neighbour
- at pixel level, the "invalid" flag is set for those pixels which could not be resampled from MERIS data (near product limits or in large gaps).

9 - MERIS Pixel Classification Algorithm

9.1. - Introduction

The last step of level 1 processing, before formatting data, consists of partitioning pixels in three classes relevant to the main compartments of level 2 processing, i.e. bright (including clouds), land and ocean, by assigning binary flags to each product pixel. This section describes the methods and algorithms proposed for the achievement of this task.

9.2. - Algorithm Overview

Based on a geo-location interpolated from values at the Tie Points, each pixel is assigned an *a priori* surface type, extracted from an atlas, through two Boolean flags :

- a flag noted **"Land"** (when true means : land, when false : ocean);
- a flag noted **"Coastline"** (true : coastline, false : not coastline).

Based on radiometry, another flag is affected to the pixel to identify **"Bright"** pixels which encompass a wide range of geo-physical categories including :

- clouds (full or partly cover above a pixel)
- thick aerosols
- bright land surfaces: sand, snow, ice
- bright water surfaces : Sun glint

A complete surface identification requires more complex modelling and falls in the scope of Level 2 processing.

9.3. - Algorithm Description

9.3.1. - Theoretical Description

9.3.1.1. - Physics of the Problem

9.3.1.1.1. - Land/ocean map

Knowledge of the geographical co-ordinates of a product pixel allows to address a data base of a priori classification. That data base (described in AD1) provides at any longitude, latitude, at a spatial resolution close to that of MERIS imaging, two attributes :

1. land : set to true when emerged land is found at the point; non-land pixels will be hereafter called "ocean" which may include lakes;
2. coastline : set to true when at the land /non-land boundary;

9.3.1.1.2. - Bright pixels screening

Bright pixels screening is based on the comparison of the pixel total TOA reflectance in a user-selected band with a threshold depending on the illumination/observation geometry.

9.3.1.2. - Mathematical Description of the Algorithm

The classification algorithm follows the logic in the functional breakdowns in figures 9.3.2-1 (top level), 9.3.2-2 and 9.3.2-3 below. It should be noted that Full and Reduced Resolution processing are identical.

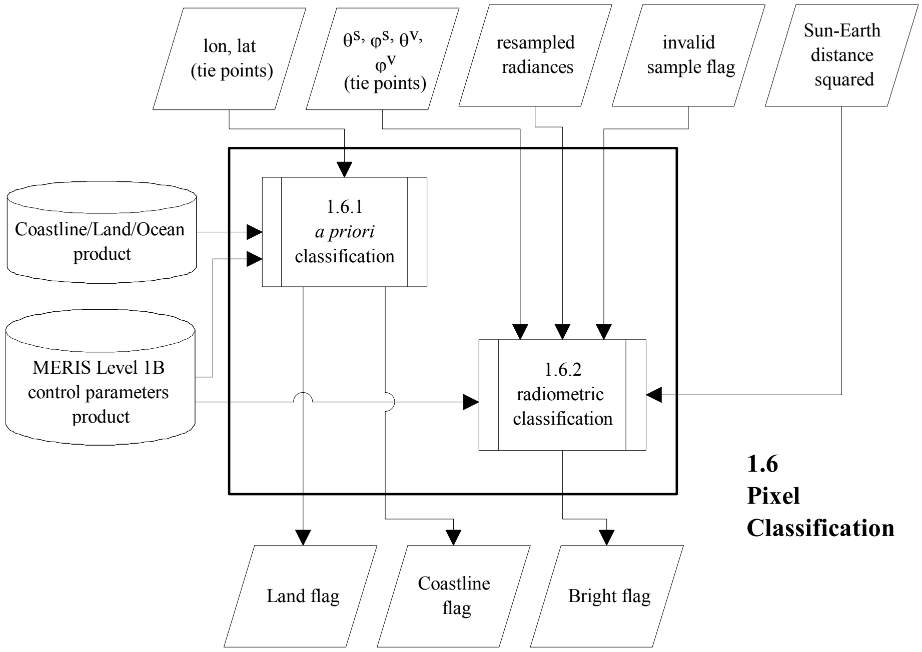


Figure 9.3.1.2-1 *: Functional breakdown of the pixel classification scheme.*

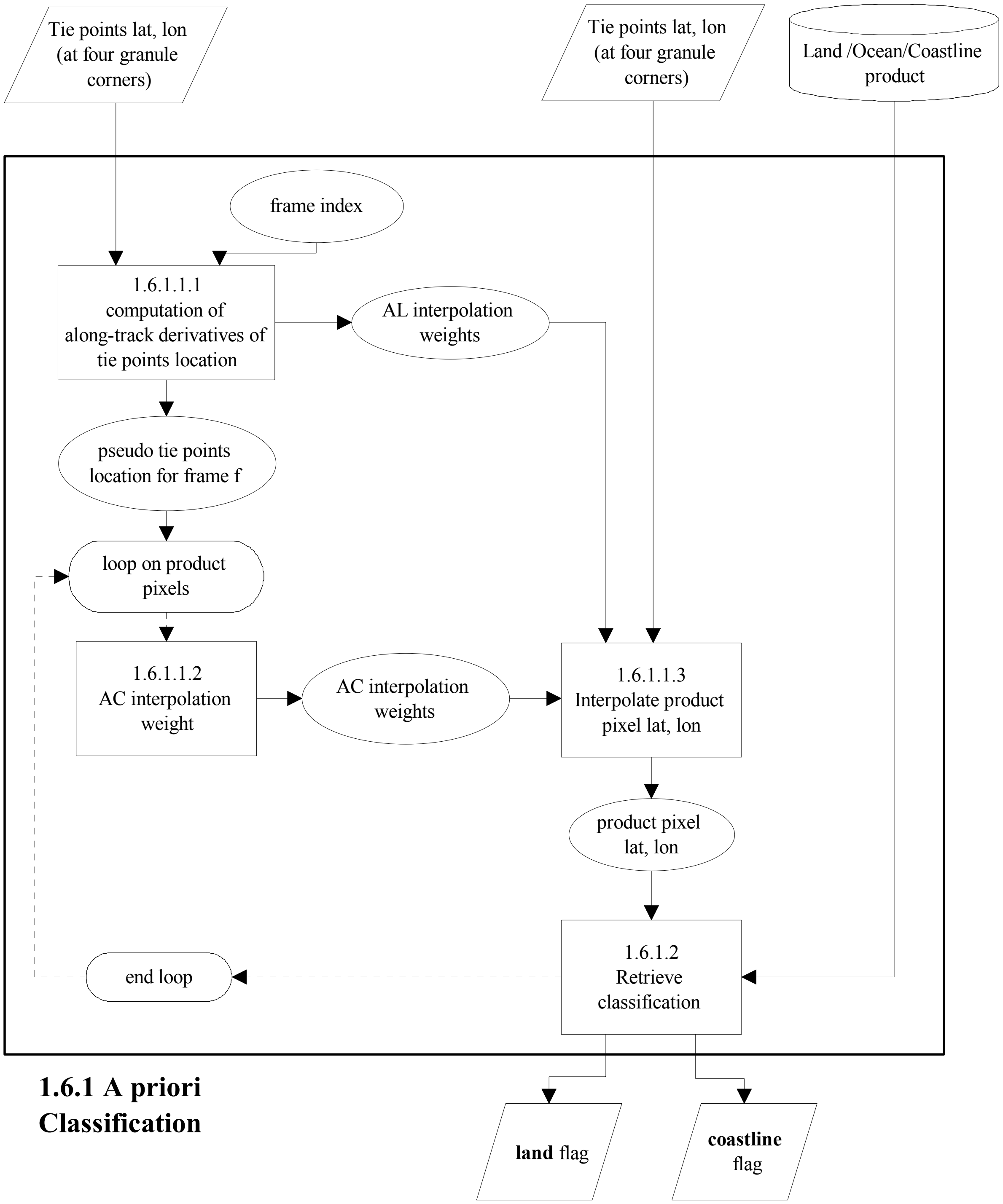


***Figure 9.3.1.2-2** : Functional breakdown of the a priori pixel classification scheme (step 1.6.1).*

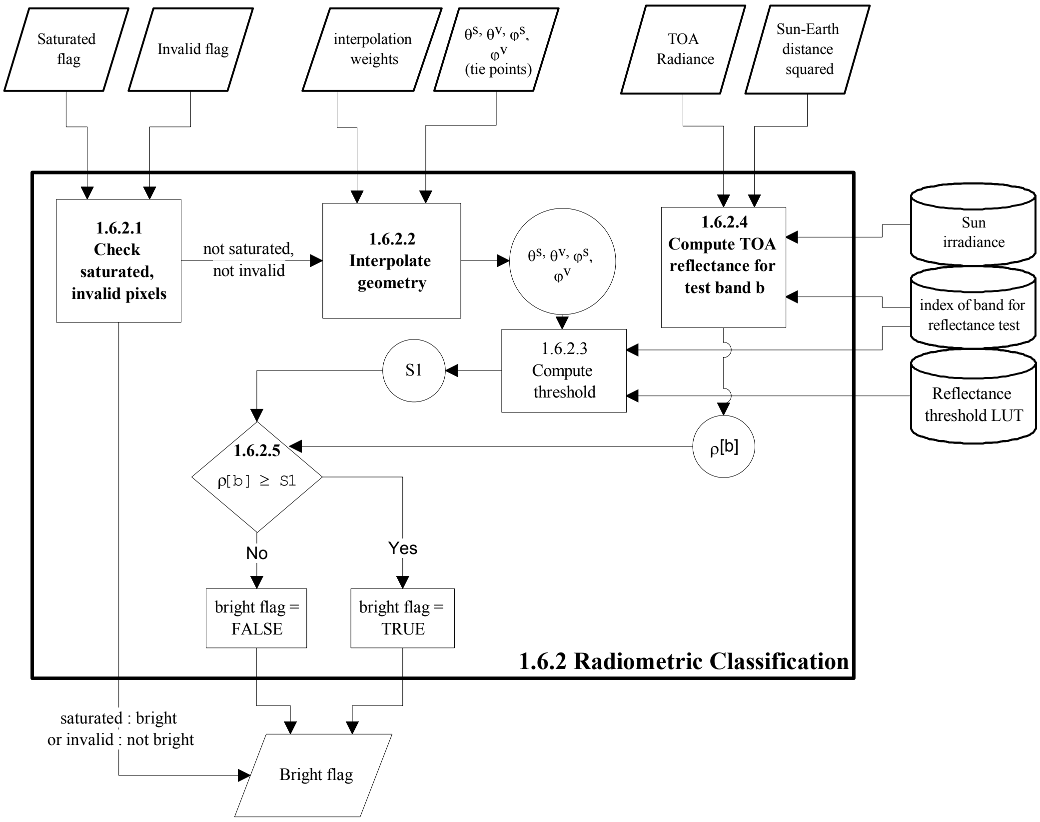


Figure 9.3.1.2-3. : Functional breakdown of the radiometric classification (step 1.6.2)

9.3.1.2.1. - *A priori* Classification Algorithm (1.6.1.)

That algorithm is performed at each frame of the product. The data and control flow within the algorithm are shown in fig. 9.3.1.2-2 above.

The *a priori* classification algorithm computes the Earth location of all product pixels by interpolation from the tie points, in order to retrieve classification information from a data base.

Its principle, as shown in fig. 9.3.1.2-7 below, is to compute the latitude and longitude of a product pixel using bi-linear interpolation on the co-ordinates of the four surrounding tie points and then to address a Land /Ocean data base using that location.

9.3.1.2.1.1. - Product Pixel Earth location

The Earth location $\{\lambda_{j,f}, \phi_{j,f}\}$ of a product pixel at column j, frame f is interpolated bi-linearly from latitude, longitude at the surrounding tie points :

$$X(J+j, F+f) = \left(\frac{J+\Delta J - j}{\Delta J}\right) \cdot \left[\left(\frac{F+\Delta F - f}{\Delta F}\right) \cdot X(J,F) + \left(\frac{f-F}{\Delta F}\right) \cdot X(J, F+\Delta F)\right]$$
$$+ \left(\frac{j-J}{\Delta J}\right) \cdot \left[\left(\frac{F+\Delta F - f}{\Delta F}\right) \cdot X(J+\Delta J,F) + \left(\frac{f-F}{\Delta F}\right) \cdot X(J+\Delta J, F+\Delta F)\right]$$

where X is either longitude or latitude,
(J,F) are the tie point co-ordinates verifying (j-ΔJ≤J≤j ; f-ΔF≤F≤f),
ΔF is the tie points frame spacing,
ΔJ is the tie points column spacing.

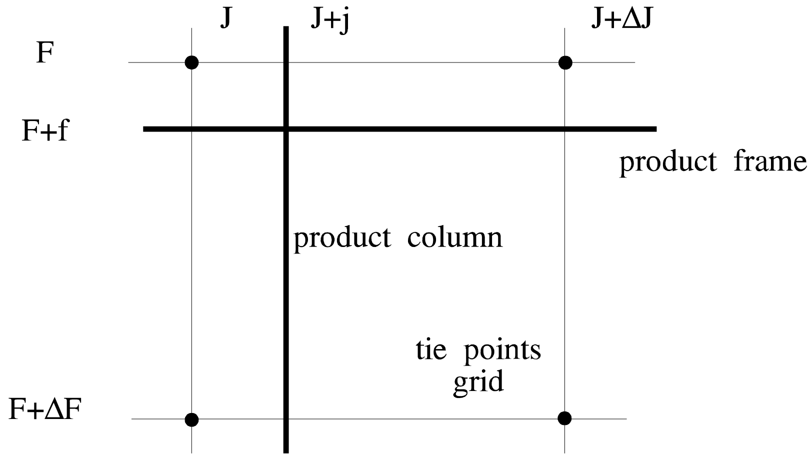


***Figure 9.3.1.2.-7** : product pixel location interpolation.*

9.3.1.2.1.2. - Land /Ocean mask retrieval

The MERIS pixel Earth location is transformed (by affine functions) into :

1. a line and column index referring to the low resolution (1degree by 1 degree) cell it belongs to,
2. an index corresponding to the mid-resolution (0.1 degree by 0.1 degree) cell within the low resolution cell listed above,
3. and an index corresponding to the low-resolution (0.01 degree by 0.01 degree) cell within the mid-resolution cell listed above.

Then for each atlas, the low resolution grid is addressed to retrieve the classification of the corresponding cell : True, False or Mix (-2, -1 or >0, see AD1 ; to be applied to Land or Coastline depending of the selected atlas). If classification is True or False, retrieval is completed ; if it is Mix , the returned value refers to a given record of the mid-resolution (0.1 degree by 0.1 degree) grid of the same atlas. This record contains 100 classification values corresponding to subdivision of the 1 degree by 1 degree cell which are addressed using the mid-resolution index. In the same way, cell classification can be True, False or Mix . If it is Mix, the returned value refers to the record of the high resolution grid corresponding to the current cell, record containing 100 values, one for each of the 0.01 degree by 0.01 degree sub-cells. The classification value for the current MERIS pixel is retrieved within the record using the high resolution index and can only be True or False.

9.3.1.2.2. - Radiometric classification (1.6.2.)

That algorithm is performed at each pixel of the product. The functional breakdown of the algorithm are shown in fig. 9.3.1.2-3 above.

9.3.1.2.2.1. - Saturation checks (1.6.2.1)

For each pixel, the calibrated radiance is compared for all the bands to the instrument saturation level. Any pixel with radiance of one or more band equal to or greater tan the instrument theoretical saturation level is classified as bright. Any pixel with its "Invalid" flag set is classified as non-bright. For all other pixels the processing continues as described below.

9.3.1.2.2.2 - Pixel Observation and Illumination geometry (1.6.2.2)

For each pixel, values of θ_S, θ_V, φ_S and φ_V are interpolated from tie point annotations as in 1.6.1.1. The azimuth difference $\Delta\varphi$ is computed from φ_S and φ_V.

9.3.1.2.2.3. - Reflectance computation (1.6.2.4)

The screening scheme applies to each pixel (whatever the resolution) and uses as input top-of-the-atmosphere radiance for the user-selected band b_{test}.corresponding to wavelength λ_{test} Reflectance ρ is calculated from :

$$\rho(\lambda_{test}) = \frac{\pi L_{TOA}(\lambda_{test})}{F'_0(\lambda_{test}) \cos \theta_s}$$

where $L_{TOA}(\lambda)$ is the top-of-atmosphere radiance measured by the sensor, $F'_0(\lambda)$ is the extraterrestrial solar irradiance, corrected for the data acquisition date and θ_S is the Sun zenith angle. Correction of the extraterrestrial solar irradiance relies on the squared Sun-Earth distance at a reference date ($Dsun_0^2$, read from a data base) and at the day of acquisition ($Dsun^2$, computed with the **pl_sun** CFI, see AD11) following :

$$F'_0(\lambda) = F_0(\lambda) . \frac{Dsun_0^2}{Dsun^2}$$

This correction is made for all bands, once per product processing as the variation of the distance during one segment is negligible.

9.3.1.2.2.4. - Reflectance threshold (1.6.2.3)

Thresholds S1, to be compared directly to ρ value is read from a look-up table as a function of θ_S, θ_V, and $\Delta\varphi$. Interpolation between grid nodes at (θ_S, θ_V, $\Delta\varphi$) is multi-linear.

9.3.1.2.2.5. - Bright Pixels discrimination (1.6.2.5)

Bright pixel screening relies on a thresholds applied to $\rho(\lambda_{test})$. The test (1.6.2.5) assumes that any pixel wit a TOA reflectance $\rho(\lambda_{test})$ higher than $S1(\theta_s, \theta_v, \Delta\varphi)$ denotes a surface pertaining to one of the following category: .

- clouds (full or partly cover above a pixel)
- thick aerosols
- bright land surfaces: sand, snow, ice
- bright water surfaces : Sun glint

9.3.2. - List of parameters

Indexing convention :

- subscript b stands for the band index, in range [1,B] or specified
- subscript j for the product pixel index, in range [1,NC]
- subscript f for the product line index, in range [1,NF]
- subscript J for the tie point column index in range [1,N_{TP}]
- subscript F for the tie point line index

Variable	Descriptive Name	T	U	Range - References
$TOAR^{FR}[b,j,f]$	FR resampled TOA radiance at pixel j,f	i	LU	from 1.5.5
$TOAR^{RR}[b,j,f]$	RR resampled TOA radiance at pixel j,f	i	LU	from 1.5.5
$Invalid\_f^{FR}[j,f]$	FR "invalid pixel" flag	i	dl	from 1.5.5
$Invalid\_f^{RR}[j,f]$	RR "invalid pixel" flag	i	dl	from 1.5.5
$\lambda[J,F]$	longitude at tie points	i	deg	from 1.5.2
$\phi[J,F]$	latitude at tie points	i	deg	from 1.5.2
$\theta_s[J,F]$	Sun zenith angle at tie points	i	deg	from 1.5.2
$\varphi_s[J,F]$	Sun azimuth angle at tie points	i	deg	from 1.5.2
$\theta_v[J,F]$	observation zenith angle at tie points	i	deg	from 1.5.2
$\varphi_v[J,F]$	observation azimuth angle at tie points	i	deg	from 1.5.2
begin_JD	UTC time of first Level 1b frame	i	jd	from 1.5.1
N_{TP}	number of tie points in Level 1B product	i	dl	from 1.5.1
NF	number of frames in Level1b product	i	dl	from 1.5.1
NC	number of columns in Level1b product	i	dl	from 1.5.1
ΔF^{FR}	tie points frame spacing	s	dl	
ΔF^{RR}	tie points frame spacing	s	dl	
ΔJ^{RR}	tie points column spacing	s	dl	
ΔJ^{FR}	tie points column spacing	s	dl	
Land_Sea_Map.land	A priori classification atlas structure, land/ocean field	s	dl	True/False "Land" and "Coastline" flags
Land_Sea_Map.coast	A priori classification atlas, true/false coastline field	s	dl	True/False "Land" and "Coastline" flags
b_{test}	band index for reflectance test	s	dl	
$class\_thr\_t[\theta_s,\theta_v,\Delta\varphi]$	Reflectance Threshold look-up table	s	dl	
Sat_rad_b	Saturation radiance values	s	LU	b=1,...,B
F_{0b}	Extra-terrestrial Sun irradiance at reference date	s	EU	
$Dsun0^2$	Square of Sun-Earth distance at reference date	s	m2	
$Dsun^2$	Square of Sun-Earth distance	c	m2	
$\lambda[j,f]$	longitude at pixel j,f	c	deg	interpolated
$\phi[j,f]$	latitude at pixel j,f	c	deg	interpolated
$\theta s[j,f]$	Sun zenith angle at pixel j,f	c	deg	interpolated
$\varphi s[j,f]$	Sun azimuth angle at pixel j,f	c	deg	interpolated
$\theta v[j,f]$	observation zenith angle at pixel j,f	c	deg	interpolated
$\varphi v[j,f]$	observation azimuth angle at pixel j,f	c	deg	interpolated
$\Delta\varphi$	absolute azimuth difference	c	deg	$\Delta\varphi \in [0,180]$
p	along-track interpolation weight	c	dl	
q	across-track interpolation weight	c	dl	
saturated	flag set when one of the bands used by the algorithm is saturated	c	dl	
$\rho T[b,j,f]$	Pixel TOA reflectance	c	dl	$b = b_{test}$
S1	Threshold for test 1.6.2.6.1	c	dl	

Table 9.3.2-1 : Parameters used in the pixel classification algorithm

Variable	Descriptive Name	T	U	Range - References
$F'_0[b]$	Extra-terrestrial Sun irradiance	o	EU	to 1.8
$Coast\_f^{FR}[j,f]$	FR pixels coastline classification flag	o	dl	Boolean; to 1.8
$Coast\_f^{RR}[j,f]$	RR pixels coastline classification flag	o	dl	Boolean; to 1.8
$Land\_f^{FR}[j,f]$	FR pixels land/ocean classification flag	o	dl	Boolean; to 1.8
$Land\_f^{RR}[j,f]$	RR pixels land/ocean classification flag	o	dl	Boolean; to 1.8
$Bright\_f^{FR}[j,f]$	FR pixels bright classification flag	o	dl	Boolean; to 1.8
$Bright\_f^{RR}[j,f]$	RR pixels bright classification flag	o	dl	Boolean; to 1.8

Table 9.3.2-1 : Parameters used in the pixel classification algorithm (cont.)

9.3.3. - Equations

NOTES:

1. FR and RR processing being identical, the superscript RR or FR of the parameters will be omitted in all equations.
2. for clarity, the subscript j,f may be omitted from the equations written for each pixel.
3. in equations 1.6.1.2.-1 and 1.6.1.2.-2, the land/sea and coastline maps are assumed uncompressed for clarity but this must not be taken as a coding specification : choices for maps data management, including data decompression, are matters of implementation.

for each product frame f

Step 1.6.1.1.1 - Tie points column interpolation

let F and F+DF be the previous and following tie frames (F ≤ f ≤ F+DF)

$$p = (F+\Delta F - f)/\Delta F \qquad (1.6.1.1.1\text{-}1)$$

for each product pixel (j,f), j ∈ [1,NC]

Step 1.6.1.1.2 - AC Interpolation weight

let J and J+ΔJ be previous and following tie points columns (J ≤ j ≤ J+ΔJ)

$$q = (J+\Delta J - j)/\Delta J \qquad (1.6.1.1.2\text{-}1)$$

Step 1.6.1.1.3 - MERIS pixel Earth location

interpolate longitude *(1.6.1.1.3-1)*

$$\lambda_{j,f} = p.q.\lambda_{J,F} + p.(1-q).\lambda_{J+\Delta J,F} + (1-p).q.\lambda_{J,F+\Delta F} + (1-p).(1-q).\lambda_{J+\Delta J,F+\Delta F}$$

interpolate latitude *(1.6.1.1.3-2)*

$$\phi_{j,f} = p.q.\phi_{J,F} + p.(1-q).\phi_{J+\Delta J,F} + (1-p).q.\phi_{J,F+\Delta F} + (1-p).(1-q).\phi_{J+\Delta J,F+\Delta F}$$

Step 1.6.1.2 - Land /ocean mask retrieval

$$Land\_f_{j,f} = Land\_Sea\_Map.land[\lambda_{j,f}, \phi_{j,f}] \qquad (1.6.1.2\text{-}1)$$

$$Coast\_f_{j,f} = Land\_Sea\_Map.coastline[\lambda_{j,f}, \phi_{j,f}] \qquad (1.6.1.2\text{-}2)$$

Step 1.6.2.1 Test saturation, invalid flags

```
if (Invalid_f[j,f]) then
  Bright_f_{j,f} = FALSE                                        (1.6.2.1-1)
else
  saturated = FALSE                                             (1.6.2.1-2)
  for all b
    saturated = saturated OR TOAR[b,j,f] > Sat_Rad_b            (1.6.2.1-3)
  endfor
  if (saturated) then
    Bright_f_{j,f} = TRUE                                       (1.6.2.1-4)
  else
```

Step 1.6.2.2 Geometry interpolation

interpolate Sun and viewing angles at current pixel from tie points values

$\theta_{sj,f} = p.q.\theta_{sJ,F} + p.(1-q).\theta_{sJ+\Delta J,F} + (1-p).q.\theta_{sJ,F+\Delta F} + (1-p).(1-q).\theta_{sJ+\Delta J,F+\Delta F}$ *(1.6.2.2-1)*

$\varphi_{sj,f} = p.q.\varphi_{sJ,F} + p.(1-q).\varphi_{sJ+\Delta J,F} + (1-p).q.\varphi_{sJ,F+\Delta F} + (1-p).(1-q).\varphi_{sJ+\Delta J,F+\Delta F}$ *(1.6.2.2-2)*

$\theta_{vj,f} = p.q.\theta_{vJ,F} + p.(1-q).\theta_{vJ+\Delta J,F} + (1-p).q.\theta_{vJ,F+\Delta F} + (1-p).(1-q).\theta_{vJ+\Delta J,F+\Delta F}$ *(1.6.2.2-3)*

$\varphi_{vj,f} = p.q.\varphi_{J,F} + p.(1-q).\varphi_{J+\Delta J,F} + (1-p).q.\varphi_{J,F+\Delta F} + (1-p).(1-q).\varphi_{J+\Delta J,F+\Delta F}$ *(1.6.2.2-4)*

$\Delta\varphi = | \varphi_s - \varphi_v |$ *(1.6.2.2-5)*

if $(\Delta\varphi > 180)$ then $\Delta\varphi = 360 - \Delta\varphi$ *(1.6.2.2-6)*

Step 1.6.2.3 – Reflectance Threshold

The threshold is read from look-up table :

S1 = class_thr_t **interp**: (θ_s, θ_v, $\Delta\varphi$) *(1.6.2.3-1)*

Step 1.6.2.4 – Reflectance

DELETED *(1.6.2.4-1)*

Correct extra-terrestrial irradiance for current day (once for the whole product) *(1.6.2.4-2)*

```
if( j=1)
  call pl_sun input: begin_JD, output: sun_pos
  Dsun^2 = || sun_pos ||^2
  for all b
    F'_0[b] = F_0[b].Dsun_0^2/Dsun^2
  end for
endif
```

compute reflectance $\rho_T[b_{test}]$:

$\rho_T[b_{test}] = \pi.TOAR[b_{test},j,f] / (F'_0[b_{test}].\cos\theta_s)$ *(1.6.2.4-3)*

Step 1.6.2.5 - Tests Reflectance against Threshold

if $\rho_T[b_{test}] \geq S1$ then

```
          Bright_f_{j,f} = TRUE                    (1.6.2.5-1)
        else
          Bright_f_{j,f} = FALSE                   (1.6.2.5-2)
        endif
```

end of band saturation tests :

```
      endif
```

end of invalid pixel test :

```
    endif
```

end of loop on columns :

```
  endfor
```

end of loop on frames

```
endfor
```

9.3.4. - Accuracy Requirements

All comparisons of classification flags with reference test values shall be exact.

9.3.5. - Product Confidence Data Summary

N/A

10. - External Data Assimilation Algorithm

10.1. -Introduction

This chapter describes the processing to be applied to external environment data for assimilation into the MERIS Level 1 processing system, and for annotation of the MERIS Level 1b product.

10.2. - Algorithm Overview

External environment data relevant to the processing of MERIS Level 1b product are stored in a data base at given spatial and temporal resolutions. Environment data, extracted from the data base for the time closest to the MERIS product time, are spatially interpolated to location of the product tie points and submitted to annotation.

10.3. - Algorithm Description

10.3.1. - Theoretical Description

10.3.1.1. - Physics of The Problem

10.3.1.1.1 - External data requirements

MERIS Level 2 processing requires knowledge of :

- atmosphere pressure at mean sea level (everywhere);
- wind speed and direction at sea surface level (over ocean);
- total ozone column contents (everywhere);
- relative humidity (over ocean);

at the time and location of every pixel.

Level 1B processing is in charge of assimilating these quantities for every tie point. Simple interpolation (see §8 above) is then adequate to derive these quantities at every pixel.

These parameters are derived from dedicated models of the environment, fed by measurements (including space-borne remote sensing). Models do not in general provide parameter data sets contemporary and co-located with the MERIS samples; interpolation is necessary. Also, such models are able to provide a global prediction of a future situation (hereafter called global forecast) as well as a global view of a past situation, consolidated with observation data such as in situ measurements and remote sensing data (global analysis). At the time of writing this report :

- numerical weather prediction models routinely provide global analyses and forecasts of pressure, wind speed and direction at 10m (expressed as u and v components of the wind vector, see note 1), relative humidity at 1000 hPa (see note 2 below); we have taken as a representative candidate the model operated by the European Centre for Medium-term Weather Forecast (ECMWF), located at Reading (UK);

- there seems to be no well established for the global short-term prediction of the total ozone column contents. On the other hand, the Total Ozone Mapping Spectrometer instrument series on board of the Nimbus satellites has been providing total ozone measurements for years so that modelling seems feasible.

The following assumptions are taken in the prospect of the ENVISAT-1 mission starting in 1999 (**bold face** denotes capabilities not yet implemented):

1. global forecasts are delivered operationally by ECMWF for Pressure at mean sea level, wind at 10m u and v components, relative humidity at 1000 hPa, **total ozone**;
2. global analyses are distributed operationally by ECMWF for Pressure at mean sea level, wind at 10m u and v components, relative humidity at 1000 hPa, **total ozone**;
3. analyses and forecasts (also called "meteo products") cover the whole globe with a bi-dimensional grid which is the same for all, provides a spatial resolution of approximately 55 km and is described in AD9;
4. analyses and forecasts are generated every six hours, and distributed every 24 hours, with the following timeline :

UT date	day n-1				day n				day n+1...		
& time	00:00	06:00	12:00	18:00	00:00	06:00	12:00	18:00	00:00	06:00	...
generation	analysis	analysis	analysis	analysis	forecast	forecast	forecast	forecast	forecast		
distribution					between 00:00 and 06:00						
generation					analysis	analysis	analysis	analysis	forecast	forecast	...
distribution									between 00:00 and 06:00		

It is assumed that the products described in AD9 are available as a unique and complete set of files, corresponding to the best available at the request time.
If any of the file is not available, or if all files do not correspond to the same data and time, process is stopped and an error report is sent.

Note 1 : the u and v components of the wind correspond in principle to the zonal (Easterly) and meridional (Northerly) directions.
Note 2 : relative humidity is distributed for several pressure levels, the 1000 hPa level, lowest level, is selected.

10.3.1.1.2 - ECMWF Grids

ECMWF data are distributed on either regular or Gaussian latitude-longitude grids. The selected one is regular with a latitude-longitude step of 1° (see AD10).

The parameters that can be found in ECMWF file are :
- The initial value of longitude λ_0 and of latitude ϕ_0
- The latitude step $\Delta\phi$ and the longitude step $\Delta\lambda$
- The number of latitude nodes $nmax\phi$ and of longitude nodes $nmax\lambda$

Thus a node $(n\phi,n\lambda)$ of the latitude-longitude grid have the index $n=(n\phi-1)*nmax\lambda+n\lambda$ in the spatial grid array and its co-ordinates are given by :

- $\phi=\phi_0+(n\phi-1)\Delta\phi$
- and $\lambda=\lambda_0+(n\lambda-1)\Delta\lambda$

Note that n, $n\phi$ and $n\lambda$ run from 1 (not 0).

10.3.1.2. - Mathematical Description of Algorithm

The functional breakdown of the algorithm is shown in figure 10.3.1.2-1 below.

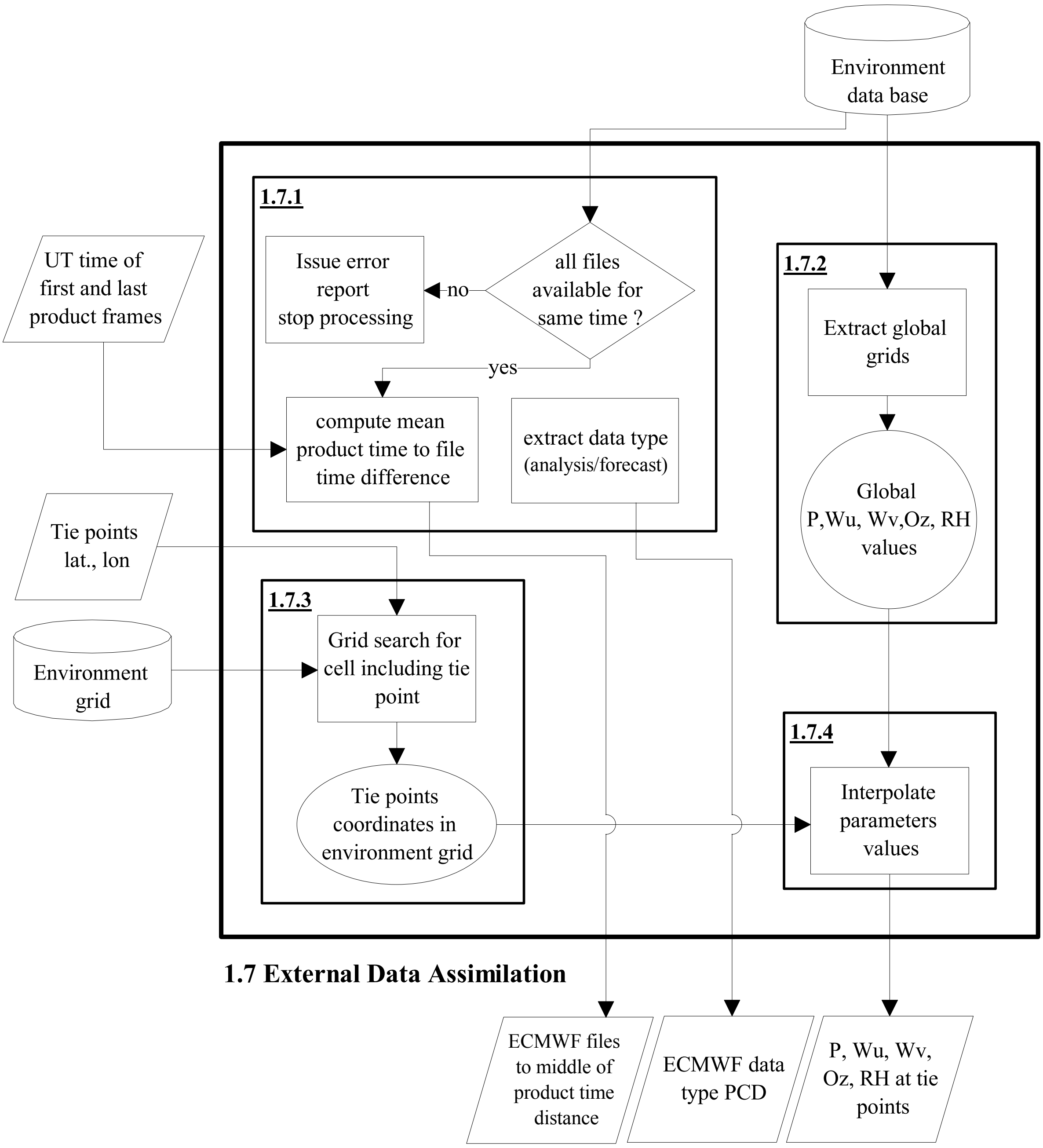


Figure 10.3.1.2-1 : External Data Assimilation functional block diagram

At initialisation, the MERIS level 1b processing checks for ECMWF files availability. If a file is not found, the processing is stopped and a error report is issued.

If processing goes on, PCDs will reflect data quality level :
1. ECMWF_DT_PCD will reflect the difference between product time and slice time;
2. ECMWF_TYPE_PCD will reflect the quality of the data : analysis or forecast.

Then for each tie point :
1. the coordinates of the four environment spatial grid enclosing the tie point are computed;
2. the parameters P, Wu, Wv, Oz, Rh are extracted at the four grid points
3. their values are spatially interpolated at the tie point location by a bi-linear method and copied to the product annotation.

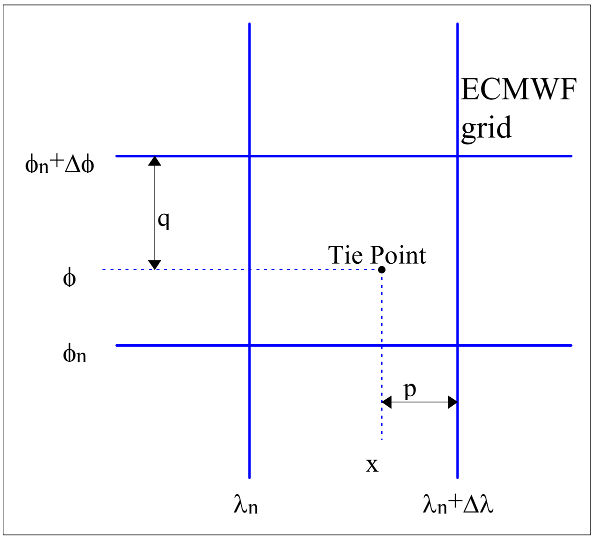


Figure 10.3.1.2-2 : geometry of tie point annotation interpolation

10.3.2. - List of Variables

Variable	Descriptive Name	T	U	Range - References
N_{TP}	number of tie pointsper frame in Level 1B product	i	dl	from 1.5.1
NF	number of frames in Level1b product	i	dl	from 1.5.1
begin_JD	UTC time of first product frame	i	jd	from 1.5.1
end_JD	UTC time of last product frame	i	jd	from 1.5.1
ϕ[J,F]	Latitude at tie point J,F	i	deg	from 1.5.2
λ[J,F]	Longitude at tie point J,F	i	deg	from 1.5.2
weather_product	Incoming weather product	s	-	from operational numerical weather prediction centre (ECMWF)
Weather_grid	Spatial sampling grid for weather products	s	-	weather_grid_type
T_ECMWF	Time of used weather products	s	jd	
kind	Kind of weather product used	s	dl	forecast or analysis
λ_0	longitude of first ECMWF grid point	s	deg.	
ϕ_0	latitude of first ECMWF grid point	s	deg.	
$\Delta\lambda$	ECMWF grid longitude steps	s	deg.	
$\Delta\phi$	ECMWF grid latitude steps	s	deg.	
P_db[loc]	Discretised global field of pressure at mean sea level	s	hPa	Environment data base, loc: index in ECMWF grid
Wu_db[loc]	Discretised global field of wind at 10m u component	s	$m.s^{-1}$	
Wv_db[loc]	Discretised global field of wind at 10m v component	s	$m.s^{-1}$	
Oz_db[loc]	Discretised global field of total ozone	s	$kg.m^{-2}$	
Rh_db[loc]	Discretised global field of relative humidity at 1000 hPa	s	%	
Oz_conv	Convertion factor from $kg.m^{-2}$ to DU for total ozone	s	$kg.m^{-2}$ DU^{-1}	Hard coded : value 4.6696.10^{4}
x_0	longitude of first grid point	c	deg	
y_0	latitude of first grid point	c	deg	
Δx	grid longitude step	c	deg	
Δy	grid latitude step	c	deg	
x	longitude of tie point J,F	c	deg	
y	latitude of tie point J,F	c	deg	
ilon	greatest grid longitude index "before" x	c	dl	before wrt grid variation direction
ilat	greatest grid latitude index "before" y	c	dl	before wrt grid variation direction
x1	greatest ECMWF grid longitude £ x	c	deg	
y1	greatest ECMWF grid latitude £ y	c	deg	
loc_k	grid indices of 4 ECMWF grid points closest to tie point J,F	c	dl	k:1..4
p,q	interpolation weights	c	dl	k:1..3
ECMWF_TYPE_PCD	ECMWF quality (forecast, analysis) PCD	o	dl	to 1-8
P_tie[J,F]	Mean sea level Pressure at tie point J,F	o	hPa	to 1.8
Wu_tie[J,F]	Wind u component at tie point	o	$m.s^{-1}$	to 1.8
Wv_tie[J,F]	Wind v component at tie point	o	$m.s^{-1}$	to 1.8
Oz_tie[J,F]	Total Ozone at tie point	o	DU	to 1.8
RH_tie[J,F]	Relative Humidity at tie point	o	%	to 1.8

Table 10.3.2-1 : Parameters used in the External Data Assimilation algorithm

10.3.3. - Equations

Step 1.7.1 - Check data availability

check availability of environment files (1.7.1-1)

```
if all files has been found then
  if not all files have the same time T_ECMWF then
    send error report                                    (1.7.1-2)
    stop processing                                      (1.7.1-3)
  endif
  retrieve kind (analysis/forecast) from data base slice (1.7.1-4)
else
  send error report                                      (1.7.1-5)
  stop processing                                        (1.7.1-6)
end if
```

NOTE : It is assumed that external software GRIBEX from ECMWF is available to retrieve T_ECMWF, kind (analysis/forecast). Detail Interfaces are provided in 10.3.6 below.

Equations 1.7.1-7 to 1.7.1-9 deleted

set data type PCD according to `kind` *:*

```
if (kind==forecast) then
  ECMWF_TYPE_PCD=0;                                      (1.7.1-10)
else
  ECMWF_TYPE_PCD=1;                                      (1.7.1-11)
endif
```

Extract available environment information :

Step 1.7.2 - Load environment data

`Load P_db, Wu_db, Wv_db, Oz_db and Rh_db` *(1.7.2-1)*

NOTE : It is assumed that external software from ECMWF, is available to perform that function. Detail Interfaces are provided in section 10.3.6 below.

let $x_0 = \lambda_0$; $y_0 = \phi_0$; $\Delta x = \Delta\lambda$; $\Delta y = \Delta\phi$ *(1.7.2-2)*

Loop on tie points

for each product tie point J,F

let $y = \phi_{J,F}$; *(1.7.2-3)*

if ($\lambda_{J,F} \geq 0$) then

$x = \lambda_{J,F}$ *(1.7.2-4)*

else

$x = 360 + \lambda_{J,F}$ *(1.7.2-5)*

endif

Step 1.7.3 - Compute tie point co-ordinate in data grid

compute the index of the four grid points surrounding the tie point : loc_i , i=1,..,4 : *(1.7.3-1)*

ilat = int((y-y_0)/Δy)

make sure we have another parallel for interpolation :

if(ilat == nmaxϕ -1) ilat--

ilon = int((x-x_0)/Δx)

loc_1 = ilat * nmaxλ +ilon

loc_2 = loc_1 + 1

check for Greenwich Meridian crossing :

if(ilon==nmaxλ-1) loc_2 -= nmaxλ

loc_3 = loc_1 + nmaxλ

loc_4 = loc_2 + nmaxλ

compute greatest grid column longitude lower than x: *(1.7.3-2)*

$x_1 = x_0 + ilon*\Delta x$

compute greatest grid row latitude lower than y: *(1.7.3-3)*

$y_1 = y_0 + ilat*\Delta y$

compute corresponding interpolation weights p & q :

$p = (x_1 + \Delta x - x) / \Delta x$ *(1.7.3-4)*

$q = (y_1 + \Delta y - y) / \Delta y$ *(1.7.3-5)*

Step 1.7.4 - Interpolate data to tie point co-ordinates

interpolate annotation products at tie point :

```
P_tie[J,F]   = p*q*P_db[loc1]       + (1-p)*q*P_db[loc2]
             + p*(1-q)*P_db[loc3]   + (1-p)*(1-q)*P_db[loc4]          (1.7.4-9)
Wu_tie[J,F]  = p*q*Wu_db[loc1]      + (1-p)*q*Wu_db[loc2]
             + p*(1-q)*Wu_db[loc3]  + (1-p)*(1-q)*Wu_db[loc4]         (1.7.4-10)
Wv_tie[J,F]  = p*q*Wv_db[loc1]      + (1-p)*q*Wv_db[loc2]
             + p*(1-q)*Wv_db[loc3]  + (1-p)*(1-q)*Wv_db[loc4]         (1.7.4-11)
OZ_tie[J,F]  = p*q*OZ_db[loc1]      + (1-p)*q*OZ_db[loc2]
             + p*(1-q)*OZ_db[loc3]  + (1-p)*(1-q)*OZ_db[loc4]         (1.7.4-12)
Rh_tie[J,F]  = p*q*RH_db[loc1]      + (1-p)*q*RH_db[loc2]
             + p*(1-q)*RH_db[loc3]  + (1-p)*(1-q)*RH_db[loc4]         (1.7.4-13)
OZ_tie[J,F]  = OZ_tie[J,F] * Oz_conv                                  (1.7.4-14)

end for                                          end of loop on tie points
```

10.3.4. - Accuracy Requirements

P_tie shall be computed with an accuracy of 0.1 hPa.
Wu_tie shall be computed with an accuracy of 0.1 $m.s^{-1}$.
Wv_tie shall be computed with an accuracy of 0.1$m.s^{-1}$.
Oz_tie shall be computed with an accuracy of 1 DU.
RH_tie shall be computed with an accuracy of 1 %.

10.3.5. - Product Confidence Data Summary

ECMWF_DT_PCD is an integer parameter, set to the time difference between the ECMWF product and the MERIS product when that difference is above 6 hours (in 6 hours unit), 0 otherwise.

ECMWF_TYPE_PCD is an integer parameter, set to 0 if the ECMWF product is a forecast, 1 if it is an analysis.

10.3.6 - Interfaces with ECMWF GRIBEX software

A summary of the main keys needed to access ECMWF data through the GRIBEX routines is given below. However the reader is refered to RD15 for more details, particularly for i/o parameters sizing. Access to data need four elementary functions corresponding to : opening and closing a file, read data, and decode data.

Function pbopen : open a weather product file

Argument number	Parameter	I/O	u
1	file identifier	o	-
2	file name	i	-
3	open flag ="R" (read)	i	-
4	error flag	o	-

Table 10.3.6-1 : Software interface with pbopen

Function pbclose : close a weather product file

Argument number	Parameter	I/O	u
1	file identifier	i	-
2	error flag	o	-

Table 10.3.6-2 : Software interface with pbclose

Function pbgrid : read gridded data

Argument number	Parameter	I/O	u
1	file identifier	i	-
2	GRIB data : kgrib	o	-
3	size of kgrib : kleng	i	-
4	number of data in kgrib	o	-

Table 10.3.6-3 : Software interface with pbgrid

Function gribex : extract gridded data

Argument number	Parameter	I/O	u
1	product definition section : ksec1	o	-
2	grid description section : ksec2	o	-
3	grid description section : dummy	o	-
4	bitmap section : dummy	o	-
5	bitmap section : dummy	o	-
6	binary data section : dummy	o	-
7	data values : psec4	o	-
8	number of data values in psec4: klenp	o	-
9	GRIB data : kgrib	i	-
10	size of kgrib : kleng	i	-
11	number of data in kgrib	i	-
12	mode flag : hoper (='D')	i	-
13	error flag : kret (=0)	i/o	-

Table 10.3.6-4 : Software interface with gribex

The useful elements of ksec1 are given by the following table :

Element	Contents
6	Parameter indicator
7	level type indicator
8	pressure level *
11	Month
12	Day
13	Hour
14	Minute

* : when applicable

Table 10.3.6-5 : key parameters for product description

The useful elements of ksec2 are given by the following table :

Element	Contents
2	Number of longitudes
3	Number of latitudes
4	Latitude of the first grid point
5	Longitude of the first grid point
7	Latitude of the last grid point
8	Longitude of the last grid point
9	Latitude step
10	Longitude step

Table 10.3.6-6 : key parameters for grid description

Notes :

longitude and latitude values and steps are given in millidegrees ;
steps are absolute values and must be affected by the sign of (*value of last point - value of first point*).

The following table gives the data layout in the psec4 vector which contains the values of the parameter defined in ksec1(6).

Lat. 1			Lat. 2			...etc.		
Long 1	Long 2	...etc.	Long 1	Long 2	...etc.	Long 1	Long 2	...etc.

Table 10.3.6-7 : Data layout in psec4 vector.

Important Note :
The relative humidity file, shared by different instruments, contains data at several pressure levels. In consequence humidity data needed for MERIS processing cannot be accessed through a single call to the functions pbgrig and gribext, as it is the case for all other files. As each call in sequence allows access to a whole level, pbgrib / gribext must be called as many time as necessary to reach the 1000.0 hPa level ; check must be done on the 8^{th} element of ksec1 (see table 10.3.6-4 above).

11. - MERIS Level 1b Product Formatting Algorithm

11.1. -Introduction

This chapter describes the processing to be applied to parameters used or created during the MERIS Level 1 processing, to generate the MERIS Level 1b products.

11.2. - Algorithm Overview

MERIS processed data samples corresponding annotations and flags are collected from previous steps and formatted according to Level 1b product description in AD1. Per sample flags are merged into per pixel flags, collapsing the spectral dimension.

11.3. - Algorithm Description

11.3.1 - Theoretical Description

11.3.1.1 - Physics of The Problem

The MERIS Level 1b product is composed of : the Main Product Header (MPH), the Specific Product Header (SPH), one Global Annotation Data Sets (GADS), two Annotation Data Sets and sixteen Measurement Data Sets. The MPH allows to identify the product and some of its main characteristics.

The SPH contains references to external data files and Data Sets descriptors, as well as general information applicable to the product such as sensor characteristics, PCD and metrics summary. The GADS contains all the data scaling factors and general information like reference extraterrestrial solar flux and some instrument settings which may be useful to analyse results.

The first ADS (LADS for location ADS) contains information on geolocation, measurement viewing and illumination geometry and auxiliary environment parameters for the tie points, a subset of the product pixels.
The second ADS (SQADS for summary quality ADS) contains quality information, aggregated at the level of a group of granules.

The first fifteen MDS are dedicated to top of atmosphere radiance measured in the 15 MERIS spectral bands and the last one to the associated flags : classification and measurement quality indicators.

Information coming either from input Level 0 product, from external data sources, or generated by any processing step are gathered, organised, scaled and coded according to AD1 specifications to build the Level 1b product file.

11.3.1.2 - Mathematical Description of Algorithm

The algorithm follows the logic shown in the block diagram in figure 11.3.1.2-1 below.

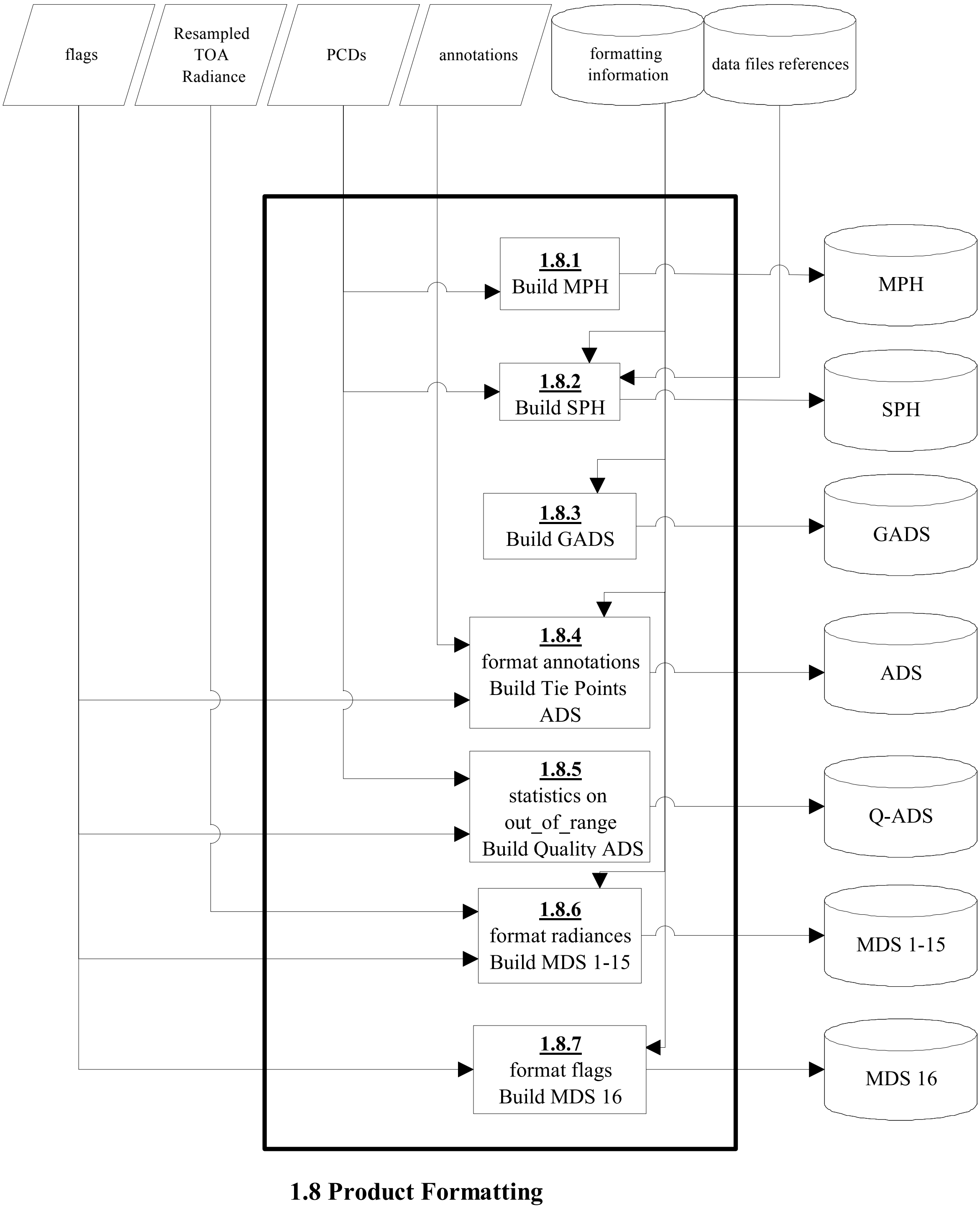


Figure 11.3.1.2-1 : MERIS Level 1b product formatting

11.3.1.2.1 - Main Product Header

Main product header is formatted as described in AD1. Only time of first and last frames of the product are input from the processing to the MPH formatting.

11.3.1.2.2 - Specific Product Header

Specific product header is formatted as described in AD1. The PCDs, issued by the previous steps 1.1 to 1.7, as well as the geolocation of first and last tie frames, from step 1.5.2, are inputs to the SPH (note that the transmission errors and the format errors counters are transformed into flags set if the mean numbers of errors per packet exceed given thresholds). In the case of the FR Scene Product, for which there is an even number of tie points, linear interpolation between the closest tie points is considered sufficiently accurate to compute geolocation of the mid sample of first and last frames.

11.3.1.2.3 - Global Annotation Data Set

Global Annotation Data Set is formatted as described in AD1. Inputs come either from algorithm step 1.6 (solar flux corrected according to day of year) or from auxiliary data bases (gain settings, scaling factors).

11.3.1.2.5 - Annotation Data Set "Tie Points Location and corresponding Auxiliary Data"

The annotation data set is composed of one Annotation Data Set Record (ADSR) for every 16 (Reduced Resolution) or 64 (Full Resolution) product frame (time sample), plus one at the last product frame. This leads to 925 ADSR per orbital product in Reduced Resolution (RR) and 36 ADSR per scene product in Full Resolution (FR), or 19 per FR imagette.

Each ADSR is composed of :

- MJD, modified Julian Day of time sample
- attachment flag : set when the MDSR corresponding to the ADSR are present in the product
- one annotation set for every tie point : 71 in RR, 36 in FR scene, 19 in FR imagette.

An annotation set includes :

1. tie point longitude
2. tie point latitude
3. tie point altitude
4. tie point surface roughness parameter
5. tie point longitude correction due to altitude
6. tie point latitude correction due to altitude
7. tie point sun zenith angle
8. tie point sun azimuth angle
9. tie point viewing zenith angle
10. tie point viewing azimuth angle

    all the above quantities from Geolocation Processing (see chapter 7)
11. ECMWF zonal wind components
12. ECMWF meridional wind components
13. ECMWF pressure
14. ECMWF total ozone
15. ECMWF relative humidity

all the above quantities from External Data Assimilation (see chapter 10)
Note : for all Tie Points with a negative altitude, fields 4 to 6 are forced to zero.

11.3.1.2.6 - Annotation Data Set "Product Quality"

The annotation data set is composed of one Annotation Data Set Records (ADSR) for every 128 (Reduced Resolution) or 512 (Full Resolution) product line, i.e. every 8 tie frames. This leads to 114 ADSR per orbital product in Reduced Resolution (RR) and 5 ADSR per scene product in Full Resolution (FR).

Each ADSR is composed of :

- MJD, modified Julian Day of time sample
- attachment flag
- one "out of range" flag register for the image pixels
- one "out of range" flag register for the blank pixels

An "out of range" flag register is composed of one flag per band and per MERIS module. A given flag is set if the number of "out of range" image or blank band samples for the given module in the region between this Quality Annotation Frame and the next one (or the product end) is above a given threshold (in %). Specific thresholds are used for image pixels and blank pixels.
Note : both "out of range" PCDs are actually linked with MERIS frames instead of Level 1b product's ones. The alignment of the Quality Annotations with the latter is equivalent to a zero along-track depointing assumption.

11.3.1.2.7 - Measurement Data Sets

There are 16 MDS, 15 for the radiances of the 15 MERIS bands and 1 for the associated flags, with the same record structure : an MDS is composed of one Measurement Data Set Record (MDSR) by product time sample.

The radiance MDSR contains :

- MJD, modified Julian Day of time sample
- quality flag : set to 0 when all data in the MDSR are invalid.
- one (scaled) radiance value per pixel (1121 in RR, 2241 in FR, 1153 in FR imagette).

Radiances are expressed in counts using the scaling factor stored in the SPH. Each value is stored in a two bytes unsigned integer.

The flag MDSR contains :

- MJD, modified Julian Day of time sample
- quality flag : set to 0 when all data in the MDSR are invalid.
- one flag set (one byte) per pixel (1121 in RR, 2241 in FR, 1153 in FR imagette).

The flag set contains 8 binary values meaning :

Flag Name*	Bit	1	0
cosmetic	0	cosmetic pixel	fully measured pixel
duplicated	1	duplicated pixel value	not duplicated pixel value
glint risk	2	glint risk	no glint risk
suspect	3	suspect pixel	not suspect pixel
land/ocean	4	land	ocean
bright	5	bright	clear sky
coastline	6	coastline	not coastline
invalid	7	invalid	valid

* : as per AD1.

Each value is coded on 1 bit of the same byte, from least significant bit for flag 1 to most significant bit for flag 8 (see AD1, section 5.3.1.8.2).

The "land/ocean", "bright" and "coastline" flags are direct inputs from Pixel Classification (see section 9) ; the "duplicate" flag is a direct input of the Radiance Resampling (see section 8) ; the "glint risk" flag is a direct input from geolocation (see section 8) ; they are stored without further processing and do not need new definitions.

The "invalid" flag is a direct input, logically recombined with other flags, in order to gather all pixels satisfying any one of the following conditions :

- ♦ samples of all bands are saturated ;
- ♦ out-of-swath product pixels ;
- ♦ pixels added at the end of the product to reach the last tie frame ;
- ♦ pixels added to fill a transmission gap of more than sixteen packets.

The "cosmetic" flag coming from the processing chain is a per band flag ; the "suspect" flag is a new flag gathering pixels with diverse internal flags configurations ; they are defined below :

- are considered "cosmetic" those pixels for which at least one radiance sample has been replaced by interpolation from neighbours as described in section 5 ;
- are considered "suspect" those pixels satisfying one of the following conditions :
  - ♦ for any pixel, if it is flagged "stray light risk" ;
  - ♦ for a "clear sky" and "ocean" pixel, at least one of the radiance samples is "saturated" or "dubious" ;
  - ♦ for a "clear sky" and "land" pixel, at least one of the radiance samples of the bands dedicated to "land" is "saturated" or "dubious" (list of "land" bands a processing parameter) ;

11.3.2 - List of Variables

Indexing convention :

- subscript b stands for the band index, in range [1,B]
- subscript j for the product pixel index, in range [1,NC]
- subscript f for the product line index, in range [1,NF]
- subscript m for the MERIS module index, in range [1,Mt]
- subscript J for the tie point pixel index, in range [1,N_{TP}]
- subscript F for the tie point line index, in range [1,1+NF/DF]

Variable	Descriptive Name	T	U	Range - References
NC	Number of samples per line	i	dl	from 1.5.1
NF	Number of frames in product	i	dl	from 1.5.1
N_{TP}	number of tie points in product	i	dl	from 1.5.1
M	number of MERIS modules to process	i	dl	from 1.5.1
F'_0[b]	Extra-terrestrial Sun irradiance	i	EU	from 1.6
T_JD'[f]	MJD2000 time for frame j (j in 1,..,NF)	i	jd	from 1.5.5
Dubious_f^{FR}[b,j,f]	FR resampled "dubious sample" flag	i	dl	from 1.5.5
Dubious_f^{RR}[b,j,f]	RR resampled "dubious sample" flag	i	dl	from 1.5.5
Saturated_f^{FR}[b,j,f]	FR resampled "saturated sample" flag	i	dl	from 1.5.5
Saturated_f^{RR}[b,j,f]	RR resampled "saturated sample" flag	i	dl	from 1.5.5
Cosmetic_f^{FR}[b,j,f]	FR resampled "cosmetic sample" flag	i	dl	from 1.5.5
Cosmetic_f^{RR}[b,j,f]	RR resampled "cosmetic sample" flag	i	dl	from 1.5.5
Glint_f^{FR}[j,f]	FR sun glint risk flag	i	dl	from 1.5.6
Glint_f^{RR}[j,f]	RR sun glint risk flag	i	dl	from 1.5.6
Stray_f^{FR}[j,f]	RR straylight risk flag for frame f	i	dl	from 1.5.5
Stray_f^{RR}[j,f]	RR straylight risk flag for frame f	i	dl	from 1.5.5
Duplicated_f^{FR}[j,f]	FR duplicated pixel flag	i	dl	from 1.5.5
Duplicated_f^{RR}[j,f]	RR duplicated pixel flag	i	dl	from 1.5.5
Bright_fFR[j,f]	FR pixels bright classification flag	i	dl	from 1.6
Bright_fRR[j,f]	RR pixels bright classification flag	i	dl	from 1.6
Land_fFR[j,f]	FR pixels land/ocean classification flag	i	dl	from 1.6
Land_fRR[j,f]	RR pixels land/ocean classification flag	i	dl	from 1.6
Coast_fFR[j,f]	FR pixels coastline classification flag	i	dl	from 1.6
Coast_fRR[j,f]	RR pixels coastline classification flag	i	dl	from 1.6
Invalid_f^{FR}[j,f]	FR "invalid pixel" flag	i	dl	from 1.5.5
Invalid_f^{RR}[j,f]	RR "invalid pixel" flag	i	dl	from 1.5.5
TOARFR[b,j,f]	FR resampled TOA radiance at pixel i,j	i	dl	from 1.5.5
TOARRR[b,j,f]	RR resampled TOA radiance at pixel i,j	i	LU	from 1.5.5
$Detector^{FR}$[j,f]	FR Detector index resampled at pixel j,f	i	dl	from 1.5.5
$Detector^{RR}$[j,f]	RR Detector index resampled at pixel j,f	i	dl	from 1.5.5
λ[J,F]	longitude at tie point J,F	i	deg	from 1.5.2
ϕ[J,F]	latitude at tie point J,F	i	deg	from 1.5.2
θ_s[J,F]	Sun zenith angle at tie point J,F	i	deg	from 1.5.2
φ_s[J,F]	Sun azimut angle at tie point J,F	i	deg	from 1.5.2
θ_v[J,F]	Observer zenith angle at tie point J,F	i	deg	from 1.5.2
φ_v[J,F]	Observer zenith angle at tie point J,F	i	deg	from 1.5.2
z[J,F]	altitude at tie point J,F	i	deg	from 1.5.4
σz[J,F]	altitude standard deviation at tie point J,F	i	deg	from 1.5.4
dlon[J,F]	longitude correction at tie point J,F	i	deg	from 1.5.4
dlat[J,F]	latitude correction at tie point J,F	i	deg	from 1.5.4
P_tie[J,F]	Surface pressure at tie point J,F	i	hPa	from 1.7
Wu_tie[J,F]	Wind U component at tie point J,F	i	$m.s^{-1}$	from 1.7
Wv_tie[J,F]	Wind V component at tie point J,F	i	$m.s^{-1}$	from 1.7
Oz_tie[J,F]	Total Ozone at tie point J,F	i	DU	from 1.7
RH_tie[J,F]	Relative humidity at tie point J,F	i	deg	from 1.7

Table 11.3.2-1: Parameters used in the Formatting algorithm

Variable	Descriptive Name	T	U	Range - References
transmission_PCD	counter of transmission errors in the segment	i	dl	from 1.1
format_PCD	counter of format errors in the segment	i	dl	from 1.1
database_PCD	flag indicating incompatibility with auxiliary parameters data base	i	dl	from 1.1
coarse_PCD	flag set if coarse offsets are above a thresold	i	dl	from 1.1
blank_PCD[b,m,f]	counter of out-of-range blank pixels	i	dl	from 1.1
out_r_PCD[b,m,f]	counter of out-of-range image samples	i	dl	from 1.3
ECMWF_TYPE_PCD	ECMWF Quality PCD	i	dl	from 1.7
B	number of MERIS bands	s	dl	
transmission_thresh	threshold for transmission errors flag (mean number of errors per packet)	s	dl	
format_thresh	threshold for format errors flag (mean number of errors per packet)	s	dl	
Tie_scale	structure of scaling factors for annotations	s	mix	see note 1
Rad_scale_b	scaling factor for radiances	s	nc/LU	
pc_thresh_image	threshold for out_of_range flagging of image pixels	s	dl	
pc_thresh_blank	threshold for out_of_range flagging of blank pixels	s	dl	
$Wavelengths_b$	Band wavelengths	s	nm	
$Widths_b$	Band widths	s	nm	
$IFOV^{RR}$	RR Instantaneous FOV	s	deg	
$IFOV^{FR}$	FR Instantaneous FOV	s	deg	
OB_R	Reference for on-board processing switch	s	dl	0: on ground, 1: on board
$BAND\_GAIN\_R_{b,m}$	Reference for band gain settings	s	dl	
OCL_R	Reference for OCL switch	s	dl	
DT^{RR}	Delay between two frames	s	s	
DT^{FR}	Delay between two frames	s	s	
K^{RR}	Number of columns per MERIS module	s	dl	
K^{FR}	Number of columns per MERIS module	s	dl	
KB	Number of blank pixel columns per module	s	dl	
Pix^{RR}	RR product pixel AC size	s	m	
Pix^{FR}	FR product pixel AC size	s	m	
DF^{RR}	RR product frame to tie frame sub-sampling factor	s	dl	
DF^{FR}	FR product frame to tie frame sub-sampling factor	s	dl	
DJ^{RR}	RR product column to tie point sub-sampling factor	s	dl	
DJ^{FR}	FR product column to tie point sub-sampling factor	s	dl	
DFSQ	tie frame to SQADS frame sub-sampling factor	s	dl	
Land_bands	set of bands used for land observation	s	dl	
ϕ_1, ϕ_2	intermediate variables for longitudes	c	deg	
λ_1, λ_2	intermediate variables for latitudes	c	deg	
$nvalid_f$	flag indicating that all pixels of frame f are invalid	c	dl	Boolean
$nvalid_F$	flag indicating that all pixels related to tie frame F are invalid	c	dl	Boolean
nvalid_Q	flag indicating that all pixels related to an ADS "Product Quality" are invalid	c	dl	Boolean
npix_blank	counter of blank pixels	c	dl	
npix_image	counter of image pixels	c	dl	
$Susp\_f_{j,f}$	Dubious sample flag after band reduction	c	dl	
$Cos\_f_{j,f}$	Cosmetic sample flag after band reduction	c	dl	
$pc\_out\_image_{b,m}$	Percentage of Out of Rangeimage pixels for band band and module m in sub-tie grid area	c	dl	
$pc\_out\_blank_{b,m}$	Percentage of Out of Range blank pixels for band band and module m in sub-tie grid area	c	dl	
$out\_image\_f_{b,m}$	out-of-range flag register for image pixels	c	dl	b=1,...,B ; m=1,...,M
$out\_blank\_f_{b,m}$	out-of-range flag register for blank pixels	c	dl	b=1,...,B ; m=1,...,M
X[b,j,f]	formatted TOA radiance	c	nc	
F[j,f]	formatted flag register	c	dl	
Outputs are the fields of the Level 1B product tables as per AD1				

Table 11.3.2-1 (cont.) : Parameters used in the Formatting algorithm

Note : the tie points scaling factor data structure can be expressed as follows :

```
struct scaling_factor_struct {
   float P;               (hPa/count)
   float Wu;              (m.s-1)/count
   float Wv;              (m.s-1/count)
   float Oz;              (DU/count)
   float RH;              (%/count)
   float Altitude;        (m/count)
   float Roughness;       (m/count)
} Tie_scale;
```

11.3.3 - Equations

Notes:

- MPH, SPH, GADS, are described precisely in AD1 and/or AD7, in a way that allows to avoid a redundant description here. Are mentionned only those fields for which either a calculation or an input from another algorithm step is needed.

- Conversions of floating point values into integers are always done using nearest integer rounding, after scaling if applicable.

- The symbol ∩ means logical AND operation on a set of Boolean values.
- The symbol ∪ means logical OR operation on a set of Boolean values.

Step 1.8.1 Build MPH

Notes:

the field names in this section refer to "Contents" column in table 5.2.2.1 of AD7;
pl_pmjd is a routine converting time expressed in mjd200 to UTC format.

UTC start time of the data sensing field = pl_pmjd(T_JD_1) *(1.8.1-1)*

UTC stop time of the data sensing field = pl_pmjd(T_JD_{NF}) *(1.8.1-2)*

write MPH *(1.8.1-3)*

Step 1.8.2 Build SPH

Note: the field names in this section refer to "Description" column in table 5.3.1.4a of AD1.

FIRST_LINE_TIME field = pl_pmjd(T_JD_1) *(1.8.2-1)*

LAST_LINE_TIME field = pl_pmjd(T_JD_{NF}) *(1.8.2-2)*

FIRST_FIRST_LAT field = $\phi[1,1]$; *(1.8.2-3)*

FIRST_FIRST_LONG field = $\lambda[1,1]$; *(1.8.2-4)*

if(mod(N_{TP},2)==1) then

  FIRST_MID_LAT field = $\phi[(N_{TP}+1)/2,1]$; *(1.8.2-5)*

  FIRST_MID_LONG field = $\lambda[(N_{TP}+1)/2,1]$; *(1.8.2-6)*

else

  $\phi_1 = \phi[N_{TP}/2,1]$; *(1.8.2-7)*

  $\phi_2 = \phi[N_{TP}/2+1,1]$; *(1.8.2-8)*

```
    FIRST_MID_LAT field = (φ1+φ2)/2;                                  (1.8.2-9)
    λ1 = λ[NTP/2,1];                                                  (1.8.2-10)
    λ2 = λ[NTP/2+1,1];                                                (1.8.2-11)
    if(|λ1-λ2|>100) then
    must cross 180 degrees meridian : change longitude range to [0,360]
      λ1 = mod(λ1+360,360);                                           (1.8.2-12)
      λ2 = mod(λ2+360,360);                                           (1.8.2-13)
    endif
    FIRST_MID_LONG field = (λ1+λ2)/2;                                 (1.8.2-14)
endif
FIRST_LAST_LAT field = φ[NTP,1];                                      (1.8.2-15)
FIRST_LAST_LONG field = λ[NTP,1];                                     (1.8.2-16)
LAST_FIRST_LAT field = φ[1,1+NF/DF];                                  (1.8.2-17)
LAST_FIRST_LONG field = λ[1,1+NF/DF];                                 (1.8.2-18)
if(mod(NTP,2)==1) then
    LAST_MID_LAT field = φ[(NTP+1)/2,1+NF/DF];                        (1.8.2-19)
    LAST_MID_LONG field = λ[(NTP+1)/2,1+NF/DF];                       (1.8.2-20)
else
    φ1 = φ[NTP/2,1+NF/DF];                                            (1.8.2-21)
    φ2 = φ[NTP/2+1,1+NF/DF];                                          (1.8.2-22)
    LAST_MID_LAT field = (φ1+φ2)/2;                                   (1.8.2-23)
    λ1 = λ[NTP/2,1+NF/DF];                                            (1.8.2-24)
    λ2 = λ[NTP/2+1,1+NF/DF];                                          (1.8.2-25)
    if(|λ1-λ2|>100) then
    must cross 180 degrees meridian : change longitude range to [0,360]
      λ1 = mod(λ1+360,360);                                           (1.8.2-26)
      λ2 = mod(λ2+360,360);                                           (1.8.2-27)
    endif
    LAST_MID_LONG field = (λ1+λ2)/2;                                  (1.8.2-28)
endif
LAST_LAST_LAT field = φ[NTP,1+NF/DF];                                 (1.8.2-29)
LAST_LAST_LONG field = λ[NTP,1+NF/DF];                                (1.8.2-30)
if (transmission_PCD/(NF*(B+1)) > transmission_thresh)
    TRANS_ERR_FLAG field = 1;                                         (1.8.2-31)
else
    TRANS_ERR_FLAG field = 0;                                         (1.8.2-32)
endif
if (format_PCD/(NF*(B+1)) > format_thresh)
    FORMAT_ERR_FLAG field = 1;                                        (1.8.2-33)
else
    FORMAT_ERR_FLAG field = 0;                                        (1.8.2-34)
endif
DATABASE_FLAG field = database_PCD;                                   (1.8.2-35)
```

COARSE_ERR_FLAG field = coarse_PCD; *(1.8.2-36)*

ECMWF_TYPE field = ECMWF_TYPE_PCD; *(1.8.2-37)*

NUM_TRANS_ERR field = transmission_PCD; *(1.8.2-38)*

NUM_FORMAT_ERR field = format_PCD; *(1.8.2-39)*

TRANS_ERR_THRESH field = transmission_thresh; *(1.8.2-40)*

FORMAT_ERR_THRESH field = format_thresh; *(1.8.2-41)*

NUM_BANDS field = B; *(1.8.2-42)*

BAND_WAVELEN field = Wavelengths; *(1.8.2-43)*

BANDWIDTH field = Widths; *(1.8.2-44)*

INST_FOV field = IFOV; *(1.8.2-45)*

PROC_MODE field = OB_R; *(1.8.2-46)*

OFFSET_COMP field = OCL_R; *(1.8.2-47)*

LINE_TIME_INTERVAL field = DT; *(1.8.2-48)*

LINE_LENGTH field = NC *(1.8.2-49)*

LINES_PER_TIE_PT field = DF *(1.8.2-50)*

SAMPLES_PER_TIE_PT field = DJ *(1.8.2-51)*

COLUMN_SPACING field = Pix *(1.8.2-52)*

copy description field of level 0 product in DSD field *(1.8.2-53)*

copy description field of each auxiliary product in DSD fields *(1.8.2-54)*

write SPH *(1.8.2-55)*

Step 1.8.3 Build GADS

Note: the field names in this section refer to "Description" column in table 5.3.1.5 of AD1.

scaling factor for pressure field = Tie_scale.P *(1.8.3-1)*

scaling factor for wind zonal field = Tie_scale.Wu *(1.8.3-2)*

scaling factor for wind meridional field = Tie_scale.Wv *(1.8.3-3)*

scaling factor for Ozone field = Tie_scale.Oz *(1.8.3-4)*

scaling factor for Relative humidity field = Tie_scale.RH *(1.8.3-5)*

scaling factor for Altitude field = Tie_scale.Altitude *(1.8.3-6)*

scaling factor for Roughness field = Tie_scale.Roughness *(1.8.3-7)*

scaling factor for Radiance field = Rad_scal *(1.8.3-8)*

gain settings field = BAND_GAIN_R; *(1.8.3-9)*

sampling rate field = DT *(1.8.2-10)*

Sun spectral flux field = (F'_0[b],b=1,B) *(1.8.2-11)*

write GADS to product *(1.8.3-12)*

Step 1.8.4 Build ADS "Tie Points Annotations and corresponding Auxiliary Data"

Build Annotation Data Set

Loop on tie points grid lines

```
for each tie point line F
  time tag field of ADSR = T_JD[F], formatted to Transport format
  using pl_pmjd CFI;
```
(1.8.4-1)

raise attachment flag only if no data are attached :

$$\text{nvalid}_F = \bigcap_{\substack{j \in 1..NC, \\ f \in F..F+DF-1}} \text{Invalid\_f}_{j,f} \;; \qquad (1.8.4\text{-}2)$$

```
  if (nvalid_F) then
    attachment flag field of ADSR = 1;                          (1.8.4-4)
  else
    attachment flag field of ADSR = 0;                          (1.8.4-5)
  end if
```

Loop on tie points

```
  for each tie point J
```

scale all annotation fields

```
    longitude field[J] = λ[J,F];                                (1.8.4-6)
    latitude field[J] = ϕ[J,F];                                 (1.8.4-7)
    sun zenith angle field[J] = θS[J,F];                        (1.8.4-8)
    sun azimuth angle field[J] = φS[J,F];                       (1.8.4-9)
    observer zenith angle field[J] = θV[J,F];                   (1.8.4-10)
    observer azimuth angle field[J] = φV[J,F];                  (1.8.4-11)
    DEM altitude field[J] = z[J,F]/Tie_scale.Altitude;          (1.8.4-12)
    if (z[J,F]≥0)
      DEM roughness field[J] = σz[J,F]/Tie_scale.Roughness;     (1.8.4-13)
      DEM longitude correction field[J] = dlon[J,F];            (1.8.4-14)
      DEM latitude correction field[J] = dlat[J,F];             (1.8.4-15)
    else
      DEM roughness field[J] = 0;                               (1.8.4-22)
      DEM longitude correction field[J] = 0;                    (1.8.4-23)
      DEM latitude correction field[J] = 0;                     (1.8.4-24)
    end if
    pressure field[J] = P_tie[J,F]/Tie_scale.P;                 (1.8.4-16)
    zonal wind field[J] = Wu_tie[J,F]/Tie_scale.Wu;             (1.8.4-17)
    meridional wind field[J] = Wv_tie[J,F]/Tie_scale.Wv;        (1.8.4-18)
    ozone field[J] = Oz_tie[J,F]/Tie_scale.Oz;                  (1.8.4-19)
    relative humidity field[J] = RH_tie[J,F]/Tie_scale.RH;      (1.8.4-20)
  end for
  write ADSR;                                                   (1.8.4-21)
end for
```

Step 1.8.5 Build ADS "Product Quality"

Build Annotation Data Set

Loop on tie points sub-grid lines

```
for each tie point grid line F with step of DF * DFSQ
```

raise attachment flag only if no data are attached :

$$nvalid\_Q = \bigcap_{k=0,DFSQ-1} nvalid_{F+k \cdot DF} \; ; \qquad (1.8.5\text{-}0a)$$

```
if (nvalid_Q) then
    attachment flag field of ADSR = 1;                          (1.8.5-0b)
else
    attachment flag field of ADSR = 0;                          (1.8.5-0c)
end if
```

reset "out of range" counters

```
pc_out_image = 0;                                               (1.8.5-1)
pc_out_blank = 0;                                               (1.8.5-2)
npix_image = 0;                                                 (1.8.5-3)
npix_blank = 0;                                                 (1.8.5-4)
```

loop on tie points grid lines between two sub-grid lines

```
for each product line f in F..F+DF.DFSQ-1
```

loop on all samples in image zone

```
    for each module m
        for each band b
            pc_out_image_{b,m} = pc_out_image_{b,m} + out_r_PCD[b,m,f];   (1.8.5-5)
            pc_out_blank_b = pc_out_blank_b + blank_PCD[b,m,f] ;        (1.8.5-6)
        end for
        npix_blank = npix_blank + KB;                               (1.8.5-7)
        npix_image = npix_image + K;                                (1.8.5-8)
    end for
```

if end of product reached, break the product line loop

```
    if (f == NF) then
        break;                                                      (1.8.5-9)
    end if
end for
```

compute percentage and update flags

```
for each module m
    for each band b
        pc_out_image_b = pc_out_image_b /npix_image;                (1.8.5-10)
        if (pc_out_image_b > pc_thresh_image)
            out_image_f_{b,m} = TRUE;                               (1.8.5-11)
        else
            out_image_f_{b,m} = FALSE;                              (1.8.5-12)
        end if
        pc_out_blank_b = pc_out_blank_b/npix_blank;                 (1.8.5-13)
        if (pc_out_blank_b > pc_thresh_blank)
            out_blank_f_{b,m} = TRUE;                               (1.8.5-14)
        else
            out_blank_f_{b,m} = FALSE;                              (1.8.5-15)
        end if
    end for
```

```
  end for
```
build QADSR with MJD and flags registers
```
  JD field of QADSR = T_JD[F], formatted to Transport format
  using pl_pmjd CFI;                                              (1.8.5-16)
  Out of range flag field of QADSR = out_image_f;                 (1.8.5-17)
  Out of range blank flag field of QADSR = out_blank_f;           (1.8.5-18)
```
write Q-ADSR into L1b product
```
  write_QADSR();                                                  (1.8.5-19)
end for
```

Step 1.8.6 Build TOA MDS

Build Measurements Data Sets

Data Sets 1 to 15 : radiance

Process frames according to the presence of valid samples
```
for each product line f
  DELETED                                                         (1.8.6-0)

  for each product pixel j in line f
    if (NOT Invalid_fj,f) then
```
$$Invalid\_f_{j,f} = \bigcap_{b \in 1..B} Saturated\_f_{b,j,f} \; ; \qquad (1.8.6\text{-}1)$$
```
    end if
  end for
```
$$nvalid_f = \bigcap_{j \in 1..NC} Invalid\_f_{j,f} \; ; \qquad (1.8.6\text{-}2)$$
```
  for each band b
    JD field of MDSR f in MDS b = T_JD'[f]], formatted to Transport
    format using pl_pmjd CFI;                                     (1.8.6-3)
```
No valid sample has been read, no need to go further
```
    if (nvalidf) then
      quality flag field of MDSR f in MDS b = -1;                 (1.8.6-4)
      for each product column j
        X[b,j,f] = 0;                                             (1.8.6-5)
      end for
```
Valid samples exist :
```
    else
      quality flag field of MDSR f in MDS b = 0;                  (1.8.6-6)
      for each product column j
        X[b,j,f] = TOAR[b,j,f] / rad_scale[b];                    (1.8.6-7)
      end for
      radiance field of MDSR f in MDS b = X;                      (1.8.6-8)
    end if                                   end (NOT nvalid) branch
  write MDSR f in MDS b;                                          (1.8.6-9)
  end for                                              end loop on bands
```

Step 1.8.7 Build flags MDS

Data Set 16 : pixel flags

```
JD field of MDSR f in MDS 16 = T_JD'[f]], formatted to Transport
 format using pl_pmjd CFI;                                          (1.8.7-1)
```

build "summary" flags

```
if (nvalid_f) then
```

No valid samples exist : set quality flag to -1 and all pixels flags to "invalid" only (1.8.7-2)

```
    quality flag field of MDS = -1;
    for each product column j
      F[j,f,0] = 1 << 7;
    end for
  else
```

Valid samples exist :

```
    quality flag field of MDS = 0;                                  (1.8.7-3)
    for each pixel column j
      if (NOT Invalid_f_j,f) then
```

$$Cos\_f_{j,f} = \bigcup_{b \in 1..B} Cosmetic\_f[b,j,f]; \quad (1.8.7\text{-}4)$$

```
        for each b in Land_bands
          if(Saturated_f[b,j,f] || Dubious[b,j,f]) then
            Susp_f_j,f = TRUE;                                      (1.8.7-5)
          end if
        end for
```

DELETED (1.8.7-6)

DELETED (1.8.7-11)

Also flag any stray light risk pixel as suspect

```
        if(Stray_f[j,f])
          Susp_f_j,f = TRUE;                                        (1.8.7-7)
        end if
      end if
```

end of (NOT Invalid_$f_{j,f}$) branch

Combine all flags in one byte : (1.8.7-8)

```
      F[j,f,0] =  Cos_f_j,f +
                  Dupl_f_j,f<<1 +
                  Glint_f_j,f<<2 +
                  Susp_f_j,f<<3 +
                  Land_f_j,f<<4 +
                  Bright_f_j,f<<5 +
                  Coast_f_j,f<<6 +
                  Invalid_f_j,f<<7;
    end for
```

end of loop on product columns

```
  end if
```

end of "Valid samples exist" branch

```
  flags field of MDSR f in MDS 16 = F;                              (1.8.7-12)
  Detector index field of MDSR f in MDS 16 = Detector[*,f];         (1.8.7-10)
  write MDSR f in MDS 16                                            (1.8.7-9)
end for
```

end of loop on product frames

11.3.4 - Accuracy Requirements

JD date of MDSR records shall be computed with an accuracy of 1 ms.
Formatted TOA radiance fields shall be computed with an accuracy of 1 Least Significant Digit.
All tie point annotation fields shall be computed with an accuracy of 1 Least Significant Digit.

11.3.5 - Product Confidence Data Summary

Product Formatting raises no PCD of its own.

ANNEX A - PARAMETERS DATA LIST

Variable	Descriptive Name	IODD section	IODD table	Parameter number	Product Name	ADS	DPM section	Algorithm step
UTC_REF_FOR_OBT	UTC reference time for OBT conversion	5.2	N/A	N/A	Level 0	MPH	4	1.1
OBT_REF	OBT counter value corresponding to the reference UTC	5.2	N/A	N/A	Level 0	MPH	4	1.1
OBT_TICK	Duration of one tick of the OBT counter	5.2	N/A	N/A	Level 0	MPH	4	1.1
B	Number of MERIS bands	6.3	4	2	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1
KB	Number of blank pixels in one module	6.1	5	23	Instrument	Instrumental Parameters	4	1.1
K^{RR}	Number of columns in one RR module	6.1	5	3	Instrument	Instrumental Parameters	4	1.1
K^{FR}	Number of columns in one FR module	6.1	5	2	Instrument	Instrumental Parameters	4	1.1
Mt	Number of MERIS modules	6.1	5	1	Instrument	Instrumental Parameters	4	1.1
DFH_LENGTH_R	Ref. value for data field header length	6.1	4	4	Instrument	Configuration Reference Values	4	1.1
MODE_MASK	Binary mask for the APID dependent bits in the instrument mode field	6.1	4	5	Instrument	Configuration Reference Values	4	1.1
$MODE\_BITS\_R^{RR}$	dictionary of ref. values for APID dependent bits in instrument mode field	6.1	4	9	Instrument	Configuration Reference Values	4	1.1
$MODE\_BITS\_R^{FR}$	dictionary of ref. values for APID dependent bits in instrument mode field	6.1	4	7	Instrument	Configuration Reference Values	4	1.1
OCL_MASK	Binary mask for the OCL dependent bits in the instrument mode field	6.1	4	12	Instrument	Configuration Reference Values	4	1.1
OCL_R	OCL switch reference	6.3	4	10	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1
OB_MASK	Binary mask for the on-board correction switch dependent bits in the instrument mode field	6.1	4	13	Instrument	Configuration Reference Values	4	1.1
OB_R	on-board correction switch reference	6.3	4	11	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1
OTHER_MASK	Binary mask for the other bits in the instrument mode field	6.1	4	14	Instrument	Configuration Reference Values	4	1.1
OTHER_BITS_R	Ref. value for other bits in instrument mode field	6.1	4	10	Instrument	Configuration Reference Values	4	1.1
REDUND_VECTOR_R	Ref. value for redundancy vector	6.3	4	1	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1
BAND_POS_R[b]	Ref. values for band position	6.3	4	5	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1
BAND_LEN_R[b]	Ref. values for band length	6.3	4	7	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1
BAND_GAIN_R[m,b]	Ref. values for band gain settings	6.3	4	20	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1
BAND_MB_R[b]	Ref. values for no. of micro-bands	6.3	4	6	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1
COARSE_THR[l]	Upper threshold for coarse offsets	6.3	4	21	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1

Variable	Descriptive Name	IODD section	IODD table	Parameter number	Product Name	ADS	DPM section	Algorithm step
RELAX_COF_R[b]	Weights for on-board Spatial and Temporal Relaxation (per band)	6.3	4	12	Radiometric Calibration	Radiometric Correction Control Parameters	4	1.1
BLANK_THR[b]	Upper threshold for blank pixels	6.2	9	1	Level1b Control Parameters	Level 0 Extraction	4	1.1
BLANK_DIF_THR[b]	Difference threshold for blank pixels	6.2	9	2	Level1b Control Parameters	Level 0 Extraction	4	1.1
MS_TO_JD	Expression of 1 ms in MJD2000	6.2	4	1	Level1b Control Parameters	General	4	1.1
PC_WRAPAROUND	Wraparound value for PC	6.1	4	11	Instrument	Configuration Reference Values	4	1.1
MAX_GAP_P	Maximum gap between two packets allowing cosmetic filling	6.2	4	2	Level1b Control Parameters	General	4	1.1
DT^{RR}	Delay between two RR frames	6.1	5	5	Instrument	Instrumental Parameters	4	1.1
DT^{FR}	Delay between two FR frames	6.1	5	4	Instrument	Instrumental Parameters	4	1.1
PK_LEN^{RR}	Packet length field for RR	6.1	4	3	Instrument	Configuration Reference Values	4	1.1
PK_LEN^{FR}	Packet length field for FR	6.1	4	2	Instrument	Configuration Reference Values	4	1.1
PK_SCALE	scaling factor for packet header float data coding	6.2	9	3	Level1b Control Parameters	Level 0 Extraction	4	1.1
B	Number of MERIS bands	6.3	4	2	Radiometric Calibration	Radiometric Correction Control Parameters	5	1.2
K^{RR}	number of columns in a RR module	6.1	5	3	Instrument	Instrumental Parameters	5	1.2
K^{FR}	number of columns in a FR module	6.1	5	2	Instrument	Instrumental Parameters	5	1.2
$SAT\_REC\_K^{FR}$	Number of following samples affected by an FR pixel saturation during read-out	6.2	11	3	Level1b Control Parameters	Flagging	5	1.2
$GLINT\_BLOOM\_K^{FR}$	Number of neighbour pixels affected by saturation in a pixel	6.2	11	1	Level1b Control Parameters	Flagging	5	1.2
$SAT\_SAMPLE^{FR}[b]$	Saturation value for a MERIS FR sample	6.2	11	7	Level1b Control Parameters	Flagging	5	1.2
RELAX_COF_R[b]	Weights for on-board Spatial and Temporal Relaxation (per band)	6.3	4	12	Radiometric Calibration	Radiometric Correction Control Parameters	5	1.2
$SAT\_REC\_K^{RR}$	Number of following samples affected by an RR pixel saturation during read-out	6.2	11	4	Level1b Control Parameters	Flagging	5	1.2
$GLINT\_BLOOM\_K^{RR}$	Number of neighbour pixels affected by saturation in a pixel	6.2	11	2	Level1b Control Parameters	Flagging	5	1.2
K^{RR}	number of columns in a RR module	6.1	5	3	Instrument	Instrumental Parameters	6	1.3
K^{FR}	number of columns in a FR module	6.1	5	2	Instrument	Instrumental Parameters	6	1.3
B	number of bands	6.3	4	2	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
Mt	number of MERIS modules	6.1	5	1	Instrument	Instrumental Parameters	6	1.3
$T\_JD_{ref}[d]$	Reference time for temperature models	6.3	4	19	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
RR_NONLIN_F	Switch to apply non-linearity correction to RR data	6.2	12	2	Level1b Control Parameters	Radiometric	6	1.3
FR_NONLIN_F	Switch to apply non-linearity correction to FR data	6.2	12	1	Level1b Control Parameters	Radiometric	6	1.3

Variable	Descriptive Name	IODD section	IODD table	Parameter number	Product Name	ADS	DPM section	Algorithm step
$NonLinLUT_{b,m}[x]$	Inverse non-linearity LUT at micro-band level	6.3	9	all	Radiometric Calibration	Non Linearity LUT	6	1.3
Aij_b	Weights for on-board Spatial and Temporal Relaxation (per band)	6.3	4	12	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
MB_b	Number of micro-bands for each band	6.3	4	6	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
$C^{0FR}_{b,k,m}$	FR Dark signal characterisation data	6.3	7	1	Radiometric Calibration	FR Offset	6	1.3
$AL^{0FR-1}_{b,k,m}$	FR Inverse Absolute gain coefficients	6.3	5	1	Radiometric Calibration	FR Gain	6	1.3
$C^{0RR}_{b,k,m}$	RR Dark signal characterisation data	6.3	8	1	Radiometric Calibration	RR Offset	6	1.3
$AL^{0RR-1}_{b,k,m}$	RR Inverse Absolute gain coefficients	6.3	6	1	Radiometric Calibration	RR Gain	6	1.3
g_{c0}	0-order coeff. of dark temp. correction	6.3	4	16	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
g_{c1}	1st order coeff. of dark temperature correction	6.3	4	17	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
g_{c2}	2nd order coeff. of dark temperature correction	6.3	4	18	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
g_0	0-order coeff. of gain temp. correction	6.3	4	13	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
g_1	1st order coeff. of gain temperature correction	6.3	4	14	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
g_2	2nd order coeff. of gain temperature correction	6.3	4	15	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
Ksm^{RR}_b	Smear weighting factor for RR	6.3	4	8	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
$Ksm^{FR}_{b,j}$	Smear weighting factor for FR	6.3	4	9	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
Sat_rad_b	Saturation radiance values	6.2	11	8	Level1b Control Parameters	Flagging	6	1.3
Def_rad_b	Default radiance value for saturated samples	6.2	8	1	Level1b Control Parameters	Exception Handling	6	1.3
$Def\_rad\_O_b$	Default radiance value for samples above range limits	6.2	8	2	Level1b Control Parameters	Exception Handling	6	1.3
$dead\_pix^{RR}[b,k2,m]$	dead pixels map for RR	6.1	7	3	Instrument	RR Pointing	6	1.3
$dead\_pix^{FR}[b,k2,m]$	dead pixels map for FR	6.1	6	3	Instrument	FR Pointing	6	1.3
$ALB^{-1}_{b,m}$	Inverse mean absolute gain	6.3	4	23	Radiometric Calibration	Radiometric Correction Control Parameters	6	1.3
$A\_JD^{FR}_{ref}$	Reference time for FR Instrument response degradation model	6.3	12	1	Radiometric Calibration	FR Degradation	6	1.3
$\beta^{FR}_{b,k,m}$	Degradation Model amplitude for FR	6.3	12	2	Radiometric Calibration	FR Degradation	6	1.3
$\gamma^{FR}_{b,k,m}$	Degradation model time shift for FR	6.3	12	3	Radiometric Calibration	FR Degradation	6	1.3
$\delta^{FR}_{b,k,m}$	Degradation model time scale for FR	6.3	12	4	Radiometric Calibration	FR Degradation	6	1.3

Variable	Descriptive Name	IODD section	IODD table	Parameter number	Product Name	ADS	DPM section	Algorithm step
$A\_JD^{RR}_{ref}$	Reference time for RR Instrument response degradation model	6.3	13	1	Radiometric Calibration	RR Degradation	6	1.3
$\beta^{RR}_{b,k,m}$	Degradation Model amplitude for RR	6.3	13	2	Radiometric Calibration	RR Degradation	6	1.3
$\gamma^{RR}_{b,k,m}$	Degradation model time shift for RR	6.3	13	3	Radiometric Calibration	RR Degradation	6	1.3
$\delta^{RR}_{b,k,m}$	Degradation model time scale for RR	6.3	13	4	Radiometric Calibration	RR Degradation	6	1.3
K^{RR}	Number of columns in a MERIS RR module	6.1	5	3	Instrument	Instrumental Parameters	7	1.4
K^{FR}	Number of columns in a MERIS FR module	6.1	5	2	Instrument	Instrumental Parameters	7	1.4
B	Number of MERIS bands	6.3	4	2	Radiometric Calibration	Radiometric Correction Control Parameters	7	1.4
Mt	number of MERIS modules	6.1	5	1	Instrument	Instrumental Parameters	7	1.4
SR	Number of spectral regions for spectrometer stray light evaluation	6.2	16	1	Level1b Control Parameters	Straylight Evaluation Parameters	7	1.4
λ[b]	band central wavelength	6.3	4	3	Radiometric Calibration	Radiometric Correction Control Parameters	7	1.4
Bs	index of bands that can be used for radiance estimation of saturated samples	6.2	16	7	Level1b Control Parameters	Straylight Evaluation Parameters	7	1.4
Stray_corr_AC_s	Switch to enable ACxSP stray light correction	6.2	12	3	Level1b Control Parameters	Radiometric	7	1.4
$SAT\_STRAY\_THR^{RR}$	Threshold on saturated RR samples count to flag for stray light risk	6.2	16	9	Level1b Control Parameters	Straylight Evaluation Parameters	7	1.4
$SAT\_STRAY\_THR^{FR}$	Threshold on saturated FR samples count to flag for stray light risk	6.2	16	8	Level1b Control Parameters	Straylight Evaluation Parameters	7	1.4
$SRDF^{RR}_{m,sr,b}[k]$	RR Spectral Region Distribution Function for region sr contribution to stray light of band b	6.1	9	3 to 17	Instrument	FR Spectral Region Distribution Function	7	1.4
$SRDF^{FR}_{m,sr,b}[k]$	FR Spectral Region Distribution Function for region sr contribution to stray light of band b	6.1	8	3 to 17	Instrument	RR Spectral Region Distribution Function	7	1.4
$Nright^{RR}$, $Nleft^{RR}$	half-extent in forward and backward directions respectively of RR SRDF (total extent is Nleft+1+Nright)	6.1	5	from 17 & 18	Instrument	Instrumental Parameters	7	1.4
$Nright^{FR}$, $Nleft^{FR}$	half-extent in forward and backward directions respectively of FR SRDF (total extent is Nleft+1+Nright)	6.1	5	from 15 & 16	Instrument	Instrumental Parameters	7	1.4
α^{RR}[b,k,m]	product of optics transmission by CCD spectral response	6.3	11	all	Radiometric Calibration	RR Optics x CCD response	7	1.4
α^{FR}[b,k,m]	product of optics transmission by CCD spectral response	6.3	10	all	Radiometric Calibration	FR Optics x CCD response	7	1.4

Variable	Descriptive Name	IODD section	IODD table	Parameter number	Product Name	ADS	DPM section	Algorithm step
W^{RR}[sr,k]	radiance across-track weighting factors for RR	6.1	9	18	Instrument	RR Spectral Region Distribution Function	7	1.4
W^{FR}[sr,k]	radiance across-track weighting factors for FR	6.1	8	18	Instrument	RR Spectral Region Distribution Function	7	1.4
P[b,sr]	interpolation coeff for spectral region flux estimation	6.2	16	6	Level1b Control Parameters	Straylight Evaluation Parameters	7	1.4
b_{ref}	Band index for default radiance R_{ref}	6.2	16	2	Level1b Control Parameters	Straylight Evaluation Parameters	7	1.4
R_{ref}	Default radiance for pixels with all bands saturated	6.2	16	3	Level1b Control Parameters	Straylight Evaluation Parameters	7	1.4
F_{0b}	Extra-terrestrial Sun irradiance at reference date	6.2	6	1	Level1b Control Parameters	Solar Parameters	7	1.4
Def_rad_O_b	Default radiance value for samples above range limits	6.2	8	2	Level1b Control Parameters	Exception Handling	7	1.4
VECTOR_SOURCE	code for type of Orbit File	5.2	N/A	N/A	Level 0	MPH	8	1.5
VECTOR_FILE	Orbit File name	5.2	N/A	N/A	Level 0	SPH	8	1.5
Mt	number of MERIS modules	6.1	5	1	Instrument	Instrumental Parameters	8	1.5
K^{FR}	number of FR columns in a MERIS module	6.1	5	2	Instrument	Instrumental Parameters	8	1.5
K^{RR}	number of RR columns in a MERIS module	6.1	5	3	Instrument	Instrumental Parameters	8	1.5
JD0, JD1	JD of first and last frames in Level0 product	5.2	N/A	N/A	Level 0	MPH	8	1.5
ϕ_{SSP0}, ϕ_{SSP1}	latitude of SSP for first and last frames of the Level 0 product	5.2	N/A	N/A	Level 0	SPH	8	1.5
ϕ_{centre}, λ_{centre}	latitude, longitude of FR scene centre	N/A	N/A	N/A	Work Order	N/A	8	1.5
image_type	FR image type : imagette or scene	N/A	N/A	N/A	Work Order	N/A	8	1.5
begin_time, end_time	time of first and last frame to process	N/A	N/A	N/A	Work Order	N/A	8	1.5
Consolidated_processing	Switch enabling Consolidated Processing options	N/A	N/A	N/A	Work Order	N/A	8	1.5
NC^{IM}	Image AC size for FR imagette	6.2	14	2	Level1b Control Parameters	Resampling	8	1.5
NC^{FR}	Image AC size for FR scene	6.2	14	3	Level1b Control Parameters	Resampling	8	1.5
NC^{RR}	Image AC size for RR product	6.2	14	4	Level1b Control Parameters	Resampling	8	1.5
DT_frame^{FR}	Bias for FR frame time correction	6.1	5	6	Instrument	Instrumental Parameters	8	1.5
DT_frame^{RR}	Bias for RR frame time correction	6.1	5	7	Instrument	Instrumental Parameters	8	1.5
Re	Mean Earth radius	6.2	10	1	Level1b Control Parameters	Geolocation	8	1.5
resampling_switch	switch enabling re-sampling process	6.2	14	1	Level 1b Control Parameters	Resampling	8	1.5
NJ	Number of tie points for full swath	6.2	10	2	Level1b Control Parameters	Geolocation	8	1.5
Dx_t	Across-track tie points pitch	6.2	10	3	Level1b Control Parameters	Geolocation	8	1.5
DJ^{FR}	Across-track pixel to tie point subsampling factor in FR	6.2	14	5	Level1b Control Parameters	Resampling	8	1.5

Variable	Descriptive Name	IODD section	IODD table	Parameter number	Product Name	ADS	DPM section	Algorithm step
DJ^{RR}	Across-track pixel to tie point subsampling factor in RR	6.2	14	6	Level1b Control Parameters	Resampling	8	1.5
DF^{FR}	Along-track frame to tie frame subsampling factor in FR	6.2	14	7	Level1b Control Parameters	Resampling	8	1.5
DF^{RR}	Along-track frame to tie frame subsampling factor in RR	6.2	14	8	Level1b Control Parameters	Resampling	8	1.5
DT^{FR}	Delay between two FR frames	6.1	5	4	Instrument	Instrumental Parameters	8	1.5
DT^{RR}	Delay between two RR frames	6.1	5	5	Instrument	Instrumental Parameters	8	1.5
$max\_d\psi^{FR}$	Maximum across track angular distance allowing pixel selection in FR	6.2	14	10	Level1b Control Parameters	Resampling	8	1.5
$max\_d\psi^{RR}$	Maximum across track angular distance allowing pixel selection in RR	6.2	14	11	Level1b Control Parameters	Resampling	8	1.5
$resamp\_pix^{FR}_{k,m}$	FR pixels resampling selection map	6.1	6	3	Instrument	FR Pointing	8	1.5
$resamp\_pix^{RR}_{k,m}$	RR pixels resampling selection map	6.1	7	3	Instrument	RR Pointing	8	1.5
$U^{FR}_{x\ k,m}$	x component of MERIS FR pixels pointing unit vectors	6.1	6	1	Instrument	FR Pointing	8	1.5
$U^{FR}_{y\ k,m}$	y component of MERIS FR pixels pointing unit vectors	6.1	6	2	Instrument	FR Pointing	8	1.5
$U^{RR}_{x\ k,m}$	x component of MERIS RR pixels pointing unit vectors	6.1	7	1	Instrument	RR Pointing	8	1.5
$U^{RR}_{y\ k,m}$	y component of MERIS RR pixels pointing unit vectors	6.1	7	2	Instrument	RR Pointing	8	1.5
$\Delta\phi^{FR}$	Along-track depointing of MERIS pixelangle corresponding to one FR frame	6.1	6	4	Instrument	FR Pointing	8	1.5
$\Delta\phi^{RR}$	Along-track depointing angle corresponding to one RR frame	6.1	7	4	Instrument	RR Pointing	8	1.5
AOCS[3]	Pitch, roll, yaw amplitude	6.9	4	1 to 3	Platform Attitude	AOCS Parameters	8	1.5
Att_error_model[]	Attitude error model data base	6.9	5	all	Platform Attitude	MERIS Attitude Perturbation	8	1.5
DEM[lon,lat]	Digital elevation model	6.4.1	N/A	N/A	Digital Elevation	N/A	8	1.5
DRM[lon,lat]	Digital roughness model for land pixels	6.4.2	N/A	N/A	Digital Roughness	N/A	8	1.5
glint_thr_zen	threshold on zenith angle difference for glint mask	6.2	11	6	Level1b Control Parameters	Flagging	8	1.5
glint_thr_azi	threshold on azimuth angle difference for glint mask	6.2	11	5	Level1b Control Parameters	Flagging	8	1.5
sp_shift^{FR} [k,m]	spectral shift index for MERIS FR pixels	6.1	12	1	Instrument	FR Spectral Shift	8	1.5
sp_shift^{RR} [k,m]	spectral shift index for MERIS RR pixels	6.1	13	1	Instrument	RR Spectral Shift	8	1.5
ΔF^{FR}	tie points frame spacing	6.2	14	7	Level1b Control Parameters	Resampling	9	1.6
ΔF^{RR}	tie points frame spacing	6.2	14	8	Level1b Control Parameters	Resampling	9	1.6
ΔJ^{RR}	tie points column spacing	6.2	14	5	Level1b Control Parameters	Resampling	9	1.6
ΔJ^{FR}	tie points column spacing	6.2	14	6	Level1b Control Parameters	Resampling	9	1.6

Variable	Descriptive Name	IODD section	IODD table	Parameter number	Product Name	ADS	DPM section	Algorithm step
Land_Sea_Map.land	A priori classification atlas structure, land/ocean field	6.5	4 to 6	all	Land/Sea Mask	see AD12	9	1.6
Land_Sea_Map.coast	A priori classification atlas, true/false coastline field	6.5	7 to 9	all	Land/Sea Mask	see AD12	9	1.6
b_{test}	band index for reflectance test	6.2	13	8	Level1b Control Parameters	Classification	9	1.6
class_thr_t[θ_s,θ_v,$\Delta\varphi$]	Look-up table of threshold values	6.2	20	1	Level1b Control Parameters	Radiometric Thresholds LUT	9	1.6
Sat_rad_b	Saturation radiance values	6.2	11	8	Level1b Control Parameters	Flagging	9	1.6
F_{0b}	Extra-terrestrial Sun irradiance at reference date	6.2	6	1	Level1b Control Parameters	Solar Parameters	9	1.6
$Dsun0^2$	Square of Sun-Earth distance at reference date	6.2	6	2	Level1b Control Parameters	Solar Parameters	9	1.6
weather_product	Incoming weather product	6.6	N/A	N/A	ECMWF	N/A	10	1.7
Weather_grid	Spatial sampling grid for weather products	6.6	N/A	N/A	ECMWF	N/A	10	1.7
T_ECMWF	Time of used weather products	6.6	N/A	N/A	ECMWF	N/A	10	1.7
kind	Kind of weather product used	6.6	N/A	N/A	ECMWF	N/A	10	1.7
λ_0	longitude of first ECMWF grid point	6.6	N/A	N/A	ECMWF	N/A	10	1.7
ϕ_0	latitude of first ECMWF grid point	6.6	N/A	N/A	ECMWF	N/A	10	1.7
$\Delta\lambda$	ECMWF grid longitude steps	6.6	N/A	N/A	ECMWF	N/A	10	1.7
$\Delta\phi$	ECMWF grid latitude steps	6.6	N/A	N/A	ECMWF	N/A	10	1.7
P_db [loc]	Discretised global field of pressure at mean sea level	6.6	N/A	N/A	ECMWF	N/A	10	1.7
Wu_db [loc]	Discretised global field of wind at 10m u component	6.6	N/A	N/A	ECMWF	N/A	10	1.7
Wv_db [loc]	Discretised global field of wind at 10m v component	6.6	N/A	N/A	ECMWF	N/A	10	1.7
Oz_db [loc]	Discretised global field of total ozone	6.6	N/A	N/A	ECMWF	N/A	10	1.7
Rh_db [loc]	Discretised global field of relative humidity at 850 hPa	6.6	N/A	N/A	ECMWF	N/A	10	1.7
B	number of MERIS bands	6.3	4	2	Radiometric Calibration	Radiometric Correction Control Parameters	11	1.8
transmission_thresh	threshold for transmission errors flag	6.2	11	12	Level1b Control Parameters	Flagging	11	1.8
format_thresh	threshold for format errors flag	6.2	11	13	Level1b Control Parameters	Flagging	11	1.8
Tie_scale	structure of scaling factors for annotations	6.2	15	1 to 7	Level1b Control Parameters	Scaling Factors	11	1.8
Rad_scale_b	scaling factor for radiances	6.2	15	8	Level1b Control Parameters	Scaling Factors	11	1.8
pc_thresh_image	threshold for out_of_range flagging of image pixels	6.2	11	9	Level1b Control Parameters	Flagging	11	1.8
pc_thresh_blank	threshold for out_of_range flagging of blank pixels	6.2	11	10	Level1b Control Parameters	Flagging	11	1.8
$Wavelengths_b$	Band wavelengths	6.3	4	3	Radiometric Calibration	Radiometric Correction Control Parameters	11	1.8

Variable	Descriptive Name	IODD section	IODD table	Parameter number	Product Name	ADS	DPM section	Algorithm step
$Widths_b$	Band widths	6.3	4	4	Radiometric Calibration	Radiometric Correction Control Parameters	11	1.8
$IFOV^{RR}$	RR Instantaneous FOV	6.1	7	4	Instrument	RR Pointing		
$IFOV^{FR}$	FR Instantaneous FOV	6.1	6	4	Instrument	FR Pointing	11	1.8
OB_R	Reference for on-board processing switch	6.3	4	11	Radiometric Calibration	Radiometric Correction Control Parameters	11	1.8
$BAND\_GAIN\_R_{b,m}$	Reference for band gain settings	6.3	4	20	Radiometric Calibration	Radiometric Correction Control Parameters	11	1.8
OCL_R	Reference for OCL switch	6.3	4	10	Radiometric Calibration	Radiometric Correction Control Parameters	11	1.8
DT^{RR}	Delay between two frames	6.1	5	5	Instrument	Instrumental Parameters	11	1.8
DT^{FR}	Delay between two frames	6.1	5	4	Instrument	Instrumental Parameters	11	1.8
K^{RR}	Number of columns per MERIS module	6.1	5	3	Instrument	Instrumental Parameters	11	1.8
K^{FR}	Number of columns per MERIS module	6.1	5	2	Instrument	Instrumental Parameters	11	1.8
KB	Number of blank pixel columns per module	6.1	5	23	Instrument	Instrumental Parameters	11	1.8
Pix^{RR}	RR product pixel AC size	6.2	10	5	Level1b Control Parameters	Geolocation	11	1.8
Pix^{FR}	FR product pixel AC size	6.2	10	4	Level1b Control Parameters	Geolocation	11	1.8
DF^{RR}	RR product frame to tie frame sub-sampling factor	6.2	14	8	Level1b Control Parameters	Resampling	11	1.8
DF^{FR}	FR product frame to tie frame sub-sampling factor	6.2	14	7	Level1b Control Parameters	Resampling	11	1.8
DJ^{RR}	RR product column to tie point sub-sampling factor	6.2	14	6	Level1b Control Parameters	Resampling	11	1.8
DJ^{FR}	FR product column to tie point sub-sampling factor	6.2	14	5	Level1b Control Parameters	Resampling	11	1.8
DFSQ	tie frame to SQADS frame sub-sampling factor	6.2	14	9	Level1b Control Parameters	Resampling	11	1.8
Land_bands	set of bands used for land observation	6.2	11	11	Level1b Control Parameters	Flagging	11	1.8