Title: MERIS Level 2 Detailed Processing Model

Doc. no: PO-TN-MEL-GS-0006

Issue: 8

Revision: 0B

Date: 24 June 2011

	Function	Name	Company	Signature	Date
Prepared:		MERIS Team	ACRI		24/06/11
Approved:	Project Manager	L. BOURG	ACRI		24/06/11
Released:	Project Manager		ESA		

Copyright © 2011 ACRI S.T.

External Distribution

Name	Quantity
P. GORYL (ESA /ESRIN)	5
J-P. HUOT (ESA /ESTEC)	1

Internal Distribution

Name	Quantity
All ESL laboratories	1
L. BOURG	1
C. MAZERAN	1
C. LEREBOURG	1

Copyright © 2011 ACRI S.T.

Change Record

Issue	Revision	Date	Description	Approval
1	0	17/10/96	Initial issue	
1	1	23/10/96	Revised at end of DPDAS phase	
2	0	25/9/96	Complete re-issue	
3	Draft	29/11/96	Draft final specification	
3	0	19/12/96	Final issue	Yes
3	1	06/06/97	Revised following prototyping phase	Yes
3	2	15/10/97	Revision of data flow, new section on neural networks	Yes
3	3	24/11/97	Minor typos corrections; change pages: 3-3, 5-1, 5-2, 6-1, 7-5, 7-19, 7-45 to 7-75, 9-14	No (internal release)
3	4	15/12/97	Revision of neural network interface: 2-6; 5-5, 5-6; 7-52, 7-53, 7-67, 7-77; 9-2 to 9-17	Yes
4	Draft	23/07/98	Major upgrade /revision of all algorithms	
4	0	13/11/98	revision and cleanup; simplification (consolidation, coastal zone); completion (no TBDs)	
4	1	18/12/98	following review (PO-MN-ACR-GS-0026)	Yes
4	2	17/12/99	All pages: revision of document structure and of all algorithms	
4	3	17/12/99	Revision after ESA comments. All pages: supersedes 4.2	
4	4	25/02/00	Case 2 waters products PCD (section 10, p 10-16 & 10-17)	

4	5	07/09/01	Revision after ESA comments	
	Page #	**Section #**	**Comments**	
	3-3	§3.4	Origin/Destination of Variables clarified	
	4-3	§4.4	Origin/Destination of Variables clarified	
	5-6	§5.4	Origin/Destination of Variables clarified	
	5-11	§5.4, table of variables, outputs	Variables ORINP0_F and OROUT0_F added to list of outputs	

Copyright © 2011 ACRI S.T.

	5-15	Eq. 2.6.12.1-2	U_{O3}4x4 divided by 1000. to convert ozone content in proper unit	
	5-18	§5.5.6	text "table 3.4" changed to "table of variables in section 3.4"	
	6-4	§6.4	Origin/Destination of Variables clarified	
	7-3	§7.4	Origin/Destination of Variables clarified	
	7-4	§7.5.1	text "RD7, 3.1.4" changed to "RD7, 3.17.2.3"	
	8-6	§8.2.2	Origin/Destination of Variables clarified	
	8-13	§8.3.2	Origin/Destination of Variables clarified	
	8-14	§8.3.2	Variable ang_0 changed from type s (specified) to type c (computed)	
	8-16	§8.3.2	Variable SPM_{br}(j,f) changed from type o to type c	
	8-16	§8.3.2	Variable ang(j,f) changed from type o to type c	
	8-17	§8.3.2	Variable ia1(j,f) changed from type o to type c	
	8-17	§8.3.2	Variable ia2(j,f) changed from type o to type c	
	8-31	§8.4.2	Origin/Destination of Variables clarified	
	8-31	§8.4.2	Variable μ_s changed from type i to type c	
	8-31	§8.4.2	Variable μ changed from type i to type c	
	8-61	§8.4.3.8.3	Equations 2.6.9.2.4-10 & -11 added	
	8-76	§8.4.3.13.4.1	"fatab_LUT" -> "$f_a$tab_LUT" "ωatab_LUT" -> "$\omega_a$tab_LUT"	
	8-86	§8.4.4	Bits 3 and 4 of "ANNOT" flag register added in description	
	8-90	§8.5.4	Origin/Destination of Variables clarified	
	8-91	§8.5.4	"par_LUT" -> "PAR_LUT"	
	9-3	§9.2.2	Origin/Destination of Variables clarified	
	9-4	§9.2.3.1	**Endif** end of bloc moved	

Copyright © 2011 ACRI S.T.

	9-7	§9.3.1.3	Added description of derivation of aerosol epsilon	
	9-11	§9.3.2	Origin/Destination of Variables clarified	
	9-15	§9.3.2	Variables $\alpha(j,f)$ and ia(j,f) changed from type o (output) to c (computed); variable $\varepsilon\_775\_865(j,f)$ added as type o and $\rho_a(b,j,f)$ as type c	
	9-15	§9.3.3.1	Note added specifying purpose and location of ***ref_rayleigh*** procedure	
	9-15	§9.3.3.1	Clarification that LUT interpolations are performed on 4x4 window geometry	
	9-19	§9.3.3.2	Note added specifying purpose and location of ***ref_aerosol*** procedure	
	9-20	§9.3.3.2	Equations 2.6.17.4-5 to –8 added for computation of aerosol epsilon	
	9-22	§9.3.3.6.1	Note added specifying purpose and location of ***ref_aerosol*** procedure	
	9-26	§9.4.2	Origin/Destination of Variables clarified	
	9-26	§9.4.2 & 9.4.3	Input variable ρ_G from step 2.6.18 (deleted step) changed to ρ_{top} from step 2.6.23	
	10-2	§10.3.1.2.3	Flag construction clarified	
	10-4	§10.4	variables P_CONFIDENCE_F and HIINLD_F of type i added	
	10-4	§10.4	variables TETAS_LIMIT of type s added	
	10-4 &10-5	§10.4	Origin of Input Variables clarified	
	10-10 & 10-11	§10.5.4	Flag construction clarified	
	10-17	§10.5.13	Fields ABSOA_CONT and ABSOA_DUST of table 10.5.13 updated (now filled from bits of "ANNOT" flag register)	
	12-1 & 12-2	§12	Cloud top pressure corrected to cloud optical thickness & table showing flags corrected accordingly	

Copyright © 2011 ACRI S.T.

5	0	14/09/01	Revision after SAG Recommendations	
	Page #	**Section #**	**Comments**	
	3-4	§3.5.1	Pressure is corrected for altitude only for land pixels (otherwise coastal pixels or small islands may get as altitude their bathymetry).	
	4-6	§4.5.2	Albedo for not land pixels is set to 0.	
	8-91	8.5.4	InvAbs_Chl2[2] and InvScat_SPM added as auxiliary parameters	
	8-98 8-99	§8.5.5.4.2	Case 2 IMT Neural Network computes optical properties that are latter converted to concentrations using aux. Param.	
	All pages of §9	§9	Infrared band is replaced by red band and Thresh_infr2nrinfr is renamed Thresh_nir2red	
	9-3, 9-4, 9-5, 10-14, 10-15	§9.2.2 §9.2.3.1 §10.4 §10.5.9	The two bands of rectified reflectances are made available in the L2 product	
	10-17, 10-18	§10.5.13	WATER_CLASS, LAND_CLASS, and CLOUD_CLASS are taken into account in the PCD_16 and PCD_17 computations	
	5-6, 5-8	§5.4	TOAR(b,j,f) as input param.; SATURATION_L(b) as aux. Param.	
	10-5, 10-6	§10.4	TOA radiances, rectified reflectances for red/near infrared bands are added as input parameters. Scaling factors and offsets for rectified reflectances, as well as radiance saturation levels are added as auxiliary parameters. Radiance saturation flag is computed.	
	5-11,5-12, 10-12, 10-17, 10-18	§5.5 §10.5.6 §10.5.13	PCD computations and exceptions are adjusted using radiance saturation levels.	
	12-1, 12-2	§12	Include Cloud Top Pressure MDS in MER_RRC_2P and MER_LRC_2P	

Copyright © 2011 ACRI S.T.

5	**1**	**26/07/02**	**Revision after Trouble-shooting activities**	
	Page #	**Section #**	**Comments**	
	5-9, 5-15,5-16	§5.5.4	Gaseous corrections performed in reflectances instead of radiances	
	10-6	§10.4	Table showing input parameters to product formatting (Table 10.4-1) shows the band to be considered in the formatting (blue_band_N, boavi_red_band, etc.) (SPR-3L000-1060-GMV)	
	8-99	§8.5.5.5	Note has been added saying that the ozone indexing values for the PAR LUT should be scaled to reflect the unit of the ozone used in the processing (SPR-3L000-1062-GMV)	
	8-21	§8.3.3.4.2	The criteria of linear system degeneration (Eq. 2.6.10.2-15) is more explicit (det<1.e-9) (SPR-3L000-1063-GMV)	
	9-15 to 9-24	§9.3.3	Replaced θ_s by θ_{s_4x4} to remove any ambiguity (same for θ_v, $\Delta\phi$ and M)	
	8-9	§8.2.3.1	Corrected bands on which to perform the glint correction	
	8-72	§8.4.3.12.3	Modified exception processing condition and added Note for bands not shown	
	8-37	§8.4.3.1.3	Call to **Process initialisation** function moved	
	8-63	§8.4.3.9.3	**Final_couple** function consolidated	
	8-79 and 8-80	§8.4.3.13.6	μ_s removed from function call and table of variables because not used	
	8-57	§8.4.3.7.3	Computation of τ_a(b865) specified and call to **Denormaliser** function corrected	
	8-53	§8.4.3.6.2	$\tau_{a\ loc}$ replaced by $\tau_{a\ 865}$ for clarity	
	8-82	§8.4.3.13.8.2	Added comment for clarity	
	8-74	§8.4.3.13.2.2	Clarified exception handling	
	8-75	§8.4.3.13.3.2	Clarified exception handling	
	8-34	§8.4.2	Added ORINP0_F(j,f) in table	

Copyright © 2011 ACRI S.T.

6	**0**	**8/11/02**	**Revision after verification activities**	
	Page #	**Section #**	**Comments**	
	1-4, 1-8	1.3	Update of tables 1.3-1 & 1.3-2 for new algorithms steps in sections §4 & §5 and for sub-sections re-numbering in section 4	
	3-3, 3-5	§3.4 & §3.5	Modification of TOA reflectance computation: use of per pixel Sun irradiances (selected using the MERIS Detector index now carried along in the Level 1b flags MDS instead of the spectral shift) instead of mean values.	
	4-3 to 4-8	§4.4 & §4.5	Improvement of pressure estimations: surface pressure is linearly interpolated between the results of two polynomials, CTP neural net now takes b761 wavelength as input.	
	5-2, 5-4 to 5.10, 5-14 to 5-17, 5-21	§5.4 & §5.5	Addition of a reflectance correction for Smile Effect, modification of TOA radiance passed to §6 & §7	
	8-9	§8.2.3.1	MEDGLINT_F flag set to TRUE when high glint to pursue the glint correction even above the highest glint threshold	
	8-72 8-80	§8.4.3.12 §8.4.3.13.6	Calls to *transmittance_d* replaced by additional calls to *transmittance_up*	
	8-97	§8.5.5.3	ORINP1 not positioned anymore in Equations 2.9.6-6 and 2.9.6.8	
	10-17 and 10-18	§10.5.13	MIXR1 removed from the construction of PCD_1_13, PCD_15, PCD_18 and PCD_19	
	10-17 and 10-18	§10.5.13	CASE2_ANOM removed from the construction of PCD_15	
	11-1	§ 11	Added smile corrected reflectance to breakpoints	

Copyright © 2011 ACRI S.T.

6	**1**	**28/03/03**	**Revision after first Validation Results**	
	Page #	**Section #**	**Comments**	
	5-7 & 5-17	5.5	Modified steps 2.6.26-4 & -5 using additional variables read from auxiliary products	
	6-4, 6-6 & 6-7 8-3, 8-6, 8-7, 8-9 & 8-10	6.4 & 6.5 8.2.2 & 8.2.3	Water vapour processing now uses a dedicated high glint flag (instead of one of the water processing ones previously. To minimise modification impact, computation of new flag has been kept within water processing.)	
	8-8	8.2.3	Computation of angular distance between wind and Sun directions modified (step 2.6.5.11-2)	
	8-13, 8-14 & 8-18	8.3.2 & 8.3.3	Use of a threshold on pixel altitude (bathymetry) to force CASE2_F flag for shallow waters	
	8.57 8.69 8.79	8.4.3.7.3 8.4.3.11.3 8.4.3.13.4.2	Use of weighted averages for aerosol couples properties computations	
	8-74 to 8-77	8.4.3.13.2 8.4.3.13.3	Modified interpolation scheme for the ρ_{path}/ρ_R LUT for both direct and inverse use.	
	9-3 9-4 & 9-5	9.2.2 9.2.3	New flags defined to sort out pixels within TOAVI processing, those explaining algorithm failure are passed to formatting.	
	10-5 & 10-18	10.4 10.5	Addition of TOAVI dedicated Science Flags to table 10.5.13	

6	**1a**	**16/05/03**	**Revised after comments on issue 6.1**	
	Page #	**Section #**	**Comments**	
	4-6	4.5	Typo corrected in step 2.1.3.-3	
	4-7	4.5	Exception processing corrected in step 2.1.12-5	
	8.75	8.4.3.13.2	Exception processing updated	
	9.4	9.2.3.1	Applicability of exception processing extended	
	9.5	9.2.3.1	Additional exception processing	

Copyright © 2011 ACRI S.T.

7	**0**	**18/06/04**	**Algorithm revisions from validation inputs**	
	Page #	**Section #**	**Comments**	
	4-7	4.5.3	Exception processing after step 2.1.12-4 deleted	
	5-7, 5-9, 5-10	5.4	Additional *s* and *c* types variables for upgraded gas absorption correction (§ 5.5.4, step 2.6.12)	
	5-9, 5-10	5.4	Additional *c* and *o* types variables for upgraded Land identification (§ 5.5.4, step 2.6.26)	
	5-17 to 5-19	5.5.4	Upgraded gas absorption correction (step 2.6.12)	
	5-20 to 5	5.5.5	Upgraded Land identification (step 2.6.26)	
	6-4	6.4	Additional *i* type variable, modified *s* type LUT (one additional dimension) for upgraded Water Vapour retrieval over Ocean (§ 6.5.1, step 2.3.2)	
	6-6	6.5.1	upgraded Water Vapour retrieval over Ocean (§ 6.5.1, step 2.3.2): LUT of polynomial coefficients now includes a wind speed dimension	
	8-1	8.1	Epsilon product replace by Angström exponent	
	8-2	8.1	Revised overall block diagram	
	8-3	8.2	A note on transfer of Glint reflectance computation to §5 has been added.	
	8-4	8.2	Description of new step 2.6.5.4	
	8-5, 8-6	8.2.1	Revised flowcharts 8.2-1 & 8.2-2	
	8-7, 8-8	8.2.2	Additional *i*, *s*, & *c* type variables to account for deletion of and addition of step 2.6.5.4	
	8-8, 8-9	8.2.3.1	Steps 2.6.12.1-5 to -8 and 2.6.5.1.1 deleted (actually moved to §5.5.4, step 2.6.26)	
	8-9	8.2.3.1	Step 2.6.5.1.8-2 added	
	8-10	8.2.3.4	Step 2.6.5.4 added as new section 8.2.3.4	
	8-13 to 8-21	8.3	Implementation of ATBD 2.6 totally reviewed	
	8-25	8.4.1.2	Test on atmosphere correction error at 705 deleted	

Copyright © 2011 ACRI S.T.

	8-27	8.4.2	Symbolic value CONT_LIKE renamed as BLUE_LIKE to reflect changes in aerosol classes	
	8-30	8.4.2	Output parameter ε_775_865(j,f) replaced by α_775_865(j,f)	
	8-33	8.4.3.1.2	Output parameter ε_775_865(j,f) replaced by α_775_865(j,f)	
	8-35	8.4.3.3.1	Flowchart updated (test on CASE2_S flag replaced by test on BPAC_ON_F flag)	
	8.36	8.4.3.3.2	CASE2_S replaced by BPAC_ON_F	
	8.37	8.4.3.3.3	CASE2_S replaced by BPAC_ON_F	
	8-38	8.4.3.4.1	Flowchart updated (test on absorbing aerosols by-passed when CASE2_S flag is set)	
	8-39	8.4.3.4.1	Flowchart updated (ε_775_865 replaced by α_775_865)	
	8-41	8.4.3.4.2	Output parameter ε_775_865(j,f) replaced by α_775_865(j,f)	
	8-42	8.4.3.4.3	Step 2.6.9.2-10 updated (test on absorbing aerosols by-passed when CASE2_S flag is set)	
	8-45 to 8-47	8.4.3.5.2 & 8.4.3.5.3	Classification of aerosols modified, hence climatology switches	
	8-47	8.4.3.5.3	Revision of aerosol database classes and scan logic	
	8-51	8.4.3.7.1	Flowchart updated (b705 no more used)	
	8-52, 8-53	8.4.3.7.2	New *i*, *s* & *c* type parameters related to	
	8-53, 8-54	8.4.3.7.3	b705 no more used	
	8-54	8.4.3.7.3	Computation of climatologic water leaving radiance at 510 nm	
	8-55	8.4.3.8.1	Flowchart updated (b705 no more used)	
	8-56	8.4.3.8.2	New *i*, *s* & *c* type parameters related to climatology of $\rho_w(510)$, parameters related to b705 deleted, aerosol classes updated	
	8-57	8.4.3.8.3	Algorithm revision: use climatology of $\rho_w(510)$, do not use b705 anymore	
	8-59	8.4.3.9.2	parameters related to b705 deleted, new parameter related to revised aerosol classes	

	8-59, 8-60	8.4.3.9.3	b705 no more used, new steps for setting bits AERO_B and ABSO_D of ANNOT flag register	
	8-62, 8-63	8.4.3.10.2 & 8.4.3.10.3	Updated after revision of aerosol database classes	
	8-64	8.4.3.11.1	Flowchart updated (ε_775_865 replaced by α_775_865)	
	8-65	8.4.3.11.2	*o* type parameter α_775_865 (j,f) replaces ε_775_865 (j,f), new *s* type parameter needed for its computation	
	8-66	8.4.3.11.3	Computation of α_775_865	
	8-68	8.4.3.12.3	Exception processing modified	
	8-84	8.4.4	ANNOT flag register description updated	
	8-84	8.4.5	α_775_865 replaces ε_775_865	
	8-86	8.5.2	Flowchart updated (ε_775_865 replaced by α_775_865)	
	8-88	8.5.4	*i* type parameter α_775_865 replaces ε_775_865)	
	8-89	8.5.4	New *s* & *c* type parameters supersedes old ones for anomalous scattering detection	
	8-89	8.5.4	Additional *s* type parameters for case2 IMT NN	
	8-95	8.5.5.3	Step 2.9.6 fully revised	
	8-96, 8-97	8.5.5.4.2	Case2 IMT NN inputs checks and transformation modified	
	8-97	8.5.5.5	PAR computation uses α_775_865	
	9-7	9.3.1.2	Blue band used in DDV screening (step 2.6.13) changed to 442	
	9-11, 9-12	9.3.2	Description of some *s* type variables aligned to cope with increased aerosol data base size	
	9-12	9.3.2	New *s* type variables	
	9-13, 9-14	9.3.2	New or modified *c* type variables	
	9-15	9.3.2	*o* type variable ε_775_865 replaced by α	
	9-17	9.3.3.2	Blue band used in DDV screening (step 2.6.13) changed to 442	
	9-17	9.3.3.2	New steps 2.6.13-10 & -11	
	9-17	9.3.3.2	DDV flag setting modified	
	9-18 to 9-20	9.3.3.3	Pseudo-code aligned to cope with increased aerosol data base size	

Copyright © 2011 ACRI S.T.

	9-20		Step 2.6.17.4-14 renumbered to 2.6.17.4-10, new steps 2.6.17.4-11 to 2.6.17.4-16 added	
	9-20, 9-21	9.3.3.3	output ε_775_865 replaced by α, corresponding exception processing revised	
	9-22	9.3.3.4	All steps within procedure calc_robar_ag renumbered for clarity	
	9-23	9.3.3.5	All steps within procedure calc_robar_ra renumbered for clarity	
	9-24 to 9-26	9.3.3.6	Procedure top_of_Rayleigh_ref deeply revised, including new *i* and *s* type variables. All steps within the procedure renumbered for clarity.	
	9.28 to 9.29	9.4	Whole section revised following complete algorithm change	
	10-3	10.3.1	Description of MDS 19 aligned with Angström exponent replacing epsilon	
	10-3	10.3.1	MDS width (pixels) for FR Full swath product added	
	10-4, 10-5	10.4	New/modified *i* parametesr	
	10-17, 10-18	10.5.13	Table 10.5.13 modified	
	11-2, 11-3	11	New/modified breakpoints fields	
	12-1	12	Clarification on Water Vapour averaging	
	Annex A	All	Parameters Data List reviewed	

Copyright © 2011 ACRI S.T.

7	**1**	**28/02/05**	**Algorithm revisions from validation inputs**	
	Page #	**Section #**	**Comments**	
	8-13	8.3.1.2	Addition of the White Scatterer Identification to Step 2.6.8.2	
	8-16	8.3.2	Additional s and c type variables	
	8-17	8.3.2	Additional o type variable	
	8-18 & 8.19	8.3.3.3	Section title (step name) revised, equations added to include “White Scatterer Identification”	
	8-24	8.4.1.1	Correction of Path Reflectance step overview	
	8-25	8.4.1.2	Clarification of MERIS Aerosol Model step overview	
	8-26	8.4.2	Additional i type variables	
	8-27	8.4.2	Correction of s type variable name Modification of s type variable	
	8-39	8.4.3.4.2	Additional i type variables	
	8-41	8.4.3.4.3	Modification of step 2.6.9.2-10	
	8-54	8.4.3.8.2	Correction of s type variable name	
	8-64	8.4.3.11.3	Correction of step 2.6.9.2.7-8	
	8-87	8.5.3.2	Modification Chl1 retrieval Algorithm description	
	8-88	8.5.3.3	Modification of an s type variable	
	8-89	8.5.3.3	Modification of an s type variable, addition of 2 s & 2 c type variables	
	8-92 & 8-93	8.5.5.2	Revision of step 2.9.7, including complete equation re-numbering	
	8-95 & 8-96	8.5.5.4.2	Modification of steps 2.6.11-6 to 2.6.11-11	
	8-96	8.5.5.4.2	Modification of steps 2.6.11-13	
	9-12	9.3.2	Modification of 4 s type variables	
	9-17	9.3.3.2	Modification of step 2.6.13-10	
	9-19	9.3.3.3	Modification of step 2.6.17.3-49	
	9-24	9.3.3.6.1	Additional i type variables	
	9-25	9.3.3.6.2	Modification of steps 2.6.17.3.3-14, 2.6.17.3.3-15 and 2.6.17.3.3-17	
	9-29	9.4.2	Modified i, and s type variables	
	9-29	9.4.3	Modification of test raising exception processing steps 2.8-1 and 2.8-2	
	9-30	9.4.3	Modification of step 2.8-5	
	10-4	10.4	Additional i type variable	
	10-14	10.5.9	Modification of step 2.10.10-1	
	10-17	10.5.13	Modification of the definitions of PCD_1_13 and PCD_15	
	10-18	10.5.13	Modification of the definitions of PCD_19, BLUE_AERO flag, LOW_PRESSURE flag	

Copyright © 2011 ACRI S.T.

7	**2**	**30/06/05**	**Algorithm revisions from validation inputs**	
	Page #	**Section #**	**Comments**	
	9-1	9.1	Name of aerosol products corrected	
	9-7	9.3.1.3	Mathematical description of step 2.6.17 updated	
	9-13	9.3.2	c type variables iaer2 and $\alpha 2$ removed	
	9-14	9.3.2	c type variables $\cos\Theta_{scatt}$, $Px\omega_0$, $Px\omega_{0\_2}$, C, and ρ_a deleted Name and description of o type variable τ_a modified	
	9-19	9.3.3.3	Step 2.6.17.4 modified	
	10-5	10.4	description of i variable τ_{a865} modified, new i variable τ_{a442} added	
	10-6	10.4	name of s variables epsilon_scale and epsilon_offset changed to alpha_scale and alpha_offset. Descriptions updated	
	10-8	10.5.3	step 2.10.3-20b updated	
	10-9	10.5.3	step 2.10.3-34 updated	
	10-16	10.5.12	Step/section name updated Step logic and equations adapted to the land/water differences for the optical thickness product	
	10-18	10.5.13	Science Flag BLUE_AERO replaced by OADB, equation updated.	
	11-2	11	Applicability of breakpoints variables aer_mix, f_a, ω_a, τ_a and ρ_a restricted to water pixels	
	11-2	11	Applicability of breakpoints variables SPM_{br} and $t\rho_w$_C2 now linked to flag BPAC_ON instead of CASE2_S	
	11-2	11	Applicability of breakpoints variable BOAVI no more linked to the DDV_F flag	

7	**2A**	**30/10/06**	**Clarifications & corrections after IPF/prototype convergence**	
	Page #	**Section #**	**Comments**	
	4-10	4.5.5	Clarification of the procedure *pressure_func*	
	6-6 & 6-7	6.5	Modification of the pixel selection criteria to switch between Land and Ocean Water Vapour algorithms	
	8-19	8.3.3.3	Correction of step 2.6.8-13	
	8-20	8.3.3.4.2	Use correct name for variable ρ_w(b705) in step *(2.6.10.2-10)*	
	8-22	8.3.4 & 8.3.5	Update of sections Quality Control and Diagnostics and Exception Processing to align with Equations sections	
	8-34	8.4.3.3.3	Added missing step *(2.6.9.1-5)*	
	8-44	8.4.3.5.3	Corrected the bit number value applicable to PRE_BLUE	
	8-50	8.4.3.7.2	Corrected the definition of variables JDCurrMonth and JDNextMonth (from first days of current and next month to mid-month days)	
	8-54	8.4.3.8.2	Corrected the definition of variables JDCurrMonth and JDNextMonth (from first days of current and next month to mid-month days)	
	8-60	8.4.3.10.2	Wording: “Continental” replaced by “Blue”	
	9-11 & 9-19	9.3.2 & 9.3.3.3	Clarification on which wavelengths shall be used to compute the Angström exponent in step *(2.6.17.3-46)*	
	12-2	12	Chapter 12 corrected to be in-line with chapter 11.	

Notes:

- change bars left relative to issue 7.1.
- Unchanged sections left to issue 7.2.

Copyright © 2011 ACRI S.T.

8	**0**	**15/07/09**	**Algorithm revisions from validation inputs**	
	Page #	**Section #**	**Comments**	
	1-2	1.2	Figure 1.2-1modified to reflect changes in Meris level 2 general control flow	
	1-3, 1-4	1.3	Table 1.3-1 “Algorithm step index, hierarchical order” updated	
	1-7, 1-9	1.3	Table 1.3-1 “Algorithm step index, numerical order” updated	
	3-3 & 3-4	3.4	Changes in i/o table due to saturation checks moved to section 3.5.2	
	3-6	3.5.2	Check saturation of TOA radiances, set flags	
	4-2	4.3.1	Mathematical description of Atmosperic Pressure Estimates modified	
	4-3	4.4	Changes in list of type *i* variables linked to change in general control flow: pressure estimates are now computed ***after*** pixel identification	
	4-3 to 4-4	4.4	New/modified *s* & *c* type variables linked to evolutions of Pressure algorithms	
	4-4	4.4	Changes in list of type *o* variables linked to changes in Pressure Estimates as well as in downstream algorithms	
	4-5	4.5	Introductory text modified	
	4-5	4.5.1	NN initialisation now includes Surface Pressure NN	
	4-5 to 4-7	4.5.2	Full section revised: step 2.1.12, Surface Pressure now estimated by a Neural Net	
	4-7, 4.8	4.5.3	Evolutions of Cloud Top Pressure algorithm	
	5-2	5.2	Flowchart of Pixel Identification modified	
	5-3	5.3	Cloud screening description modified	
	5-6 to 5-9	5.4	New type *i* and type *s* variables	
	5-9 to 5-10	5.4	New type *c* variables	
	5-11	5.4	New type *o* variables	
	5-12 to 5-14	5.5.1 & .2	Modifications in step 2.1.7 & 2.1.8	
	5-14	5.5.3	Clarification in Eq. 2.1.9-5	
	5-16, 5-17	5.5.3	Eq. 2.1.9-17, 2.1.9-19 & 2.1.9-21 deleted	
	5-17	5.5.4	Eq. 2.6.12.1-1 moved, existing unnumbered equations numbered, list of bands for average reflectance computation updated.	
	5-17, 5-18	5.5.4	Step 2.6.12.2 deeply modified	
	5-20	5.5.5	Glint reflectance estimation now done at every pixel	
	5-21	5.5.5	Systematic land/water re-classification over a priori land now subject to a condition on surface elevation	
	5-21, 5-22	5.5.5	Radiometric tests for land/water classification clarified	
	5-22 to 5-23	5.5.5	Step 2.1.6 modified for water pixels	
	5-28 to 5-29	5.5.9	Procedure Pscatt added	
	5-29 to 5-30	5.5.10	Procedure P1 added	
	5-30 to 5-31	5.5.11	Procedure compute_pressure added	

Copyright © 2011 ACRI S.T.

	6-1, 6-2	6.2	Algorithm Overview reviewed including Fig. 6.2-1	
	6-3	6.3	Mathematical description reviewed to account for new Water Vapour algorithm over Land pixels	
	6-4, 6-5	6.4	List Of Variables reviewed to account for new Water Vapour algorithm over Land pixels	
	6-6	6.5	Neural Net initialisation added, computation of radiance from reflectance added, GoodPix condition revised	
	6-7	6.5.1	Condition modified for WV over Land or Cloud	
	6-7	6.5.2	Range checks for WV over Land or Cloud modified	
	6-7, 6-8	6.5.3	Full revision of step 2.3.1	
	6-8	6.5.4	Surface albedo computation introduced	
	6-9	6.5.6	Range checks for Land algorithm modified	
	6-10	6.5.7.2	Exception processing in function Water_Vapour_Polynomial improved	
	7-3	7.4	Lst Of Variables updated to account for algorithm evolutions	
	7-4	7.5	Eq. 2.4-1 added	
	8-3	8.2.1	Mention of processing on 4x4 sub-windows removed	
	8-4	8.2.1.4	Description of Vicarious Adjustment step added	
	8-7	8.2.2	List of variables updated	
	8-8 to 8-10	8.2.3	Processing on 4x4 sub-windows removed	
	8-9	8.2.3.1	SNOW_ICE_F flag combined to BRIGHT_F flag	
	8-10	8.2.3.5	Vicarious adjustment (step 2.6.5.5) added	
	8-35	8.4.2	Band 885 added to the set that are corrected for atmosphere, Note added on variables not used anymore but still required as breakpoints	
	8-35 to 8-38	8.4.2	List of Variables updated.	
	8.4.3.1.2	8-41, 8-42	Extension of applicable band set to 885	
	8.4.3.3.2	8-45	Extension of applicable band set to 885	
	8.4.3.3.3	8-46	Pressure correction moved to section 8.3.3.2	
	8.4.3.4.2	8-49, 8-50	Extension of applicable band set to 885	
	8.4.3.4.3	8-52	Extension of applicable band set to 885	
	8.4.3.6.2	8-59	Cosines replace angles as inputs variables for Sun and View zenith (List of Variables)	
	8.4.3.6.3	8-59	Cosines replace angles as inputs variables for Sun and View zenith (call to ***spectral interpolation of*** ζ)	
	8.4.3.7.2	8-62	New type *c* variables	
	8.4.3.7.3	8-63	Cosines replace angles as inputs variables for Sun and View zenith (call to ***Aerosol parameters per band***)	
	8.4.3.7.3	8-64	Calls to ***transmittance_up*** and ***transmittance_d*** and ***Denormaliser*** modified according to new interfaces	
	8.4.3.11.2	8-75	Cosines replace angles as inputs variables for Sun and View zenith, extension of applicable band set to 885 (List of Variables)	

Copyright © 2011 ACRI S.T.

	8.4.3.11.3	8-76	Extension of applicable band set to 885, cosines replace angles as inputs variables for Sun and View zenith (call to ***Aerosol parameters per band***), comments added	
	8.4.3.12.2	8-78	Several changes in variables list	
	8.4.3.12.3	8-78	Several Eq. modified (setting of RWNEG, transmittance computations)	
	8.4.3.13.1	8-79	Cosines replace angles as inputs variables for Sun and View zenith (list of variables, call interface to routines)	
	8.4.3.13.2	8-80, 8-81	Cosines replace angles as inputs variables for Sun and View zenith (list of variables, interpolation in LUT)	
	8.4.3.13.3	8-82, 8-83	Cosines replace angles as inputs variables for Sun and View zenith (list of variables, interpolation in LUT)	
	8.4.3.13.4	8-84, 8-85	Cosines replace angles as inputs variables for Sun and View zenith (list of variables, call interface to routines)	
	8.4.3.13.5	8-86, 8-87	Fully revised routine	
	8.4.3.13.6	8-88	Updated interface, some steps removed (moved to call level)	
	8.4.3.4.13.7	8-89	Extension of applicable band set to 885	
	8.4.3.4.13.8	8-90	Extension of applicable band set to 885	
	8.4.3.13.10	8-93, 8-94	Fully revised routine	
	8.5.3.2	8-98	Parameter f0 introduced in description of bi-directionality correction	
	8.5.4	8-99	New variables added	
	8.5.5.2	8-104	Exception processing updated (Chl added), f0 computation added, use of irradiance reflectance (instead of bb_over_a) in reflectance ratio for Chl1 retrieval, range chack on ratio added.	
	9-3	9.2.2	Size of several *s* type variables updated to follow algorithm changes, one new *c* type variable.	
	9-4	9.3.2.1	Computation of normalised and rectified reflectance extended to TOAVI_BRIGHT class.	
	9-5, 9-6	9.3.2.1	Rectification and normalisation parameters now class-dependent, some Eq. Numbers added, Exception processing upon negative TOAVI added.	
	9-6	9.3.2.2	Class dependence introduced in function **Normalize_F**	
	9-7	9.3.2.3	Identification of parameters set for function **Polynomial_ratio** clarified.	
	9-12, 9-15	9.3.2	Introduction of a flexible set of bands (2 or 3 among 4) for the Land Aerosol algorithm, several auxiliary data now provided at 4 bands instead of 3 previously.	
	9-18	9.3.3.3	Addition of step 2.6.17.0 to load and check the user defined set of bands	
	9-19, 9-20	9.3.3.3	Introduction of flexible band set in step 2.6.17.3, introduction of the **RefCorr** function.	
	9-20	9.3.3.3	New exception processing upon ρ_G value	
	9-20	9.3.3.3	Modified call interface to ***top_of_rayleigh_ref***	

Copyright © 2011 ACRI S.T.

	9-21	9.3.3.3	Modified computation of angstrom exponent to account for variable number of bands	
	9.22	9.3.3.3	Modified exception processing	
	9-25	9.3.3.6	Modified interface and Equations for **top_of_rayleigh_ref**: C_Corr being passed as an input its computation has been deleted.	
	9-28, 9-29	9.3.3.7	New procedure RefCorr	
	10-4, 10-5	10.4	New/modified/deleted inputs, modified c type variables	
	10-9	10.5.4	Quality indicators pc_low_pol_press and pc_low_NN_press replaced by pc_bad_surf_press and pc_bad_cloud_press respectively and restricted to their relevant surface class.	
	10-12	10.5.6	Steps 2.6.10 7, 2.6.10 7a and 2.6.10 7b deleted	
	10-18	10.5.13	Flag SNOW_ICE added for LAND_CLASS pixels, typo corrected, flag LOW_PRESSURE deleted	
	11-1 to 11-4	11	Several variables added, deleted or modified in breakpoints list	

8	**0A**	**28/02/11**	**Algorithm revisions from validation inputs**	
	Page #	**Section #**	**Comments**	
	2-1	2.2	New applicable and reference documents AD7 and RD11	
	2-3	2.4.2	Symbol b779 added to the list of bands as equivalent to b12 or b775, as closer to the actual wavelength (use of b775 should be avoided)	
	3-1	3.2	Flowchart of Pre-Processing step updated.	
	4-7	4.5.3	Equation (2.1.5-2b) corrected	
	5-8	5.4	Rayleigh reflectance LUT rhoR_LUT renamed as $\rho_R$tab_LUT for consistency with other sections. A note is added regarding interpolation on cosines rather than angles.	
	5-9	5.4	Flag CloseToCoast_f added to computed variables	
	5-10	5.4	iaer_sa(i,j) added to computed variables	
	5-11	5.4	ρ_{R1}(b,j,f) added to output variables	
	5-13	5.5.1	Step 2.17-3 now applies to all bands but b900	
	5-14	5.5.1	Step 2.1.7-11 modified (calculation of BRIGHT_TOA_F)	
	5-14	5.5.1	Step 2.1.4-14: typo corrected	
	5-14, 5-15	5.5.1	Steps 2.1.7-20 and -21 added, step 2.1.7-17 to 2.1.7-19 corrected	
	5-15	5.5.1	Steps 2.6.26-14 to 2.6.26-19 moved from §5.5.5 to §5.5.1	
	5-15	5.5.2	Steps 2.1.8-0 added, 2.1.8-1 to -3 modified	
	5-16, 5-17	5.5.3	Typos in steps 2.1.9-10 & -14corrected, steps 2.1.9-17, -21 & -22 added	
	5-18	5.5.4	Typo corrected in band loop limits for step 2.6.12.1-8	
	5-18, 5-19	5.5.4	index $\Delta\lambda_{b705}$ replaced by iλ_{b705} for clarity steps (steps 2.1.3-2-b705 to 2.1.3-4-b705)	
	5-19	5.5.4	Use of band symbol b779 (instead of b775) generalised in steps 2.6.12.2-*	
	5-20	5.5.4	Steps 2.6.12.3-2-$a_{h\ abv}$, -$b_{h\ abv}$, -$c_{h\ abv}$, & -$d_{h\ abv}$ corrected	
	5-21	5.5.5	Steps 2.6.26-14 to 2.6.26-19 have been moved at the end of section 5.5.1	
	5-22	5.5.5	Typo in 2.6.26-22 corrected	
	5-23	5.5.5	LUT rhoR_LUT renamed as $\rho_R$tab_LUT for consistency with other sections. Interpolation on zenith angles **cosines** rather than angles (step 2.1.6-6)	
	5-28	5.5.9	Pscatt now takes ρ_{R1} as input for band 1 instead of (re)computing it internally: changes in input variables list and steps 2.1.7.1-3 & -4 deleted.	
	5-31	5.5.11	Exception processing added on step 2.1.7.3-1	
	5-31	5.5.11	Step 2.1.7.3-6 modified	
	6-4	6.4	List of variables revised.	
	6-7	6.5.1	step 2.6.5.1.9 transferred from §8.2.3 to §6.4	
	6-8	6.5.1	SNOW_ICE_F water pixels excluded from the Water algorithm scope	

Copyright © 2011 ACRI S.T.

	6-8	6.5.1	Typo corrected in 2.3.2-10 to -12	
	6-9	6.5.3	SNOW_ICE_F water pixels added to the Land algorithm scope	
	6-9	6.5.3	Steps 2.3.1-11, -13 & next exception processing modified	
	6-11	6.5.7.2	Exception processing modified	
	8-7	8.2.2	Variable ρ deleted, ρ^{*}_{ng} clarified, SNOW_ICE type corrected, τ_{R0} added	
	8-8	8.2.2	tau_atm and flag WV_HIGLINT deleted	
	8-8	8.2.3	Equations 2.6.12-1 & 2.6.12.1-10 renumbered 2.6.5-1 & 2.6.5-2 respectively.	
	8-9	8.2.3.1	Computation of tau_atm removed; Note about 4x4 sub-window removed	
	8-10	8.2.3.1	Step 2.6.5.1.9 move to section 6	
	8-12	8.3.1.2	Comment added about non-corrected glint pixels	
	8-14	8.3.2	Index of variable τ_{R0} corrected; ORINP0_F deleted	
	8-15	8.3.2	Typo in variable bs; additional breakpoint for ρ_w_fe	
	8-16	8.3.2	Variable ang_exp moved and include dependence on bandset; annotation flag ANNOT_BPAC added	
	8-17	8.3.3.1	BPAC_ON flag initialised; Transfer BPAC flags to annotation	
	8-18	8.3.3.2	Use τ_{R0} variable in transmittance computation; initialise internal flags do_bandset, error and converge	
	8-20	8.3.3.2	Update exception processing	
	8-21	8.3.3.2	Update determination of bandset	
	8-23	8.3.3.3.1	Include bandset dependence in ang_exp	
	8-24	8.3.3.3.2	Activate BPAC_ON flag in case of success, and remove ACFAIL in case of failure	
	8-26	8.3.3.5	Add N_Fp_coeff and use it instead of N_Fp_ord; change computation of Fp, with a warning	
	8-27	8.3.3.6	ρ_a type is only output	
	8-28, 8-29	8.3.3.7	Update initialisation and break on bb_old	
	8-30	8.3.4	Quality control and diagnostic has been totally revised with new ANNOT_BPAC flag. Note ORINP0 has been removed. Exception handling 1, 2 and 3 have been corrected.	
	8-31 to 8-33	8.4.1	Several typo corrections and clarifications about the use of tρ_w_C2	
	8-34	8.4.2	Source step of variable ρ_R changed	
	8-35	8.4.2	Range of tρ_w_C2 extended to 885 nm	
	8-37	8.4.2	Range/Remarks on τ_a updated	
	8-40	8.4.3.1.2	Variables ρ_{R0} and τ_{R0} updated as input	
	8-44	8.4.3.3.2	Variable ρ_{R0} added	
	8-45	8.4.3.3.3	Text clarification about the reflectance over clear waters.	
	8-48	8.4.3.4.2	Variable τ_{R0} used instead of $\tau_{R\ a}$	
	8-60	8.4.3.7.2	idem	

Copyright © 2011 ACRI S.T.

	8-73	8.4.3.11.2	Variables ROGC and tpw_C2 added; breakpoint for ρ_a	
	8-74	8.4.3.11.3	Distinguish b775, b865 and b885 for ρ_a computation	
	8-75	8.4.3.12.1	Loop on bands up to 885	
	8-76, 8-77	8.4.3.12.2	Add tpw_C2; change ρ^*_w computation for bands 779, 865 and 885 nm but keep transmittance computation for all bands.	
	8-79, 8-81	8.4.3.13.2 8.4.3.13.3	XC coefficients are now indexed by m	
	8-80	8.4.3.13.2	Change the way τ_a is computed, by first interpolating XC coefficients	
	8-81, 8-82	8.4.3.13.3	Simplify ζ computation with XC coefficients interpolation	
	8-83	8.4.3.13.4	Correct a typo about τ_a in equation *(apb-2)*	
	8-85	8.4.3.13.5	Use τ_{R0} instead of τ_R_a; Important clarification about warning 1 and warning 2	
	8-90	8.4.3.13.9	Add equality in positive testing of x1	
	8-92	8.4.3.13.10	Use τ_{R0} instead of τ_R_a; Important clarification about warning 1 and warning 2	
	8-95	8.5.2	Removal of the IMT algorithm	
	8-96	8.5.2	Update of the functional breakdown, without IMT alg.	
	8-97	8.5.3.4	Section deleted	
	8-98	8.5.4	Flag ICE_HIGHAERO added; range of τ_a and f_0_LUT updated;	
	8-99	8.5.4	Replace bb_over_a by R (irradiance reflectance)	
	8-101	8.5.5	RWNEG condition replaced by positivity test; CASE2ANOM flag moved to other section	
	8-102	8.5.5.2	RWNEG condition replaced by positivity test	
	8-104	8.5.5.2	End of loop added	
	8-105	8.5.5.3	Change anomalous scattering condition	
	8-105	8.5.5.4	Section deleted	
	8-106	8.5.5.5	Typo corrected	
	8-107	8.5.6 and 8.5.7	Flags ORINP2 and OROUT2 removed	
	8-108 to 8-113	8.6	Completely new section, replaces 8.5.5.4	
	9-4	9.2.3.1	Symbols used for channels in measurement (15 channels) and auxiliary data (3 channels here) are now distinct.	
	9-7	9.2.3.3	Exception processing to step 2.2-19 clarified	
	9-12	9.3.2	New variables added (i & c), obsolete ones (i & c) removed, several renamed to remove obsolete 4x4 average subscript, additional breakpoints defined.	
	9-16 to 9-19	9.3.3.1	Loop on 4x4 sub-windows deleted for Rayleigh correction	
	9-16	9.3.3.1	Steps 2.6.23-2 to -5 and 2.6.15.1-3 to -5 deleted	
	9-17	9.3.3.1	Steps 2.6.15.4-1 to -4 deleted	

Copyright © 2011 ACRI S.T.

	9-17	9.3.3.2	Loop on 4x4 sub-windows clarified here as deleted above.	
	9-18	9.3.3.3	Band set definition moved out of the pixel loops	
	9-18, 9-19	9.3.3.3	Loop on 4x4 sub-windows and internal variables clarified locally.	
	9-20, 9-25, 9-26, 9-29	9.3.3.3, 9.3.3.6	Variable ρ_G renamed ρ_{Ground} and ρ_R renamed ρ_{R1} for consistency and to avoid confusion other DPM sections.	
	9-20	9.3.3.3	Wavelength to be used in 2.6.17.3-7 changed to theoretical	
	9-20	9.3.3.3	Repeat loop break condition 2.6.17.3-25 modified	
	9-21	9.3.3.3	Additional Exception Processing after step 2.6.17.3-45	
	9-21	9.3.3.3	Step 2.6.17.3-53 clarified	
	9-21	9.3.3.3	Step 2.6.17.3-47 clarified	
	9-21	9.3.3.3	Step 2.6.17.4-1 split into -1a & -1b and clarified	
	10-2	10.3.1.2.3	Definition of 2 fields of the Summary Product Quality ADSR updated	
	10-4	10.4	Obsolete input variables deleted, new ones added.	
	10-17	10.5.13	Obsolete symbol b890 replaced by b885	
	10-17	10.5.13	Definition of PCD_17 (Land & Water) & PCD_18 (Land) modified	
	10-18	10.5.13	Definition of Science flags OADB and ABSOA_DUST clarified	

Note:

- change bars left relative to issue 7.2A.

Copyright © 2011 ACRI S.T.

8	**0B**	**24/06/11**	**Clarifications after convergence with IPF**	
	Page #	**Section #**	**Comments**	
	4-5	4.5.1	Step *2.1.12-1* clarified: fSL indexed by Detector(j,f)	
	4-7	4.5.3	Step *2.1.5-2a* clarified: fSL indexed by Detector(j,f)	
	5-6	5.4	Sun azimuth angles added to inputs in list of variables.	
	5-10	5.4	Description of $\cos\theta_{scat}$ variable clarified	
	5-10	5.4	Variable $\cos\theta_{scat\ 4x4}$ added	
	5-13	5.5.1	Spectral domain of step 2.1.7-2 extended up to b900	
	5-14	5.5.1	Step 2.1.7-22 added (from previous step 2.1.9-8, deleted)	
	5-14	5.5.1	SATURATED_F flag explicitly added to P1 function interface	
	5-14	5.5.1	Equation 2.1.7-20 corrected	
	5-15	5.5.1	Equation 2.1.8-1 corrected	
	5-16	5.5.3	Equation 2.1.9-8 deleted (moved up as 2.1.7-22)	
	5-23	5.5.5	Step 2.1.6-9 clarified	
	5-28	5.5.9	Variable $\Delta\phi$ added to list of inputs of function Pscatt	
	5-29	5.5.9	Equation 2.1.7.1-12 clarified	
	6-7	6.5	Typo corrected in equation 2.6.5.1.9-1 & -2 (indexing)	
	7-4	7.4	Variable L_T added to c types list.	
	8-7	8.2.2	τ_R(b) removed from list of variables	
	8-21	8.3.3.2	Font problem corrected in equation 2.6.8.5-5	
	8-24	8.3.3.3.2	Font problem corrected in condition above equation 2.6.8.7-7a	
	8-28	8.3.3.7.2	typo corrected in exception condition above equation rtbb-1	
	8-50	8.4.3.4.2	taua(b) removed from list of variables	
	8-50	8.4.3.4.3	Extension of step 2.6.9.2-15 to 885 removed.	
	8-52	8.4.3.5.2	Domain of variable UseA clarified	
	8-56	8.4.3.6.2	Domain of variable UseA clarified	
	8-61	8.4.3.7.3	Calls to transmittance_up and transmittance_d clarified	
	8-66	8.4.3.9.2	Domain of variable τa_865 clarified	
	8-67	8.4.3.9.3	Indexing of variable τa_865 clarified (in 2.6.9.2.5-7 & -8)	
	8-72	8.4.3.11.2	Domain of variable τa_865 clarified	
	8-72	8.4.3.11.2	Spectral domain of variable ρa clarified	
	8-73	8.4.3.11.3	Indexing of variable τa_865 clarified (in 2.6.9.2.7-1 & -3)	
	8-75	8.4.3.12.2	Variables ia1 & ia2 added to replace ia_candidate	
	8-75	8.4.3.12.2	Variable τa_865 added to replace τa_candidate	
	8-76	8.4.3.12.3	Calls to transmittance_up and transmittance_d clarified	
	8-77	8.4.3.13.1	Function interface clarified	
	8-82	8.4.3.13.4	Function interface clarified	
	8-82	8.4.3.13.4.1	Domain of variable τa_865 clarified	
	8-82	8.4.3.13.4.2	Steps apb-0a & apb0b added to clarify aerosol indexing	
	8-82 & -83	8.4.3.13.4.2	Indexing of variable τa_865 & τa_vis clarified (in apb-1 to -3)	

Copyright © 2011 ACRI S.T.

	8-84	8.4.3.13.5	Function interface clarified	
	8-84	8.4.3.13.5.1	Variables ia1 & ia2 added to replace ia_candidate	
	8-84	8.4.3.13.5.2	Indexing of LUT tup_LUT clarified (in tup-1 & -2)	
	8-91	8.4.3.13.10	Function interface clarified	
	8-84	8.4.3.13.10.1	Variables ia1 & ia2 added to replace ia_candidate	
	8-84	8.4.3.13.10.2	Indexing of LUT t_{down}_LUT clarified (in td-1 & -2)	
	9-14	9.3.2	Added index b and list of bands to variable C_Corr	
	9-19, 9-20	9.3.3.3	Added index b to variable C_Corr in steps 2.6.17.3-51, .2.6.17.3-49, .2.6.17.3-50	
	9-20	9.3.3.3	Range clipping to ≥1e-6 added in exception processing below step .2.6.17.3-45	
	9-21	9.3.3.3	Font problem corrected in equation 2.6.17.3-47	

Note:

- change bars left relative to issue 7.2A.

Table of Contents

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

1. - Introduction

1.1 - General

This document is the Detailed Processing Model document for the MERIS data processing. It covers the MERIS Level 2 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 description of the MERIS processing algorithms in terms of algorithms and data structures, following the guidelines in AD1. This detailed description is intended to serve as :

- a functional requirements specification for the MERIS data processing entities within the ENVISAT-1 ground segment;
- a basis for the estimate of the computation resources requirements for the MERIS data processing.

This document describes in detail data processing to be applied to the MERIS pixels, in order to derive **the MERIS level 2 Reference Products**, in Reduced Resolution as well as in Full Resolution. It provides detailed descriptions of the algorithms and parameters in the MERIS level 2 processing architecture.

The Level 2 processing is in charge of processing TOA radiance measurements into geophysical parameters. These parameters depend on the observed pixel (water, land, cloud) and provide information on:

- the surface properties : normalised reflectance at surface, chlorophyll and other water constituents concentration (ocean); reflectance at surface, vegetation indices (land);
- the properties of the atmosphere above the surface : aerosol type and optical thickness, water-vapour column content, cloud top height, optical thickness and albedo.

The general structure of Level 2 processing and products is presented in the following flow chart (figure 1.2-1) The box numbers refer to the different step numbers of the MERIS Level 2 processing breakdown presented in RD9. Note that, in the diagram below, the paths labelled in Arial type indicate the main control flow, according to pixel type; those labelled in *Times italic* indicate the product flow.

Copyright © 2011 ACRI S.T.

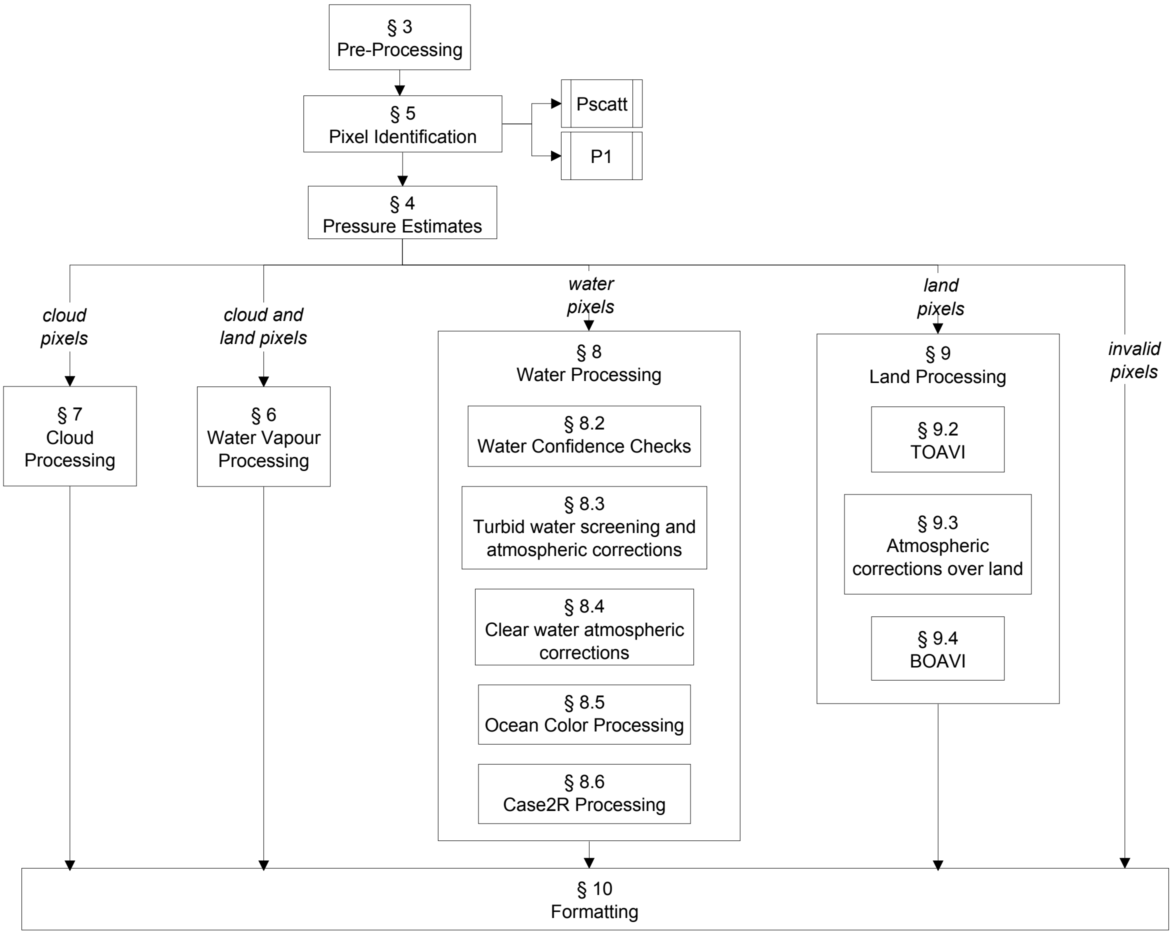


Figure 1.2-1: Meris level 2 general control flow.

1.3 - Guide to this document

This document is organised as follows:

- **Section 2** lists the applicable and reference documents, abbreviations, notations and conventions.

  Section 2.4, notations and conventions, is essential reading and reference for this specification.
- **Sections 3 to 10** provide the detailed specification of the MERIS level 2 processing. Level 2 processing is hierarchically broken down into algorithm steps. The top level breakdown is shown in fig. 1.2-1 above. The following tables 1.3-1 and 1.3-2 provide the correspondence between step number and the section number for detailed specification, respectively in hierarchical order and in numerical order.
- **Section 11** lists all breakpoints to be made available in testing and diagnostic.
- **Section 12** specifies the extraction of the low resolution Level 2 product from the Reference product.
- **Annex A**, the Parameters data list, provides the correspondence between algorithm auxiliary parameters (specified parameters, symbol "s" in column T of "List of variables" tables) and the databases specified by the Input /Output Definition Document (AD4).

Step no.	**Title**	**Section**
2.1a	Pre-Processing	3
2.1.0	Pre-processing for geometry and meteo.	3.5.1
2.1.11	Pixel = valid ?	3.5.1
2.1.4	Reflectance conversion	3.5.2
2.1c	Pixel Classification	5
2.1.7	Cloud screening tests	5.5.1
2.1.17	Procedure to estimate Rayleigh reflectance	5.5.6
2.1.7.1	procedure Pscatt to retrieve apparent pressure over ocean	5.5.9
2.1.7.2	procedure P1 to retrieve apparent surface pressure over land	5.5.10
2.1.7.3	Generic procedure compute_pressure to retrieve apparent pressure from LUTs	5.5.11
2.1.8	Set Cloud and Snow/Ice flags	5.5.2
2.1.9	Correction for stratospheric aerosol	5.5.3
2.1.18	Procedure to estimate Aerosol reflectance	5.5.7
2.6.12	Gaseous absorption corrections	5.5.4
2.6.12.1	Estimation of O3 transmittance	5.5.4
2.6.12.2	Estimation of O2 transmittance	5.5.4
2.6.12.3	Estimation of H2O transmittance	5.5.4
2.6.12.4	Estimation of corrected reflectance	5.5.4
2.6.26	Land identification	5.5.5
2.1.6	Reflectance Corrections for Smile Effect	5.5.5 & 5.5.8
2.1b	Pressure Estimates	4
2.1.12	Surface Pressure Neural Net method	4.5.3
2.1.12.0	Neural Network Initialisation	4.5.1
2.1.5	Cloud Top Pressure Neural Net method	4.5.2
2.1.5.0	Neural Network Initialisation	4.5.1
2.3	Water vapour retrieval	6
2.3.0	Range check for land and cloud pixels	6.5.2
2.3.1	Water vapour processing over land	6.5.3
2.3.2	Water vapour processing over water macro-pixels	6.5.1
2.3.3	Water vapour processing over cloud	6.5.4
2.3.5	Water macro-pixels to pixels	6.5.5
2.3.6	Range check on water vapour product	6.5.6
2.4	Cloud processing	7
2.4.1	Cloud albedo processing	7.5.1
2.4.3	Cloud Optical thickness processing	7.5.2
2.4.8	Cloud type processing	7.5.3

Table 1.3-1: Algorithm step index, hierarchical order

Copyright © 2011 ACRI S.T.

Step no.	Title	Section
N/A	Water Processing	8
2.6.5	Waters confidence checks	8.2
2.6.5.1	Glint Processing	8.2.3.1
2.6.5.1.1	Sun glint estimation	8.2.3.1
2.6.5.1.3	Low sun glint ?	8.2.3.1
2.6.5.1.4	Medium sun glint ?	8.2.3.1
2.6.5.1.5	Uncorrected Glint flagging	8.2.3.1
2.6.5.1.7	Glint correction	8.2.3.1
2.6.5.1.8	Pixel = bright ?	8.2.3.1
2.6.5.2	Low pressure water flagging	8.2.3.2
2.6.5.3	Whitecaps identification	8.2.3.3
2.6.8	Turbid water screening and corrections	8.3
2.6.8.1	Turbid screening and atmospheric correction	8.3.3.1
2.6.8.2	Coarse Rayleigh correction and diffuse transmittance computation	8.3.3.2
2.6.8.3	White Scatterer identification	8.3.3.2
2.6.8.4	Turbid water identification and initial estimates	8.3.3.2
2.6.8.5	Determination of bandset and radiometric flagging	8.3.3.2
2.6.8.6	Iterative estimate of α, IOPs and rhow	8.3.3.3
2.6.8.7	Combine and check estimates	8.3.3.3
2.6.8.8	Estimate of TOA marine reflectances and TSM	8.3.3.4
2.6.9	Clear water atmospheric corrections	8.4.3.1
2.6.9.0	Process initialisation	8.4.3.2
2.6.9.1	Path reflectance	8.4.3.3
2.6.9.2	MERIS aerosol model	8.4.3.4
2.6.9.2.1	Select aerosols	8.4.3.5
2.6.9.2.2	Candidate	8.4.3.6
2.6.9.2.3	Store candidate models	8.4.3.7
2.6.9.2.4	Test absorbing aerosol	8.4.3.8
2.6.9.2.5	Final couple	8.4.3.9
2.6.9.2.6	Check climatology	8.4.3.10
2.6.9.2.7	Aerosol parameters	8.4.3.11
2.6.9.3	Correction	8.4.3.12
2.9	Ocean colour processing	8.5.5
2.9.4	Case 2 yellow substance dominated waters flagging	8.5.5.1
2.9.6	Anomalous scattering water flagging	8.5.5.2
2.9.7	Algal pigment index retrieval in Case 1 waters	8.5.5.3
2.9.12	Case 2 R processing	8.6.4
2.9.8	PAR processing	8.5.5.5

Table 1.3-1: Algorithm step index, hierarchical order (cont.)

Copyright © 2011 ACRI S.T.

Step no.	Title	Section
	Land Processing	9
2.2	TOAVI processing	9.2.3.1
2.6.23	Atmospheric correction aver land	9.3
2.6.15	Rayleigh correction 2	9.3.3.1
2.6.15.1	Estimation of Rayleigh reflectance	9.3.3.1
2.6.15.2	Estimation of Rayleigh transmittance	9.3.3.1
2.6.15.3	Estimation of Rayleigh spherical albedo	9.3.3.1
2.6.15.4	Estimation of reflectance corrected for Rayleigh scattering	9.3.3.1
2.6.13	DDV screening	9.3.3.2
2.6.17	Aerosols above DDV	9.3.3.3
2.6.17.1	Read climatology and retrieve aerosol model as a first guess for optical thickness at 550nm	9.3.3.3
2.6.17.2	Select refractive index corresponding to the aerosol model found in climatology and 3 additional aerosol models having the same refractive index	9.3.3.3
2.6.17.3	Derive optimal aerosol model within the set of 4 models, and its optical thickness, by iterative procedure	9.3.3.3
2.6.17.4	Compute aerosol parameters over DDV pixels	9.3.3.3
2.8	BOAVI	9.4

Table 1.3-1: Algorithm step index, hierarchical order (cont.)

Step no.	Title	Section
2.10	Product formatting	10
2.10.1	Build MPH	10.5.1
2.10.2	Build SPH	10.5.2
2.10.3	Build GADS “Scaling Factors and Offsets”	10.5.3
2.10.4	Build ADS “Summary Quality”	10.5.4
2.10.5	Build ADS "Tie Points Annotations and corresponding Auxiliary Data"	10.5.5
2.10.6	Build Normalised Surface Reflectance MDS 1 to 13	10.5.6
2.10.7	Build Total water vapour MDS 14	10.5.7
2.10.9	Build Algal index I or Top of Atmosphere Vegetation Index MDS 15	10.5.8
2.10.10	Build Yellow Substance and Total Suspended Matter MDS 16	10.5.9
2.10.11	Build Algal index II or Bottom of Atmosphere Vegetation Index MDS 17	10.5.10
2.10.12	Build Pressure or PAR or Cloud Albedo MDS 18	10.5.11
2.10.13	Build Aerosol epsilon or Cloud type and optical thickness MDS 19	10.5.12
2.10.14	Build flags MDS	10.5.13

Table 1.3-1: Algorithm step index, hierarchical order (cont.)

Copyright © 2011 ACRI S.T.

Step no.	Title	Section
	Land Processing	9
	Ocean Processing	8
2.1a	Pre-Processing	3
2.1b	Pressure Estimates	4
2.1c	Pixel Identification	5
2.1.0	Pre-processing for geometry and meteo.	3.5.1
2.1.11	Pixel = valid ?	3.5.1
2.1.12	Surface Pressure Neural Net method	4.5.3
2.1.12.0	Neural Network Initialisation	4.5.1
2.1.16	Polynomial procedure to estimate pressure	4.5.5
2.1.17	Procedure to estimate Rayleigh reflectance	5.5.6
2.1.18	Procedure to estimate Aerosol reflectance	5.5.7
2.1.2	Tests on pressure	4.5.4
2.1.3	Spectral Shift index computation	4.5.2
2.1.4	Reflectance conversion	3.5.2
2.1.5	Cloud top pressure determination	4.5.3
2.1.5.0	Neural Network Initialisation	4.5.1
2.1.6	Reflectance Corrections for Smile Effect	5.5.5 & 5.5.8
2.1.7	Cloud screening tests	5.5.1
2.1.7.1	procedure Pscatt to retrieve apparent pressure over ocean	5.5.9
2.1.7.2	procedure P1 to retrieve apparent surface pressure over land	5.5.10
2.1.7.3	Generic procedure compute_pressure to retrieve apparent pressure from LUTs	5.5.11
2.1.8	Set Cloud and Snow/Ice flags	5.5.2
2.1.9	Correction for stratospheric aerosol	5.5.3

Table 1.3-2: Algorithm step index, numerical order

Copyright © 2011 ACRI S.T.

2.10	Product formatting	10
2.10.1	Build MPH	10.5.1
2.10.10	Build Yellow Substance and Total Suspended Matter MDS 16	10.5.9
2.10.11	Build Algal index II or Bottom of Atmosphere Vegetation Index MDS 17	10.5.10
2.10.12	12 Build Pressure or PAR or Cloud Albedo MDS 18	10.5.11
2.10.13	Build Aerosol epsilon or Cloud type and optical thickness MDS 19	10.5.12
2.10.14	Build flags MDS	10.5.13
2.10.2	Build SPH	10.5.2
2.10.3	Build GADS “Scaling Factors and Offsets”	10.5.3
2.10.4	Build ADS “Summary Quality”	10.5.4
2.10.5	Build ADS "Tie Points Annotations and corresponding Auxiliary Data"	10.5.5
2.10.6	Build Normalised Surface Reflectance MDS 1 to 13	10.5.6
2.10.7	Build Total water vapour MDS 14	10.5.7
2.10.9	Build Algal index I or Top of Atmosphere Vegetation Index MDS 15	10.5.8

Table 1.3-2: Algorithm step index, numerical order (cont.)

Copyright © 2011 ACRI S.T.

Step no.	Title	Section
2.2	TOAVI processing	9.2
2.3	Water vapour retrieval	6
2.3.0	Range check for land and cloud pixels	6.5.2
2.3.1	Water vapour processing over land	6.5.3
2.3.2	Water vapour processing over water macro-pixels	6.5.1
2.3.3	Water vapour processing over cloud	6.5.4
2.3.5	Water macro-pixels to pixels	6.5.5
2.3.6	Range check on water vapour product	6.5.6
2.4	Cloud processing	7
2.4.1	Cloud albedo processing	7.5.1
2.4.3	Cloud Optical thickness processing	7.5.2
2.4.8	Cloud type processing	7.5.3
2.6.8	Turbid water screening and corrections	8.3
2.6.8.1	Turbid screening and atmospheric correction	8.3.3.1
2.6.8.2	Coarse Rayleigh correction and diffuse transmittance computation	8.3.3.2
2.6.8.3	White Scatterer identification	8.3.3.2
2.6.8.4	Turbid water identification and initial estimates	8.3.3.2
2.6.8.5	Determination of bandset and radiometric flagging	8.3.3.2
2.6.8.6	Iterative estimate of α, IOPs and rhow	8.3.3.3
2.6.8.7	Combine and check estimates	8.3.3.3
2.6.8.8	Estimate of TOA marine reflectances and TSM	8.3.3.4
2.6.12	Gaseous absorption corrections	5.5.4
2.6.12.1	Estimation of O3 transmittance	5.5.4
2.6.12.2	Estimation of O2 transmittance	5.5.4
2.6.12.3	Estimation of H2O transmittance	5.5.4
2.6.12.4	Estimation of corrected reflectance	5.5.4
2.6.13	DDV screening	9.3.3.2
2.6.15	Rayleigh correction 2	9.3.3.1
2.6.15.1	Estimation of Rayleigh reflectance	9.3.3.1
2.6.15.2	Estimation of Rayleigh transmittance	9.3.3.1
2.6.15.3	Estimation of Rayleigh spherical albedo	9.3.3.1
2.6.15.4	Estimation of reflectance corrected for Rayleigh scattering	9.3.3.1

Table 1.3-2: Algorithm step index, numerical order (cont.)

Copyright © 2011 ACRI S.T.

Step no.	Title	Section
2.6.17	Aerosols above DDV	9.3.3.3
2.6.17.1	Read climatology and retrieve aerosol model as a first guess for optical thickness at 550nm	9.3.3.3
2.6.17.2	Select refractive index corresponding to the aerosol model found in climatology and 3 additional aerosol models having the same refractive index	9.3.3.3
2.6.17.3	Derive optimal aerosol model within the set of 4 models, and its optical thickness, by iterative procedure	9.3.3.3
2.6.17.4	Compute aerosol parameters over DDV pixels	9.3.3.3
2.6.23	Atmospheric correction aver land	9.3
2.6.26	Land identification	5.5.5
2.6.5	Waters confidence checks	8.2
2.6.5.1	Glint processing	8.2.3.1
2.6.5.1.1	Sun glint estimation	8.2.3.1
2.6.5.1.3	Low sun glint ?	8.2.3.1
2.6.5.1.4	Medium sun glint ?	8.2.3.1
2.6.5.1.5	Uncorrected Glint flagging	8.2.3.1
2.6.5.1.7	Glint correction	8.2.3.1
2.6.5.1.8	Pixel = bright ?	8.2.3.1
2.6.5.2	test ECMWF pressure	8.2.3.2
2.6.5.3	Whitecaps identification	8.2.3.3
2.6.8 & 2.6.10	Turbid water screening and atmospheric corrections	8.3
2.6.8.1	Rayleigh reflectance estimate	8.3.3.2
2.6.8.2	Turbid water screening	8.3.3.3
2.6.9	Clear water atmospheric corrections	8.4.3
2.6.9.0	Process initialisation	8.4.3.2
2.6.9.1	Path reflectance	8.4.3.3
2.6.9.2	MERIS aerosol model	8.4.3.4
2.6.9.2.1	Select aerosols	8.4.3.5
2.6.9.2.2	Candidate	8.4.3.6
2.6.9.2.3	Store candidate models	8.4.3.7
2.6.9.2.4	Test absorbing aerosol	8.4.3.8
2.6.9.2.5	Final couple	8.4.3.9
2.6.9.2.6	Check climatology	8.4.3.10
2.6.9.2.7	Aerosol parameters	8.4.3.11
2.6.9.3	Correction	8.4.3.12

Table 1.3-2: Algorithm step index, numerical order (cont.)

Copyright © 2011 ACRI S.T.

Step no.	Title	Section
2.8	BOAVI	9.4
2.9	Ocean colour processing	8.5.5
2.9.12	Case 2 R processing	8.6.4
2.9.4	Case 2 yellow substance dominated waters flagging	8.5.5.1
2.9.6	Anomalous scattering water flagging	8.5.5.2
2.9.7	Algal pigment index retrieval in Case 1 waters	8.5.5.3
2.9.8	PAR processing	8.5.5.5

Table 1.3-2: Algorithm step index, numerical order (cont.)

Copyright © 2011 ACRI S.T.

2. - References, abbreviations and definitions

2.1 - Applicable Documents

AD1. Guidelines for the specification of ground processing algorithms, PO-RS-ESA-GS-0252, Iss. 1
AD2. ENVISAT-1 Product Format Guidelines, PO-TN-ESA-GS-0242.
AD3. ESA Software Engineering Standards, ESA PSS-05.
AD4. MERIS I/O Data Definition (IODD), PO-TN-MEL-GS-0003
AD5. ENVISAT Product Specification, PO- RS-MDA-GS-2009

Note: AD4 supersedes AD5 in case of conflict between these two documents.

AD6. Neural Network Interface Document, PO-TN-MEL-GS-0025
AD7 Case2R CFI Interface Document, PO-TN-MEL-GS-0048

2.2 - Reference Documents

RD1. ENVISAT-1 Ground Segment Concept, ESA/PB-EO(94)24, Iss. 5 rev.3
RD2. Level 1B Detailed Processing Model, PO-TN-MEL-GS-0002, Iss. 4
RD3. MERIS Assumptions on the Ground Segment, PO-RS-DOR-SY-0029, Iss. 1, Vol. 6
RD4. Measurement Data Definition and Format Description for MERIS, PO-ID-DOR-SY-0032, Iss. 4.0, Vol. 4, 7
RD5. *deleted*
RD6. *deleted*
RD7. Tables Generation Requirements Document (TGRD), PO-TN-MEL-GS-0012, Iss.2.1
RD8. Algorithm Theoretical Basis Document (ATBD), PO- TN-MEL-GS-0005, Iss. 4.1
RD9. System Architecture Theoretical Basis Document (SATBD), PO-TN-MEL-GS-0001, Iss. 4
RD10 W. Press *et al.* Numerical Recipes in C, Cambridge University Press, second edition, 1992
RD11 MERIS Regional Coastal and Lake Case 2 Water Project Atmospheric Correction ATBD, GKSS-KOF-MERIS-ATBD01

2.3 - Abbreviations

ARVI	Atmosphere Robust Vegetation Index
ATBD	Algorithm Theoretical Basis Document
BOA	Bottom of Atmosphere
BRDF	Bi-directional Reflectance Distribution Function
CFI	Customer Furnished Item
DDV	Dense Dark Vegetation
ECMWF	European Centre for Medium-term Weather Forecast
FR	Full resolution
FOV	Field Of View
IMT	Inverse Modelling Technique
IR	Infra Red
IRTM-NN	Inverse Radiative Transfer Model – Neural Network
ISLSCP	International Satellite Land Surface Climatology Project

L1B	Level 1B
L2	Level 2 processing
LUT	Look Up Table
MERIS	Medium Resolution Imaging Spectrometer
MGVI	MERIS Global Vegetation Index
MOMO	Matrix Operator MOdel
MPH	Main Product Header
NDVI	Normalised Differential Vegetation Index
NIR	Near Infra Red
NN	Neural Networks
ODOC	Optical Dissolved Organic Compounds (synonyms: Yellow substance, gelbstoff)
PAR	Photosynthetically Available Radiation
POLDER	Polarisation and Directionality of Earth Reflectance
RD	Reference Document
RR	Reduced Resolution
SATBD	System Architecture Theoretical Basis Document
SP	spectral (dimension of the sensor)
SPM	Suspended Particulate Matter (equivalent to TSM)
TBC	To Be Confirmed
TBD	To Be Defined
TOA	Top Of Atmosphere
TSM	Total Suspended Matter (equivalent to SPM)

2.4 - Notations and Conventions

2.4.1- Look-Up Tables

The term Look-Up Table (LUT) is used within this document in the wide acceptance of a table with multiple index variables, pre-computed and loaded into the MERIS processor from a file. LUTs as a rule correspond to data sets identified in the I/O DD (AD4).

2.4.2 - Table indexing and Spectral Bands

The subscripts of the array data structures shall be
b band (see below); j column ($j \in \{1..J\}$); f frame ($f \in \{1..F\}$);

In general spectral bands shall be indexed by variable b. $\lambda(b)$ denotes the wavelength for band **b**, with :

Number	Symbols	$\lambda(b)$ in nm	$\Delta\lambda(b)$ in nm
1	b1 or b412	412.5	10
2	b2 or b442	442.5	10
3	b3 or b490	490.0	10
4	b4 or b510	510.0	10
5	b5 or b560	560.0	10
6	b6 or b620	620.0	10
7	b7 or b665	665.0	10
8	b8 or b681	681.25	7.5
9	b9 or b705	708.75	10
10	b10 or b753	753.75	7.5
11	b11 or b761	760.625	3.75
12	b12 or b775 or b779	778.75	15
13	b13 or b865	865.0	20
14	b14 or b885	885.0	10
15	b15 or b900	900.0	10

Table 2.4-1 : Band wavelength correspondence

Symbol notations found in this document for band index are b, b1..b15 (indices of bands 1 to 15), b412..b900 (indices of bands whose central wavelength is 412nm..900nm).

2.4.3 - Table Interpolation

Interpolation in look-up tables shall be noted :
<*result*> = <*table*> [**interpol:** (<*linear interpolation indices*>)] [**nearest:** (<*nearest neighbour interpolation indices*>)] **select:** (<*indices*>)
Example : let T_LUT be a table indexed by (u , v , β_k , γ_k).
α = T_LUT **interpol:** (u , v) **select:** (β_k , γ_k) denotes that a linear interpolation has been performed to obtain α as a function of (u,v) on the selection of T_LUT corresponding to indices β_k and γ_k.

Copyright © 2011 ACRI S.T.

α = T_LUT **nearest:** (u , v) **select:** (β_k , γ_k) denotes that a selection has been performed to obtain α, at indices nearest to (u,v) and matching exactly β_k and γ_k.

IMPORTANT NOTES

1. Exception processing: when any of the index parameters is out of the corresponding index range, the value of the index parameter shall be replaced by the nearest value in the index range.

Example: if T_LUT is indexed by (a, b) and the index range for a is $a_{min}..a_{max}$, then
if ($a <= a_{min}$) T_LUT **interpol**:(a, b) = T_LUT **interpol**:(a_{min}, b)
if ($a >= a_{max}$) T_LUT **interpol**:(a, b) = T_LUT **interpol**:(a_{max}, b)

Flagging of such out-of-index-range condition shall be performed only when explicitly specified.

2. In some cases, interpolation may be specified on indices which need transformation of the original LUT grid vector provided in the auxiliary data files: e.g. a vector of angles is provided to describe one of the LUT dimension while interpolation is required on the angle cosine. To keep the present document as independent as possible from AD4, these simple transformations will not be described explicitly and the responsibility of establishing correspondence, and transformation if required, between AD4 grid vectors and interpolation parameters specified hereafter is left to the user.

2.4.4 - Parameter type

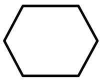
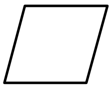
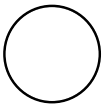
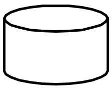
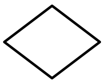
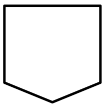
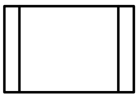
The column labelled "T" (for Type) in the lists of variables in each algorithm description section below describes the type of the variable :

i	input to the algorithm step (from a previous step)
c	intermediate result
s	specified in Look Up Tables (LUTs) - The correspondence between such parameters and the auxiliary databases specified in AD4 is defined in Appendix A: Parameters Data List. Also includes implementation-dependent constants (not found in AD4).
o	output of the algorithm step

2.4.5 - Block diagram symbols

In all functional block diagrams the symbols in table 2.4.5-I below are used.

Copyright © 2011 ACRI S.T.

	processing
	processing on multiple pixel ("window processing")
	data, e.g internal data, flags, L2 product data
	data, same as above, for consistency with other
	a data base (i.e. auxiliary product database, configuration controlled, used as input by processing).
	decision
	connector, indicating the connection between steps
	subroutine /procedure /function

Control flow precedence	→
Data flow precedence	⇢
Exception control flow	▷
The relation between caller and called is symbolised by the double arrow	↠

Table 2.4.5-1 : Diagram symbols

2.4.6 - Units

Table 2.4.6-1 below describes the units used in this document, shown in column "U" in the lists of variables :

Unit symbol	Name
dl	dimensionless
EU or W.m^{-2}.µm^{-1}	spectral irradiance
LU or W.m^{-2}.sr^{-1}.µm^{-1}	spectral radiance
s	seconds
%	percentage
K	degree Kelvin (temperature)
° or deg	degree (angle)
rad	radian
hPa	hectoPascal
DU	Dobson Unit (= 10^{-3} atm.cm)
IU or W.m^{-2}	irradiance
µeinstein.m^{-2}.s^{-1}	micro Einstein (=10^{-6} mol.photon) per square metre per second (unit for PAR)

Table 2.4.6-1 - Units and notation

2.4.7 - Parameter coding

For all flag (boolean) fields, the flag is considered to be TRUE when it is set to 1, and it is considered to be FALSE when it is set to 0.

The value BAD_VALUE is a convention value meant to represent the output when an algorithm fails. Its choice is left to implementation but should fulfil the following constraints:

1. BAD_VALUE must be meaningfully outside the validity range of any parameter in MERIS processing;
2. the IEEE floating-point representation of BAD_VALUE must be exact so as to permit equality (== or !=) comparisons;

The value BAD_PRODUCT is a code written to the MERIS L2 product MDS to indicate an invalid parameter. Following the convention in AD5, BAD_PRODUCT shall be 0.

2.4.8 - Geometry auxiliary parameters

The geometry (latitude, longitude, θ_s, θ_v, $\Delta\phi$, ...) and the meteorological data (wind speed, pressure , ...) are not given for all pixels, they are given at tie points or at grid points of the ECMWF grid. Therefore, interpolations will have to be performed when needed. In this case, the notation will remain unchanged, e.g. θ_s(j,f) is the viewing zenith angle interpolated from the viewing zenith angles of the surrounding tie points). When interpolation is not needed and the value of the closest tie point is sufficient, the variable name will be given a "_tie" extension, e.g. θ_s_tie(i,j) , θ_v_tie(i,j) , lat_tie(i,j) , long_tie(i,j) , ...

Copyright © 2011 ACRI S.T.

2.4.9 - Pseudo-code

In the sections where pseudo-code is needed, all text in *italic type* has to be considered as comments. So does text enclosed between C-style comment delimiters `/* */` .

2.4.10 - Requirements numbering

Statements or equations to be considered as requirements for L2 processing implementation will be followed by a unique number composed of step number, followed by a requirement number within the step, between parentheses.

IMPORTANT NOTE

For the sake of backward compatibility with earlier versions of the DPM and processor code, statement numbers DO NOT follow a logical sequence. The sequence of document chapters is not representative of control flow: for control flow, refer to the block diagrams. Within a given DPM section, the sequence of statements is representative of control flow..

2.4.11- Standard mathematical functions

Mathematical functions used in this document, assumed to be provided by the standard libraries applicable to the MERIS processor, will be noted in **bold type**. Table 2.4.11-1 below provides a summary of the functions used.

Symbol	Name
arc cos	arc cosine in degrees
arc sin	arc sine in degrees
arc tan	arc tangent in degrees
cos	cosine of an angle in degrees
exp	exponential
floor	largest integer smaller than a number
int	nearest integer to a number
ln	Neperian logarithm
log	Neperian logarithm
$\log_{10}$	Decimal logarithm
MAX	largest element of a list
MIN	smallest element of a list
sin	sine of an angle
tan	tangent of an angle

Table 2.4.11-1: Standard mathematical functions

2.4.12 - Illumination and observation geometry convention

The following illumination and observation geometry conventions are used in MERIS processing :

Copyright © 2011 ACRI S.T.

- A point on Earth observed by MERIS is taken as a reference
- The sun zenith angle θ_S is the angle between the local outward normal and the vector from the point towards the Sun.
- The view zenith angle θ_V is the angle between the local outward normal and the vector towards the MERIS sensor.
- The azimuth difference $\Delta\phi$ is the angle between the half-plane containing the local normal and the Sun, and the half-plane containing the local normal and MERIS. In the principal plane,
    - there may be *specular reflection* of a point source into the MERIS sensor when the azimuth difference is **180°** (and the zenith angles are equal)
    - there may be *back-scatter* from a point source into the MERIS sensor when the azimuth difference is **0°** (and the zenith angles are equal).

In general, we assume that an azimuth difference of N degrees is equivalent wrt. MERIS radiometry, to 360° - N, so that the $\Delta\phi$ ranges from 0° to 180°.

IMPORTANT NOTE

All Look-up tables used by the processor assume the above convention for indexing. Should a radiative transfer tool used to compute these tables follow a different convention, it is essential that reordering of the values in order to respect the above convention should be performed **before** integration of the table into the MERIS processor.

2.4.13 - Exception handling

This Document combines two ways of specifying exception processing:

1. in line with algorithm specification, blocks of statements labelled as follows:

exception processing: <condition>:
    <actions>
end of exception processing

Such blocks may refer explicitly to the statement(s) where the exception can occur, or they refer to the statement immediately above.

2. common exception handling routines in a section "Exception processing" for a given algorithm step, which may be referred from several places in the algorithm.

2.5- List of TBDs

This document does not contain any TBD.

3.- MERIS pre-processing

3.1 - Introduction

This section describes:

- the checks to be applied to the Level 1b product submitted to processing to ensure that the data it contains respect the processing constraints
- data extraction
- preliminary processing to be applied to Annotation Data Sets: interpolation of annotations from the tie-points grid to the pixel grid
- preliminary processing to be applied to Measurement Data Sets: radiance to reflectance conversion.

3.2 - Algorithm Overview

The MERIS level 2 pre-processing is applied to each pixel and follows the logic shown in the flow chart in figure 3.2-1.

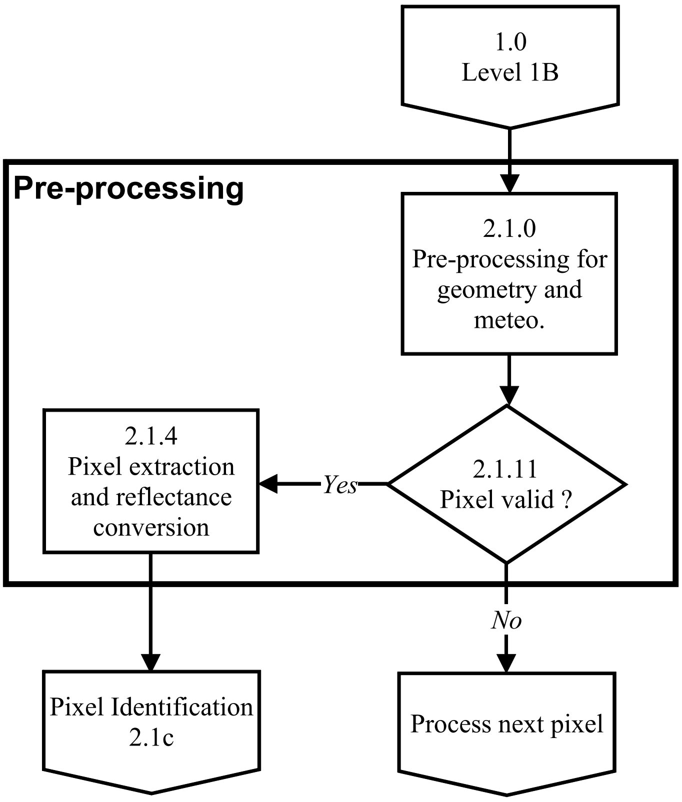


Figure 3.2-1 : MERIS level 2 pre-processing (step 2.1a)

Copyright © 2011 ACRI S.T.

3.3 - Mathematical Description Of Algorithm

3.3.1 - Level 1b product check

If one or more band within the Level 1b product is not one of those in table 2-1, the product shall not be processed to Level 2.

If the Solar irradiance in the Level 1B product (GADS scaling factors) is 0 in any band, the product shall not be processed to Level 2.

3.3.2 - Pre processing step

3.3.2.1 - Pre-processing for geometry and meteorological parameters (step 2.1.0)

This step is done in order to derive geometry and meteorological parameters (pressure, wind) at each pixel, including invalid ones, from those provided at each tie point of the Level 1b annotation product.

3.3.2.2 - Level 1b pixel classification screening (step 2.1.11)

The Level 2 pixel identification starts with the reading of the INVALID flag of the Level 1b product. If it is set to TRUE, then no further processing of the current pixel is done, the L2 product shall contain fixed values for all MDS (see section 10 below), with the "Invalid" flag set, and the next pixel is examined ; otherwise, the processing of the current pixel is pursued.

3.3.2.3 - Pixel extraction and reflectance conversion (step 2.1.4)

If the Level 1b pixel is not flagged INVALID, the other L1B flags and the Top Of Atmosphere radiances at all bands are extracted from the L1B product. Radiances are converted to reflectances, using the Sun zenith angle cosine interpolated at the pixel and the Sun spectral flux read from the L1B product annotations.

Copyright © 2011 ACRI S.T.

3.4 - List of Variables

Note: in this section and section 3.5 below, J and F are the notation for the index of column and frame corresponding to tie points, j and f are (as in the rest of this DPM) the product column and frame indices.

Variable	Descriptive Name	T	U	Range - References
INVALID_F(j,f)	Invalid flag	i/o	dl	from Level1b (Flags MDS), to Breakpoint
LAND_F(j,f)	Land flag (TRUE when land, FALSE when ocean)	i/o	dl	from Level1b (Flags MDS), to Breakpoint
TOAR(b,j,f)	TOA Radiance	i	LU	all bands; from level 1b MDS
Detector(j,f)	MERIS detector associated to pixel	i	dl	from Level1b (Flags MDS)
JD1, JD2	UTC times of first and last frames in product	i	jd	from Level1b (Flags MDS)
θ_s(J,F)	Sun zenith angle at tie point	i	deg	from Level 1b (LADS)
θ_v(J,F)	Viewing zenith angle at tie point	i	deg	from Level 1b (LADS)
ϕ_s(J,F)	Azimuth of sun angle at tie point	i	deg	from Level 1b (LADS)
ϕ_v(J,F)	Azimuth of view angle at tie point	i	deg	from Level 1b (LADS)
lat(J,F)	Latitude at tie points	i	deg	from Level 1b (LADS)
lon(J,F)	Longitude at tie points	i	deg	from Level 1b (LADS)
z(J,F)	Altitude at tie points	i	m	from Level 1b (LADS)
W_u(J,F), W_v(J,F)	Wind vector components at tie points	i	$m.s^{-1}$	from Level 1b (LADS)
U_{O3}(J,F)	Actual ozone content at tie points	i	DU	from Level 1b(LADS)
P_{Sea_ECMWF}(J,F)	ECMWF mean sea level pressure at tie points	i	hPa	from Level1b (LADS)
H_p	Pressure scale height	s	m	From data base
$Dsun_0^2$	Square of Sun-Earth distance at reference date	s	m	From data base
F_0^{RR}(b,k)	Extra-terrestrial Sun irradiance at reference date for all MERIS RR detectors and band	s	EU	From data base
F_0^{FR}(b,k)	Extra-terrestrial Sun irradiance at reference date for all MERIS FR detectors and band	s	EU	From data base
SATURATION_L(b)	Default radiance for saturated pixels	s	LU	All bands
p, q	linear interpolation weights for ancillary data	c	dl	
ϕ_v (j, f)	View azimuth angle at (j, f)	c	deg	
P_{Sea_ECMWF}(j,f)	ECMWF mean sea level pressure interpolated at pixel (j,f)	c	hPa	
sun_pos	Sun centre location in Geocentric reference frame at mid-product date	c	m	Mission CFI output
seasonal_fact	Correction factor for seasonal variation of Sun irradiance	c	dl	

ρ_{TOA} (b,j,f)	TOA reflectance at pixel (j,f)	o	dl	all bands, to steps 2.1b (§4.4), 2.1c (§5.4), to Breakpoint
θ_s(j,f)	Sun zenith angle for pixel (j,f)	o	deg	to steps 2.1b (§4.4), 2.1c (§5.4), 2.2 (§9.2.2), 2.3 (§6.4), 2.4 (§7.4), 2.6.5 (§8.2.2), 2.6.8 (§8.3.2), 2.6.10 (§8.3.2), 2.6.9 (§8.4.2), 2.9 (§8.5.4), to Breakpoint
θ_v(j,f)	Viewing zenith angle for pixel (j,f)	o	deg	*idem*
$\Delta\phi$(j,f)	Azimuth difference angle for pixel (j,f)	o	deg	*idem*
ϕ_s(j,f)	Sun azimuth angle for pixel (j,f)	o	deg	to step 2.6.5 (§8.2.2), to Breakpoint
lat(j,f)	Latitude for pixel (j,f)	o	deg	to step 2.1b (§4.4), 2.1c (§5.4), 2.6.9 (§8.4.2), to Breakpoint
lon(j,f)	Longitude for pixel (j,f)	o	deg	*idem*
z(j,f)	Altitude at interpolated at pixel (j,f)	o	m	to step 2.6.9 (§8.4.2)
W_u(j,f), W_v(j,f)	Wind vector components for pixel (j,f)	o	$m.s^{-1}$	to steps 2.1c (§5.5), 2.6.5 (§8.2.2)
P_{ECMWF}(j,f)	surface pressure for pixel (j,f) (from interpolated ECMWF sea level pressure, corrected for pixel altitude)	o	hPa	to step 2.1b (§4.4), 2.1c (§5.4), 2.6.5 (§8.2.2), 2.6.8 (§8.3.2), 2.6.10 (§8.3.2), 2.6.9 (§8.4.2), 2.9 (§8.5.4), 2.3 (§6.4), to Breakpoint
U_{O3}(j,f)	Actual ozone content for pixel (j,f)	o	DU	to step 2.1c (§5.4), 2.9 (§8.5.4)
SATURATED_F(b,j,f)	Saturated pixel flag	o	-	to step 2.1b (§4.4),), 2.1c (§5.4), to Breakpoint

NOTE: subscript FR or RR for type s variable F_0 is omitted in equation section below. Proper variable shall be selected according to processed product resolution.

3.5 – Equations (step 2.1a)

Product processing :
For each pixel (in column j, frame f):

3.5.1 - Pre-processing (step 2.1.0, 2.1.11)

interpolate geometry from values at the 4 surrounding tie points for current pixel
let J be the tie point column such that $J <= j < J + DJ$
let F be the tie point frame such that $F <= f < F + DF$

$p = (J+DJ - j) / DJ$ *(2.1.0-1)*

$q = (F+DF - f) /DF$ *(2.1.0-2)*

let **interpolate**(P) = $p.q.P(J,F) + p.(1 - q).P(J,F+DF) + (1 - p).q.P(J+DJ,F) + (1 - p).(1 - q).P(J+DJ, F+DF)$

where P is any parameter defined at the tie points and stored in the geo-location ADS of the L1B product

$\theta_s(j,f)$ = **interpolate** (θ_s) /* *sun zenith angle* */ *(2.1.0-3)*

$\theta_v(j,f)$ = **interpolate** (θ_v) /* *view zenith angle* */ *(2.1.0-4)*

$\phi_v(j,f)$ = **interpolate** (ϕ_v) /* *view azimuth angle* */ *(2.1.0-5)*

Copyright © 2011 ACRI S.T.

$\phi_s(j,f)$ = **interpolate** (ϕ_s) */* sun azimuth angle */* *(2.1.0-6)*
lat(j,f) = **interpolate** (lat) */* latitude */* *(2.1.0-7)*
lon(j,f) = **interpolate** (lon) */* longitude must stay in [-180, 180]: if lon(J) < lon(J+DJ) then 360 must be added to lon(J) prior to interpolation */* *(2.1.0-8)*
z(j,f) = **interpolate** (z) */* altitude */* *(2.1.0-9)*

interpolate meteorological ancillary data from value at the 4 surrounding tie points for current pixel
$P_{Sea\_ECMWF}(j,f)$ = **interpolate** (P_{Sea_ECMWF}) */* Ecmwf sea level pressure */* *(2.1.0-10)*
W_u(j,f) = **interpolate** (W_u) */* zonal wind */* *(2.1.0-11)*
W_v(j,f) = **interpolate** (W_v) */* longitudinal wind */* *(2.1.0-12)*
$U_{O3}(j,f)$ = **interpolate** (U_{O3}) */* ozone content */* *(2.1.0-13)*

compute azimuth difference angle
$\Delta\phi(j,f) = \mathbf{arc\ cos}[\mathbf{cos}(\phi_v(j,f) - \phi_s(j,f))]$ *(2.1.0-14)*

Correct sea level pressure for pixel altitude if pixel is land or has a positive altitude

```
If (LAND_F(j,f)) then
    $P_{ECMWF}(j,f) = P_{Sea\_ECMWF}(j,f).\ \exp(-\max(0,z(j,f))/H_P)$        (2.1.0-15)
Else
    $P_{ECMWF}(j,f) = P_{sea\_ECMWF}(j,f)$
End if
```

Test validity of pixel (step 2.1.11)

```
    If (INVALID_F(j,f) == TRUE) then                                     (2.1.11-1)

            no further processing is performed on that pixel

    Else
```

3.5.2 - Reflectance conversion (step 2.1.4)

Computation of Irradiance seasonal variation correction factor (once for the whole product)
call pl_sun input: (JD1+JD2)/2 output: sun_pos *(2.1.4-2)*

$$\text{seasonal\_fact} = \frac{\|\text{sun\_pos}\|^2}{\text{Dsun}_0^2} \qquad (2.1.4\text{-}3)$$

exception processing: $\theta_s >= 90°$:
Note: this exception is not expected ever to happen in nominal MERIS operation
INVALID_F (j, f) = TRUE
no further processing is performed on that pixel
end of exception processing

For each band b in { b412..b900 }

$$\rho_{TOA}(b,j,f) = \frac{\pi.\text{TOAR}(b,j,f)\cdot\text{seasonal\_fact}}{\cos(\theta_s(j,f)).F_0(b,\text{Detector}(j,f))} \qquad (2.1.4\text{-}1)$$

```
            SATURATED_F(b,j,f)= ( TOAR(b,j,f) >= SATURATION_L(b) )          (2.1.4-4)
        Endfor

    Endif     end of processing for current valid pixel
Endfor      end of processing for current pixel
```

3.6 - Quality Control and Diagnostics

N/A.

3.7 - Exception Handling

See the blocks labelled "exception processing:... end of exception processing" in section 3.5 above.

Copyright © 2011 ACRI S.T.

4. - MERIS pressure processing

4.1 – Introduction

This section describes the two independent algorithms used to derive pressure estimates from the radiometric measurements of each pixel.

4.2 - Algorithm Overview

The MERIS level 2 pressure processing is applied to each pixel and follows the logic shown in the flow chart in figure 4.2-1.

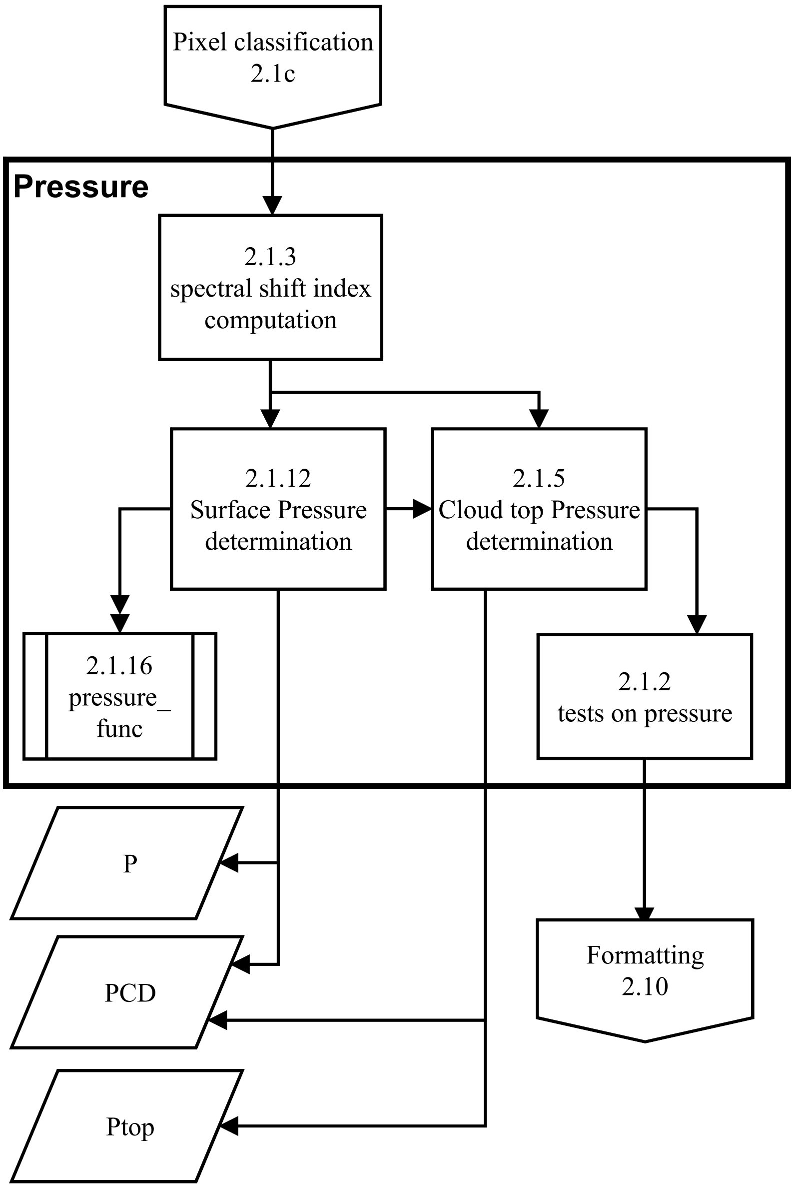


Figure 4.2-1 : MERIS level 2 Pressure processing (step 2.1b)

Copyright © 2011 ACRI S.T.

4.3 - Mathematical Description Of Algorithm

4.3.1 - Atmospheric pressure estimate (steps 2.1.5, 2.1.12)

The pressure is estimated over Land and Cloud pixels from the MERIS bands 10: 753.75 nm and 11: 760.625 nm, using dedicated Neural Net (NN) algorithms.
The basic inputs of both Neural Nets are:

- The transmittance of molecular oxygen at 761 nm, estimated from bands 11 – with absorption – and a non-absorbing reference: band 10 over clouds or a combination of bands 10 & 12 over land.
- The wavelength of band 11 at every pixel is an input of the neural nets.
- The signal in band 10 (either in radiance – CTP – or normalised radiance – P_S)
- The illumination and observation geometry.

- The "cloud top pressure" NN also uses a priori knowledge of the surface albedo, where appropriate.
- The "surface pressure" NN also uses an a priori knowledge of the aerosol optical thickness.

To retrieve the Cloud Top Pressure, P_{top},a neural net (NN) approach is used. The algorithm is detailed in RD 8, 2.3. The MERIS signals in channel 10, 11, the surface albedo α_{surf} and the geometry (sun zenith angle, viewing zenith angle and azimuth angle) are used as input of the Neural Network. The net produces the cloud Top pressure P_{top}. Depending on the surface albedo two different neural nets are used (one for surface albedo equal to zero, one for non-zero surface albedo). Neural Nets are selected according to spectral shift index.

The Neural Nets apply generic Neural Net functions, as specified in AD6, to specific auxiliary parameters and inputs, to obtain the required outputs. All specific aspects of the application are described in section 4.5 below.

Each pressure estimate produces Product Confidence Data (PCD).

Copyright © 2011 ACRI S.T.

4.4 - List of Variables

Note: in this section and section 4.5 below, J and F are the notation for the index of column and frame corresponding to tie points, j and f are (as in the rest of this DPM) the product column and frame indices.

Variable	Descriptive Name	T	U	Range - References
LAND_F(j,f)	*A priori* Land flag (TRUE when land, FALSE when ocean)	i	dl	from Level1b (Flags MDS)
LANDCONS_F(j,f)	Consolidated Land flag	i	dl	From 2.1c (§5.4)
CLOUD_F(j,f)	Cloud flag	i	dl	From 2.1c (§5.4)
Detector(j, f)	MERIS detector associated to pixel	i	dl	from Level 1b (Flags MDS)
month	Month of acquisition	i	-	from L1B (MPH)
θ_s(j,f)	Sun zenith angle for pixel (j,f)	i	deg	from 2.1a (§3.4)
θ_v(j,f)	Viewing zenith angle for pixel (j,f)	i	deg	idem
$\Delta\phi$(j,f)	Azimuth difference angle for pixel (j,f)	i	deg	idem
lat(j,f)	Latitude for pixel (j,f)	i	deg	idem
lon(j,f)	Longitude for pixel (j,f)	i	deg	idem
ρ(b,j,f)	stratospheric aerosol corrected reflectance at pixel (j,f)	i	dl	b753 & b761, from step 2.1c (§5.4
P_{ECMWF}(j,f)	surface pressure for pixel (j,f) (from interpolated ECMWF sea level pressure, corrected for pixel altitude)	i	hPa	from 2.1a (§3.4)
SATURATED_F(b,j,f)	Saturated pixel flag	i	-	from step 2.1a (§3.4)
f_{SL}	Correction factor for residual stray-light in band 11	s	-	
λ_{theo}(b)	Nominal wavelengths of MERIS bands	s	nm	
λ^C_{761}(detector)	band 11 wavelength optimised for pressure retrievals	s	nm	Select for FR or RR according to input L1
SP_{NN}_min, SP_{NN}_max	Validity ranges for surface pressure NN inputs	s		
AOT_P	Default aerosol optical depth	s	dl	
P_{Smin}, P_{Smax}	Validity range for surface pressure NN output	s	hPa	
ΔP_{Smax}	Max allowed difference between P_S & P_{ECMWF}	s	hPa	
E^P_{753}	Solar flux reference value at b753 consistent with CTP NN	s	EU	
E^{CTP}_{ratio}	Solar flux ratio, consistent with CTP NN, to convert reflectance ration into normalised radiance ratio	s	dl	
Surfalb_b11_LUT [lat,lon,month]	LUT of surface albedo at b761as a function of latitude, longitude and month of year	s	dl	lat: latitude, lon: longitude, month: 1..12
min_TOARb753	Minimum acceptable value for TOAR(b753)	s	LU	

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
max_TOARb753	Maximum acceptable value for TOAR(b753)	s	LU	
MAX_PRESSURE	Maximum acceptable value for pressure	s	hPa	
	Cloud Top Pressure Neural networks tables	s	-	see step 2.1.5.0 in § 4.5.1 below
	Surface Pressure Neural network table	s	-	See step 2.1.12.0 in § 4.5.1 below
CTP_NoAlbedo_Net	Neural network object	c	-	CTP NN for null surface albedo
CTP_Albedo_Net	Neural network object	c	-	CTP NN for positive surface albedo
SurfP_Albedo_Net	Neural network object	c	-	Surface Pressure NN
ρ^{C}_{761}	Reflectance in b761 corrected for residual stray-light	c	dl	
$\frac{\partial\rho}{\partial\lambda}$	Local spectral slope of reflectance	c	nm^{-1}	
$\rho^{761\ no\ abs}$	Estimation of the reflectance at b761 without O_2 absorption	c	dl	
L_N753	Normalised radiance at b753	c	sr^{-1}	
NN_Input	Neural Network input vector	c		
NN_Output	Neural Network output vector	c		
L^{C}_{753}	TOA radiance at b753 rebuilt using solar flux consistent with CTP NN	c	LU	
R_761_753	O2 transmittance at b761 (radiance ratio)	c	dl	
ORIN_NN	Out of range input flag for neural net	c	dl	Boolean
OROUT_NN	Out of range output flag for neural net	c	dl	Boolean
$\alpha_{surf}(j, f)$	Surface albedo at band 11 for pixel (j,f)	o	dl	to step 2.4 (§7.5)
$\eta_{LN}(j,f)$	O2 transmittance at b761 (reflectance or normalised radiance ratio)	o	dl	to step 2.3.1 (§6.5.3)
$P_S(j,f)$	Surface pressure	o	hPa	to steps 2.3 (§6), 2.10 (§10.4), to Breakpoint
P_{top} (j,f)	Cloud top pressure	o	hPa	to steps 2.4 (§7.4), 2.10 (§10.4), to Breakpoint
PCD_NN_F (j,f)	Pressure products confidence flag	o	dl	Boolean; to step 2.10 (§10.4), to Breakpoint
errcode(j,f)	Cloud top pressure error flags	o	dl	to Breakpoint

NOTE:

1. all calculated and output Boolean parameters shall be initialised to FALSE (0).
2. subscript FR or RR for type s variable λpix is omitted in equation section below. Proper variable shall be selected according to processed product resolution.

Copyright © 2011 ACRI S.T.

4.5 – Equations (step 2.1b)

Surface Pressure and Cloud Top Pressure processing uses two Neural Network structures to derive Pressure from TOA radiances at bands b753, b761, wavelength of band 11, viewing and illumination geometry, surface albedo or aerosol optical thickness. The Neural Network structures must be created at process initialisation by decoding tables in NNff format (see AD6), read from the auxiliary parameters data files. They are then activated in turn for each relevant pixel.

4.5.1 - Neural Networks Initialisation (steps 2.1.5.0, 2.1.12.0)

```
At process initialisation
read GADS- Cloud Neural network for non-zero albedo into memory (see AD6)      (2.1.5.0-1)
call NN_CreateNetFromMemFile routine (see AD6)                                 (2.1.5.0-2)
    input: address of memory copy of GADS
    return value: CTP_Albedo_Net
read GADS- Cloud Neural network for null albedo into memory (see AD6)          (2.1.5.0-3)
call NN_CreateNetFromMemFile routine (see AD6)                                 (2.1.5.0-4)
    input: address of memory copy of GADS
    return value: CTP_NoAlbedo_Net
read GADS- Surface Pressure Neural network into memory          (see AD6)      (2.1.12.0-1)
call NN_CreateNetFromMemFile routine (see AD6)                                 (2.1.12.0-2)
    input: address of memory copy of GADS
    return value: SurfP_Albedo_Net
end of process initialisation section
```

Product processing :

For each VALID pixel (in column j, frame f):

Deleted (step 2.1.3)

```
compute reflectance ratio
    If ( (ρ(b753,j,f) > 0) and not SATURATED_F(b753,j,f)
          and not SATURATED_F(b761,j,f) then
        $\rho_{761}^{C} = \rho(b761,j,f) + f_{SL}(Detector(j,f)) \cdot \rho(b753,j,f)$      (2.1.12-1)
        If not SATURATED_F(b779,j,f) then
            $\frac{\partial\rho}{\partial\lambda} = \frac{\rho(b775,j,f) - \rho(b753,j,f)}{\lambda_{theo}(b775) - \lambda_{theo}(b753)}$
                                                                                (2.1.12-2)
            $\rho^{761\text{ no abs}} = \rho(b753,j,f) + \left(\lambda_{761}^{C}(Detector(j,f)) - \lambda_{theo}(b753)\right) \cdot \frac{\partial\rho}{\partial\lambda}$
        Else
            $\rho^{761\text{ no abs}} = \rho(b753,j,f)$
        Endif
```

Copyright © 2011 ACRI S.T.

```
    $\eta_{LN}(j,f) = \frac{\rho^C_{761}}{\rho^{761\ \text{no abs}}}$                                   (2.1.12-3)
else
    $\eta_{LN}(j,f) = 0$                                                   (2.1.12-5)
Endif
```

4.5.2 - Surface pressure neural net method (step 2.1.12)

Note: Step 2.1.12 has been fully revised, all equations are new and there is no correspondence with previous issues of this document.

```
Check pixel classification
  If LANDCONS_F(j,f) then

    Check input ranges
    $L_N753 = \cos\theta_s.\rho(b753, j, f)/\pi$
    If ( $L_N753 <$ SP_NN_min(1) OR $L_N753 >$ SP_NN_max(1) OR
         $\eta_{LN}(j,f) <$ SP_NN_min(2) OR $\eta_{LN}(j,f) >$ SP_NN_max(2) OR
         $\cos\theta_S(j,f) <$ SP_NN_min(4) OR $\cos\theta_S(j,f) >$ SP_NN_max(4) OR
         $\cos\theta_V(j,f) <$ SP_NN_min(5) OR $\cos\theta_V(j,f) >$ SP_NN_max(5) OR
         $\sin\theta_v(j,f)).\cos(180-\Delta\phi(j,f)) <$ SP_NN_min(6) OR $\sin\theta_v(j,f)).\cos(180-\Delta\phi(j,f)) >$ SP_NN_max(6) OR
         $\lambda^C_{761}$(Detector(j,f)) < SP_NN_min(7) OR $\lambda^C_{761}$(Detector(j,f)) > SP_NN_max(7) )    then

      ORIN_NN = TRUE                                                (2.1.12-6)

    Endif
    If $\eta_{LN}(j,f) <= 0$ then ORIN_NN = TRUE                             (2.1.12-4)
    If ORIN_NN == FALSE then
      NN_Input(1) = $L_N753$                                        (2.1.12-7)
      NN_Input(2) = $\eta_{LN}(j,f)$                                (2.1.12-8)
      NN_Input(3) = $AOT_P$                                         (2.1.12-9)
      NN_Input(4) = $\cos\theta_s(j,f)$                             (2.1.12-10)
      NN_Input(5) = $\cos\theta_v(j,f)$                             (2.1.12-11)
      NN_Input(6) = $\sin\theta_v(j,f)).\cos(180-\Delta\phi(j,f))$  (2.1.12-12)
      NN_Input(7) = $\lambda^C_{761}$(Detector(j,f))                (2.1.12-13)

      call Nn_ProcessNet routine (see AD6)                          (2.1.12-14)
        Network: SurfP_Net; input: NN_Input; number of input elements: 7; output:
        NN_output; number of output elements: 1

    Post-processing after Neural Network call:
      $P_S(j, f)$ = NN_output(1)                                    (2.1.12-15)

      If: ($P_S(j, f) < P_{Smin}$ OR $P_S(j, f) > P_{Smax}$ OR $|P_S(j, f) - P_{ECMWF}(j,f)| > \Delta P_{Smax}$) then
        PCD_NN_F( j,f) = TRUE                                       (2.1.12-16)
      Endif
    Else
      $P_S(j, f)$ = BAD_VALUE                                       (2.1.12-17)

      PCD_NN_F( j,f) = TRUE                                         (2.1.12-18)
```

Copyright © 2011 ACRI S.T.

```
    Endif
```

4.5.3 – Cloud top pressure neural net method (step 2.1.5)

```
  Else If CLOUD_F (j, f) then
```

The surface albedo is read from a LUT as a function of geographical latitude, geographical longitude and month of measurement derived from the time index found in the Level 1b product header.

```
    If LAND_F(j,f) then                                                          (2.1.5-1)
        $\alpha_{surf}$ = Surfalb_b11_LUT nearest:(lat (j, f), lon(j, f)) select:(month)
    Else
        $\alpha_{surf}$ =0.
    Endif

    deleted                                         (2.1.5-2)
    Correct b761 for residual stray-light
    $\rho_{761}^{C} = \rho(b761, j, f) + f_{SL}(Detector(j, f)) \cdot \rho(b753, j, f)$          (2.1.5-2a)
    Compute radiance at bands b753 and b761 with dedicated solar flux values
    $L_{753}^{C} = \frac{1}{\pi} \rho(b753, j, f) \cdot E_{753}^{P} \cdot \cos \theta_s (j, f)$          (2.1.5-2b)

    If ( $L_{753}^{C}$ <= max(0,min_TOARb753)) then
        bit 0 of errcode =1; endif                                              (2.1.5-3)
    If ( $L_{753}^{C}$ > max_TOARb753 OR SATURATED_F(b753,j,f)) then
        bit 1 of errcode = 1; endif                                             (2.1.5-4)
    If ( $\rho_{761}^{C}$ <= 0) then
        bit 2 of errcode = 1; endif                                             (2.1.5-5)
    If (SATURATED_F(b761,j,f)) then
        bit 3 of errcode = 1; endif                                             (2.1.5-6)

    deleted                     (2.1.5-7)
exception processing: errcode != 0:
        bit 0 of errcode =1
        ORIN_NN = TRUE
        $P_{TOP}(j, f)$ = BAD_VALUE
        Continue processing at Equation 2.1.5-25
end of exception processing

    Compute ratio, convert to radiance using dedicated solar flux ratio
    R_761_753= $\frac{\rho_{761}^{C}}{\rho(b753, j, f)} \cdot E_{ratio}^{CTP}$          (2.1.5-8)
```

Copyright © 2011 ACRI S.T.

```
        If ($\alpha_{surf}$ == 0) then
            NN_Input(1) = $L^C_{753}$                                        (2.1.5-9)
            NN_Input(2) = R_761_753                                          (2.1.5-10)
            NN_Input(3) = cos $\theta_s$                                     (2.1.5-11)
            NN_Input(4) = cos $\theta_v$                                     (2.1.5-12)
            NN_Input(5) = sin $\theta_v$.cos $\Delta\phi$                    (2.1.5-13)
            NN_Input(6) = $\lambda^C_{761}$(Detector(j,f))                   (2.1.5-14)

            call Nn_ProcessNet routine (see AD6)                             (2.1.5-15)
                Network: Ctp_NoAlbedo_Net; input: NN_Input; number of input elements: 6;
                output: NN_output; number of output elements: 1

        else
            NN_Input(1) = $\alpha_{surf}$                                    (2.1.5-16)
            NN_Input(2) = $L^C_{753}$                                        (2.1.5-17)
            NN_Input(3) = R_761_753                                          (2.1.5-18)
            NN_Input(4) = cos$\theta_s$                                      (2.1.5-19)
            NN_Input(5) = cos $\theta_v$                                     (2.1.5-20)
            NN_Input(6) = sin $\theta_v$.cos $\Delta\phi$                    (2.1.5-21)
            NN_Input(7) = $\lambda^C_{761}$(Detector(j,f))                   (2.1.5-22)

            call Nn_ProcessNet routine (see AD6)                             (2.1.5-23)
                Network: Ctp_Albedo_Net; input: NN_Input; number of input elements: 7;
                output: NN_output; number of output elements: 1
        Endif
    Post-processing after Neural Network call:
        $P_{top}$ (j, f) = NN_output(1)                                      (2.1.5-24)
exception processing: ($P_{top}$ (j, f) <= 0 OR $P_{top}$ (j, f) > MAX_PRESSURE):
        If ($P_{top}$ (j, f) <= 0) then $P_{top}$ (j, f) = 0; Endif
        If ($P_{top}$ (j, f) > MAX_PRESSURE) then $P_{top}$ (j, f) = MAX_PRESSURE; Endif
        OROUT_NN = TRUE
        continue at next equation
end of exception processing
        PCD_NN_F (j, f) = ORIN_NN OR OROUT_NN                                (2.1.5-25)

    Endif Pixel is cloud
Endfor    end of processing for current pixel
```

Copyright © 2011 ACRI S.T.

4.5.4 - deleted (step 2.1.2)

4.5.5 - deleted (proc. 2.1.16)

4.6- Quality Control and Diagnostics

The flag PCD_NN_F signals out-of-range input or output in the NN pressure estimates (eq. 2.1.12-16, 2.1.12-18 and 2.1.5-25).

4.7 - Exception Handling

See the blocks labelled "exception processing:... end of exception processing" in section 4.5 above.

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

5. - MERIS pixel identification

5.1 - Introduction

Before pursuing the processing of the pixels towards **Cloud, Land or Water processing**, or to **exempt them from further processing**, it is necessary to screen sort into four categories:

- cloud,
- land,
- water,
- invalid

This chapter describes the processing to be applied to the MERIS pixels in order to achieve this classification (see RD8, §2.17), starting from the level 1b geo-location based land/water a priori classification:

- Cloud pixels identification
- Stratospheric aerosol correction, when applicable, i.e. in the periods following a massive release of particles into the stratosphere, such as may occur after volcanic eruptions.
- Corrections of reflectance for absorption by ozone, molecular oxygen and water vapour.
- Consolidation, using radiometry, of the land/water *a priori* classification of non cloudy pixels for which *a priori* surface classification has low confidence (because of geo-location uncertainty, high tide zones, seasonal variations of water level…).

5.2 - Algorithm Overview

The MERIS level 2 pixel identification is applied to each pixel and follows the logic shown in the flow chart in figure 5.2-1.

Copyright © 2011 ACRI S.T.

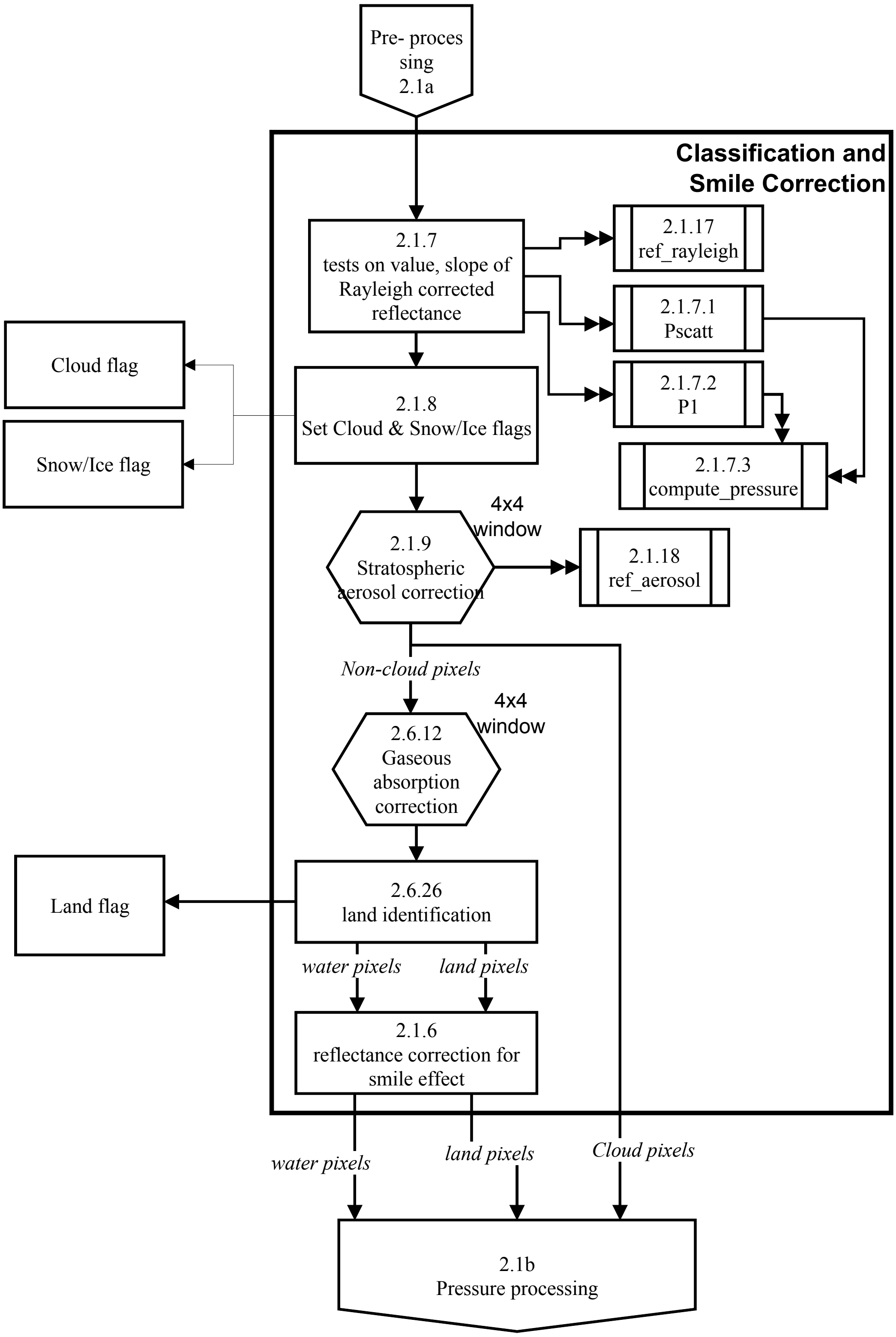


Figure 5.2-1 : MERIS level 2 pixel classification (step 2.1c)

Copyright © 2011 ACRI S.T.

5.3 - Mathematical Description Of Algorithm

5.3.1 - Cloud screening (steps 2.1.2, 2.1.7, 2.1.8)

The algorithm to screen clouds from other surfaces is described in RD8, section 2.17.

As clouds are often bright, a first stage screening is done for the determination of bright pixels using tests and auxiliary data values depending on a priori knowledge classification if the surface among Land and Water. Apparent pressure is also computed using specific procedures over Land and Water to allow detection of high altitude thin clouds not captured by the bright tests.
Those two tests, plus additional tests on the spectral slope of the Rayleigh corrected reflectance over Land, aiming at discarding sand and snow/ice, are used to derive a Cloud mask, as well as a Snow/Ice flag applicable to non-cloudy pixels.

5.3.2- Stratospheric Aerosol Correction (step 2.1.9)

When the switch to perform stratospheric aerosol corrections is set, all valid pixels are corrected for stratospheric aerosol transmission and scattering. Correction applies to all TOA reflectance bands.
The correction algorithm is similar to the one described in section 9.3.3 below; the stratospheric aerosol parameters are read from auxiliary parameters sets. The algorithm runs on a 4x4 pixels window.

5.3.3 - Gaseous absorption corrections (step 2.6.12)

Gaseous correction processing is organised in three steps, O_3, O_2 and H_2O correction. Input is the TOA reflectance for the MERIS channels, corrected for stratospheric aerosols when applicable: ρ (b, j, f). Its output is the reflectance corrected for gaseous absorption (ρ_{ng}). All three algorithms apply polynomial expressions using LUT technique. In step 2.6.12.1 the O_3 transmittance is estimated over a 4*4 pixel window [*]. In step 2.6.12.2 and 2.6.12.3 the O_2 and H_2O transmittances are estimated for each land pixel. Because the signals used are weak, an average of radiances is performed on water pixels to compute the transmittances.

The block diagram in fig. 5.3.3-1 below shows the control flow for step 2.6.12.

---

[*] this correction uses the assumption, that both total Ozone and zenith angles variations within the 4x4 pixels window are so small that they have no effect on the computation, either in RR or in FR mode.

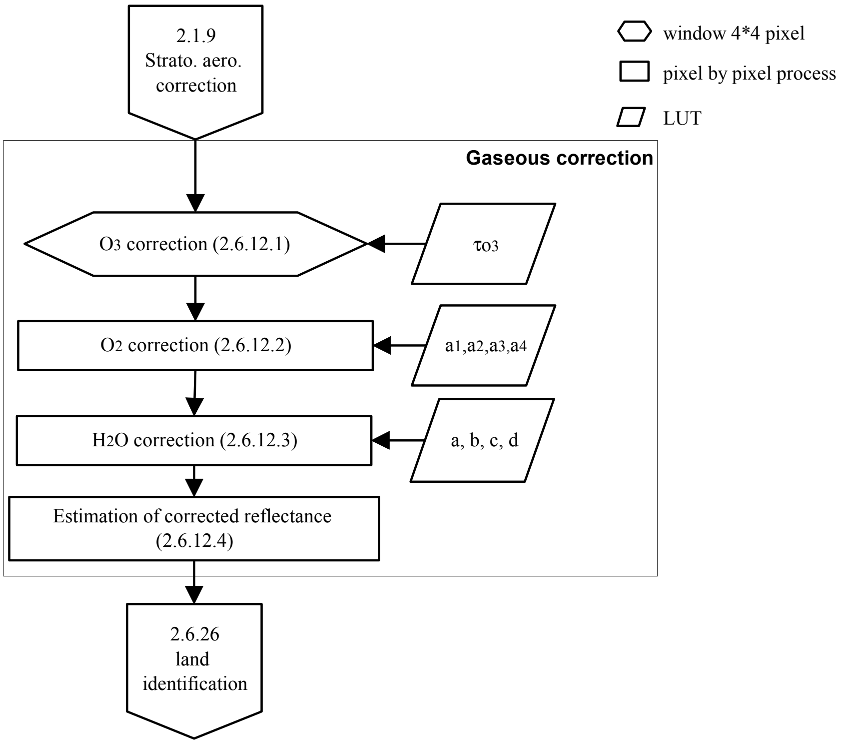


Figure 5.3.3-1: Gaseous correction processing (step 2.6.12)

5.3.4 - Land Identification (step 2.6.26) and Smile Effect Correction (step 2.1.6)

The purpose of this classification is to identify using geo-physical data, land from water pixels in cases where the Level1b *a priori* classification leads to ambiguities which may occur from:

- geo-location error;
- land /ocean atlas error: uncharted land or water, etc.;
- transient emerged land: tidal flats, etc.

This cases are identified using the Surface Confidence Map, an atlas identifying zones of low-confidence in the *a priori* land/water classification map used in the level1b. When the Surface Confidence Map indicates high confidence classification, the Land Identification radiometric tests are by-passed and the *a priori* classification is kept.

Inland water

First, a test on the reflectance corrected for gaseous absorption at 665 nm is performed to identify the darkest pixels. The TOA reflectance at 665 nm is compared to a threshold interpolated from a LUT.

For the pixels having a reflectance smaller than this threshold, a second test is made to compare the TOA reflectance at 665 nm with the TOA reflectance at 865 nm ; if the TOA reflectance at 665 nm is greater than the reflectance at 865 nm, the pixel is classified as water.

Land in water

The purpose of this test is to identify pixels of emerged land, flagged as "water" in the L1B product. It is the opposite of the Inland water test.

The purpose of the Smile Effect Correction is to correct TOA reflectance (already corrected for stratospheric aerosols and gaseous absorption) for small scale variations due to non-constant central wavelength of a given band across the field of view. Correction is made only for a subset of bands for which those variations can induce severe distortions after corrections based on fixed wavelength scheme (e.g. Rayleigh diffusion correction). This subset of bands, which is specific to each land and water surface type, should ensure smoothness of reflectance local variations with wavelength and allow a good estimation of the reflectance derivative using neighbour bands.

The block diagram in figure 5.3.4-1 below shows the control flow in step 2.6.26.

Note : Steps 2.6.4 and 2.6.31, because of their simplicity, are only shown in the corresponding equation section.

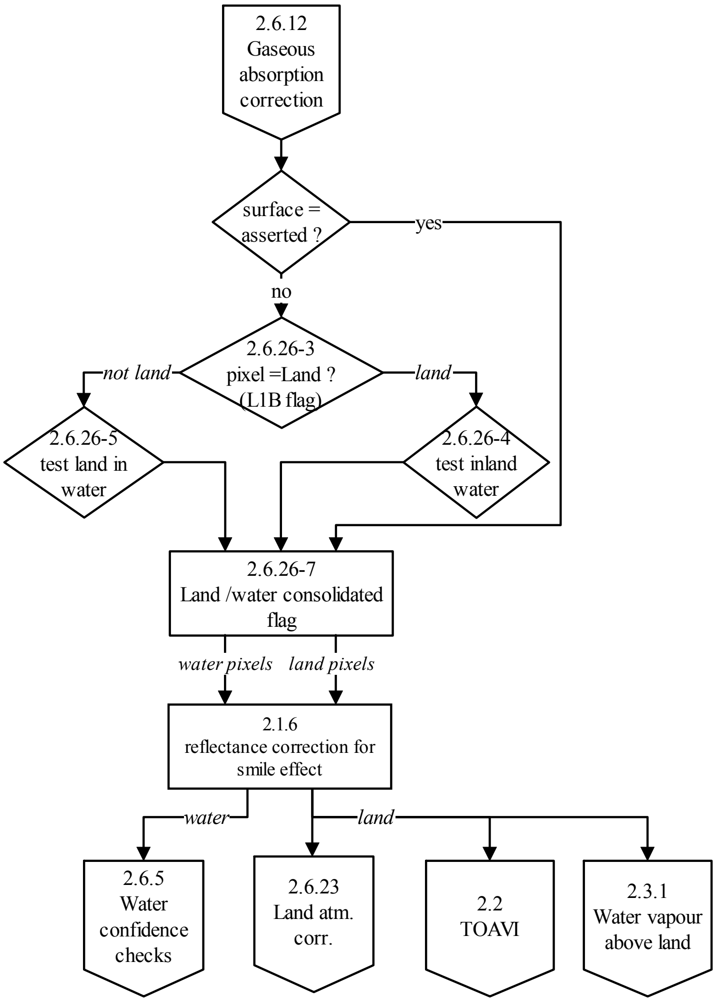


Figure 5.3.4-1 : Step 2.6.26 Land identification and Smile Correction
(Note: numbers in figure 5.3.4-1 above may refer directly to equation numbers in §5.5 below).

Copyright © 2011 ACRI S.T.

5.4 - List of Variables

Note: in this section and section 5.5 below, J and F are the notation for the index of column and frame corresponding to tie points, j and f are (as in the rest of this DPM) the product column and frame indices.

Variable	Descriptive Name	T	U	Range - References
LAND_F(j,f)	Land flag (TRUE when land, FALSE when ocean)	i	dl	from step 2.1a (§3.4)
INVALID_F(j,f)	Invalid flag	i	dl	idem
Detector(j, f)	MERIS detector associated to pixel	i	dl	from Level 1b (Flags MDS)
ρ_{TOA} (b,j,f)	TOA reflectance at pixel (j,f)	i	dl	all bands, from 2.1a (§3.4)
θ_s(j,f)	Sun zenith angle	i	deg	from 2.1a (§3.4)
θ_v(j,f)	Viewing zenith angle	i	deg	idem
$\Delta\phi$(j,f)	Difference of azimuth angles	i	deg	idem
ϕ_s(j,f)	Sun azimuth angle for pixel (j,f)	i	deg	idem
P_{ECMWF}(j, f)	ECMWF surface pressure	i	hPa	idem
U_{O3} (j, f)	Actual ozone content for pixel (j, f)	i	DU	idem
lat(j,f)	Latitude for pixel (j,f)	i	deg	idem
lon(j,f)	Longitude for pixel (j,f)	i	deg	idem
W_u(j,f), W_v(j,f)	Wind vector components for pixel (j,f)	i	$m.s^{-1}$	idem
SATURATED_F(b,j,f)	Saturated pixel flag	i	-	from step 2.1a (§3.4)
STRAT_CORR	Switch to perform stratospheric aerosol correction	s	dl	Boolean
b_bright	Index of band for test on Rayleigh corrected reflectance	s	dl	
b_bright2	Index of band for test on TOA reflectance	s	dl	
b_slope1_n	Index of numerator band for test 1	s	dl	
b_slope1_d	Index of denominator band for test 1	s	dl	
b_slope2_n	Index of numerator band for test 2	s	dl	
b_slope2_d	Index of denominator band for test 2	s	dl	
τ_R(b)	Rayleigh optical thickness at standard pressure for all bands	s	dl	
P_{std}	Standard pressure	s	hPa	= 1013.25
Slope_1_low	Lower limit of slope range for test 1	s	dl	RD 8, §2.17
Slope_1_high	Upper limit of slope range for test 1	s	dl	RD 8, §2.17
Slope_2_low	Lower limit of slope range for test 2	s	dl	RD 8, §2.17
Slope_2_high	Upper limit of slope range for test 2	s	dl	RD 8, §2.17
Rho_rc_LUT [k,θ_s, θ_v, $\Delta\phi$]	LUT of thresholds on Rayleigh corrected reflectance at 442nm, index k selects records for ocean or land pixels	s	dl	
rho$_{TOA}$_thresh	Threshold on TOA reflectance at band b_bright2	s	dl	

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
MDSI_Thresh	Threshold on MDSI	s	dl	
P_1_Thresh[2]	Apparent pressure threshold over Land	s	dl	First element to be used far from coastline, 2nd when close.
rho$_{753}$_thresh	Minimum b10 reflectance value to consider apparent pressure	s	dl	
P_{scatt}_Thresh	Apparent pressure threshold over Water	s	dl	
R_{10_12}_thresh	Minimum b10-b12 spectral slope value to consider apparent pressure	s	dl	
Stratospheric_LUT [lat, lon]	Map of stratospheric aerosol model	s	dl	lat: latitude, lon: longitude; value: 0..18
Strato_rad [iaer_sa]	Table of effective radius index for strato. aerosol	s	dl	iaer_sa: 1..18
Strato_multi [iaer_sa]	Table of index in multiple scattering LUT for strato. aerosol	s	dl	iaer_sa: 1..18
Strato_aerpha_LUT [cosΘ_{scat}, i_eff_radius, band]	Aerosol phase function times single scattering albedo values as a function of cosine of scattering angle and effective radius index	s	dl	i_eff_radius: 1..3 band: b412..b900
Strato_tau [iaer_sa]	Table of optical thickness at a reference band for strato. aerosol	s	dl	iaer_sa: 1..18
Strato_spectr [i_eff_radius, band]	Table of spectral dependency of optical thickness as a function of stratospheric aerosol effective radius	s	dl	i_eff_radius: 1..3 band: b412..b900
TA_Strato_LUT [iaer_sa, band, θ]	Look-up table of stratospheric aerosol transmittance	s	dl	iaer_sa: 1..18 band: b412..b900 θ: 12 values
Strato_sphalb [iaer_sa, b]	Look-up table of stratospheric aerosol spherical albedo	s	dl	iaer_sa: 1..18 band: b412..b900
$\tau_{O3\_norm}(b)$	Ozone optical thickness corresponding to 1cm.atm for all bands	s	dl	
T_{O2}_LUT[λ^{779},L_N^{779},θ_s,θ_v,$\Delta\phi$]	Look-up-table for O_2 correction at 779 nm (b779)	s	dl	λ: 21 values, L_N^{779}: 25 values θ_s, θ_v, $\Delta\phi$: 15, 10 & 19 values respectively
H_2O_{705}Corr_Poly_LUT[iλ, k]	Polynomial coefficients for H_2O transmission correction at 709nm (b705)	s	dl	iλ: ref. wavelength index, k: polynomial order
λ_{ref}[iλ]	Reference wavelengths grid for correction coefficients of H20 at 709nm	s	nm	iλ: ref. wavelength index
$a_h(b)$,$b_h(b)$,$c_h(b)$,$d_h(b)$	Polynomial coefficients for H_2O correction for all bands, except 709nm	s	dl	b: b412..b753, b779..b885 a,b,c,d: polynomial order 0..3
$b_{thresh}(b)$	Indices of bands to be used for comparison with threshold within the island and in-land waters screening	s	dl	1st value for land and ocean with glint, 2nd for ocean outside glint area
ρ_{thresh}_LUT[b,θ_s, θ_v, $\Delta\phi$]	LUT containing threshold values for island and in-land waters screening	s	dl	2 b values 78 (θ_s,θ_v), 19 $\Delta\phi$ values

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
β_L	Threshold on spectral slope used in inland waters screening over land	s	dl	
β_W	Threshold on spectral slope used in island screening over waters	s	dl	
$\alpha_{thresh}(b)$	Constant applying to threshold value derived from LUT. Allows to take into account environment and bathymetric effects	s	dl	2 b values
Surface_Confidence_Map[lat,lon]	Atlas map for confidence on a priori surface type (land/water) knowledge	s	dl	Boolean, true = uncertain surface type
SATURATION_L(b)	Default radiance for saturated pixels	s	LU	All bands
LandRefCorr_sw(b)	array of per band switches enabling Smile Effect Correction for Land pixels reflectance	s	-	b in {b412…b900}
LandRefCorr_b(b,i)	array of pairs of band indices for estimation of reflectance spectral derivative (Land pixels)	s	-	b in {b412…b900}, i in {1,2} (1 for lower wavelength, 2 for upper)
WaterRefCorr_sw(b)	array of per band switches enabling Smile Effect Correction for Water pixels reflectance	s	-	b in {b412…b900}
WaterRefCorr_b(b,i)	array of pairs of band indices for estimation of reflectance spectral derivative (Water pixels)	s	-	b in {b412…b900}, i in {1,2} (1 for lower wavelength, 2 for upper)
$\lambda_{pix}^{FR}(b,k)$	Characterised central wavelengths for each MERIS FR detector and each band	s	nm	b: any MERIS band, k: detector index
$\lambda_{pix}^{RR}(b,k)$	Characterised central wavelengths for each MERIS RR detector and each band	s	nm	idem
$\rho_R$tab_LUT $[\theta_S, b, \theta_V, \Delta\phi, W_S]$	LUT for the Rayleigh reflectance above water	s	dl	[†]
Parameters and LUTs for ref_rayleigh procedure (step 2.1.17, §5.5.6)				
{A,B}	Coefficients to correct for molecule anisotropy	s	dl	A = 0.9587256 B = 0.0412744
Rayscatt_coef_LUT $[\theta_s,\theta_v,s,k]$	LUT of polynomial coefficients for the 3 Fourier series terms used to compute the correction factor for Rayleigh multiple scattering	s	dl	78 (θs,θv) couples s : 0,1,2 k : 1..4
Parameters and LUTs for ref_aerosol procedure (step 2.1.18, §5.5.7)				
Aermult_LUT$[\theta_s,\theta_v,$iaer,s,k$]$	Polynomial coefficients for each of the Fourier terms used to compute the correcting factor for aerosol multiple scattering	s	dl	used by ref_aerosol procedure
Parameters and LUTs for ref_smile_corr procedure (step 2.1.6, §5.5.8)				
$\lambda_{pix}^{FR}(b,k)$	Characterised central wavelengths for each MERIS FR detector and each band	s	nm	used by ref_smile_corr procedure
$\lambda_{pix}^{RR}(b,k)$	Characterised central wavelengths for each MERIS RR detector and each band	s	nm	used by ref_smile_corr procedure

---

[†]: the increasing order of magnitude for θ_s and θ_v indices in the LUT files, imposes a decreasing order for the corresponding μ_s and μ_v cosines.

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
λ_{theo}(b)	Theoretical wavelengths corresponding to smile corrected reflectances	s	nm	used by ref_smile_corr procedure
Parameters and LUTs for Pscatt (step TBD, §5.5.9)				
λ_{theo}(b)	Theoretical wavelengths corresponding to smile corrected reflectances	s	nm	
TO2_Ray_LUT[$\lambda,\theta_s,\theta_v$]	O2 Rayleigh transmittance	s	dl	λ: 21 $\theta_{s,}\theta_v$: 24
TO2_Atm_Aer_LUT[$\lambda,\theta_s,\theta_v$]	O2 aerosol atmospheric transmittance for Ha=2km	s	dl	λ: 21 $\theta_{s,}\theta_v$: 24
TO2_Fresnel_LUT[$\lambda,\theta_s,\theta_v$]	O2 Aerosol Fresnel transmittance	s	dl	λ: 21 $\theta_{s,}\theta_v$: 24
APF_Junge_LUT[i]	APF of the Junge aerosol model nb 10	s	dl	i: 181
fresnel_Coeff_LUT[i]	Fresnel coefficients	s	dl	i: 91
Parameters and LUTs for P1 (step TBD, §5.5.10)				
λ_{theo}(b)	Theoretical wavelengths corresponding to smile corrected reflectances	s	nm	
Parameters and LUTs for ComputePressure (step TBD, §5.5.11)				
λ_{ref}_O2	Reference wavelength values for the Pscatt and P1 LUTs	s	nm	21 values
θ_{ref}_O2	Reference zenith angle values for the Pscatt and P1 LUTs	s	deg	24 values
pressLevel	Reference pressure levels for TO2_Atm_LUT	s	hPa	21 values
TO2_Atm_LUT[λ_{761}, layer,θs,θv]	O2 atmospheric transmittance LUT	s		λ_{761}: 21 values, layer: 21 values, θs,θv 24 values each
M(j,f)	Air mass	c	dl	
ρ_{rc1} (b)	Coarse Rayleigh corrected reflectance	c	dl	b in {b_bright, b_slope1_n, b_slope1_d, b_slope2_n, b_slope2_d}
SLOPE_1_F	Spectral slope test 1 flag	c	dl	Boolean, to Breakpoint
SLOPE_2_F	Spectral slope test 2 flag	c	dl	Boolean, to Breakpoint
rho_thresh	Threshold on Rayleigh-corrected reflectance	c	dl	
BRIGHT_RC_F(j,f)	Bright flag from Rayleigh corrected reflectance at band b_bright	c	dl	Boolean, to Breakpoint
BRIGHT_TOA_F(j,f)	Bright flag from TOA reflectance at band b_bright2	c	dl	Boolean, to Breakpoint
MDSI(j,f)	MERIS Differential Snow Index	c	dl	To Breakpoint
HIGH_MDSI_F(j,f)	Flag on high values of MDSI	c	dl	To Breakpoint
CloseToCoast_f(j,f)	Flag identifying land pixels close to coastline (less than 3 pixels)	c	dl	To Breakpoint
Papp(j,f)	Apparent pressure	c	hPa	To Breakpoint
LowP_F(j,f)	Flag on low values of Papp	c	dl	Boolean, to Breakpoint
θ_{s_4x4}	Sun zenith angle for 4x4 sub-window	c	deg	

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
θ_{v_4x4}	Sun viewing angle for 4x4 sub-window	c	deg	
$M_{\_4x4}$	Air mass for 4x4 sub-window	c	dl	
iaer_sa $_{4x4}$	Stratospheric aerosol for 4x4 sub-window	c	dl	0..18
iaer_sa(i,j)	Stratospheric aerosol for each pixel	c	dl	To Breakpoint
i_eff_radius	index of effective radius for strato. aerosol	c	dl	1..3
iaer	aerosol index in multiple scattering LUT	c	dl	13..15
$\cos\theta_{scat}$	cosine of scattering angle for current pixel	c	deg	
$P \times \omega_0$	Phase function times single scattering albedo for current pixel	c	dl	
$\tau_{sa}(b)$	Stratospheric aerosol optical thickness	c	dl	b: b412..b900
$\rho_{sa\_4x4}(b)$	Stratospheric aerosol reflectance on 4x4 sub-window	c	dl	*idem*
T_{sa} (b)	Stratospheric aerosol transmittance on 4x4 window	c	dl	*idem*
$\rho_{ac}(b)$	intermediate aerosol corrected reflectance	c	dl	*idem*
S_{sa} (b)	Stratospheric aerosol spherical albedo	c	dl	*idem*
$\Delta\phi_{\_4x4}$	Azimuth angle for 4x4 sub-window	c	deg	
$\cos\theta_{scat\_4x4}$	cosine of scattering angle on 4x4 pixels sub-window	c	deg	
U_{O3_4x4}	Actual ozone content on 4x4 pixels sub-window	c	DU	
$T_{O3\_4x4}(b)$	Ozone transmission for 4x4 sub-window	c	dl	
L_N^{779}	Normalised radiance at b779	c	sr^{-1}	
λ_{779}	Wavelength of b779 for current pixel	c	nm	
N_{wat}	Number of water pixels on 4x4 sub-window	c	-	
$\rho_{ave}(b)$	Averaged reflectance above water pixels on 4x4 sub-window	c	LU	b: { b885, b885}
X_{ave}	Ratio of averaged reflectance 900 /885	c	dl	
p_{b705}, p_{b761}	Weighting factors to compute TO_2 and TH_2O	c	dl	
a_{h_abv}, b_{h_abv}, c_{h_abv}, d_{h_abv} a_{h_blw}, b_{h_blw}, c_{h_blw}, d_{h_blw}	polynomial coefficients for H_2O corrections	c	dl	
X_2	Ratio used to compute T_{H2O}	c	dl	
$T_{O2}(b)$	Oxygen transmission for current pixel	c	dl	
$T_{H2O}(b)$	H_2O transmission for current pixel	c	dl	
tg(b)	Gaseous transmittance for current pixel	c	dl	
ρ_{thresh}value	Threshold value for reflectance of band b_{thresh} interpolated from LUT	c	dl	
b_{thersh}	Index of band that shall be used for reflectance comparison to the above threshold	c	dl	
$\theta_{s\ \_4x4}$, $\theta_{v\ \_4x4}$, $\Delta\phi_{\ \_4x4}$	Sun zenith, view zenith and azimuth difference for 4x4 pixels window	c	deg	

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
phiw	wind azimuth in topocentric frame	c	deg	RD 8, 2.13, 3.1.1
chiw	wind azimuth in local frame	c	deg	RD 8, 2.13, 3.1.1
UNCERTAIN_F	Uncertain Surface Type flag for current pixel	c	dl	Boolean, to Breakpoint
LOINLD_F	Inland water flag for current pixel	c	dl	Boolean, to Breakpoint
ISLAND_F	Land in water flag for current pixel	c	-	Boolean, to Breakpoint
ρ_{ng}(b,j,f)	TOA reflectance, corrected for stratospheric aerosol contribution and gaseous absorption	c	dl	b in {b412..b885}, to Breakpoint
ρ(b,j,f)	Stratospheric aerosol corrected reflectance for pixel (j,f)	o	dl	to steps 2.2 (§9.2.2), 2.10 (§10.4), to Breakpoint
CLOUD_F(j,f)	Flag for cloud pixel	o	dl	Boolean to steps 2.2 (§9.2.2), 2.3 (§6.4), 2.4 (§7.4), 2.6.5 (§8.2.2), 2.6.8 & 2.6.10 (§8.3.2), 2.8 (§9.4.2), 2.9 (§8.5.4), 2.10 (§10.4), to Breakpoint
BRIGHT_F(j,f)	Bright flag	o	dl	Boolean to step 2.6.5 (§8.2.2), to Breakpoint
SNOW_ICE_F(j,f)	Snow or Ice flag	o	dl	Boolean, to steps 2.3 (§ 6.4), 2.6.5 (§8.2.2), to Breakpoint
ρ^{*}_{ng}(b,j,f)	TOA reflectance, corrected for stratospheric aerosol contribution, gaseous absorption and smile effect	o	dl	b in {b412..b885} to 2.6.5 (§8.2.2), 2.6.23 (§9.3.2), to Breakpoint
LANDCONS_F(j,f)	Land/water consolidated flag	o	dl	Boolean, to 2.6.5 (§8.2.2), 2.6.8 & 2.6.10 (§8.3.2), 2.6.9 (§8.4.2), 2.10 (§10.4), to Breakpoint
ORINP0_F(j,f)	Out of range input for atmosphere corrections	o	dl	Boolean, to steps 2.6.8 & 2.6.10 (§8.3.2 for water pixels), to 2.10 (§10.4 for others) , to Breakpoint
OROUT0_F(j,f)	Out of range output for atmosphere corrections	o	dl	Boolean, to 2.10 (§10.4) , to Breakpoint
W_s(j,f)	wind speed modulus for pixel (j,f)	o	$m.s^{-1}$	to step 2.3.2 (§6), 2.6.5 (§8.2), 2.6.8 (§8.3), to Breakpoint
ROG(j,f)	Sun glint reflectance for pixel (j,f)	o	dl	to step 2.6.5 (§8.2), to Breakpoint
ρ_{R1}(b,j,f)	Coarse Rayleigh reflectance	o	dl	b in {b412 … b885}, to 2.6.23 (§ 9.3), to Breakpoint
ρ_{R}(b,j,f)	Rayleigh reflectance at nominal wavelengths	o	dl	b in {b412..b900}, to 2.6.8 (§ 8.3) & 2.6.9 (§ 8.4), to Breakpoint

Variable	Descriptive Name	T	U	Range - References
$\rho_{R0}(b,j,f)$	Rayleigh reflectance at nominal wavelengths, corrected for pressure, water pixels only	o	dl	b in {b412..b900}, to 2.6.8 (§ 8.3) & 2.6.9 (§ 8.4)
$\tau_{R0}(b,j,f)$	Rayleigh optical thickness at nominal wavelengths, corrected for pressure	o	dl	b in {b412..b885}, to 2.6.5 (§ 8.2), 2.6.9 (§ 8.4), 2.6.15 (§ 9.3)

NOTES:

1. all calculated and output Boolean parameters shall be initialised to FALSE (0).
2. subscript FR or RR for type s variable λpix is omitted in equation section below. Proper variable shall be selected according to processed product resolution.

Copyright © 2011 ACRI S.T.

5.5 – Equations (step 2.1c)

Product processing :

For each pixel (in column j, frame f) such that (INVALID_F(j, f)== FALSE)**:**

5.5.1- Cloud screening tests (step 2.1.7)

NOTE: The procedure **ref_rayleigh** computing the Rayleigh reflectance for a given pressure, geometry, is specified in section 5.5.6 below.

```
deleted                                                                  (2.1.7-1)
For each band b in {b412..b900}
    $\tau_{R0}(b,j,f) = \tau_R(b).P_{ECMWF}(j,f)/P_{std}$                 (2.1.7-2)
Endfor
compute air mass
$M(j,f) = 1/\cos(\theta_s(j,f)) + 1/\cos(\theta_v(j,f))$                 (2.1.7-20)

call ref_rayleigh ($\theta_s(j,f), \theta_v(j,f), \Delta\phi(j,f), M(j,f), \tau_{R0}$, {b412…b885}, $\rho_{R1}$)   (2.1.7-3)

For each band b in {b_bright, b_slope1_n, b_slope1_d, b_slope2_n, b_slope2_d}
    $\rho_{rc}(b) = \rho_{TOA}(b,j,f) - \rho_{R1}(b,j,f)$                 (2.1.7-4)
Endfor

If ($\rho_{rc}$(b_slope1_d) > 0) then
    $SLOPE\_1\_F = \begin{cases} SATURATED\_F(b\_slope1\_n,j,f) \quad \textbf{OR} \\ \left(\left(\frac{\rho_{rc}(b\_slope1\_n)}{\rho_{rc}(b\_slope1\_d)} \geq Slope\_1\_low\right) \textbf{ AND } \left(\frac{\rho_{rc}(b\_slope1\_n)}{\rho_{rc}(b\_slope1\_d)} \leq Slope\_1\_high\right)\right) \end{cases}$   (2.1.7-5)
else
    SLOPE_1_F = FALSE                                                    (2.1.7-6)
Endif
If ($\rho_{rc}$(b_slope2_d) > 0) then
    $SLOPE\_2\_F = \begin{cases} SATURATED\_F(b\_slope2\_n,j,f) \quad \textbf{OR} \\ \left(\left(\frac{\rho_{rc}(b\_slope2\_n)}{\rho_{rc}(b\_slope2\_d)} \geq Slope\_2\_low\right) \textbf{ AND } \left(\frac{\rho_{rc}(b\_slope2\_n)}{\rho_{rc}(b\_slope2\_d)} \leq Slope\_2\_high\right)\right) \end{cases}$   (2.1.7-7)
else
    SLOPE_2_F = FALSE                                                    (2.1.7-8)
Endif
Threshold on value of Rayleigh corrected reflectance
Interpolate threshold for current geometry and pixel surface type        (2.1.7-9)
If (LAND_F == TRUE) then
```

```
    rho_thresh = Rho_rc_LUT interpol: ($\theta_s$, $\theta_v$, $\Delta\phi$), select k for LAND
Else
    rho_thresh = Rho_rc_LUT interpol: ($\theta_s$, $\theta_v$, $\Delta\phi$), select k for OCEAN
End if
Derive bright flag by reflectance comparison to threshold                                  (2.1.7-10)
$\mathrm{BRIGHT\_RC\_F}(j,f) = \begin{matrix}(\rho_{rc}(b\_bright) > rho\_thresh) \quad \mathbf{OR} \\ (\mathrm{SATURATED\_F}(b\_bright, j, f))\end{matrix}$
Compute cosine of scattering angle
    $\cos\theta_{scat}(j,f) = -\sqrt{1-\cos^2\theta_s}\sqrt{1-\cos^2\theta_v}\cos\Delta\phi - \cos\theta_s\cos\theta_v$        (2.1.7-22)
Combine with Slope test when over (a priori) Land and with another
BRIGHT test when over (a priori) ocean                                                     (2.1.7-11)
If (LAND_F == TRUE) then
    BRIGHT_F(j,f) = BRIGHT_RC_F(j,f) AND SLOPE_1_F(j,f) AND SLOPE_2_F(j,f)
Else
    BRIGHT_TOA_F(j,f) = ($\rho_{TOA}$(b_bright2,j,f) > $rho_{TOA}$_thresh[0]+
                              $rho_{TOA}$_thresh[1]*$\cos^2(\theta_{scatt}(j,f))$)
    BRIGHT_F(j,f) = BRIGHT_RC_F(j,f) OR BRIGHT_TOA_F(j,f)
Endif

Compute MERIS Differential Snow Index and corresponding flag                               (2.1.7-12)
$\mathrm{MDSI}(j,f) = \frac{\rho_{TOA}(b_1^{MDSI},j,f) - \rho_{TOA}(b_2^{MDSI},j,f)}{\rho_{TOA}(b_1^{MDSI},j,f) + \rho_{TOA}(b_2^{MDSI},j,f)}$
HIGH_MDSI_F(j,f)= (MDSI(j,f) $\geq$ MDSI_Thresh)                                            (2.1.7-14)
Compute Apparent Pressure tests according to surface                                       (2.1.7-13)
NOTE: The procedures Pscatt and P1 are specified in sections 5.5.9 and 5.5.10 below.

Compute the Low Pressure flag using surface dependent algorithms.
$\lambda_{b761} = \lambda_{pix}(b_{761}, \mathrm{Detector}(j,f))$                                   (2.1.7-15)
If (LAND_F == TRUE) then
    Call P1($\lambda_{b761}$, $\theta_s$, $\theta_v$, $\rho_{TOA}$({b753,b761,b779},j,f),
          SATURATED_F({b753,b761,b779},j,f), Papp(j,f))                                    (2.1.7-16)
    Check distance to coastline
    $\mathrm{CloseToCoast\_f}(j,f) = \sum_{j'=\max(j-1,1)}^{\min(j+1,nj)} \sum_{f'=\max(f-1,1)}^{\min(f+1,nf)} (\mathrm{Coastline\_f}(j',f'))$        (2.1.7-20)
    Select pressure threshold accordingly
    If CloseToCoast_f(j,f) >0 then iThreshP1=2 else iThreshP1=1                            (2.1.7-21)
    LowP_F(j,f) = (Papp(j,f) $\neq$ BAD_VALUE) AND
                  (Papp(j,f)< $P_{ECMWF}$(j, f)-$P_1$_Thresh(iThresh)) AND
                  ($\frac{\rho_{TOA}(b753,j,f)}{\rho_{TOA}(b779,j,f)}$ > $R_{10\_12}$_thresh)                   (2.1.7-17)
Else
    Call Pscatt($\lambda_{b761}$, $\theta_s$, $\theta_v$, $\rho_{R1}$ (b761,j,f), SATURATED_F(b779,j,f),
          $\rho_{TOA}(\{b753,b761,b779\},j,f)$,Papp(j,f))                                   (2.1.7-18)
```

```
        LowP_F(j,f) = (Papp(j,f) $\neq$ BAD_VALUE)
                      AND (Papp(j,f)<$P_{scatt}$_Thresh)
                      AND ($\rho_{TOA}$(b753,j,f) > $rho_{753}$_thresh)          (2.1.7-19)
    End if

Estimate Glint reflectance at every pixel
    computation of wind azimuth angle from wind vector components in topocentric frame
    If (W_v(j,f) > 0) then
        phiw = arc tan (W_u(j,f) /W_v(j,f))                                (2.6.26-14)
    else if (W_v(j,f) < 0) then
        phiw = 180 + arc tan (W_u(j,f)/W_v(j,f))                           (2.6.26-15)
    else        /* W_v = 0 */
        phiw = 180 - 90. sign(W_u(j,f))                                    (2.6.26-16)
    endif

    computation of wind azimuth chiw in local frame
    chiw = arc cos[cos($\phi_s$(j,f) - phiw)]                                  (2.6.26-17)
    computation of wind speed modulus
    $W_s(j,f) = (W\_u(j,f)^2+W\_v(j,f)^2)^{1/2}$                                 (2.6.26-18)
    interpolation of glint reflectance
    ROG(j,f) = ROG_LUT interpol: ($\theta_s$(j,f), $\theta_v$(j,f), $\Delta\phi$(j,f), $W_s$(j,f), chiw)     (2.6.26-19)
```

exception processing: out of index range θ_s(j,f), θ_v(j,f), $\Delta\phi$(j,f), W_s(j,f), chiw in (2.6.26-19) above:

continue processing at next equation

end of exception processing

```
    deleted                              (2.6.26-20 & 21)
```

5.5.2- **Set Cloud and Snow/ice flags** (step 2.1.8)

```
    Set Snow/ice flag
    SNOW_ICE_F(j,f) = BRIGHT_F(j, f) AND HIGH_MDSI_ F(j, f)                (2.1.8-0)
    Set Cloud flag accounting for a priori surface
    If (LAND_F == TRUE) then
        CLOUD_F (j, f) =   (BRIGHT_F(j, f) AND LowP_F(j, f)) AND
                           (NOT (HIGH_MDSI_ F(j, f))                       (2.1.8-1)
    else
        Over Water check Sun Glint condition first
        ClassGlint_F = ( ROG(j,f) >= thres_medg . $\rho_{TOA}$(b865, j, f) )    (2.1.8-2)
        CLOUD_F (j, f) =   (NOT SNOW_ICE_F(j,f)) AND
```

Copyright © 2011 ACRI S.T.

```
            [ (BRIGHT_F(j, f) AND NOT (ClassGlint_F )) OR
             (BRIGHT_RC _F(j, f) AND (ClassGlint_F )) OR LowP_F(j, f) ]          (2.1.8-3)
    Endif

Endfor      end of processing for current pixel
```

5.5.3- Stratospheric Aerosol correction (step 2.1.9)

NOTE: The procedure **ref_aerosol** computing the aerosols reflectance for a given aerosol model, optical thickness, geometry, is specified in section 5.5.7 below.

```
If (STRAT_CORR) then
    For each 4x4 pixel window containing at least one valid pixel

        $\theta_{s\_4x4} = \theta_s$ at North-East corner[‡] pixel of window             (2.1.9-1)
        $\theta_{v\_4x4} = \theta_v$ at North-East corner pixel of window               (2.1.9-2)
        $\Delta\phi_{\_4x4} = \Delta\phi$ at North-East corner pixel of window              (2.1.9-3)
        $M_{4x4} = M(j,f)$ at North-East corner pixel of window                (2.1.9-4)
        $iaer\_sa_{\_4x4}$ = Stratospheric_LUT nearest: $(lat_0, lon_0)$                (2.1.9-5)
        where $lat_0$ & $lon_0$ are latitude and longitude at North-East corner pixel of window
        $\cos\theta_{scat\_4x4} = \cos\theta_{scat}(j,f)$ at North-East corner pixel of window   (2.1.9-23)

        If ($iaer\_sa_{\_4x4}$ != 0) then
    Compute stratospheric aerosol transmittance on each 4x4 pixels sub-window
            i_eff_radius = Strato_rad select: ($iaer\_sa_{\_4x4}$)                (2.1.9-6)
            iaer = Strato_multi select: ($iaer\_sa_{\_4x4}$)                      (2.1.9-7)
            deleted   (2.1.9-8)

            For each band b in { b412..b900 }
    Interpolate aerosol phase function times single scattering albedo
                $P\times\omega_0$ = Strato_aerpha_LUT interpol : ($\cos\Theta_{scat}$) select : (i_eff_radius, band)
                                                                                (2.1.9-9)
                $\tau_{sa}(b)$ = [Strato_tau select: ($iaer\_sa_{\_4x4}$)].[Strato_spectr select: (i_eff_radius, b)]
                                                                                (2.1.9-10)
                $\rho_{sa\_4x4}(b)$ = ref_aerosol($\theta_{s\_4x4}, \theta_{v\_4x4}, \Delta\phi_{\_4x4}, M_{\_4x4}$, iaer, $\tau_{sa}(b)$,
                $P\times\omega_0$)                                                     (2.1.9-12)
                $T_{sa}(b) = \text{TA\_strato\_LUT } \textbf{interpol}: (\theta_{s\_4x4})\ \textbf{select}: (iaer\_sa_{\_4x4}, b)$
                         $\times \text{TA\_strato\_LUT } \textbf{interpol}: (\theta_{v\_4x4})\ \textbf{select}: (iaer\_sa_{\_4x4}, b)$   (2.1.9-13)
    exception processing: $T_{sa}(b) = 0$:
        exit loop
        process all valid pixels in 4x4 window as if there were no stratospheric aerosol
        (equations 2.1.9-18, 2.1.9-19)
        process next window
```

[‡] As column numbering increases from East to West and line numbering increases with satellite motion, i.e. from North to South, this corresponds to the smaller line and column indices within the window.

Copyright © 2011 ACRI S.T.

```
end of exception processing
        $S_{sa}(b)$ = Strato_sphalb select: (iaer_sa_4x4, b)                        (2.1.9-14)
      Endfor

      For each pixel (j, f) in 4x4 pixel window such that (INVALID_F(j, f) == FALSE)
        Apply stratospheric aerosol correction to reflectance in ALL bands
        For each band b in { b412..b900 }
          $\rho_{ac}(b) = \frac{1}{T_{sa}(b)} . \left(\rho_{TOA}(b,j,f) - \rho_{sa\_4x4}(b)\right)$      (2.1.9-15)
exception processing: $\rho_{ac}(b) < 0$:
    exit loop
    process all valid pixels in 4x4 window as if there were no stratospheric aerosol
    (equations 2.1.9-18, 2.1.9-19)
    process next window
end of exception processing
          $\rho(b,j,f) = \frac{\rho_{ac}(b)}{1 + S_{sa}(b) . \rho_{ac}(b,j,f)}$                   (2.1.9-16)
        Endfor
        iaer_sa(i,j) = iaer_sa_4x4                                                (2.1.9-17)

      Endfor    end of processing for current valid pixel

    Else
  iaer_sa_4x4 == 0 : conventional value for absent aerosol
      For each pixel (j, f) in 4x4 pixel window such that (INVALID_F(j, f) == FALSE)
        For each band b in { b412..b900 }
          $\rho(b, j, f) = \rho_{TOA}(b, j, f)$                                   (2.1.9-18)
        Endfor
  deleted            (2.1.9-19)
        iaer_sa(i,j) = BAD_VALUE                                                 (2.1.9-21)
      Endfor
    Endif
  Endfor    end of processing for current 4x4 window

Stratospheric aerosol correction is disabled
else
  For each pixel (j, f) in 4x4 pixel window such that (INVALID_F(j, f) == FALSE)
      For each band b in { b412..b900 }
        $\rho(b, j, f) = \rho_{TOA}(b, j, f)$                                     (2.1.9-20)
      Endfor
        iaer_sa(i,j) = BAD_VALUE                                                 (2.1.9-22)
  Endfor    end of processing for current 4x4 window
Endif
```

Copyright © 2011 ACRI S.T.

5.5.4- Gaseous absorption correction (step 2.6.12)

```
For each 4 (across-track) x4 (along-track) pixels[§] window containing at least 1 pixel such
that (INVALID_F(j, f)== FALSE) AND (CLOUD_F(j, f)==FALSE)                  (2.6.12-1)

let j0, f0 be the column and frame co-ordinates of the North-East corner[**] pixel of window
   $\theta_{s\_4x4} = \theta_s(j0, f0)$                                    (2.6.12.1-3)
   $\theta_{v\_4x4} = \theta_v(j0, f0)$                                    (2.6.12.1-4)
   $\Delta\phi_{\_4x4} = \Delta\phi(j0, f0)$                               (2.6.12.1-5)
   $U_{O3\_4x4} = U_{O3}(j0, f0)$                                          (2.6.12.1-6)
   $M_{\_4x4} = M(j0, f0)$                                                 (2.6.12.1-1)

   1. Averaging of TOA reflectances on water pixels
   $N_{wat} = \sum_{\substack{j,f\ \mathrm{in} \\ \mathrm{window}}} (1-INVALID\_F(j,f)).(1-LAND\_F(j,f)).(1-CLOUD\_F(j,f))$   (2.6.12.1-7)
   If ($N_{wat} > 0$) then
      For each band b in {b885, b900}
         $\rho_{ave}(b) = \frac{1}{N_{wat}} \sum_{\substack{j,f\ \mathrm{in} \\ \mathrm{window}}} (1-INVALID\_F(j,f)).(1-LAND\_F(j,f)).(1-CLOUD\_F(j,f)).\rho(b,j,f)$
                                                                           (2.6.12.1-8)
      End for
      $X_{ave} = \rho_{ave}(b900) / \rho_{ave}(b885)$                      (2.6.12.1-9)
   End if

   2. Estimation of O3 transmittance (step 2.6.12.1)
   For each band b in {b412..b753, b779..b885}
      $T_{O3\_4x4}(b) = e^{-(U_{O3\_4x4}/1000.0).M_{\_4x4}.\tau_{O3\_norm}(b)}$   (2.6.12.1-2)
   Endfor  end loop over bands

   3. Estimation of O2 transmittance (step 2.6.12.2)
   For each pixel (j, f) within 4x4 window such that
   (INVALID_F(j, f)== FALSE) AND (CLOUD_F(j, f)==FALSE)

   deleted    (2.1.3-1-b761), (2.1.3-2-b761), (2.1.3-3-b761), (2.1.3-4-b761)

find $i\lambda_{b705}$ such that:
   $\lambda_{ref}(i\lambda_{b705}) \le \lambda pix(b705, Detector(j,k)) < \lambda_{ref}(i\lambda_{b705}+1)$   (2.1.3-1-b705)
exception processing:
   if $\lambda pix(b705, Detector(j,k)) < \lambda_{ref}(0)$   then $i\lambda_{b705} = 0$   (2.1.3-2-b705)
```

[§] If processing by windows with odd width or height, sub-window size may be reduced to 4x1, 1x4, 1x1 at borders without impact.

[**] As column numbering increases from East to West and line numbering increases with satellite motion, i.e. from North to South, this corresponds to the smaller line and column indices within the window.

Copyright © 2011 ACRI S.T.

```
    if $\lambda pix(b705, Detector(j,k)) > \lambda_{ref}(N_\lambda - 1)$ then $i\lambda_{b705} = N_{\Delta\lambda}-2$        (2.1.3-3-b705)
end of exception processing
    $p_{b709} = \frac{\lambda pix(b705, Detector(j,f)) - \lambda_{ref}(i\lambda_{b705})}{\lambda_{ref}(i\lambda_{b705}+1) - \lambda_{ref}(i\lambda_{b705})}$        (2.1.3-4-b705)

        deleted        (2.6.12.2-1)
        deleted        (2.6.12.2-3-blw), (2.6.12.2-3-abv), (2.6.12.2-4-blw), (2.6.12.2-4-abv),
        (2.6.12.2-5-blw), (2.6.12.2-5-abv), (2.6.12.2-6-blw), (2.6.12.2-6-abv)

        If (LAND_F (j, f) then
            $X_2 = \rho(b900,j,f) / \rho(b885,j,f)$        (2.6.12.3-1)
exception processing : $\rho(b885,j,f) <= 0$ in (2.6.12.3-1) :
    $X_2 = 1$
    ORINP0_F(j, f) = TRUE
    continue processing at next equation
end of exception processing
        Else
            $X_2 = X_{ave}$
        End if

        For b in {b412..b753,b779..b885}
            If (b == b779) AND NOT SATURATED_F(b779,j,f) then
                deleted        (2.6.12.2-2-blw), (2.6.12.2-2-abv), (2.6.12.2-2-fnl)
                $L_N^{775} = \frac{\rho(b779,j,f) \cdot \cos(\theta_S(j,f))}{\pi}$        (2.6.12.2-2)
                $\lambda_{779} = \lambda pix(b779, Detector(j,k))$        (2.6.12.2-3)
                $T_{O_2}(b) = T_{O2}$_LUT interpol $\lambda_{779}$, $L_N^{779}$, $\theta_S(j,f)$, $\theta_V(j,f)$, $180-\Delta\phi(j,f)$        (2.6.12.2-4)
            Else
                $T_{O_2}(b) = 1$        (2.6.12.2-7)
            Endif
```

3. Estimation of H_2O transmittance (step 2.6.12.3)

Compute set of polynomial coefficients for H2O transmission retrieval at 709nm

```
If (b==b705) then
    $a_{h\_blw} = H_2O_{b705}$Corr_Poly_LUT select: ($\Delta\lambda = \Delta\lambda_{b705}$, k=1)        (2.6.12.3-2- $a_{h\_blw}$)
    $a_{h\_abv} = H_2O_{b705}$Corr_Poly_LUT select: ($\Delta\lambda = \Delta\lambda_{b705}+1$, k=1)        (2.6.12.3-2- $a_{h\_abv}$)

    $b_{h\_blw} = H_2O_{b705}$Corr_Poly_LUT select: ($\Delta\lambda = \Delta\lambda_{b705}$, k=2)        (2.6.12.3-2- $b_{h\_blw}$)
    $b_{h\_abv} = H_2O_{b705}$Corr_Poly_LUT select: ($\Delta\lambda = \Delta\lambda_{b705}+1$, k=2)        (2.6.12.3-2- $b_{h\_abv}$)

    $c_{h\_blw} = H_2O_{b705}$Corr_Poly_LUT select: ($\Delta\lambda = \Delta\lambda_{b705}$, k=3)        (2.6.12.3-2- $c_{h\_blw}$)
    $c_{h\_abv} = H_2O_{b705}$Corr_Poly_LUT select: ($\Delta\lambda = \Delta\lambda_{b705}+1$, k=3)        (2.6.12.3-2- $c_{h\_abv}$)
```

Copyright © 2011 ACRI S.T.

d_{h_blw} = H_2O_{b705}Corr_Poly_LUT **select**: ($\Delta\lambda = \Delta\lambda_{b705}$, k=4) *(2.6.12.3-2- d_{h_blw})*

d_{h_abv} = H_2O_{b705}Corr_Poly_LUT **select**: ($\Delta\lambda = \Delta\lambda_{b705}+1$, k=4) *(2.6.12.3-2- d_{h_abv})*

$$T_{H_2O\_blw} = a_{h\_blw} + b_{h\_blw}.X_2 + c_{h\_blw}.X_2^{\ 2} + d_{h\_blw}.X_2^{\ 3} \qquad (2.6.12.3\text{-}2\text{-}blw)$$

$$T_{H_2O\_abv} = a_{h\_abv} + b_{h\_abv}.X_2 + c_{h\_abv}.X_2^{\ 2} + d_{h\_abv}.X_2^{\ 3} \qquad (2.6.12.3\text{-}2\text{-}abv)$$

$$T_{H_2O}(b) = (1 - p_{b705})T_{H_2O\_blw} + (p_{b705})T_{H_2O\_abv} \qquad (2.6.12.3\text{-}2\text{-}fnl)$$

Else

$$T_{H_2O}(b) = a_h(b) + b_h(b).X_2 + c_h(b).X_2^{\ 2} + d_h(b).X_2^{\ 3} \qquad (2.6.12.3\text{-}2)$$

Endif

5. Estimation of reflectance corrected for gaseous absorption (step 2.6.12.4)

deleted *(2.6.12.4-1)*

$$tg(b) = T_{O3\_4x4}(b).T_{O_2}(b).T_{H_2O}(b) \qquad (2.6.12.4\text{-}2)$$

exception processing: tg(b) <=0 OR tg(b) > 1. in (2.6.12.4-2) :
tg(b) = 1
OROUT0_F(j, f) = TRUE
continue processing at next equation
end of exception processing

$$\rho_{ng}(b, j, f) = \frac{\rho(b, j, f)}{tg(b)} \qquad (2.6.12.4\text{-}3)$$

Endfor *End of loop over bands*

Endfor *End of loop over pixels within 4x4 window*

Copyright © 2011 ACRI S.T.

5.5.5 - Land identification (step 2.6.26) and smile effect correction (step 2.1.6)

```
$N_{proc} = \sum_{j,\, f \text{ in window}} (1-INVALID\_F(j,f)).(1-CLOUD\_F(j,f))$        (2.6.26-8)
    If ($N_{proc} > 0$) then
let j0, f0 be the column and frame co-ordinates of the North-East corner[††] pixel of window
        $\theta_{s\_4x4} = \theta_s\ (j0, f0)$                                  (2.6.26-9)
        $\theta_{v\_4x4} = \theta_v\ (j0, f0)$                                  (2.6.26-10)
        $\Delta\phi_{\_4x4} = \Delta\phi\ (j0, f0)$                              (2.6.26-11)
        deleted                       (2.6.26-12)
        deleted                       (2.6.26-13)

    Endif
  For each pixel (j, f) within 4x4 window such that
    (INVALID_F(j, f)== FALSE) AND (CLOUD_F(j, f)==FALSE)

    UNCERTAIN_F=Surface_Confidence_Map nearest: (lat(j,f),lon(j,f))       (2.6.26-2)

Note: steps 2.6.26-14 to 2.6.26-19 have been moved at the end of section 5.5.1.
Steps 2.6.26-20 & 2.6.26-21 have been deleted.

Check all a priori land pixels for in-land waters, below a certain altitude:
    If (LAND_F(j,f) == TRUE) then                                         (2.6.26-3)
        If ($z(j,f) < Z_{max}\_INLAND$) then
  Threshold for Inland Waters processing
          $\rho_{thresh}value = \rho_{thresh}\_LUT$ select (b=1) interpol: ($\theta_{s\_4x4}, \theta_{v\_4x4}, \Delta\phi_{\_4x4}$)  (2.6.26-1a)
  exception processing: out of index range $\theta_{s\_4x4}, \theta_{v\_4x4}, \Delta\phi_{\_4x4}$ in (2.6.26-1a) above):
    continue processing at next equation
  end of exception processing

  Inland waters processing
          $b_{test} = b_{thresh}[1]$
          $\alpha_{test} = \alpha_{thresh}[1]$
          $LOINLD\_F = (\rho_{ng}(b_{test},j,f) \le \alpha_{test} \cdot \rho_{thresh}value)$ AND
                       $(\beta_L \cdot \rho_{ng}(b865,j,f) < \rho_{ng}(b665,j,f))$               (2.6.26-4)
        Else altitude over the threshold
  Re-classification not reliable, make sure LANDCONS = LAND
          LOINLD_F = ISLAND_F =FALSE                                      (2.6.26-6a)
        Endif test on altitude over land
    Else
Check only those a priori water pixels for which surface type is uncertain:
        If (UNCERTAIN_F== TRUE) then
Test land in water
```

[††] As column numbering increases from East to West and line numbering increases with satellite motion, i.e. from North to South, this corresponds to the smaller line and column indices within the window.

Copyright © 2011 ACRI S.T.

```
Evaluate Glint condition on current pixel:
            If (ROG(j,f) >= thres_medg . $\rho_{ng}$(b865, j, f) then                    (2.6.26-22)
               $b_{test} = b_{thresh}[1]$
               $\alpha_{test} = \alpha_{thresh}[1]$
Threshold for Island in Glint processing
               $\rho_{thresh}$value =
                  $\rho_{thresh}$_LUT select (b=1) interpol: ($\theta_{s\_4x4}$ ,$\theta_{v\_4x4}$ ,$\Delta\phi_{\_4x4}$)      (2.6.26-1b)
exception processing: out of index range $\theta_{s\_4x4}$ ,$\theta_{v\_4x4}$ ,$\Delta\phi_{\_4x4}$ in (2.6.26-1b) above):
   continue processing at next equation
end of exception processing
            Else
               $b_{test} = b_{thresh}[2]$
               $\alpha_{test} = \alpha_{thresh}[2]$
Threshold for Island out of Glint processing
               $\rho_{thresh}$value =
                  $\rho_{thresh}$_LUT select (b=2) interpol: ($\theta_{s\_4x4}$,$\theta_{v\_4x4}$,$\Delta\phi_{\_4x4}$)        (2.6.26-1c)
exception processing: out of index range $\theta_{s\_4x4}$ ,$\theta_{v\_4x4}$ ,$\Delta\phi_{\_4x4}$ in (2.6.26-1b) above):
   continue processing at next equation
end of exception processing
            Endif
            ISLAND_F = ($\rho_{ng}(b_{test}$,j,f) > $\alpha_{test}$ . $\rho_{thresh}$value) AND
                         ($\beta_W$ . $\rho_{ng}$(b865,j,f) > $\rho_{ng}$(b665,j,f) )                      (2.6.26-5)
         Else
   Surface type is asserted, make sure LANDCONS = LAND
            LOINLD_F = ISLAND_F =FALSE                                              (2.6.26-6b)
         Endif
      Endif

Land /water consolidated flag for every valid pixel
      LANDCONS_F(j,f) = (LAND_F(j,f) AND NOT LOINLD_F) OR ISLAND_F
                                                                              (2.6.26-7)

Apply reflectance correction for smile effect according to surface type (step 2.1.6)
      If (LANDCONS_F(j,f) == TRUE) then
         call ref_smile_corr($\rho_{ng}$, LandRefCorr_sw, LandRefCorr_b, Detector(j,f),
                         $\rho^*_{ng}$)                                                 (2.1.6-1)
      Else
         For each band b in {b412..b900}
            Compute Rayleigh reflectance at all wavelengths for current pixel. Interpolation
            is done in cosine for Viewing and Sun zenith angles.
            Let $\mu_v = \cos(\theta_v)$
            Let $\mu s = \cos(\theta s)$

            $\rho_R$(b,j,f) =
               $\rho_R$tab_LUT interpol: ($\mu v$(j,f), $\mu s$(j,f), $\Delta\phi$(j,f), $W_s$(j,f)) select: (b) (2.1.6-6)
            Compute pressure corrected Rayleigh optical thicknesses
```

```
            Correct Rayleigh reflectance for pressure variation
            $\rho_{R0}(b,j,f) = \rho_R(b,j,f).\frac{1-\exp\left(-\frac{\tau_{R0}(b)}{\mu_V}\right)}{1-\exp\left(-\frac{\tau_R(b)}{\mu_V}\right)}$        (2.1.6-7)
        Endfor
        For each band b in {b412..b900}
            Get spectral derivative of log of Rayleigh reflectance at all bands
            Let b1 = max (b412, b-1), b2 = min(b900, b+1)
            $\mathrm{dlog}\rho_{R0}(b) = (\log(\rho_{R0}(b2)) - \log(\rho_{R0}(b1)))/(\lambda_{theo}(b2)-\lambda_{theo}(b1))$        (2.1.6-8)
            Get Rayleigh reflectance smile correction (from log derivative)
            $\rho'_{R0}(b) = \rho_{R0}(b).(1+\mathrm{dlog}\rho_{R0}(b).(\ \lambda_{pix}(b,det\_index(j,f)) - \lambda_{theo}(b)\ )\ )$        (2.1.6-9)
            Get Rayleigh corrected reflectance
            $\rho_{R0C}(b) = \rho_{ng}(b,j,f) - \rho'_{R0}(b)$        (2.1.6-10)
        Endfor

        Correct Rayleigh Corrected Reflectance for smile
        call ref_smile_corr($\rho_{R0C}$, WaterRefCorr_sw, WaterRefCorr_b, Detector(j,f),
                            $\rho^*_{R0C}$)        (2.1.6-2)
        Back to TOA reflectance with Rayleigh reflectance at nominal wavelength
        For each band b in {b412..b900}
            $\rho^*_{ng}(b,j,f) = \rho^*_{R0C}(b) + \rho_{R0}[b]$        (2.1.6-11)
        Endfor
    Endif
  Endfor End of loop over clear sky pixels within 4x4 window

Endfor End of loop over 4x4 windows
```

Copyright © 2011 ACRI S.T.

5.5.6 - Procedure *ref_rayleigh* to estimate Rayleigh reflectance (step. 2.1.17)

The List of Variables below identifies the dummy input, output and locally computed variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, read from an auxiliary parameters file, are listed in table of variables in section 3.4 above.

Variable	Descriptive Name	T	U	Range
θ_s	Sun zenith angle	i	deg	
θ_v	Viewing zenith angle	i	deg	
$\Delta\phi$	Azimuth angle between pixel-sensor and pixel-sun plane	i	deg	
M	air mass	i	dl	
τ_{R0} (b)	Rayleigh optical thickness at actual surface pressure	i	dl	b in {bands}
bands	list of bands for which the reflectance is to be computed	i	dl	
$P_R^{(s)}$	Fourier components of Rayleigh scattering phase function	c	dl	s: 0..2
$a^{(s)}$, $b^{(s)}$, $c^{(s)}$, $d^{(s)}$	Polynomial coefficients for each of the Fourier components of the correction factor for Rayleigh multiple scattering	c	dl	a, b, c, d: polynomial order 0 to 3 s: Fourier series order 0..2
$\rho_{R,P}^{(s)}(b)$	Fourier components of Rayleigh reflectance in single scattering approximation for 4x4 sub-window	c	dl	b in {bands} s: 0..2
$f_R^{(s)}(b)$	Fourier components of correction factor for Rayleigh multiple scattering for 4x4 sub-window	c	dl	*idem*
$\rho_R^{(s)}(b)$	Fourier components of Rayleigh reflectance for 4x4 sub-window	c	dl	*idem*
$\rho_{R1}(b)$	Rayleigh reflectance	o	dl	b in {bands}

*The **ref_rayleigh** procedure is called by steps 2.1.7, 2.6.15.1.*
*The **ref_rayleigh** procedure is defined as follows:*
compute Fourier components of Rayleigh phase function

$$P_R^{(0)} = \frac{3}{4}.A.\left[1 + \cos^2(\theta_s)\cos^2(\theta_v) + 0.5.\sin^2(\theta_s).\sin^2(\theta_v)\right] + B \qquad (2.1.17\text{-}1)$$

$$P_R^{(1)} = -\frac{3}{4}.A.\cos(\theta_s).\cos(\theta_v).\sin(\theta_s).\sin(\theta_v) \qquad (2.1.17\text{-}2)$$

$$P_R^{(2)} = \frac{3}{16}.A.\sin^2(\theta_s).\sin^2(\theta_v) \qquad (2.1.17\text{-}3)$$

```
For each Fourier series component order s in 0..2
    compute Fourier components of correcting factor for multiple scattering)
    $a^{(s)}$ = Rayscatt_coef_LUT interpol : ($\theta_s$,$\theta_v$) select : (s, k=1)    (2.1.17-4)
    $b^{(s)}$ = Rayscatt_coef_LUT interpol : ($\theta_s$,$\theta_v$) select : (s, k=2)    (2.1.17-5)
    $c^{(s)}$ = Rayscatt_coef_LUT interpol : ($\theta_s$,$\theta_v$) select : (s, k=3)    (2.1.17-6)
```

Copyright © 2011 ACRI S.T.

```
        $d^{(s)}$ = Rayscatt_coef_LUT interpol : $(\theta_s,\theta_v)$ select : (s, k=4)        (2.1.17-7)
Endfor    End of loop on Fourier series order

For each band b in {bands}

    For each Fourier series component order s in 0..2
compute components of Rayleigh reflectance for primary scattering
        $\rho_{R,P}^{(s)}(b) = \frac{P_R^{(s)}}{4(\cos\theta_s + \cos\theta_v)}\left(1 - e^{-\tau_{R0}(b).M}\right)$        (2.1.17-8)
        $f_R^{(s)}(b) = a^{(s)} + b^{(s)}\left(\tau_{R0}(b)\right) + c^{(s)}\left(\tau_{R0}(b)\right)^2 + d^{(s)}\left(\tau_{R0}(b)\right)^3$        (2.1.17-9)
compute Fourier components of Rayleigh reflectance
        $\rho_R^{(s)}(b) = \rho_{R,P}^{(s)}(b) * f_R^{(s)}(b)$        (2.1.17-10)
    Endfor End of loop over index s

      compute Rayleigh reflectance as a Fourier sum
      $\rho_{R1}(b) = \rho_R^{(0)}(b) + 2\,\rho_R^{(1)}(b)\cos(\Delta\phi) + 2\rho_R^{(2)}(b)\cos(2\Delta\phi)$        (2.1.17-11)
Endfor End of loop over band b

End of ref_rayleigh procedure
```

5.5.7 - Procedure *ref_aerosol* to estimate Aerosol reflectance (step 2.1.18)

The List of Variables below identifies the dummy input, output and locally computed variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, read from an auxiliary parameters file, are listed in table 3.4 above.

Variable	Descriptive Name	T	U
θ_s	Sun zenith angle	i	deg
θ_v	Viewing zenith angle	i	deg
$\Delta\phi$	Azimuth angle between pixel-sensor and pixel-sun plane	i	deg
M	air mass	i	dl
Iaer	index for aerosol multiple scattering LUTs	i	dl
τ_a	aerosol optical thickness	i	dl
$Px\omega_0$	Phase function times single scattering albedo	i	dl
$\rho_{a,P}$	Aerosol reflectance in primary scattering approximation	c	dl
$aa^{(s)}$, $ba^{(s)}$, $ca^{(s)}$, $da^{(s)}$	Polynomial coefficients for each of the Fourier components of the correction factor for multiple scattering	c	dl
$f_a^{(s)}$	Fourier components of the correction factor for multiple scattering	c	dl
f_a	Correction factor for multiple scattering	c	dl
ρ_a	aerosol reflectance	o	dl

The ***ref_aerosol*** *procedure is defined as follows:*
(deleted 2.1.18-1, 2.1.18-2)
Estimate aerosol reflectance for primary scattering

$$\rho_{a,P} = \frac{Px\omega_0}{4(\mathbf{cos}(\theta_s)+\mathbf{cos}(\theta_v))}\left(1-e^{-\tau_a.M}\right) \qquad (2.1.18\text{-}3)$$

Compute Fourier components of correcting factor for aerosol multiple scattering

```
For each Fourier series component order s in 0..5
    $aa^{(s)}$ = Aermult_LUT interpol :($\theta_s,\theta_v$) select :(iaer, s, k=1)    (2.1.18-4)
    $ba^{(s)}$ = Aermult_LUT interpol :($\theta_s,\theta_v$) select :(iaer, s, k=2)    (2.1.18-5)
    $ca^{(s)}$ = Aermult_LUT interpol :($\theta_s,\theta_v$) select :(iaer, s, k=3)    (2.1.18-6)
    $da^{(s)}$ = Aermult_LUT interpol :($\theta_s,\theta_v$) select :(iaer, s, k=4)    (2.1.18-7)
    $f_a^{(s)} = aa^{(s)} + ba^{(s)}(\tau_a) + ca^{(s)}(\tau_a)^2 + da^{(s)}(\tau_a)^3$    (2.1.18-8)
Endfor  End of loop over index s
```

Compute correcting factor for aerosol multiple scattering as Fourier sum

```
$f_a = f_a^{(0)} + 2f_a^{(1)}\cos(\Delta\phi) + 2f_a^{(2)}\cos(2\Delta\phi) + 2f_a^{(3)}\cos(3\Delta\phi) + 2f_a^{(4)}\cos(4\Delta\phi) + 2f_a^{(5)}\cos(5\Delta\phi)$    (2.1.18-9)
Compute aerosol reflectance, corrected for multiple scattering
$\rho_a = \rho_{a,P}.f_a$    (2.1.18-10)
return $\rho_a$    (2.1.18-11)

End of ref_aerosol procedure
```

5.5.8 - procedure *ref_smile_corr* to correct reflectance for smile effect (step 2.1.6)

The List of Variables below identifies the dummy input, output and locally computed variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, read from an auxiliary parameters file, are listed in table 3.4 above.

Variable	Descriptive Name	T	U
$\rho_{ng}(b)$	TOA reflectance, corrected for stratospheric aerosol contribution and gaseous absorption	i	deg
RefCorr_sw(b)	Array of switch controlling reflectance correction for each band	i	deg
RefCorr_b(b,i)	Array of band indices to be used as lower (i=1) and upper (i=2) points for reflectance derivative computation	i	deg
Detector	MERIS detector index corresponding to current pixel	i	dl
$\partial\rho/\partial\lambda$	reflectance derivative wrt wavelength	c	dl
$\rho^*_{ng}(b)$	TOA reflectance corrected for stratospheric aerosol contribution, gaseous absorption and smile effect	o	dl

```
The ref_aerosol procedure is defined as follows:
For b in {b412..b900}
    If RefCorr_sw(b) == TRUE then
        $\frac{\partial\rho}{\partial\lambda} = \frac{\rho_{ng}(\mathrm{RefCorr\_b}(b,2),j,f) - \rho_{ng}(\mathrm{RefCorr\_b}(b,1),j,f)}{\lambda\mathrm{pix}(\mathrm{RefCorr\_b}(b,2),\mathrm{Detector}(j,f)) - \lambda\mathrm{pix}(\mathrm{RefCorr\_b}(b,1),\mathrm{Detector}(j,f))}$    (2.1.6-3)
        $\rho^*_{ng}(b,j,f) = \rho_{ng}(b,j,f) + \frac{\partial\rho}{\partial\lambda}\cdot\left(\lambda_{theo}(b) - \lambda\mathrm{pix}(b,\mathrm{Detector}(j,f))\right)$    (2.1.6-4)
    Else
        $\rho^*_{ng}(b,j,f) = \rho_{ng}(b,j,f)$    (2.1.6-5)
    Endif
Endfor End of loop over bands
End of ref_smile_corr procedure
```

Copyright © 2011 ACRI S.T.

5.5.9 - procedure *Pscatt* to retrieve apparent pressure of main contributor to atmospheric scattering over ocean (step 2.1.7.1)

The List of Variables below identifies the dummy input, output and locally computed variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, read from an auxiliary parameters file, are listed in table 3.4 above.

Variable	Descriptive Name	T	U	Range - References
λ_{b761}	Actual wavelength of band b761	i	nm	
θ_s	Sun zenith angle	i	deg	
θ_v	Viewing zenith angle	i	deg	
$\Delta\phi(j,f)$	Difference of azimuth angles	i	deg	
$\rho_{R1}(b761)$	Rayleigh reflectance at nominal wavelength of b761, corrected for pressure	i	dl	b: b761
SATURATED_F(b)	Level 1b radiance saturation flag	i	dl	b: b779
$\rho_{TOA}(b)$	TOA reflectance	i	dl	b: b753, b761, b779
Pscatt	apparent pressure of main contributor to atmospheric scattering over ocean	o	hPa	

The **Pscatt** *procedure is defined as follows:*

NOTE: The procedure ComputePressure is specified in sections 5.5.11 below.

Compute the reference RO_TOA at 761

If not SATURATED_F(b779) **then**

$$\frac{\partial\rho}{\partial\lambda} = \frac{\rho_{TOA}(b779) - \rho_{TOA}(b753)}{\lambda_{theo}(b779) - \lambda_{theo}(b753)} \qquad (2.1.7.1\text{-}1)$$

$$\rho_{TOA}^{761\text{ no abs}} = \rho_{TOA}(b753) + \left(\lambda_{b761} - \lambda_{theo}(b753)\right)\cdot\frac{\partial\rho}{\partial\lambda}$$

Else

$$\rho_{TOA}^{761\text{ no abs}} = \rho_{TOA}(b753, j, f) \qquad (2.1.7.1\text{-}2)$$

Endif

deleted *(2.1.7.1-3)*

deleted *(2.1.7.1-4)*

Compute the Rayleigh O2 transmittance

trO2 = TO2_Ray_LUT **interpol**: (λ_{b761}, θ_s, θ_v) *(2.1.7.1-5)*

Rayleigh correction on the O2 transmittance

$$\text{TO2RCorrected} = \frac{\rho_{TOA}(b761) - \rho_{R1}(b761)\cdot trO2}{\rho_{TOA}^{761\text{ no abs}} - \rho_{R1}(b761)} \qquad (2.1.7.1\text{-}6)$$

Copyright © 2011 ACRI S.T.

Determination of the aerosol pressure after surface correction:
Compute the aerosol O2 transmittance
trAerosol = TO2_Atm_Aer_LUT **interpol**: (λ_{b761}, θ_s, θ_v) *(2.1.7.1-7)*
Compute the aerosol fresnel O2 transmittance for direct to diffuse
trFresnel1 = TO2_Fresnel_LUT **interpol**: (λ_{b761}, θ_s, θ_v) *(2.1.7.1-8)*
Compute the aerosol fresnel O2 transmittance for diffuse to direct
trFresnel2 = TO2_Fresnel_LUT **interpol**: (λ_{b761}, θ_v, θ_s) *(2.1.7.1-9)*
Compute the APF ratio between forward and backward scattering
compute scattering angle
$\theta = \mathrm{acos}(-\cos(\theta_s)*\cos(\theta_v) - \sin(\theta_s)*\sin(\theta_v)*\cos(\Delta\varphi))$
compute wave angle (angle between View and Fresnel reflection)
$xsi = \mathrm{acos}(\cos(\theta_s)*\cos(\theta_v) - \sin(\theta_s)*\sin(\theta_v)*\cos(\Delta\varphi))$
compute ratio of PhaseFunction for these two directions
pfb = APF_Junge_LUT **nearest**: (xsi) / APF_Junge_LUT **nearest**: (θ) *(2.1.7.1-10)*
Compute the contribution of the aerosol-Fresnel
caf = 1.0 + pfb * (fresnel_Coeff_LUT **nearest**: (θ_s)
+ fresnel_Coeff_LUT **nearest**: (θ_v)) *(2.1.7.1-11)*
Correction of the O2 transmittance for the coupling aerosol-Fresnel
xx = (trAerosol + (pfb/caf) *
(trFresnel1*fresnel_Coeff_LUT **nearest**: (θ_s) +
trFresnel2 * fresnel_Coeff_LUT **nearest**: (θ_v))) *(2.1.7.1-12)*
to2Rf = TO2RCorrected * trAerosol / xx *(2.1.7.1-13)*

call compute_pressure(λ_{b761}, θ_s, θ_v, to2Rf, Pscatt) *(2.1.7.1-14)*

*End of **Pscatt** procedure*

5.5.10 - procedure *P1* to retrieve apparent surface pressure over land (step 2.1.7.2)

The List of Variables below identifies the dummy input, output and locally computed variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, read from an auxiliary parameters file, are listed in table 3.4 above.

Variable	Descriptive Name	T	U	Range - References
λ_{b761}	Actual wavelength of band b761	i	nm	
θ_s	Sun zenith angle	i	deg	
SATURATED_F(b)	Level 1b radiance saturation flag	i	dl	b: b779
θ_v	Viewing zenith angle	i	deg	
$\rho_{TOA}(b)$	TOA reflectance	i	dl	b: b753, b761, b779
P1	apparent surface pressure	o	hPa	

*The **P1** procedure is defined as follows:*
NOTE: The procedure ComputePressure is specified in sections 5.5.11 below.

Copyright © 2011 ACRI S.T.

```
Compute the reference RO_TOA at 761
If not SATURATED_F(b779,j,f) then
    $\frac{\partial\rho}{\partial\lambda} = \frac{\rho_{TOA}(b779) - \rho_{TOA}(b753)}{\lambda_{theo}(b779) - \lambda_{theo}(b753)}$        (2.1.7.2-1)
    $\rho_{TOA}^{761\text{ no abs}} = \rho_{TOA}(b753) + \left(\lambda_{b761} - \lambda_{theo}(b753)\right) \cdot \frac{\partial\rho}{\partial\lambda}$
Else
    $\rho_{TOA}^{761\text{ no abs}} = \rho_{TOA}(b753, j, f)$        (2.1.7.2-2)
Endif

Ratio of the two bands
$\text{to2Ratio} = \frac{\rho_{TOA}(b761)}{\rho_{TOA}^{761\text{ no abs}}}$        (2.1.7.2-3)
Computation of the apparent pressure P1
call compute_pressure($\lambda_{b761}$, $\theta_s$, $\theta_v$, to2Ratio, P1)        (2.1.7.2-4)
End of P1 procedure
```

5.5.11 - Procedure compute_pressure to compute pressure (step 2.1.7.3)

The List of Variables below identifies the dummy input, output and locally computed variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, read from an auxiliary parameters file, are listed in table 3.4 above.

Variable	Descriptive Name	T	U	Range - References
λ_{b761}		i	nm	
θ_s	Sun zenith angle	i	deg	
θ_v	Viewing zenith angle	i	deg	
ratio	O2 transmittance in band 11	i	dl	
Pressure	Computed pressure	o	hPa	

The **compute_pressure** *procedure is defined as follows:*

```
Retreive the bracketing indices for viewing geometry and filter
Find iFilter such that $\lambda_{ref}$_O2[iFilter] =< $\lambda_{b761}$ =< $\lambda_{ref}$_O2[iFilter+1]        (2.1.7.3-1)

exception processing: when $\lambda_{b761}$ < $\lambda_{ref}$_O2[1] OR $\lambda_{b761}$ > $\lambda_{ref}$_O2[nFilter]
    Pressure = BAD_VALUE
    Exit procedure
end of exception processing
Find jθs such that $\theta_{ref}$_O2[jθs] =< $\theta_s$ =< $\theta_{ref}$_O2[jθs+1]        (2.1.7.3-2)
Find kθv such that $\theta_{ref}$_O2[kθv] =< $\theta_v$ =< $\theta_{ref}$_O2[kθv+1]        (2.1.7.3-3)
```

Copyright © 2011 ACRI S.T.

```
exception processing: when any index variable $\lambda_{b761}$, $\theta_S$ or $\theta_V$ is out of LUT index range
    select the corresponding extreme index (e.g 1 if below minimum, maximum index-1 if above range)
    continue processing
end of exception processing
exception processing: if ratio <= 0
    Pressure = BAD_VALUE
    Exit procedure
end of exception processing
t = ln(ratio)                                                            (2.1.7.3-4)

For i=1..2
    For j=1..2
        For k=1..2
            iLayer = 1                                                   (2.1.7.3-5)
            While ( iLayer < nLayers-1 AND
                  ratio < TO2_Atm_LUT select:( $\lambda_{ref}$_O2[ifilter+i-1], iLayer+1,
                           $\theta_{ref}$_O2[j$\theta$s+j-1], $\theta_{ref}$_O2[k$\theta$s+k-1]) )
                iLayer = iLayer + 1                                      (2.1.7.3-6)
            EndWhile

            t1 = ln( TO2_Atm_LUT select:( $\lambda_{ref}$_O2[ifilter+i-1], iLayer,
                           $\theta_{ref}$_O2[j$\theta$s+j-1], $\theta_{ref}$_O2[k$\theta$s+k-1]) )   (2.1.7.3-7)
            t2 = ln( TO2_Atm_LUT select:( $\lambda_{ref}$_O2[ifilter+i-1], iLayer+1,
                           $\theta_{ref}$_O2[j$\theta$s+j-1], $\theta_{ref}$_O2[k$\theta$s+k-1]) )   (2.1.7.3-8)
            p1 = pressLevel[iLayer]                                      (2.1.7.3-9)
            p2 = pressLevel[iLayer+1]                                    (2.1.7.3-10)
            Press_LUT(i,j,k) = p2 + ((p2 - p1)/(t2 - t1)) * (t - t2)     (2.1.7.3-11)
        Endfor
    Endfor
Endfor

Pressure = Press_LUT interpol: ($\lambda_{b761}$, $\theta_S$, $\theta_V$)               (2.1.7.3-12)

End of compute_pressure procedure
```

5.6 - Quality Control and Diagnostics

N/A.

5.7 - Exception Handling

See the blocks labelled "exception processing:... end of exception processing" in section 5.5 above.

Copyright © 2011 ACRI S.T.

6. - TOTAL WATER VAPOUR RETRIEVAL

6.1. - Introduction

This chapter describes the algorithm to be applied to the MERIS signals in order to retrieve the atmospheric water vapour content (see RD8, section 2.4).

6.2. - Algorithm Overview

This algorithm is applied to land, water and cloud pixels. It is based on a differential absorption method using two spectral bands close to each other (one within the absorption band and the other outside the absorption band). The retrieval algorithm depends on the target nature in order to cope with the surface reflectivity and the physical processes involved:

- Land or water with significant Sun glint (bright reflective surfaces),
- Water without significant glint (dark reflective surfaces) or
- Clouds (reflection with significant penetration depth).

The Water Vapour retrieval over Land surface uses a neural network approach.

The algorithms over water (outside significant Sun glint) and over clouds consist in polynomials of the logarithm of the ratio of TOA radiance (corrected for stratospheric aerosols if required) at band 15 (900 nm, within the water vapour absorption region) to TOA radiance at band 14 (885 nm, outside water vapour absorption). Above water surfaces, the algorithm takes into account the aerosol optical depth, except when Sun glint is significant.

Polynomial coefficients depend on illumination and viewing geometry, surface type, aerosol optical thickness (above water), cloud properties and surface albedo (above clouds), for any given pixel. That dependence is coded in look-up tables.

The diagram in figure 6.2-1 below shows the logic of the water vapour retrieval algorithm.

Copyright © 2011 ACRI S.T.

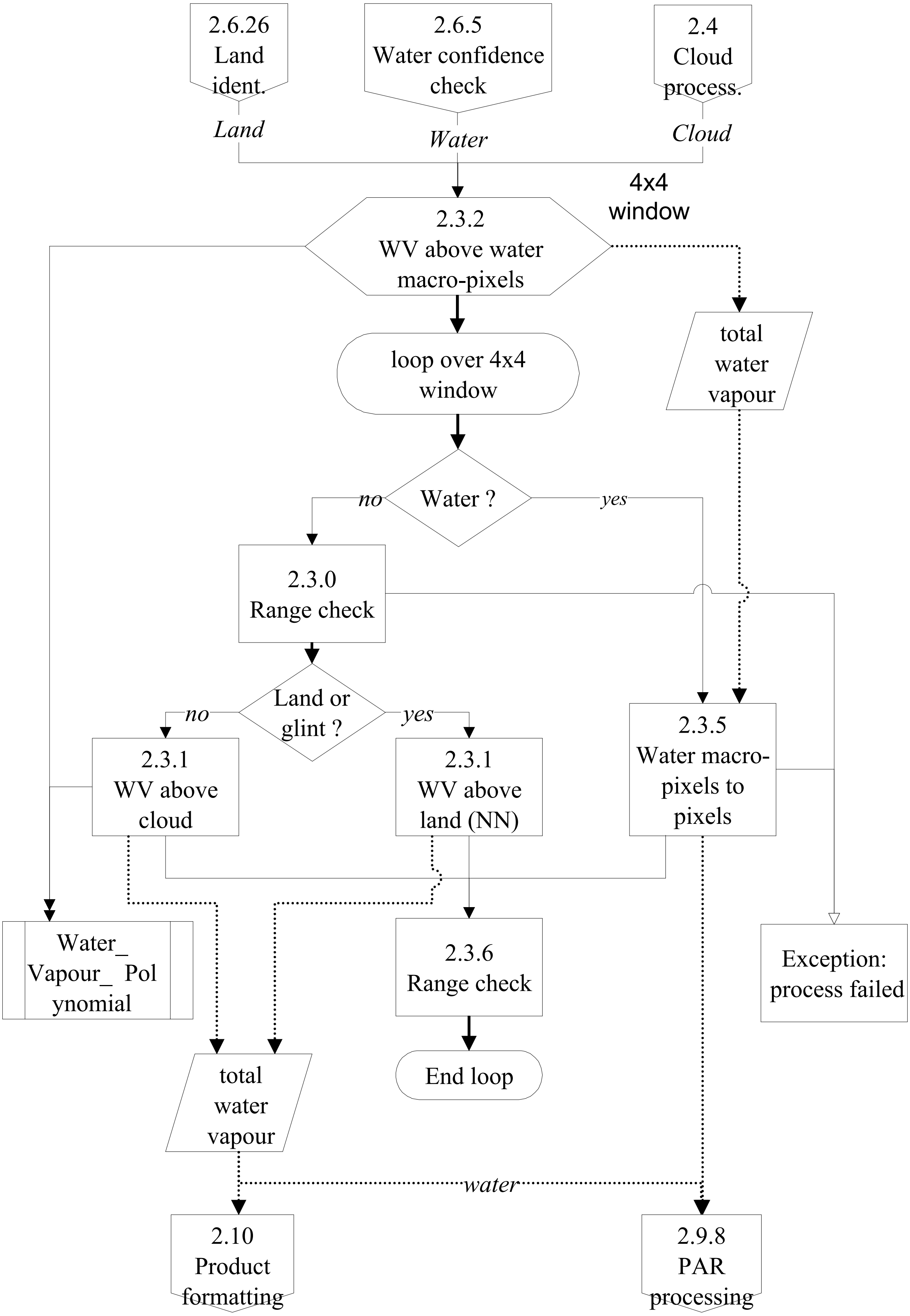


Figure 6.2-1 : MERIS Level 2 Water Vapour retrieval (step 2.3)

Copyright © 2011 ACRI S.T.

6.3 - Mathematical Description Of Algorithm

The spectral bands centred at 885 nm and 900 nm have proven to be the best suited for the retrieval of water vapour over all surfaces. The total water vapour is computed thanks to surface dependent algorithms: a Neural Network over Land and very bright waters (high Sun glint or Sea ice) and polynomials over water and clouds. In the latter case, the algorithms are implemented in two steps. In a first step, depending on the type of pixel: water or cloud, polynomial coefficients are read from LUT. The ratio of TOA radiances (corrected for stratospheric aerosols if needed) at 885nm and 900nm is corrected for tropospheric aerosols over water. The second step is a simple polynomial applied to the (corrected) radiance ratio.

6.3.1. - Water vapour retrieval over land surfaces (step 2.3.1)

The Water Vapour is retrieved over Land using a Neural Network. It takes as input variables representative of

- the illumination and viewing geometry,
- the surface albedo spectral variation between 885 and 900 nm,
- the normalised radiance in channel 14 (885nm) and the estimated transmission at 900 nm,
- the normalised radiance in channel 19 (754nm), the estimated transmission of molecular oxygen around 761nm, and the actual central wavelength of channel 11,
- the surface pressure estimated from the above three inputs (see section 4).

6.3.2. - Water vapour retrieval over water surfaces (steps 2.3.2, 2.3.5)

Outside the Sun glint region, the retrieval of the total water vapour over water surfaces is more difficult than over land surfaces because of the larger influence of aerosols. The polynomial coefficients take into account aerosol influence. Also in order to improve noise performance, 4x4 pixel averaging is performed as a pre-processing step.

In the Sun glint region, the same algorithm as above land is applied.

6.3.3. - Water vapour retrieval over clouds (step 2.3.3)

Polynomial coefficients take into account the cloud optical thickness and the albedo of the underlying surface.

6.3.4 – Range checks (steps 2.3.0, 2.3.6)

Range checks are performed on radiance at the algorithm input. Out of range radiance result in an exception, water vapour is not processed. When processed, the water vapour is also checked for range, the product is kept but a flag is raised when out of range.

6.3.5 - Water vapour polynomial (function)

The algorithm consists in a second-degree polynomial equation using the logarithm of the ratio of TOA radiance at 885 and 900 nm and coefficients. All polynomial parameters are provided by steps 2.3.1, 2.3.2, 2.3.3 depending on the type of pixel.

6.4. - List of Variables

Variable	Descriptive Name	T	U	Range - References
CLOUD_F(j,f)	Cloud flag for pixel (j,f)	i	dl	from step 2.1c (§5.4)
SNOW_ICE_F(j,f)	Snow or Ice flag	o	dl	from step 2.1c (§5.4)
INVALID_F(j,f)	"invalid pixel" flag for pixel (j,f)	i	dl	from L1B Flags MDS
LANDCONS_F(j,f)	Land /water flag for pixel (j,f)	i	dl	from step 2.1c (§5.4)
$W_s(j,f)$	wind speed modulus for pixel (j,f)	i	$m.s^{-1}$	from step 2.16 (§5.5.5) or 2.6.5 (§8.2.2),
ROG(j,f)	Sun glint reflectance for pixel (j,f)	i	dl	from step 2.6.26 (§5.5)
$\rho(b,j,f)$	Stratospheric aerosol corrected reflectance for pixel (j,f)	i	dl	From step 2.1c (§5.4)
$\rho^*_{ng}(b,j,f)$	TOA reflectance, corrected for stratospheric aerosol contribution, gaseous absorption and smile	i	dl	for water pixels; b=b865, from step 2.1c (§5.4)
$\theta_s(j,f)$	Sun zenith angle for pixel (j,f)	i	deg	from step 2.1a (§3.4)
$\theta_v(j,f)$	Viewing zenith angle for pixel (j,f)	i	deg	idem
$\Delta\phi(j,f)$	Azimuth difference for pixel (j,f)	i	deg	idem
$P_{ECMWF}(j,f)$	ECMWF pressure for pixel (j, f)	i	hPa	idem
$\tau_c(j,f)$	Cloud optical thickness for pixel (j, f)	i	dl	from step 2.4.3 (§7.4)
$\eta_{LN}(j,f)$	O2 transmittance at b761 (reflectance or normalised radiance ratio)	i	dl	from step 2.1b (§4.5.3)
$P_S(j,f)$	Surface pressure over Land pixels	i	hPa	From 2.1b (§4.5.3)
Month	Month of acquisition	i	-	from step 2.1b (§4.5.3)
Detector(j,f)	MERIS detector associated to pixel	i	dl	from Level 1b (Flags MDS)
$F_0^{WV}(b)$	Solar flux consistent with WV LUTs	s	EU	b in b775, b865, b885, b900
Aerosol_wv_LUT [μ_s, μ_v, $\Delta\phi$, k]	LUTs of polynomial coefficients for aerosol correction over water	s	dl	µs :27 values* µv : 18 values* $\Delta\phi$:25 values k: 3 values
Cloud_wv_LUT[μ_s, μ_v, $\Delta\phi$, δ, α, k]	LUTs of polynomial coefficients water vapour retrieval over cloud	s	dl	µs, µv, $\Delta\phi$:see above δ: cloud optical thickness, 20 values α: surface albedo, 10 values k: 3 values
INV_WV	Threshold on radiance at 885 nm for marking a pixel as invalid	s	dl	
λ^C_{761}	Band 11 central wavelengths for pressure retrieval	s	nm	
Water_noglint_wv_LUT [μ_s, μ_v, $\Delta\phi$, δ_A, w, k]	LUTs of polynomial coefficients for water vapour retrieval over water without glint	s	dl	µs, µv, $\Delta\phi$:see above δ_A: aerosol optical thickness, 20 values w: wind speed, 5 values k: 3 values

Table 6.4-1: List of Variables

*: the increasing order of magnitude for θ_s, θ_v indices in the LUT files, imposes a decreasing order for the corresponding μ_s, μ_v cosines.

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
α_15_14_LUT [lat,lon,month]	Surface albedo slope LUT	s	dl	lat: latitude, lon: longitude, month: 1..12
α_bad	Missing data value in α_15_14_LUT	s	dl	
Surfalb_b14_LUT [lat,lon,month]	Surface albedo at b885 LUT	s	dl	lat: latitude, lon: longitude, month: 1..12
	Water Vapour over Land Neural network table	s		See step 2.3.1.0 in § 6.5 below
OUT_MIN	Minimum acceptable output value	s	$g.cm^{-2}$	0.1 $g.cm^{-2}$
OUT_MAX	Maximum acceptable output value	s	$g.cm^{-2}$	7 $g.cm^{-2}$
BAD_VALUE	Output value when algorithm fails	s	dl	see §2 above
$WV_{NN}\_IN_{min}$, $WV_{NN}\_IN_{max}$	Validity ranges for every neural network input	s	misc	
$WV_{NN}\_OUT_{min}$, $WV_{NN}\_OUT_{min}$	Validity range for the neural network output	s	$g.cm^{-2}$	
WVLand_Net	Neural Net object	c	-	
$L_T(b,l,p)$	TOA radiance consistent with Sun Irradiance used to build WV LUTs	c	LU	b: b775, b865, b885, b900; to Breakpoint
WV_HIGLINT_F(j,f)	Flag for pixels contaminated by too much glint to use Total Water Vapour over Water algorithm	c	dl	Boolean, to breakpoints
a_{wv}	zero- order polynomial coefficient	c	dl	
b_{wv}	first order polynomial coefficient	c	dl	
c_{wv}	second order polynomial coefficient	c	dl	
δ_A	Aerosol optical depth estimate	c	dl	
d_{wv}	Coefficient of polynomial for aerosol estimate	c	dl	
e_{wv}	*Idem*	c	dl	
f_{wv}	*Idem*	c	dl	
$L_{Tave}(b)$	TOA Radiance averaged on 4x4 window	c	LU	b: {b775, b865, b885, b900}
μ_s, μ_v	Cosine of Sun and view zenith angle	c	dl	]0..1]
N_{ave}	Number of water pixels within 4x4 window on which average is performed	c	dl	
NO_VAPOUR_WATER	Flag indicating failure of water vapour above water	c	-	Boolean[1]
T	Radiance Band Ratio b900/b885	c	dl	
water_vapour_water	Total water vapour for 4x4 window	c	$g.cm^{-2}$	

---

[1] all calculated and output Boolean parameters shall be initialised to FALSE (0).

Copyright © 2011 ACRI S.T.

ORINPWV_F(j,f)	Flag for out of range input	o	dl	see 1,to step 2.10 (§10.4), to Breakpoint
OROUTWV_F(j,f)	Flag for out of range output	o	dl	idem
w_T(j,f)	Total water vapour content for pixel (j,f)	o	$g.cm^{-2}$	to step 2.10 (§10.4), to Breakpoint

Table 6.4-1: List of Variables (cont.)

Copyright © 2011 ACRI S.T.

6.5. – Equations (step 2.3)

NOTE: sub-steps and equations are re-numbered with reference to issue 3.4 of this document.

```
At process initialisation
read GADS- Water Vapour Neural Network for Land into memory (see AD6)      (2.3.1.0-1)
call NN_CreateNetFromMemFile routine (see AD6)                             (2.3.1.0-2)
    input: address of memory copy of GADS
    return value: WVLand_Net
end of process initialisation section
```

6.5.1 – Water vapour retrieval above water macro-pixels (step 2.3.2)

Water vapour is processed for a "macro-pixel" represented by the mean radiance over all water pixels in the 4x4 window.

For each 4 (across-track) x 4 (along-track) window containing at least 1 pixel (j, f) such that (INVALID_F (j, f) == FALSE)[2]

```
    For each pixel (l,p) in 4x4 window such that ( (INVALID_F (l,p) OR LANDCONS_F(l,p) OR CLOUD_F(l,p)) == FALSE )
        Compare sun glint reflectance to Water Vapour glint threshold (step 2.6.5.1.9)
            if (ROG(l,p) ≥ thres_WVhg . $\rho^{*}_{ng}$(b865, l, p)) then
                WV_HIGLINT_F(l, p) = TRUE                                  (2.6.5.1.9-1)
            else
                WV_HIGLINT_F(l, p) = FALSE                                 (2.6.5.1.9-2)
            endif
    Endfor

    Let (j0, f0) be the column, frame co-ordinates of the North-East corner pixel of window
    For each pixel (l,p) in 4x4 window such that (INVALID_F (l,p) == FALSE)
        For each band b in {b775, b865, b885, b900}
            $L_T(b,l,p) = \frac{\rho(b,l,p).F_0^{WV}(b).\cos\theta_s(l,p)}{\pi}$        (2.3.2-0)
        Endfor
    Endfor
    Glint-free water: radiances are averaged over 4x4 pixels and aerosol optical depth is taken into account (ignoring invalid or land or cloud pixels)
    Let GoodPix(l,p) = (NOT INVALID_F(l,p) AND NOT LANDCONS_F(l,p) AND NOT CLOUD_F(l,p) AND NOT WV_HIGLINT_F(l,p) AND NOT SNOW_ICE_F(l,p) AND NOT SATURATED_F(b,l,p)) for any pixel (l,p) and any band b in {b775, b865, b885, b900}
    $N_{ave} = \sum_{\substack{l,p \text{ in } 4x4 \\ \text{window}}} GoodPix(l,p)$        (2.3.2-1)
    If ($N_{ave}$ > 0) then
        For each band b in {b775, b865, b885, b900}
```

---

[2] NOTE: when reaching product boundaries the 4x4 window shall be clipped.

```
        $L_{Tave}(b) = \frac{1}{N_{ave}} \cdot \sum_{\substack{l,p \text{ in} \\ 4x4 \text{ window}}} GoodPix(l,p).L_T(b,l,p)$        (2.3.2-2)
    Endfor
Endif
Check radiance range for quality control for average over water macro-pixels
If (($N_{ave}$ == 0) OR ($L_{Tave}$(b885) ≤ 0) OR ($L_{Tave}$(b900) ≤ 0) OR ($L_{Tave}$(b775) ≤ 0)
    OR ($L_{Tave}$(b865) ≤ 0) OR ($L_{Tave}$(b885) > INV_WV)) then
        NO_VAPOUR_WATER = TRUE        (2.3.2-3)
else
    NO_VAPOUR_WATER = FALSE        (2.3.2-4)
    Compute band ratio
    $T = \frac{L_{Tave}(b900)}{L_{Tave}(b885)}$        (2.3.2-5)
    Estimate aerosol optical depth
    Let $\mu_s$ = cos ($\theta_s$ (j0, f0)); $\mu_v$ = cos ($\theta_v$ (j0, f0))
    $d_{wv}$ = Aerosol_wv_LUT interpol:($\mu_s$, $\mu_v$, $\Delta\phi$ (j0, f0)) select (k=1)        (2.3.2-6)
    $e_{wv}$ = Aerosol_wv_LUT interpol:($\mu_s$, $\mu_v$, $\Delta\phi$ (j0, f0)) select (k=2)        (2.3.2-7)
    $f_{wv}$ = Aerosol_wv_LUT interpol:($\mu_s$, $\mu_v$, $\Delta\phi$ (j0, f0)) select (k=3)        (2.3.2-8)
    $\delta_A = d_{wv} + e_{wv}.L_{Tave}$ (b775) + $f_{wv}.L_{Tave}$ (b865)        (2.3.2-9)
    Interpolate polynomial coefficients in LUT Water_noglint_wv
    $a_{wv}$ = Water_noglint_wv_LUT  interpol:($\mu_s$, $\mu_v$, $\Delta\phi$(j0,f0), $\delta_A$, $W_s$(j0,f0))
                                     select:(k=1)        (2.3.2-10)
    $b_{wv}$ = Water_noglint_wv_LUT  interpol:($\mu_s$, $\mu_v$, $\Delta\phi$(j0,f0), $\delta_A$, $W_s$(j0,f0))
                                     select:(k=2)        (2.3.2-11)
    $c_{wv}$ = Water_noglint_wv_LUT  interpol:($\mu_s$, $\mu_v$, $\Delta\phi$(j0,f0), $\delta_A$, $W_s$(j0,f0))
                                     select:(k=3)        (2.3.2-12)
    Compute total water vapour applicable to all pixels in window
    water_vapour_water = Water_Vapour_Polynomial ($a_{wv}$, $b_{wv}$, $c_{wv}$, T)        (2.3.2-13)
NOTE: the function Water_Vapour_Polynomial is specified in §6.5.7 below.
Endif

Processing of pixels in 4x4 window
For each pixel (j,f) in 4x4 window such that (INVALID_F (j, f) == FALSE)
    Let $\mu_s$ = cos ($\theta_s$ (j,f)); $\mu_v$ = cos ($\theta_v$ (j, f))
    If (LANDCONS_F(j,f) OR WV_HIGLINT_F(j,f) OR SNOW_ICE_F(j,f)
        OR CLOUD_F (j, f))
    then
```

6.5.2. – Range check for land and cloud pixels (step 2.3.0)

```
        Check radiance range for quality control, for all land or cloud pixels
        If (($L_T$(b885,j,f) <= 0) OR ($L_T$(b900, j, f) <= 0) OR ($L_T$(b885,j,f) >= INV_WV) OR
            SATURATED_F(b885,j,f) OR SATURATED_F(b900,j,f))
            then
            Exception: water vapour process failed (see 6.7 below)        (2.3.0-1)
```

Copyright © 2011 ACRI S.T.

```
Endif
```

6.5.3. - Water-vapour retrieval over land surfaces, water with high glint or Sea ice (step 2.3.1)

Note: Step 2.3.1 has been fully revised for issue 8.0 of this document, all equations being new and loosing any correspondence with earlier issues.

```
If (LANDCONS_F(j,f) OR WV_HIGLINT_F(j,f) OR SNOW_ICE_F(j,f)) then
    Rα_900_885 =
        α_15_14_LUT nearest: (lat(j, f), lon (j, f)) select: (month)          (2.3.1-1)
    If Rα_900_885 == α_bad then
        Rα_900_885 = $\frac{7\cdot\rho(b885)-3\cdot\rho(b865)}{4\cdot\rho(b885)}$          (2.3.1-2)
        Exception processing: SATURATED_F(b865,j,f)
        water vapour process failed (see 6.7 below)
        End exception processing:
    Endif

    If ( LANDCONS_F(j,f) AND NOT PCD_NN_F (j,f) )
        then Press=$P_s$(j,f) else Press=$P_{ECMWF}$(j,f)          (2.3.1-3)
    NN_Input(1) = cos $\theta_s$(j, f)          (2.3.1-4)
    NN_Input(2) = cos $\theta_v$(j, f)          (2.3.1-5)
    NN_Input(3) = sin $\theta_v$(j, f).cos(180-$\Delta\phi$(j, f))          (2.3.1-6)
    NN_Input(4) = cos$\theta_s$(j, f).$\rho$(b885, j, f)/$\pi$          (2.3.1-7)
    NN_Input(5) = Rα_900_885          (2.3.1-8)
    NN_Input(6) = ln($\rho$(b900, j, f)/$\rho$(b885, j, f))          (2.3.1-9)
    NN_Input(7) = cos$\theta_s$(j, f) .$\rho$(b753, j, f)/$\pi$          (2.3.1-10)
    NN_Input(8) = ln($\eta_{LN}$(j,f))          (2.3.1-11)
    NN_Input(9) = $\lambda^C_{761}$(Detector(j,f))          (2.3.1-12)
    NN_Input(10) = min(Press, $WV_{NN}\_IN_{max}$(10))          (2.3.1-13)

Exception processing: ($\rho$(b900, j, f)/$\rho$(b885, j, f) $\le$ 0 OR $\rho$(b885, j, f)=0
                      OR $\eta_{LN}$(j,f) $\le$ 0 OR
                      NN_Input(i) < $WV_{NN}\_IN_{min}$(i)) OR
                      (NN_Input(i) > $WV_{NN}\_IN_{max}$(i)) for any i in [1,10]
    water vapour process failed (see 6.7 below)
End exception processing:

    call Nn_ProcessNet routine (see AD6)          (2.3.1-14)
        Network: WVLand_Net; input: NN_Input; number of input elements: 10;
        output: NN_output; number of output elements: 1
    $w_T$ (j, f) = NN_output(1)          (2.3.1-15)
```

6.5.4. - Water-vapour retrieval over clouds (step 2.3.3)

```
Else if (CLOUD_F (j, f)) then
```

Copyright © 2011 ACRI S.T.

The surface albedo is read from a LUT as a function of geographical latitude, geographical longitude and month of measurement derived from the time index found in the Level 1b product header.

```
            $\alpha_{surf}$ = Surfalb_b14_LUT nearest:(lat (j, f), lon(j, f)) select:(month)        (2.3.3-0)
            $T = \frac{L_T(b900, j, f)}{L_T(b885, j, f)}$        (2.3.3-1)
            $a_{wv}$ = Cloud_wv_LUT interpol:($\mu_s$, $\mu_v$, $\Delta\phi$ (j, f), $\tau_c$ (j, f), $\alpha_{surf}$ (j, f)) select:(k=1)        (2.3.3-2)
            $b_{wv}$ = Cloud_wv_LUT interpol:($\mu_s$, $\mu_v$, $\Delta\phi$ (j, f), $\tau_c$ (j, f), $\alpha_{surf}$ (j, f)) select:(k=2)        (2.3.3-3)
            $c_{wv}$ = Cloud_wv_LUT interpol:($\mu_s$, $\mu_v$, $\Delta\phi$ (j, f), $\tau_c$ (j, f), $\alpha_{surf}$ (j, f)) select:(k=3)        (2.3.3-4)
            Compute total water vapour for cloud pixels
            $w_T$ (j, f) = Water_Vapour_Polynomial ($a_{wv}$, $b_{wv}$, $c_{wv}$, T)        (2.3.3-5)
        Endif
```

6.5.5 - Water macro-pixels to pixels (step 2.3.5)

```
For all water pixels of the 4x4 window, total water vapour is an output of step 2.3.2
    Else
        If (NO_WATER_VAPOUR) then
            Exception: water vapour process failed (see 6.7 below)        (2.3.5-1)
        Else
            $w_T$ (j, f) = water_vapour_water        (2.3.5-2)
        Endif
    Endif
```

6.5.6. - Range check on water vapour product (step 2.3.6)

```
    If (LANDCONS_F(j,f) OR WV_HIGLINT_F(j,f)) OR SNOW_ICE_F(j,f) then
        OROUTWV_F(j,f) = ($w_T$(j,f)<$WV_{NN}$_$OUT_{min}$) OR ($w_T$(j,f)>$WV_{NN}$_$OUT_{max}$)  (2.3.6-2)
    Else
        OROUTWV_F(j,f) = ($w_T$(j,f) < OUT_MIN) OR ($w_T$(j,f) > OUT_MAX)        (2.3.6-1)
    Endif
  End for      end loop on pixels in 4x4 window
End for        end loop on 4x4 windows
```

Copyright © 2011 ACRI S.T.

6.5.7. - Function Water_Vapour_Polynomial

6.5.7.1. - Inputs /Outputs

Variable	Descriptive Name	T	U
a_{wv}, b_{wv}, c_{wv}	polynomial coefficients	i	dl
T	corrected ratio $L_T(900)/ L_T(885)$	i	dl
WV	total water vapour	o	$g.cm^{-2}$

Table 6.5.7-1: List of parameters for Water_Vapour_Polynomial

6.5.7.2. – Equations

$WV = a_{wv} + b_{wv}.\mathbf{log}(T) + c_{wv}.\mathbf{log}^2(T)$

Return WV

Exception processing: $T \leq 0$

water vapour process failed (see 6.7 below)

End exception processing:

6.6. - Quality Control and Diagnostics

Input values of radiance at 885 nm, at 900 nm are checked, out of range values are not processed (see 2.3.0-1).

When any of the interpolation index variables μ_s, μ_v, $\Delta\phi$, α_{surf} , are outside of the LUTs index range, the nearest LUT values are used as explained in §2 above; the flag ORINPWV_F is set to TRUE and the pixel is processed.

Output values out of the range [OUT_MIN, OUT_MAX] $g.cm^{-2}$ are flagged by setting the OROUTWV_F flag to TRUE (see 2.3.6-1).

6.7. - Exception Handling

Exception processing: Water vapour process failed:

$W_T(j,f)$ = BAD_VALUE *(2.3-1)*

ORINPWV_F(j,f) = TRUE *(2.3-2)*

OROUTWV_F(j,f) = TRUE. *(2.3-3)*

skip the rest of water vapour processing for pixel (j, f)

end of exception processing

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

7. - Cloud processing

7.1. - Introduction

This chapter describes the MERIS level 2 cloud processing module for retrieving the albedo, optical thickness and type of clouds from TOA radiance data (RD 8, sections 2.1, 2.3).

7.2. - Algorithm Overview

The flow chart in figure 7.2-1 below describes the breakdown of the cloud processing module. Pixels output from the Pixel Identification module (step 2.1) and flagged as cloudy (CLOUD_F = true) are processed in order to retrieve the cloud albedo (step 2.4.1) and the cloud optical thickness (step 2.4.3). Cloud top pressure is already known from step 2.1. From the optical thickness and top pressure, the cloud type index is computed (step 2.4.8). It should be noted that cloud reflectance, written in the L2 product, is the TOA reflectance (corrected for stratospheric aerosol if needed) ρ (b, j, f), computed at step 2.1.

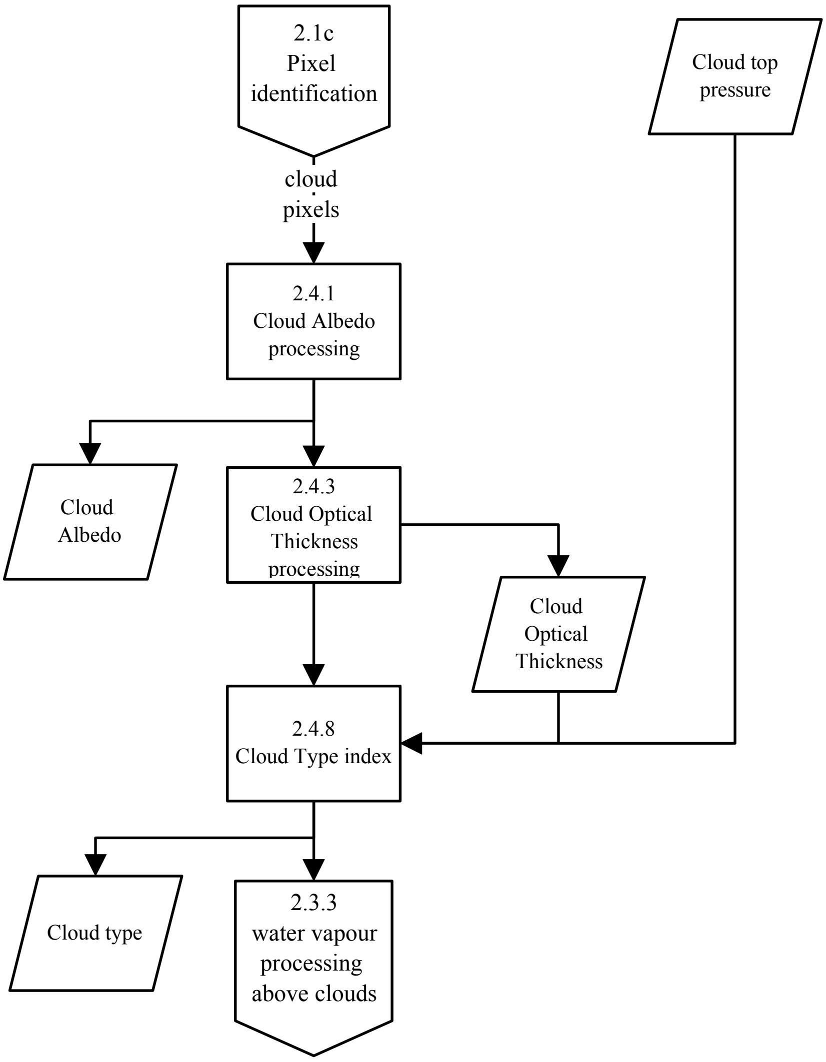


Figure 7.2-1 : MERIS Level 2 Cloud Processing (step 2.4)

Copyright © 2011 ACRI S.T.

7.3. -Mathematical Description of Algorithm

7.3.1. - Cloud Albedo processing (step 2.4.1)

The cloud albedo processing relates the cloud albedo α_c to the MERIS radiance in channel 10 (753.75 nm) using a polynomial regression technique. The polynomial coefficients are read from a Look Up Table (RD 7, 3.1.4) as a function of geometry (sun zenith angle, viewing angle and azimuth angle) and estimated surface albedo. The algorithm is detailed in RD 8, 2.1.

7.3.2. - Cloud Optical Thickness processing (step 2.4.3)

To retrieve the cloud optical thickness τ_c , the same technique is used as for retrieving the cloud albedo. A polynomial expression relates the cloud optical thickness τ_c as a function of the MERIS radiance in channel 10. The polynomial coefficients are read from a Look Up Table (RD 7, 3.1.5) as a function of geometry and estimated surface albedo. The algorithm is detailed in RD 8, 2.2.

7.3.3. - Cloud type processing (step 2.4.8)

The algorithm uses a simple classification table indexed by the cloud optical thickness and cloud top pressure, to provide a cloud type index.

7.4. - List of Variables

Variable	Descriptive Name	T	U	Range - References
$\rho(b,j,f)$	Stratospheric aerosol corrected reflectance for pixel (j,f)	i	dl	From step 2.1c (§5.4) b: 753
$\theta_s(j,f)$	Solar zenith angle for pixel (j,f)	i	deg	from step 2.1a (§3.4)
$\theta_v(j,f)$	Viewing zenith angle for pixel (j,f)	i	deg	idem
$\Delta\phi(j,f)$	Azimuth angle for pixel (j,f)	i	deg	idem
CLOUD_F(j,f)	Flag for cloudy pixels	i	dl	from step 2.1c (§5.4)
INVALID_F(j,f)	Invalid pixel flag	i	dl	from step 2.1a (§3.4)
P_{top} (j,f)	Cloud top pressure	i	hPa	from step 2.1b (§4.4)
α_{surf}	Surface albedo at band 11 for pixel (j,f)	i	dl	from step 2.1b (§4.4)
Calb_LUT [μ_s, μ_v, $\Delta\phi$, α_{surf},k]	LUTs of polynomial coefficients for estimating cloud albedo as a function of geometry and surface albedo	s	dl	μs : 27 values μv : 18 values $\Delta\phi$: 25 values α_{surf} : 9 values k : coefficient : 3 values
Cthick_LUT [μ_s, μ_v, $\Delta\phi$, α_{surf}, k]	LUTs of polynomial coefficients for estimating cloud optical thickness as a function of geometry and surface albedo	s	dl	μ_s, μ_v , $\Delta\phi$, α_{surf} :see above k : coefficient : 4 values
Ctype_n_δ_c	number of optical thickness values for cloud type classification	s	dl	
Ctype_n_P	number of pressure values for cloud type classification	s	dl	
Ctype_δ_c_range [1..Ctype_n_δ]	range of optical thickness values for cloud type classification	s	dl	
Ctype_P_range [1..Ctype_n_P]	range of pressure values for cloud type classification	s	dl	
Ctype_LUT [δ_c,P_{top}]	LUT of cloud type index	s	dl	δ_c in Ctype_δ_range Ptop in Ctype_P_range
F_0^C(b753)	Solar flux consistent with cloud LUTs	s	EU	Only at b753

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
μ_s, μ_v	Cosine of Sun, view zenith angles	c	dl	
a_{al}	Polynomial coefficient for estimating α_c	c	dl	
b_{al}	Polynomial coefficient for estimating α_c	c	LU^{-1}	
c_{al}	Polynomial coefficient for estimating α_c	c	LU^{-2}	
a_{th}	Polynomial coefficient for estimating τ_c	c	dl	
b_{th}	Polynomial coefficient for estimating τ_c	c	LU^{-1}	
c_{th}	Polynomial coefficient for estimating τ_c	c	LU^{-2}	
d_{th}	Polynomial coefficient for estimating τ_c	c	LU^{-3}	
kd, kp	Indices within Ctype_LUT	c	-	
L_T(b753,j,f)	TOA radiance for pixel (j,f),corrected for stratospheric aerosol	c	LU	
τ_c(j,f)	Cloud Optical thickness for pixel (j,f)	o	dl	to steps 2.3 (§6.4), 2.10 (§10.4), to Breakpoint
α_c(j,f)	Cloud albedo for pixel (j,f)	o	dl	to step 2.10 (§10.4), to Breakpoint
Ctype (j, f)	Cloud type index for pixel (j, f)	o	dl	*idem*
ORINP1_F(j,f)	Out of range input flag for pixel (j,f)	o	dl	Boolean, to step 2.10 (§10.4), to Breakpoint
OROUT1_F(j,f)	Out of range output flag for pixel (j,f)	o	dl	*idem*
ORINP2_F(j,f)	Out of range input flag for pixel (j,f)	o	dl	*idem*

NOTE: all calculated and output Boolean parameters shall be initialised to FALSE (0).

7.5. - Equations

For each pixel (j,f) such that (INVALID_F(j,f)== FALSE) **AND** (CLOUD_F(j,f) == TRUE)

Let $\mu_s = \mathbf{cos}\,(\theta_s, j, f)$, $\mu_v = \mathbf{cos}\,(\theta_v, j, f)$;

$$L_T(b753, j, f) = \rho(b753,j,f).\ F_0^C(b753).\mu_s / \pi \qquad (2.4\text{-}1)$$

7.5.1 - Cloud Albedo processing (step 2.4.1)

The polynomial coefficients a_{al}, b_{al}, and c_{al} are obtained by interpolation in the LUTs (RD 7, 3.17.2.3) as a function of the sun zenith angle, viewing angle and azimuth angle and the surface albedo.

a_{al} = Calb_LUT **interpol:**(μ_s, μ_v, $\Delta\phi$, j, f) **nearest**: (α_{surf}, j, f) **select:**(k=1) *(2.4.1-1)*

b_{al} = Calb_LUT **interpol:**(μ_s, μ_v, $\Delta\phi$, j, f) **nearest**: (α_{surf}, j, f) **select:**(k=2) *(2.4.1-2)*

c_{al} = Calb_LUT **interpol:**(μ_s, μ_v, $\Delta\phi$, j, f) **nearest**: (α_{surf}, j, f) **select:**(k=3) *(2.4.1-3)*

exception processing: out of LUT range μ_s, μ_v, $\Delta\phi(\varphi, \phi)$ in equations (2.4.1-1) to (2.4.1-3) above:

use extreme range index value (see section 2.4.3 above)
ORINP1_F (j, f) = TRUE

Copyright © 2011 ACRI S.T.

continue at next equation
end of exception processing

The cloud albedo α_c and the MERIS radiance at 753.75 nm are related using a second order polynomial:

$\alpha_c\,(j, f) = a_{al} + b_{al}.L_T(b753, j, f) + c_{al}.\,(L_T(b753, j, f))^2$ *(2.4.1-4)*

7.5.2. - Cloud Optical Thickness processing (step 2.4.3)

The polynomial coefficients a_{th}, b_{th}, c_{th}, and d_{th} are interpolated from a LUT as a function of μ_s, μ_v, and $\Delta\phi$ and the surface albedo α_{surf}.

a_{th} = Cthick_LUT **interpol:**(μ_s, μ_v, $\Delta\phi$(j, f)) **nearest**: (α_{surf} (j, f)) **select:**(k=1) *(2.4.3-1)*

b_{th} = Cthick_LUT **interpol:**(μ_s, μ_v, $\Delta\phi$(j, f)) **nearest**: (α_{surf} (j, f)) **select:**(k=2) *(2.4.3-2)*

c_{th} = Cthick_LUT **interpol**:(μ_s, μ_v, $\Delta\phi$(j, f)) **nearest**: (α_{surf} (j, f)) **select**:(k=3) *(2.4.3-3)*

d_{th} = Cthick_LUT **interpol**:(μ_s, μ_v, $\Delta\phi$(j, f)) **nearest**: (α_{surf} (j, f)) **select**:(k=4) *(2.4.3-4)*

exception processing: out of LUT range μ_s, μ_v, $\Delta\phi(\varphi, \phi)$ in equations (2.4.3-1) to (2.4.3-4) above:
use extreme range index value (see section 2.4.3 above)
ORINP2_F (j, f) = TRUE
continue at next equation
end of exception processing

The relationship between output τ_c and input L_T is described by the polynomial :

$\tau_c\,(j, f) = \mathbf{exp}\,[a_{th} + b_{th}.\,L_T(b753, j, f)+c_{th}.\,(L_T(b753, j, f))^2+d_{th}.\,(L_T(b753, j, f))^3]$ *(2.4.3-5)*

7.5.3. - Cloud type index processing (step 2.4.8)

A simple cloud type index is computed from the cloud geo-physical products: cloud top pressure, cloud optical thickness.

Find kd such that Ctype_δ_range[kd] <= τ_c (j, f) < Ctype_δ_range[kd+1] *(2.4.8-1)*

Find kp such that Ctype_P_range[kp] <= P_{top} (j, f) < Ctype_P_range[kp+1] *(2.4.8-2)*

exception processing: out of range optical thickness or pressure in eq. 2.4.8-1, 2.4.8-2 above:
use extreme range index
continue at next equation
end of exception processing

Ctype (j,f)= Ctype_LUT[kd][kp] *(2.4.8-3)*

Copyright © 2011 ACRI S.T.

Endfor *End of loop on pixels*

7.6. - Quality Control and Diagnostics.

When any of μ_s, μ_v, $\Delta\phi(j, f)$, α_{surf} (j, f) are outside of the LUTs index range, the nearest LUT values are used as explained in §2 above; the flag ORINP1_F or ORINP2_F is set to TRUE and the pixel is processed.

If α_c (j, f) is found negative, then the flag OROUT1_F is set to TRUE.

7.7. - Exception Handling

See the blocks "exception processing:... end of exception processing" in section 7.5 above.

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

8 - Water Processing

8.1 Overview

The processing of water pixels is intended to provide the following Level 2 products:

a) quantitative
- normalised water-leaving reflectance at bands 412.5, 442.5, 490, 510, 560, 620, 665, 681.25, 705, 753, 775, 865, 885 nm
- algal pigment index 1
- algal pigment index 2
- total suspended matter
- yellow substance absorption at 442.5nm
- photosynthetically available radiance (PAR)
- aerosol optical thickness at 865nm
- aerosol Angström exponent (775, 865)

b) qualitative
- turbid case 2 water;
- yellow substance loaded case 2 water;
- water with excessive scattering;
- continental absorbing aerosol;
- desert dust absorbing aerosol;

as well as flags relevant to the quality of all products.

The block diagram in figure 8.1-1 below shows the general logic of the main processing steps. These steps are detailed in the following sections 8.2: water confidence checks, 8.3: turbid water screening and correction, 8.4: atmosphere correction, 8.5: ocean colour processing, 8.6 Case 2 R processing.

Copyright © 2011 ACRI S.T.

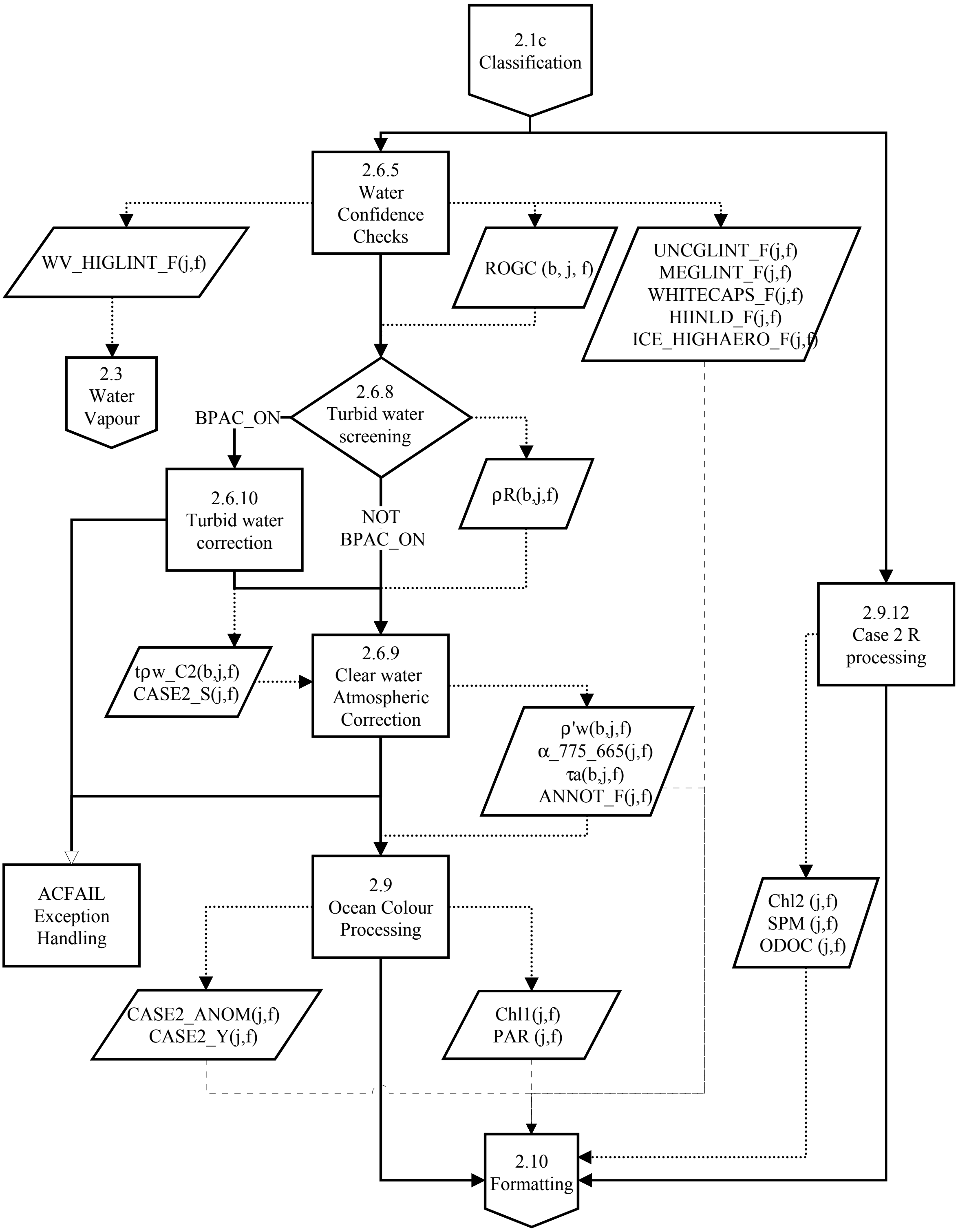


Figure 8.1-1: Water Processing Overall Block Diagram

8.2. - Water Confidence Checks (step 2.6.5)

8.2.1 - Mathematical Description of the Algorithm

Water confidence checks include processing of Sun glint, flagging of low pressure and whitecaps and vicarious adjustment. The first steps use the wind and pressure provided in the L1B product annotations and interpolated at pixel. The vicarious gains are applied at the end of these steps.

8.2.1.1 - Glint processing (step 2.6.5.1)

Glint processing applies only to water pixels. It is described in RD 8, 2.13.

- *Glint estimation*

The Sun glint reflectance ρ_g is calculated (step 2.6.5.1.1) by interpolation in LUT produced using the Cox and Munk model (1954) as a function of geometry, wind speed modulus and direction. An estimate of glint reflectance is produced. Wind speed modulus and wind direction are computed from the W_u and W_v annotations.
Note: Sun glint reflectance is now an input to current step as its computation a been moved to section 5.

- *Glint classification (low, medium or high glint ?)*

This glint reflectance is compared to a low glint threshold (step 2.6.5.1.3). If the glint reflectance is below this low glint threshold then no glint correction for this pixel is applied. If the pixel is not bright then it is further processed by step 2.6.5.3. If bright, it is flagged as ice or high aerosol load.

If the glint reflectance is above the low glint threshold then the glint reflectance is compared to a medium glint threshold (step 2.6.5.1.4).

If the glint reflectance is below the medium glint threshold then a medium glint flag is raised and the pixel is corrected for glint reflectance in step 2.6.5.1.7.

If the glint reflectance is above the medium glint threshold then no correction is applied and the pixel is flagged as uncorrected sun glint (step 2.6.5.1.5).

- *Glint correction in case of medium glint (step 2.6.5.1.7)*

The glint reflectance at surface level is transferred to the Top Of Atmosphere by applying a direct atmospheric transmission term (including Rayleigh scattering on both sun-surface and surface sensor paths). The Top Of Atmosphere glint reflectance ρ_g* is then subtracted from the TOA reflectance.

Note : step 2.6.5.1.8, because of its simplicity, is only shown in the corresponding equation section.

8.2.1.2 – Low pressure water flagging (step 2.6.5.2)

Surface pressure (from ECMWF annotation) is compared with a threshold in order to flag low pressure, typically high altitude inland waters. Above such waters, the results of the atmosphere correction are disturbed.

Copyright © 2011 ACRI S.T.

8.2.1.3 – Whitecaps Flagging (step 2.6.5.3)

The modulus of the wind speed is compared to a threshold. Above that threshold a whitecap flag is raised because white caps are likely to disturb the performance of the atmosphere correction (RD 8, 2.14)

8.2.1.3 – Reflectance threshold on reflectance at 412 nm (step 2.6.5.4)

An additional “bright” pixels screening, specific to water pixels, is performed by comparison of the Glint corrected reflectance at 412 nm with a pre-computed threshold. It is intended to further sort out and flag those pixels affected by sea-ice, partial clouds or very high aerosol load.

8.2.1.4 – Vicarious adjustment (step 2.6.5.5)

The glint corrected TOA signal is finally adjusted by a multiplicative gain (band per band) before entering the atmospheric corrections steps, i.e. before BPAC and Case 1 atmospheric corrections. Those pre-computed gains aims at removing any bias in the {instrument + processing} chain in order to optimally retrieve water-leaving reflectance.

Copyright © 2011 ACRI S.T.

Figure 8.1-1 and 8.1-2 below show the block diagrams for water confidence checks and glint processing.

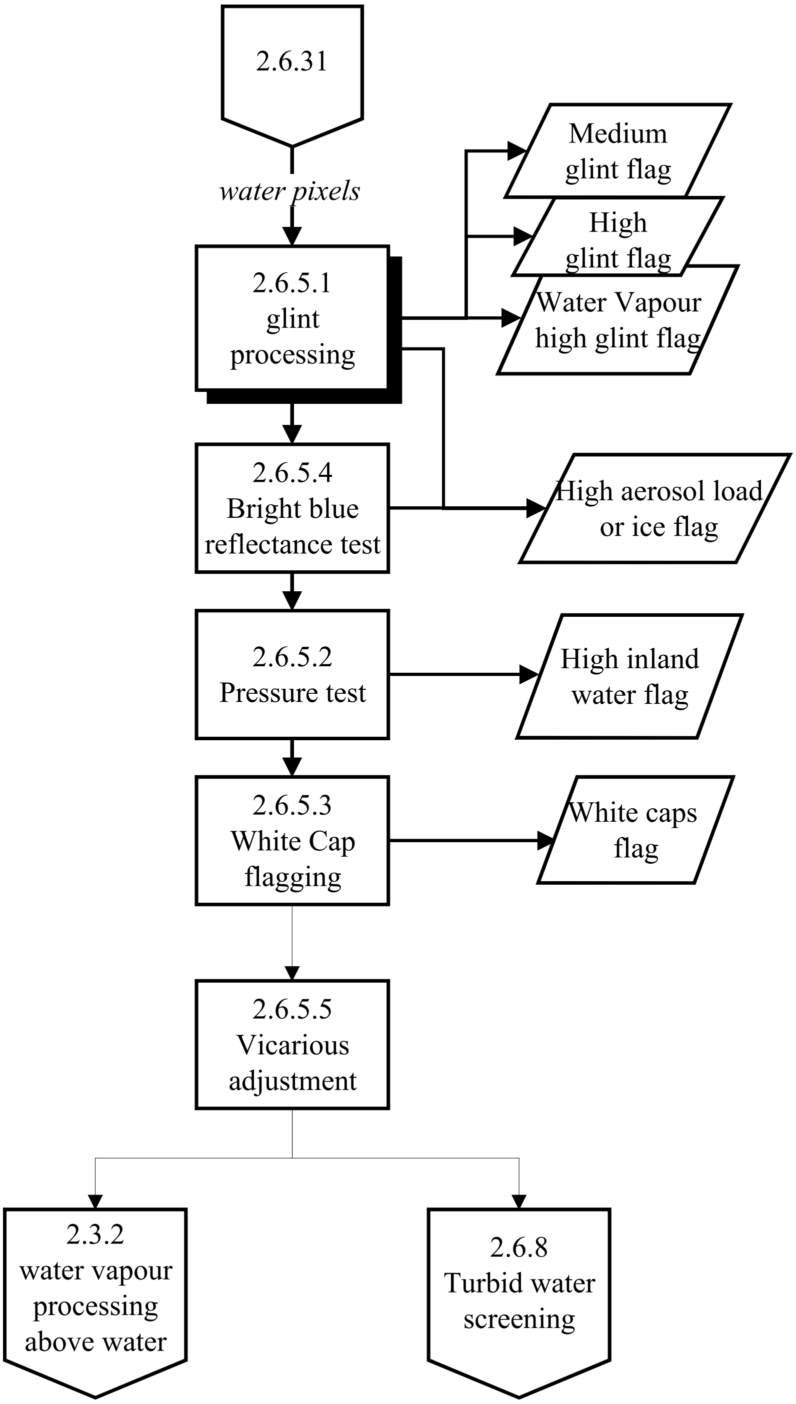


Figure 8.2-1 : Step 2.6.5 Water confidence checks

Copyright © 2011 ACRI S.T.

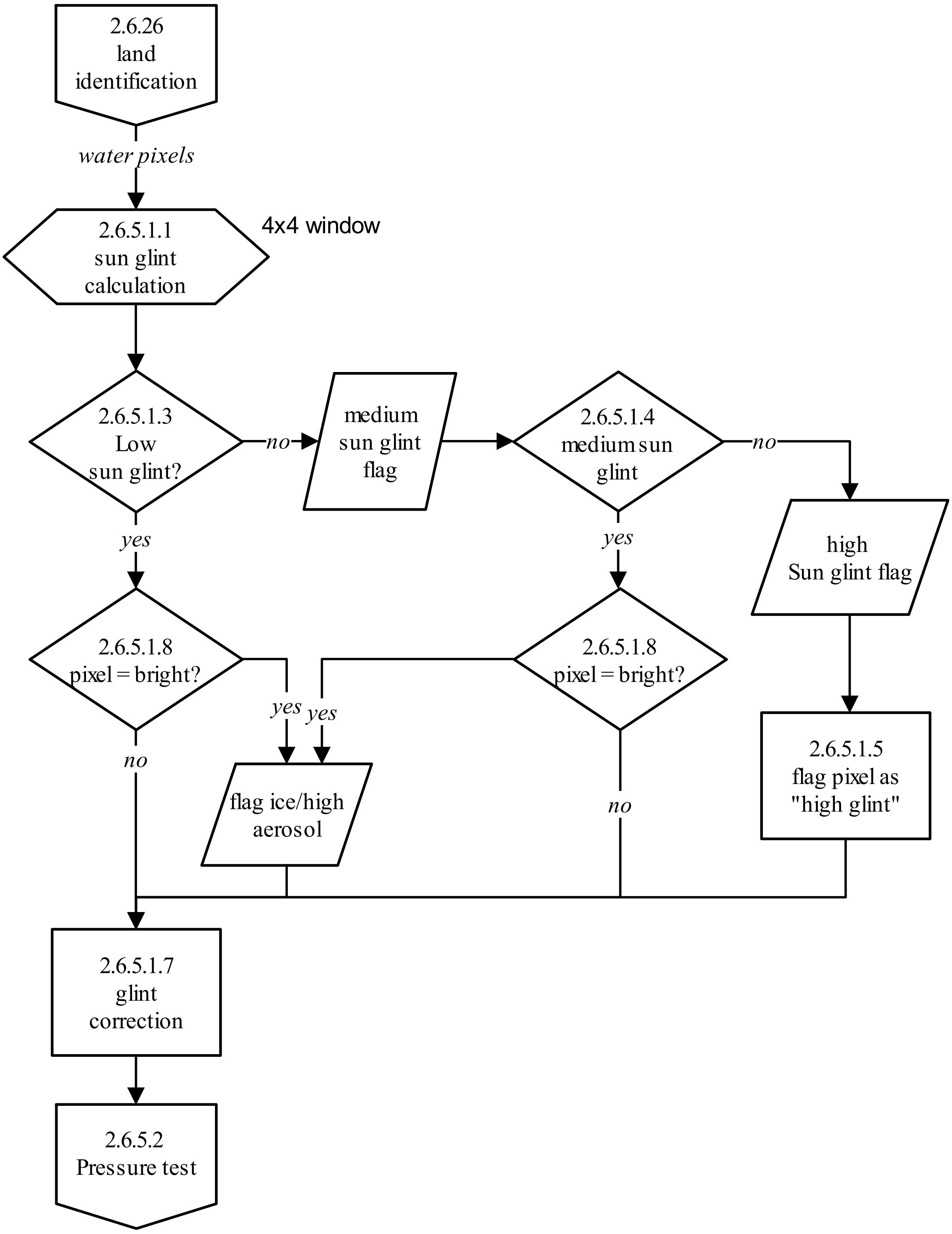


Figure 8.2-2 : Step 2.6.5.1 Glint processing

Copyright © 2011 ACRI S.T.

8.2.2 - List of variables

Variable	Descriptive Name	T	U	Range - References
INVALID_F(j,f)	Invalid pixel flag	i	-	from step 2.1a (§3.4)
LANDCONS_F(j,f)	Land/water consolidated flag	i	dl	Boolean, from step 2.1c (§5.4)
BRIGHT_F(j,f)	Bright pixel flag	i	dl	from step 2.1c (§5.4)
CLOUD_F(j,f)	Cloud flag	i	-	idem
θ_s(j,f)	Sun zenith angle	i	deg	from step 2.1a (§3.4)
θ_v(j,f)	Viewing zenith angle	i	deg	idem
$\Delta\phi$(j,f)	Difference of azimuth angles	i	deg	idem
ϕ_s(j,f)	Sun azimuth for pixel (j,f)	i	deg	idem
Ws (j,f)	wind speed modulus for pixel (j,f)	i	$m.s^{-1}$	from step 2.6.26 (§5.5)
P_{ECMWF}(j, f)	ECMWF surface pressure	i	hPa	idem
ρ^*_{ng}(b,j,f)	TOA reflectance, corrected for stratospheric aerosol contribution, gaseous absorption and smile	i	dl	for water pixels; b in {b412..b885}, from step 2.1c (§5.4)
ROG(j,f)	Sun glint reflectance for pixel (j,f)	i	dl	from step 2.6.26 (§5.5)
SNOW_ICE_F(j,f)	Snow or Ice flag	i	dl	Boolean, from steps 2.1c (§ 5.4)
τ_{R0}(b,j,f)	Rayleigh optical thickness at nominal wavelengths, corrected for pressure	i	dl	from step 2.1c (§ 5.4)
gain_vicarious(b)	Vicarious adjustment gains	s	dl	
P_{thresh}	Threshold value for low pressure water	s	hPa	
thres_lowg	Threshold value for low glint	s	dl	
thres_medg	Upper threshold value for ratio between glint and TOA reflectance	s	dl	
Thres_WVhg	Water vapour high glint threshold	s	dl	
Thresh_BlueROGC_LUT[θ_s, θ_v, $\Delta\phi$]	LUT giving glint corrected reflectance threshold at 412 nm	s	dl	
P_{std}	Standard pressure	s	hPa	1013.25 hPa
WHITECAP_THR	wind speed threshold for whitecaps	s	$m.s^{-1}$	
M	Air mass for current pixel	c	dl	

Table 8.2.2-1: List of variables for water confidence checks

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
tdir	Direct atmospheric transmission	c	dl	
TOAROG(b)	Sun glint reflectance at TOA	c	dl	to Breakpoint
Thresh_BlueROGC	Threshold on glint corrected reflectance at 412 nm for current pixel	c	dl	
ROGC (b,j,f)	TOA reflectance, corrected for stratospheric aerosol contribution, gaseous absorption, Sun glint	o	dl	for water pixels; b in {b412..b885}, to step 2.6.9 (§8.4.2), to Breakpoint
HIINLD_F(j,f)	Flag for low pressure water	o	dl	Boolean, to step 2.10 (§10.4), to Breakpoint
ICE_HIGHAERO_F(j,f)	Flag for ice or high aerosol loading pixels	o	dl	Boolean, to step 2.3 (§6.4), 2.10 (§10.4), to Breakpoint
MEGLINT_F(j,f)	Flag for pixels corrected for glint	o	dl	Boolean, to step 2.10 (§10.4), to Breakpoint
UNCGLINT_F(j,f)	Flag for pixels contaminated by glint	o	dl	Boolean, to step 2.10 (§10.4), to Breakpoint
WHITECAPS_F(j,f)	Whitecaps flag	o	dl	Boolean, to step 2.10 (§10.4), to Breakpoint

Table 8.2.2-1: List of variables for water confidence checks (cont)

NOTE: all calculated and output Boolean parameters shall be initialised to FALSE (0).

8.2.3 - Equations

For each pixel (j,f) such that (INVALID_F(j, f)== FALSE) **AND** (CLOUD_F(j, f)==FALSE) **AND** (LANDCONS_F(j,f) == FALSE)
(2.6.5-1)

deleted *(2.6.12.1-3), (2.6.12.1-4)*
deleted *(2.6.12.1-5), (2.6.12.1-7), (2.6.12.1-8)*
deleted *(2.6.12.1-9)*

$$M = 1/\cos(\theta_s (j, f)) + 1/\cos(\theta_v (j, f)) \quad (2.6.5\text{-}2)$$

8.2.3.1. - Flag Sun glint on water pixels only (step 2.6.5.1)

deleted (step 2.6.5.1.1)

Compare Sun glint reflectance to low glint threshold (step 2.6.5.1.3)

```
If (ROG(j,f) < thres_lowg) then
    MEGLINT_F(j,f) = FALSE                    (2.6.5.1.3-1)
    UNCGLINT_F(j,f) = FALSE
(2.6.5.1.3-2)
```

```
        ICE_HIGHAERO_F(j,f) = BRIGHT_F(j,f) OR SNOW_ICE_F(j,f)      (2.6.5.1.8-1)
```

Compare sun glint reflectance to low glint threshold and medium glint threshold (step 2.6.5.1.4)

```
    else if (ROG(j,f) >= thres_lowg) and (ROG(j,f) < thres_medg . $\rho^*_{ng}$(b865, j, f)) then
        MEGLINT_F(j,f) = TRUE                                       (2.6.5.1.4-1)
        UNCGLINT_F(j,f) = FALSE
    (2.6.5.1.4-2)
        ICE_HIGHAERO_F(j,f) = BRIGHT_F(j,f) OR SNOW_ICE_F(j,f)      (2.6.5.1.8-2)
    else
```

Flag pixels with high glint as uncorrected glint pixels (step 2.6.5.1.5)

```
        MEGLINT_F(j,f) = TRUE                                       (2.6.5.1.5-1)
        UNCGLINT_F(j,f) = TRUE                                      (2.6.5.1.5-2)
    Endif
```

Apply glint correction in case of medium glint (step 2.6.5.1.7)

```
    If ( MEGLINT_F(j,f) == TRUE ) then
        For each band b in {b412..b753, b775..b885}
  computation of direct atmospheric transmission on sun-surface and surface-sensor paths
            $tdir=e^{-(\tau R0(b).M)}$                               (2.6.5.1.7-1)
  computation of Top Of Atmosphere glint reflectance
            TOAROG(b) = tdir.ROG(j,f)                               (2.6.5.1.7-2)

  glint correction
            ROGC(b,j,f) = $\rho^*_{ng}$ (b,j,f) - TOAROG(b)            (2.6.5.1.7-3)
```

exception processing: ROGC(b,j,f) <= 0 in (2.6.5.1.7-3) :
MEGLINT (j, f) = FALSE
UNCGLINT (j, f) = TRUE
jump to loop (2.6.5.1.7-4) starting at b412
end of exception processing

```
        endfor      End of loop on bands
    Else
  no glint correction
        For each band b in {b412..b753, b775..b885}
            ROGC(b,j,f) = $\rho^*_{ng}$ (b,j,f)                      (2.6.5.1.7-4)
        Endfor End of loop over bands
    Endif
```

Deleted:* step 2.6.5.1.9 *(moved to section 6)

8.2.3.2. -Test ECMWF pressure (step 2.6.5.2)

$$HIINLD\_F(j,f) = (P_{ECMWF}(j,f) < P_{thresh}) \qquad (2.6.5.2\text{-}1)$$

8.2.3.3. - Whitecaps identification (step 2.6.5.3)

```
    compare wind speed modulus to wind threshold for appearance of whitecaps
    $WHITECAPS\_F(j,f) = ( W_s(j,f) \geq WHITECAP\_THR )$        (2.6.5.3-1)
        deleted                                                  (2.6.5.3-2)
        deleted                                                  (2.6.5.3-3)
```

8.2.3.4. -Compare glint corrected reflectance in the blue to a threshold (step 2.6.5.4)

$$Thresh\_BlueROGC = Thresh\_BlueROGC\ LUT\ \textbf{interpol:}\ (\theta_s, \theta_v, \Delta\phi) \qquad (2.6.5.4\text{-}1)$$

exception processing: out of index range θ_s, θ_v, $\Delta\phi$, in (2.6.5.4-1) above:
continue processing at next equation
end of exception processing

```
    if    (ROGC(b412,j,f) >= Thresh_BlueROGC)   then
          ICE_HIGHAERO_F(j,f) = TRUE                             (2.6.5.4-2)
    endif
```

8.2.3.5. -Vicarious adjustment (step 2.6.5.5)

```
Apply the adjustment factors                                     (2.6.5.5-1)
    For each band b in {b412..b753, b775..b885}
        ROGC(b,j,f)=ROGC(b,j,f)*gain_vicarious(b)
    Endfor End of loop over bands

Endfor End of loop on valid water pixels
```

8.2.4 - Quality control and Diagnostics

The flag MEGLINT_F (j, f) flags pixels contaminated by Sun glint (according to wind knowledge from ECMWF) and corrected.

The flag UNGLINT_F (j, f) flags pixels contaminated by High Sun glint for which quality of the correction is lower.

The flag ICE_HIGHAERO_F (j, f) flags bright water pixels not contaminated by Sun glint.

8.2.5 - Exception handling

See blocks "exception processing... end of exception processing" in section 8.2.3 above.

Copyright © 2011 ACRI S.T.

8.3 Turbid water screening and corrections (steps 2.6.8.1 to 2.6.8.8)

This section describes the algorithms used

1. in steps 2.6.8.2 to 2.6.8.5, to detect Case 2 turbid waters based on radiometry (reflectance corrected for stratospheric aerosol, gaseous absorption, Sun glint, Rayleigh), initiate the first iterate of NIR water-leaving reflectance and choose the appropriate band set for further steps;
2. in step 2.6.8.6 to 2.6.8.8, to compute the water-leaving reflectance for Case 2 turbid waters at b705, b775, b865 and b885, needed before entering the atmospheric corrections processing over Case 1 waters (step 2.6.9) and provide an estimate of the total suspended matter used in turn to identify sediment dominated case 2 waters through a dedicated flag.

8.3.1 - Mathematical description

8.3.1.1 - Water identification and initial estimate

8.3.1.1.1 Coarse Rayleigh correction and diffuse transmittance computation (step 2.6.8.2)

The Rayleigh reflectance corrected for pressure is removed to the rho_gc signal to compute rho_rc used further in the algorithm. The diffused transmittance is approximated taking only into account the Rayleigh contribution.

8.3.1.1.2 - White scatterer identification (step 2.6.8.3)

An estimate of the spectral slope of marine backscatter is computed using Rayleigh corrected reflectance and pure water specific absorption. This estimated spectral slope is compared to a threshold below which the White Scatterer Flag is raised. The IOPS (sediment backscatter slope and absorption) are chosen according to the white scattering flag for use in subsequent calculations.

8.3.1.1.3 - Turbid water identification (steps 2.6.8.4 and 2.6.8.5)

The turbid water identification is described in RD 8, 2.5.

If one of the Rayleigh corrected TOA reflectances at b705 or b865 is negative, then it is assumed that the ocean reflectance is insufficient to calculate the water vapour above ocean product correctly and no further computations are required by the turbid water reflectance (note the flag `F_ORINPWV` can also be used from wvo products). The BPAC flag is set to FALSE and the algorithm stops, returning the value of pure water.

Otherwise, the marine reflectances are computed at b705, b775, b865 and b885 using the Angström exponent method from Rayleigh corrected reflectances at two band sets: {b705, b775} and {b865, b885}. Then the TOA marine reflectance at b775 derived from the second bandset is compared to thresholds and raises flags to indicate whether the BPAC iterative calculations should be carried out on either or both of the {b705, b775, b865} and {b775, b865, b885} band set. Eventually, in case of the first bandset is activated, the marine reflectance at b705 is compated to a threshold determined by the mean gain and solar flux at b705. If it is below the threshold then BPAC flag is set to FALSE and algorithm stops, returning the value of pure water.

Copyright © 2011 ACRI S.T.

8.3.1.2 Turbid water correction (step 2.6.8.6 to 2.6.8.)

When a water pixel has been detected as contaminated by a water signal in the infra-red by test 2.6.8.1, the algorithm called bright pixel procedure performs an estimate of the water-leaving reflectance at four bands used later by the atmosphere corrections above water (see 8.4 below). The algorithm is based on optical properties of the water and performs an iterative procedure with a combination of:

- single scattering aerosol reflectance;
- water-leaving reflectance;

The iterative procedure is preformed on either or both of the {b705, b775, b865} and {b775, b865, b885} bandsets. The former bandset applies to low to medium turbidity waters and the latter bandset applies to highly turbid waters. There is an overlap in the applicability of the bandsets and where both bandsets are used, then a simple average of the results are used to avoid image artifacts.

Note that in condition of non-corrected glint (i.e. flag HIGH_GLINT and not MEDIUM_GLINT), only the low bandset is used, in order to retrieve proper SPM and activate correctly the CASE2_S flag.

The atmospheric attenuation of water-leaving reflectance is taken into account.

The logic of the turbid waters correction is shown in block diagram below:

Copyright © 2011 ACRI S.T.

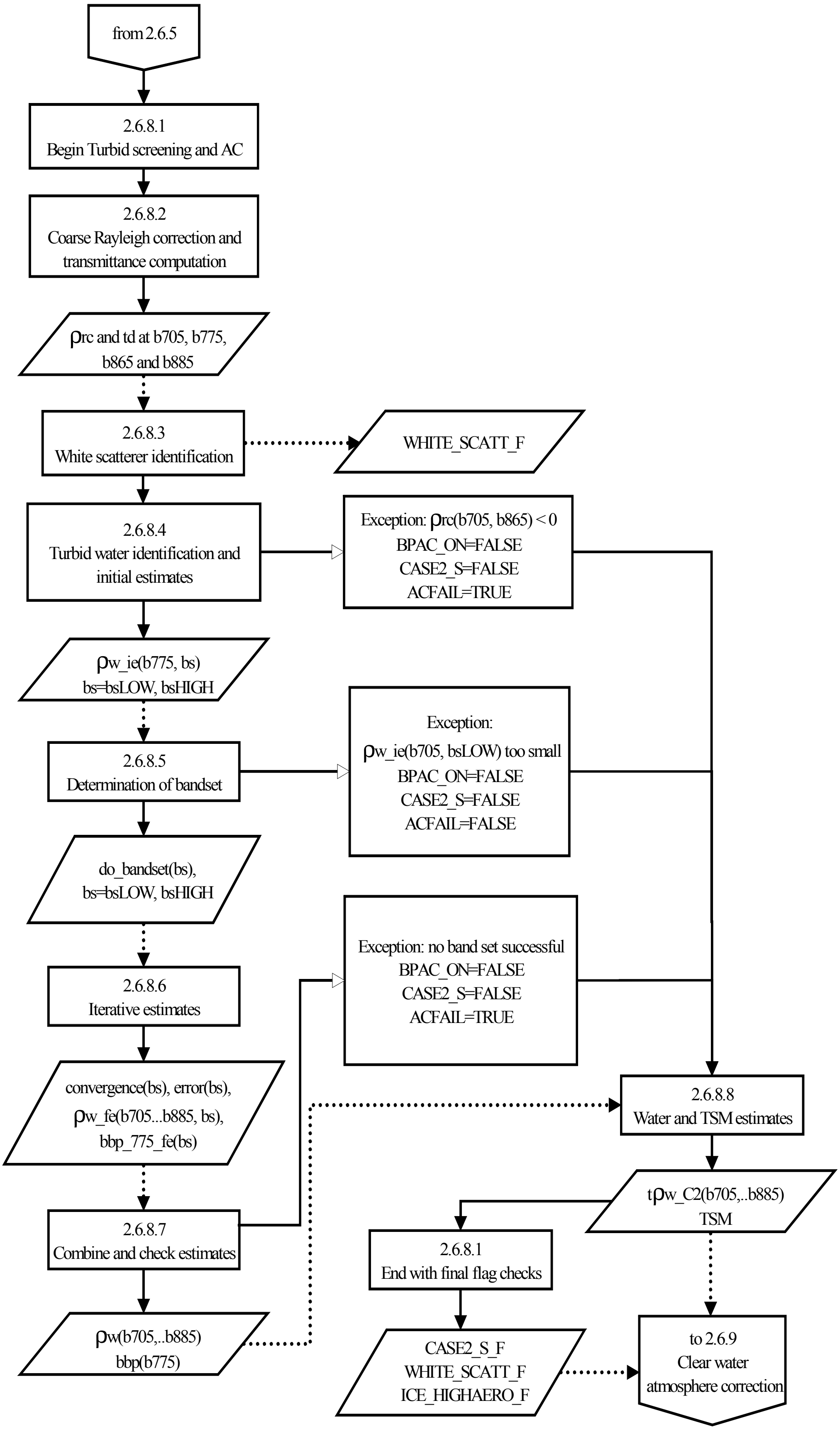


Figure 8.3.1.3 : Step 2.6.8 Turbid water screening and atmospheric corrections

Copyright © 2011 ACRI S.T.

8.3.2 - List of variables

Variable	Descriptive name	T	U	Range - References
ROGC(b,j,f)	TOA reflectance corrected for strato.aerosol, gaseous absorption, sun glint	i	dl	from step 2.6.5 (§8.2.2) ; b: {b412-b705, b775, b865, b885}
$\Delta\phi$(j,f)	Difference of azimuth angles	i	deg	from step 2.1a (§3.4)
θ_S(j,f)	Sun zenith angle	i	deg	idem
θ_V(j,f)	Satellite viewing angle	i	deg	idem
μ_S(j,f)	Cosine of Sun zenith angle	i	dl	idem
μ_V(j,f)	Cosine of satellite viewing angle	i	dl	idem
z(j,f)	Pixel altitude	i	m	idem
W_s(j,f)	Wind speed modulus	i	m.s^{-1}	from step 2.6.5 (§8.2.2)
P_{ECMWF}(j,f)	ECMWF pressure at surface	i	hPa	from step 2.1a (§3.4)
ρ_{R0}(b,j,f)	Rayleigh reflectance corrected for pressure variations, for pixel (j,f)	i	dl	b in {b412…b885} from step 2.1.c (§5.5)
τ_{R0}(b,j,f)	Molecular optical thickness corrected for actual pressure , for pixel (j,f)	i	dl	b in {b412… b885} from step 2.1.c (§5.5)
INVALID_F(j,f)	Invalid pixel flag	i	-	idem
CLOUD_F(j,f)	Cloudy pixel flag	i	-	from step 2.1c (§5.4)
LANDCONS_F(j,f)	Land pixel flag	i	-	idem
ICE_HIGHAERO	Ice or high aerosol load flag	i/o	-	
aw(b)	Absorption of pure water	s	m^{-1}	b in {b620, b705, b775, b865, b885}
bbp_star_c(b)	specific back scattering of sediment, case of coccoliths	s	m^2.g^{-1}	b in {b705, b775, b865, b885}
bbp_star_p(b)	specific back scattering of sediment, case of particulates	s	m^2.g^{-1}	*idem*
bbw(b)	back scattering of pure water	s	m^{-1}	*idem*
a_to_bb_c(b)	specific absorption, case of coccoliths	s	m^2.g^{-1}	*idem*
a_to_bb_p(b)	specific absorption, case of particulate	s	m^2.g^{-1}	*idem*
bb_775_ie(bs)	Initial estimate of backscatter at 775 for LOW and HIGH band estimate	s	m^{-1}	bs in {bLOW, bHIGH}
init_ang	Initial estimate of the Angström exponent	s	dl	
NIterBPAC(bs)	Number of iterations in BPAC for LOW and HIGH bandset	s	-	bs in {bLOW, bHIGH}
bbp_tol	Convergence criteria on bbp in the BPAC iterations	s	dl	

Copyright © 2011 ACRI S.T.

Variable	Descriptive name	T	U	Range - References
Fp_LUT[θ_S, θ_V, $\Delta\phi$, Ws, coeff, b]	LUT of coefficients of F' to IOPs relation	s	dl	
λ_{theo}(b)	Theoretical wavelengths corresponding to smile corrected reflectances	s	nm	b in {b412..b900}
BAD_VALUE	output value when algorithm fails	s	dl	see §2
SPM_Case2_Thresh	Threshold on SPM concentration to identify sediment dominated waters	s	$g.m^{-3}$	
Ln_min_705	Minimum normalised radiance measurable by MERIS at b705	s	dl	
row_775_do_both_th	Threshold on rhow at 775 to activate the HIGH bandset	s	dl	
row_775_do_high_th	Threshold on rhow at 775 to desactivate the LOW bandset	s	dl	
α_Scatt_Threshold	Threshold on marine basckscatter spectral slope estimate	s	dl	
rhow_bb_tol	Convergence criteria on bb in the rhow_to_bb routine	s	dl	
Niter_rhow_bb	Number of iterations in the rhow_to_bb routine	s	-	
bbp_init	Initial value of bbp to initialise rhow_bb_routine	s	m^{-1}	
b1, b2, b3, b4	local index of bands in the initial and iterative estimates, changing wrt the band set.	c	-	
ρ_{rc}(b,j,f)	TOA reflectance corrected for Rayleigh scattering and sun glint	c	dl	b in { b705, b775, b865, b885}
t_d(b)	Diffuse transmittance for current pixel	c	dl	b in {b705, b775, b865, b885}
bs	index of the bandset	c	-	bLOW or bHIGH
rho_min_705	Minimum reflectance measurable by MERIS at b705	c	dl	
Counter	counter of the iterative loop	c	-	[0, NiterBPAC(bs)-1]
ρ_w_ie(b,bs)	Initial reflectance estimate at band b for bandset bs	c	dl	b in {b705, b775, b865, b885}, bs in {bLOW, bHIGH}
ρ_a(b)	Aerosol reflectance	c	dl	b in { b705, b775, b865, b885}
ρ_w(b)	Estimate of ρ_w in iterative procedure	c	dl	b in {b705, b775, b865, b885}
ρ_w_fe(b,bs)	Final reflectance estimate at band b for bandset bs	c	dl	*idem* for b, bs in {bLOW, bHIGH}; to Breakpoint for b775.
bbp_775_fe(bs)	final bbp estimate at b775 for bandset bs	c	m^{-1}	bs in {bLOW, bHIGH}; to Breakpoint
ang_exp(bs)	Angström exponent in loop for bandset bs	c	dl	bs in {bLOW, bHIGH}; to Breakpoint at the end

Copyright © 2011 ACRI S.T.

Variable	Descriptive name	T	U	Range - References
				of loop
do_bandset(bs)	Flag to activate a bandset	c	-	bs in {bLOW, bHIGH}
error(bs)	flag to check errors in the iterative estimate	c	-	*idem*
converge(bs)	flag to check convergence of the iterative estimate	c	-	*idem*
a_to_bb(b)	Specific absorption (chosen)	c	dl	b in {b705, b775, b865, b885}
bbp_star(b)	Specific backscattering sediment (chosen)	c	$m^2.g^{-1}$	*idem*
Bbp	back scattering of sediment	c	m^{-1}	*idem*
bbp_old	back sediment of sediment at band b1 to check convergence			
`kw_705_775`	Ratio of water reflectance at top of atmosphere in iterative procedure	c	dl	
`ka_705_775`	Ratio of aerosol reflectance in iterative procedure	c	dl	
SPM_{br}(j, f)	Sediment load retrieved by the bright pixel method	c	$g.m^{-3}$	RD8 2.6, section 3.1.2, to Breakpoint
α_{water}	Estimate of spectral slope of water backscatter	c	m^{-1}	
$t\rho_w$_C2 (b,j,f)	TOA water-leaving reflectance	o	dl	b in {b705, b775, b865, b885}; to step 2.6.9 (§8.4.2), to Breakpoint
BPAC_ON_F	Flag triggering Bright Pixels turbid water atmosphere correction	o	-	Boolean, to 2.6.9 (§8.4.2), 2.10 (§10.4)
CASE2_S(j,f)	Flag identifying Case 2 sediment dominated waters	o	-	Boolean, to 2.6.9 (§8.4.2), 2.10 (§10.4), to Breakpoint
WHITE_SCATT_F	Flag identifying “white” scatter within water	o	-	Boolean, to 2.6.9 (§8.4.2), 2.10 (§10.4), to Breakpoint
ACFAIL_F (j,f)	Flag indicating failure of the bright pixel correction procedure	o	dl	to step 2.9 (§8.5.4), 2.10 (§10.4), to Breakpoint
ANNOT_BPAC(j,f)	Annotation flag for the quality of the BPAC	o	-	Coding: see 8.3.4; to Breakpoint

Copyright © 2011 ACRI S.T.

8.3.3 - Detailed Algorithm Specification

Note that the subscripts (j,f) may be omitted for clarity, all equations pertain to pixel(j,f).

8.3.3.1 Turbid pixel screening and atmospheric correction (step 2.6.8.1)

```
For each pixel (j,f) such that
(INVALID_F == FALSE) AND (CLOUD_F == FALSE) AND (LANDCONS_F == FALSE)
                                                                    (2.6.8.1-1)
    BPAC_ON_F(j, f) = TRUE                                          (2.6.8.1-1b)
    activate Water identification and initial estimates (steps 2.6.8.2 to 2.6.8.5)
    If (BPAC_ON_F (j, f)) then
        activate Iterative estimate of Angström exponent, IOPs and water leaving
        reflectances (steps 2.6.8.6 to 2.6.8.7)
        activate Estimate of TOA marine reflectances and TSM (step 2.6.8.8)
        Raise CASE2_S flag wrt TSM concentration
        CASE2_S(j,f) = ($SPM_{br}$(j, f)>SPM_Case2_Thresh)          (2.6.8.1-2)
    End if
    Check again WHITE_SCATT flag once BPAC has run
    WHITE_SCATT_F(j,f) = WHITE_SCATT_F(j,f) AND BPAC_ON_F(j,f)
                                                                    (2.6.8.1-3)
    Update ICE_HIGHAERO flag
    If (WHITE_SCATT_F(j,f) AND ICE_HIGHAERO_F(j,f)) then
        ICE_HIGHAERO_F(j,f) = FALSE                                 (2.6.8.1-4)
   Endif

   Transfer BPAC flags to annotation
   ANNOT_BPAC(j,f)=0                                                (2.6.8.1-5)
   If do_bandset(bsLOW) then
        set bit DO_BANDSET_LOW of ANNOT_BPAC(j,f)                   (2.6.8.1-6)
        If converge(bsLOW) then
            set bit CONVERGE_LOW of ANNOT_BPAC(j,f)                 (2.6.8.1-7)
        If error(bsLOW) then
            set bit ERROR_LOW of ANNOT_BPAC(j,f)                    (2.6.8.1-8)
   Endif
   If do_bandset(bsHIGH) then
        set bit DO_BANDSET_HIGH of ANNOT_BPAC(j,f)                  (2.6.8.1-9)
        If converge(bsHIGH) then
            set bit CONVERGE_HIGH of ANNOT_BPAC(j,f)                (2.6.8.1-10)
        If error(bsHIGH) then
            set bit ERROR_HIGH of ANNOT_BPAC(j,f)                   (2.6.8.1-11)
   endif
End for
```

8.3.3.2 Water identification and initial estimate (steps 2.6.8.2 to 2.6.8.5)

```
Coarse Rayleigh correction and diffuse transmittance computation (step 2.6.8.2)
For each band b in {b620, b705, b775, b865, b885}
    $\rho_{rc}(b,j,f) = ROGC(b,j,f) - \rho_{R0}(b,j,f)$                    (2.6.8.2-1)
                        (2.6.8.2-2)
    $t_d(b) = \exp\left[-\frac{\tau_{R0}(b)}{2}\cdot\left(\frac{1}{\mu_s(j,f)}+\frac{1}{\mu_v(j,f)}\right)\right]$    (2.6.8.2-2)
End for End of loop over bands
```

```
White Scatterer identification (step 2.6.8.3)
If ($\rho_{rc}$(b620, j, f) <= 0 OR $\rho_{rc}$(b705, j, f) <= 0) then
    WHITE_SCATT_F(j,f) = FALSE                                      (2.6.8.3-1)
Else
    $\alpha_{water} = \frac{\log\left(\frac{\rho_{rc}(b620,j,f)\cdot a_{w\_620} / t_d(b620)}{\rho_{rc}(b705,j,f)\cdot a_{w\_705} / t_d(b705)}\right)}{\log\left(\lambda_{theo}(b620) / \lambda_{theo}(b705)\right)}$    (2.6.8.3-2)
    WHITE_SCATT_F(j,f) = ($\alpha_{water}$ < α_Scatt_Threshold) AND BPAC_ON_F(j,f)
                                                                    (2.6.8.3-3)
Endif
```

```
Initialise do_bandset, error and converge
For each bs in {bsLOW, bsHIGH}
    do_bandset(bs) = error(bs) = converge(bs) = FALSE               (2.6.8.4-0)
Turbid water identification and initial estimates (step 2.6.8.4)
If ($\rho_{rc}$ (b865, j, f) <= 0) OR ($\rho_{rc}$ (b705, j, f) <= 0) then
    BPAC_ON_F(j, f) = FALSE                                         (2.6.8.4-1)
    CASE2_S (j, f) = FALSE                                          (2.6.8.4-2)
    ACFAIL_F (j,f)=TRUE                                             (2.6.8.4-3)
    continue to step 2.6.8.8 – return clear water reflectances      (2.6.8.4-4)
Endif

Adjust IOPs according to sediment type (coccoliths or particles)
If WHITE_SCATT_F(j,f) then
    For each b in {b705, b775, b865, b885} do
        bbp_star(b)=bbp_star_c(b)                                   (2.6.8.4-5)
        a_to_bb(b)=a_to_bb_c(b)                                     (2.6.8.4-6)
    Endfor
Else
    For each b in {b705, b775, b865, b885} do
        bbp_star(b)=bbp_star_p(b)                                   (2.6.8.4-7)
        a_to_bb(b)=a_to_bb_p(b)                                     (2.6.8.4-8)
    Endfor
```

Copyright © 2011 ACRI S.T.

```
Endif

First estimates for both band sets (loop)
For each bs in {bsLOW, bsHIGH}

    If bs=bsLOW then
        b1=b705    b2=b775    b3=b865    b4=b885                                    (2.6.8.4-9)
    else
        b1=b865    b2=b885    b3=b775    b4=b705                                    (2.6.8.4-10)
    endif

    Calculate IOPs
    bbp(b1)=bb_775_ie(bs)*bbp_star(b1)/bbp_star(b775)                               (2.6.8.4-11)
    bbp(b2)=bb_775_ie(bs)*bbp_star(b2)/bbp_star(b775)                               (2.6.8.4-12)
    a(b1) =aw(b1) +bbp(b1)*a_to_bb(b1)                                              (2.6.8.4-13)
    a(b2) =aw(b2) +bbp(b2)*a_to_bb(b2)                                              (2.6.8.4-14)

    Calculate F prime factor
    fp(b1)=F_ab(a(b1), bbw(b1), bbp(b1), $\theta_S$, $\theta_V$, $\Delta\phi$, Ws, b1)   (2.6.8.4-15)
    fp(b2)=F_ab(a(b2), bbw(b2), bbp(b2), $\theta_S$, $\theta_V$, $\Delta\phi$, Ws, b2)   (2.6.8.4-16)

    Calculate reflectances from bb estimate
    $\rho_w$(b1)=fp(b1) *(bbp(b1)+bbw(b1))/(a(b1)+bbp(b1)+bbw(b1))                  (2.6.8.4-17)
    $\rho_w$(b2)=fp(b2) *(bbp(b2)+bbw(b2))/(a(b2)+bbp(b2)+bbw(b2))                  (2.6.8.4-18)

    Calculate slopes of aerosol and toa water reflectances
    kw_b1_b2=($\rho_w$(b2) * $t_d$(b2)) / (($\rho_w$(b1) * $t_d$(b1))                (2.6.8.4-19)
    ka_b1_b2=$(\lambda_{theo}(b2)/ \lambda_{theo}(b1))^{init\_ang}$                   (2.6.8.4-20)

    Estimate aerosol reflectance $\rho_a$(b1), using ratios
    status=two_band_rhoa($\rho_{rc}$(b1), $\rho_{rc}$(b2), kw_b1_b2, ka_b1_b2, $\rho_a$(b1))   (2.6.8.4-21)

    If (status=0) then
        $\rho_w$_ie(b1, bs)=($\rho_{rc}$(b1)- $\rho_a$(b1))/$t_d$(b1)                (2.6.8.4-22)
        $\rho_w$_ie(b2, bs)= $\rho_w$_ie(b1, bs)* $\rho_w$(b2)/$\rho_w$(b1)          (2.6.8.4-23)
        Compute also for b3 and b4 (b3 mandatory when bsHIGH)
        For each b in {b3, b4} do
            $\rho_a(b) = \rho_a(b1)*(\lambda_{theo}(b)/\lambda_{theo}(b1))^{init\_ang}$   (2.6.8.4-24)
            $\rho_w$_ie(b, bs) =  ($\rho_{rc}$(b)- $\rho_a$(b))/ td(b)               (2.6.8.4-25)

        Endfor
    Endif

    Deal with exception processing (may be ok in case of absorbing aerosols or in case of
     very high sediments and bsLOW)
    If (status !=0) or ($\rho_w$_ie(b1, bs) < 0.0) then
        If bs=bsLOW then
```

```
                $\rho_w$_ie(b1, bs)= $\rho_w$(b1)                    (2.6.8.4-26)
                $\rho_w$_ie(b2, bs)= $\rho_w$(b2)                    (2.6.8.4-27)
            Endif
            For bsHIGH band set, rhow(b775) is set so that it does both and applies
            radiometric thresholding
            If bs = bsHIGH then
                $\rho_w$_ie(b775, bs) =  row_775_do_both_th          (2.6.8.4-28)
                $\rho_w$_ie(b865, bs) =  $\rho_w$(b1)                (2.6.8.4-29)
            Endif
        Endif

Endfor
```

Copyright © 2011 ACRI S.T.

```
Determination of bandset according to row_775 and radiometric flagging (step 2.6.8.5)
Note: only low in default configuration, and if glint is not corrected
do_bandset(bsHIGH)=FALSE                                                    (2.6.8.5-1)
do_bandset(bsLOW)=TRUE                                                      (2.6.8.5-2)
If ( not UNCGLINT_F(j,f) or MEDGLINT_F(j,f) ) then
    if ($\rho_w$_ie(b775,bsHIGH) >= row_775_do_both_th)  then do_bandset(bsHIGH)=TRUE
                                                                            (2.6.8.5-3)
    if ($\rho_w$_ie(b775,bsHIGH) >= row_775_do_high_th)  then do_bandset(bsLOW)=FALSE
                                                                            (2.6.8.5-4)
Endif

Calculate TOA minimum rho_w
rho_min_705=$\pi$*Ln_min_705/cos($\theta_S$)                                   (2.6.8.5-5)
Check radiometric threshold – note rho_w 705 estimate can be 0 at high turbidity
If (do_bandset(bsLOW) AND $t_d$(b705)*$\rho_w$_ie(b705,bsLOW) < rho_min_705 then
        BPAC_ON_F(j, f) = FALSE                                             (2.6.8.5-6)
        CASE2_S (j, f) = FALSE                                              (2.6.8.5-7)
        ACFAIL_F (j,f)= FALSE                                               (2.6.8.5-8)
        continue to step 2.6.8.8 – return clear water reflectances          (2.6.8.5-9)
Endif
```

Copyright © 2011 ACRI S.T.

8.3.3.3 - Turbid water correction (steps 2.6.8.6 to 2.6.8.7)

Only pixels with a BPAC_ON_F *flag set to TRUE will be further processed by the atmospheric corrections over Case 2 waters scheme.*

8.3.3.3.1 - Iterative estimate of Angström exponent, IOPs and rhow (step 2.6.8.6)

```
For each bs in {bsLOW, bsHIGH} do

    If bs=bsLOW then
      b1=b705    b2=b775    b3=b865    b4=b885                      (2.6.8.6-1)
    else
      b1=b865    b2=b885    b3=b775    b4=b705                      (2.6.8.6-2)
    Endif

    If (do_bandset(bs) =TRUE) then
        bbp_old=0.0                                                 (2.6.8.6-5)
        $\rho_w$(b1)= $\rho_w$_ie(b1,bs)                            (2.6.8.6-7)

        For (counter=0; counter<NIterBPAC(bs); counter++) do
            Estimate backscatter at b1 from rhow(b1)
            bbp(b1)=rhow_to_bb($\rho_w$(b1), aw(b1), bbw(b1), a_to_bb(b1),
                    $\theta_s$, $\theta_v$ , $\Delta\phi$, Ws, b1)  (2.6.8.6-8)
            Test convergence
            If (|bbp_old-bbp(b1)|/bbp(b1) < bbp_tol) then
                converge(bs)=TRUE                                   (2.6.8.6-9)
                break                                               (2.6.8.6-10)
            Endif
            Update bbp estimate for next step
            bbp_old=bbp(b1)                                         (2.6.8.6-11)

            Determine rest of spectrum
            bbp(b2)=bbp(b1)*bbp_star(b2)/bbp_star(b1)               (2.6.8.6-12)
            bbp(b3)=bbp(b1)*bbp_star(b3)/bbp_star(b1)               (2.6.8.6-13)
            a(b2)=aw(b2)+bbp(b2)*a_to_bb(b2)                        (2.6.8.6-14)
            a(b3)=aw(b3)+bbp(b3)*a_to_bb(b3)                        (2.6.8.6-15)
            fp(b2)=F_ab(a(b2), bbw(b2), bbp(b2), $\theta_s$, $\theta_v$, $\Delta\phi$, Ws, b2)   (2.6.8.6-16)
            fp(b3)=F_ab(a(b3), bbw(b3), bbp(b3), $\theta_s$, $\theta_v$, $\Delta\phi$, Ws, b3)   (2.6.8.6-17)
            $\rho_w$(b2)=fp(b2)*(bbp(b2)+bbw(b2))/(a(b2)+bbp(b2)+bbw(b2))   (2.6.8.6-18)
            $\rho_w$(b3)=fp(b3)*(bbp(b3)+bbw(b3))/(a(b3)+bbp(b3)+bbw(b3))   (2.6.8.6-19)

            Calculate aerosol reflectance
            $\rho_a$(b1)=$\rho_{rc}$(b1)- $t_d$(b1)*$\rho_w$(b1)    (2.6.8.6-20)
            $\rho_a$(b2)=$\rho_{rc}$(b2)- $t_d$(b2)*$\rho_w$(b2)    (2.6.8.6-21)
            $\rho_a$(b3)=$\rho_{rc}$(b3)- $t_d$(b3)*$\rho_w$(b3)    (2.6.8.6-22)

            Calculate aerosol slope exponent
```

Copyright © 2011 ACRI S.T.

```
                **If** (($\rho_a(b775) \le 0.0$) **OR** ($\rho_a(b865) \le 0.0$)) **then**
                    error(bs)=TRUE                                                        (2.6.8.6-23)
                    **break**                                                             (2.6.8.6-24)
                **Endif**
                ang_exp(bs)=$\log(\rho_a(b775)/\rho_a(b865))/\log(\lambda_{theo}(b775)/\lambda_{theo}(b865))$
        (2.6.8.6-25)

                *Calculate aerosol and water reflectances ratios*
                kw_b1_b2=$(\rho_w(b2)*t_d(b2))/(\rho_w(b1)*t_d(b1))$                      (2.6.8.6-26)
                ka_b1_b2=$(\lambda_{theo}(b2)/\lambda_{theo}(b1))^{ang\_exp(bs)}$         (2.6.8.6-27)

                *Compute aerosol reflectance at band b1*
                status=***two_band_rhoa***($\rho_{rc}(b1)$, $\rho_{rc}(b2)$, kw_b1_b2, ka_b1_b2, $\rho_a(b1)$)
                                                                                          (2.6.8.6-28)
                **If** (status != 0) **then**
                    error(bs)=TRUE                                                        (2.6.8.6-29)
                    **break**                                                             (2.6.8.6-30)
                **Endif**

                *Compute marine signal at band b1*
                $\rho_w(b1)=(\rho_{rc}(b1)-\rho_a(b1))/t_d(b1)$                           (2.6.8.6-31)
                **If** ($\rho_w(b1) < 0.0$) **then**
                    error(bs)= TRUE                                                       (2.6.8.6-32)
                    **break**                                                             (2.6.8.6-33)
                **Endif**

            **Endfor**    *End of iterative loop for current bandset*

            **IF** error(bs) = FALSE **then**
                *Complete the calculus for the remaining band*
                bbp(b4)=bbp(b1)*bbp_star(b4)/bbp_star(b1)                                 (2.6.8.6-34)
                a(b4)=aw(b4)+bbp(b4)*a_to_bb(b4)                                          (2.6.8.6-35)
                fp(4)=***F_ab***(a(b4), bbw(b4), bbp(b4), $\theta_s$, $\theta_v$, $\Delta\phi$, Ws, b4)    (2.6.8.6-36)
                $\rho_w(b4)$=fp(b4)*(bbp(b4)+bbw(b4))/(a(b4)+bbp(b4)+bbw(b4))             (2.6.8.6-37)

                *Final estimates – Note: store bbp_775_estimate for TSM estimation*
                **For each** b in {b705, b775, b865, b885} **do** $\rho_w$_fe(b,bs)= $\rho_w(b)$   (2.6.8.6-38)
                bbp_775_fe(bs)=bbp(b1)*bbp_star(b775)/bbp_star(b1)                        (2.6.8.6-39)
            **Endif**
        **Endif**    *End of if for bandset*
    **Endfor**       *End of loop on bandsets*
```

Copyright © 2011 ACRI S.T.

8.3.3.3.2 – Combine and check estimates (step 2.6.8.7)

```
Initialise averaged bbp and rhow
bbp(b775)=0.0                                                        (2.6.8.7-1)
For each b in {b705, b775, b865, b885} do $\rho_w$(b)=0.0             (2.6.8.7-2)
valid_data=0                                                         (2.6.8.7-3)

Loop on bandset
For each bs in {bsLOW,bsHIGH} do
    If ((do_bandset(bs) = TRUE) and (error(bs) = FALSE) and (converge(bs) = TRUE))
    then
        valid_data=valid_data+1                                      (2.6.8.7-4)
        For each b in {b705, b775, b865, b885} do
            $\rho_w$(b)=$\rho_w$(b)+ $\rho_w$_fe(b,bs)                (2.6.8.7-5)
            bbp(b775)=bbp(b775)+ bbp_775_fe(bs)                      (2.6.8.7-6)
    Endif
Endfor

Check at least one bandset has been successful
If valid_data $\geq$1 then
    BPAC_ON_F(j, f)=TRUE                                             (2.6.8.7-7a)
else
    BPAC_ON_F(j, f) = FALSE                                          (2.6.8.7-7b)
    CASE2_S (j, f) = FALSE                                           (2.6.8.7-8)
    deleted (2.6.8.7-9)
    continue to step 2.6.8.8 – return clear water reflectances       (2.6.8.7-10)
Endif

Compute average
For each b in {b705, b775, b865, b885} do $\rho_w$(b)=$\rho_w$(b)/valid_data   (2.6.8.7-11)
bbp(b775)=bbp(b775)/valid_data                                       (2.6.8.7-12)
```

8.3.3.4 - Estimate of TOA marine reflectances and TSM (step 2.6.8.8)

```
If BPAC_ON(j,f) = TRUE then
    Return tρw_C2 for b705, b775 b865 and b885
    For each b in {b705, b775, b865, b885} do
        $t\rho_w\_C2(b, j, f) = t_d(b) . \rho_w(b)$                                  (2.6.8.8-1)

    Compute TSM of bb 775
    $SPM_{br}(j,f) = bbp(b775)/bbp\_star(b775)$                                      (2.6.8.8-2)
Else
    Return pure water reflectance and null TSM
    For each b in {b705, b775, b865, b885} do
        $fp(b)=F\_ab(aw(b), bbw(b), 0, \theta_s, \theta_v, \Delta\phi, Ws, b)$           (2.6.8.8-3)
        $t\rho_w\_C2(b, j, f)= t_d(b)*fp(b)*bbw(b)/(aw(b)+bbw(b))$                   (2.6.8.8-4)
    Endfor
    $SPM_{br}(j,j)=0$                                                                (2.6.8.8-5)
Endif
```

8.3.3.5 Function F_ab(a, bbw, bbp, θ_s, θ_v, $\Delta\phi$, , Ws, b)

This function computes the F' factor which relates IOPs to marine reflectance. It is called by *Water identification and initial estimate* (step 2.6.8.4) , *Turbid water correction* (step 2.6.8.6) and *Estimate of TOA marine reflectances and TSM* (step 2.6.8.8).

8.3.3.5.1 Input/Output

Variable	Descriptive name	T	U	Range – References
a	Total absorption (pure water and sediments)	i	m^{-1}	
bbw	Back scattering of pure water	i	m^{-1}	
bbp	Back scattering of sediments	i	m^{-1}	
θ_S	Sun zenith angle	i	deg	
θ_V	Satellite viewing angle	i	deg	
$\Delta\phi$	Difference of azimuth angles	i	deg	
W_s	Wind speed modulus	i	m/s	
b	Band index	i	-	
Fp_LUT[θ_S, θ_V, $\Delta\phi$, Ws, coeff, b]	LUT of coefficients of F' to IOPs relation	s	dl	
N_Fp_coeff	Number of coefficients for fitting the F' function	s	-	
η	Ratio of pure water back scattering to total back scattering	c	dl	
ω	Single-scattering albedo	c	dl	
coeff(i)	Interpolated coefficients of Fp_LUT	c	dl	i:0… N_Fp_coeff-1
Fp	F' value	o	dl	

Table8.3.3.5-1: List of variables for function F_ab

8.3.3.5.2 Algorithm

```
Interpol F' LUT
For (i=0; i<N_Fp_coeff) do
        coeff(i) = Fp_LUT interpol: ($\theta_s$, $\theta_v$, $\Delta\phi$, Ws) select: (i, b)        (fp-ab-1)

Compute F' value
Warning: F' is the addition of a linear function in $\eta$ and a polynomial of order 4 in $\omega$ – note however that the third coefficient of the table Fp_LUT is never used.
$\eta$=bbw/(bbw+bbp)                                        (fp-ab-2)
$\omega$=(bbw+bbp)/(bbw+bbp+a)                              (fp-ab-3)
Fp=coeff(0) + coeff(1) $\cdot$ $\eta$                       (fp-ab-4)
For (i=0; i<N_Fp_coeff-3; i++) do Fp = Fp + coeff(i+3)$\cdot$ $\omega^i$        (fp-ab-5)

Return Fp
```

Copyright © 2011 ACRI S.T.

8.3.3.6 Function two_band_rhoa(ρ_{rc}(bLOW), ρ_{rc}(bHIGH), kw, ka, ρ_a(bLOW))

Given the Rayleigh TOA reflectance at two bands bLOW and bHIGH and the spectral slopes of both the TOA marine and the aerosol signal, this function computes the aerosol reflectance at the first band.

8.3.3.6.1 Input/Output

Variable	Descriptive name	T	U	Range – References
ρ_{rc}(bLOW)	Rayleigh corrected TOA reflectance at first band	i	dl	
ρ_{rc}(bHIGH)	Rayleigh corrected TOA reflectance at second band	i	dl	
kw	Ratio of TOA water reflectance at band bHIGH and bLOW	i	dl	
ka	Ratio of aerosol reflectance at band bHIGH and bLOW	i	dl	
ρ_a(bLOW)	Aerosol reflectance at the first band	o	dl	
status	Status of the computation	o	-	

Table8.3.3.6-1: List of variables for function two_band_rhoa

8.3.3.6.2 Algorithm

```
If (($\rho_{rc}$(bLOW) <=0) || ($\rho_{rc}$(bHIGH) <=0)) then
        return 1                                                    (tbr-1)
Endif

Return $\rho_a$(bLOW) given the slope in water reflectance and atmospheric reflectance
If ( |ka-kw| ≠ 0) then
        $\rho_a$(bLOW)=($\rho_{rc}$(bHIGH) - kw*$\rho_{rc}$(bLOW))/(ka-kw)      (tbr-2)
else
        $\rho_a$(bLOW)=0                                            (tbr-3)
        return -1                                                   (tbr-4)
Endif

exception processing: $\rho_a$(bLOW)<0:
        $\rho_a$(bLOW)=0                                            (tbr-5)
        return -1                                                   (tbr-6)
end of exception processing

return 0
```

Copyright © 2011 ACRI S.T.

8.3.3.7 Function rhow_to_bb(ρ_w, aw, bbw, a_to_bb, θ_s, θ_v, $\Delta\phi$, Ws, b)

This function computes by an iterative loop the sediments backscattering at a given band starting from the marine reflectance and the specific absorption of sediments (a to bb ratio).

8.3.3.7.1 Input/Output

Variable	Descriptive name	T	U	Range – References
ρ_w	Marine reflectance at band b	i	dl	
a_to_bb	Specific absorption	i	dl	
θ_S	Sun zenith angle	i	deg	
θ_V	Satellite viewing angle	i	deg	
$\Delta\phi$	Difference of azimuth angles	i	deg	
W_s	Wind speed modulus	i	m/s	
b	Band index	i	-	
aw(b)	Absorption of pure water	s	m^{-1}	
bbw(b)	Back scattering of pure water	s	m^{-1}	
rhow_bb_tol	Convergence criteria of the algorithm	s	dl	
Niter_rhow_bb	Number of iterations in the iterative loop	s	-	
bbp_init	Initial value of bbp to initialise F’	s	m^{-1}	
a	total absorption	c	m^{-1}	
bb_old	Sediment back scattering for convergence check	c	m^{-1}	
f	F factor	c	dl	
fp	F’ factor	c	dl	
iter	Iterative counter	c	-	[0, Niter_rhow_bb-1]
bb	Sediment back scattering	o	m^{-1}	

Table8.3.3.7-1: List of variables for function rhow_to_bb

8.3.3.7.2 Algorithm

```
exception processing: $\rho_w$ <0:
        return 0                                                    (rtbb-1)
end of exception processing

Initialise
bb_old=-1.0                                                         (rtbb-2)
a=aw                                                                (rtbb-3)
fp=F_ab(a, bbw, bbp_init, $\theta_s$, $\theta_v$, $\Delta\phi$, Ws, b)
        (rtbb-4)
bb=($\rho_w$ *a)/fp-bbw                                             (rtbb-5)
For (iter=0; iter < Niter_rhow_bb; iter++) do
        a = aw + a_to_bb*bb                                         (rtbb-6)
        fp = F_ab(a, bbw, bb, $\theta_s$, $\theta_v$, $\Delta\phi$, Ws, b)   (rtbb-7)
        f = (fp*a) / (a + bb + bbw)                                 (rtbb-8)
        bb = ($\rho_w$*a)/f – bbw                                   (rtbb-9)
        if (|bb - bb_old|/bb < rhow_bb_tol)  then break             (rtbb-10)
        bb_old=bb                                                   (rtbb-11)
Endfor
```

Copyright © 2011 ACRI S.T.

```
If bb <0. then bb=0.                                        (rtbb-12)

return bb
```

Copyright © 2011 ACRI S.T.

8.3.4 - Quality control and diagnostics

The annotation flag has been introduced within the BPAC scheme, to provide the breakpoints with indications about the course of the algorithm. This flag (variable "ANNOT_BPAC") is represented by 2 bytes. Each bit implements a Boolean flag, corresponding to the activation and course of the iterative loop of a given bandset:

symbol	**bit pos.**	**Meaning**
DO_BANDSET_LOW	0	Bandset Low is activated
DO_BANDSET_HIGH	1	Bandset High is activated
CONVERGE_LOW	2	Error in the iterative estimate for bandset Low
CONVERGE_HIGH	3	Error in the iterative estimate for bandset High
ERROR_LOW	4	Convergence of the iterative estimate for bandset Low
ERROR_HIGH	5	Convergence of the iterative estimate for bandset Low

Table 8.3.4-1: Coding of ANNOT_BPAC flag

8.3.5 - Exception handling

To summarise exception handling in section 8.3.3 above:

1. When ρ_{rc}<= 0 at b705 or b865 in step 2.6.8.4 then go to step 2.6.8.8, set BPAC_ON_F(j,f) and CASE2_S (j, f) to FALSE and ACFAIL_F(j,f) to TRUE.

2. When the LOW band set is selected and the initial value of ρ_w(b705) in step 2.6.8.5 is below the MERIS threshold, then go to step 2.6.8.8, set BPAC_ON_F(j,f), CASE2_S(j, f) and ACFAIL_F(j,f) to FALSE, ρ_w({b705, b775, b865, b885}) to clear water values and SPM_{br} to 0.

3. Iterative loop shall break when aerosol reflectances at 775nm or 865nm are negative or null in step 2.6.8.6.

4. Iterative loop shall break when water leaving reflectance at band b1 is negative in step 2.6.8.6.

5. When step 2.6.8.7 detects that none of the band set iterations is successful, then go to step 2.6.8.8, set BPAC_ON_F(j,f) and CASE2_S(j, f) to FALSE, ρ_w({b705, b775, b865, b885}) to clear water values and SPM_{br} to 0.

Copyright © 2011 ACRI S.T.

8.4 - Clear water atmospheric corrections (step 2.6.9)

8.4.1 - Overview

The objective of the clear water atmosphere correction is to identify and subtract from the TOA reflectances (corrected for stratospheric aerosols, gaseous absorption and Sun glint), the contribution of the atmosphere, which consists of molecular (Rayleigh) and particulate (aerosol) scattering and extinction. The correction is performed in order to provide normalised water-leaving reflectances.

A secondary objective is to estimate aerosol products: type and optical thickness.

The principle of the clear water atmosphere correction is to identify aerosol models which, together with a tabulated model of the molecular scattering and assumptions on the surface reflectance, fit the observed glint-corrected reflectance in the infra-red part of the spectrum (bands 775, 865nm) and in a visible band (510nm).

The assumptions for Case 1 waters are that reflectance is null at all wavelengths beyond 700nm, and that reflectance at 510nm is nearly constant.

The output of the turbid water atmosphere correction (step 2.6.10, see section 7.3.4.2 above) provides as input estimates of the water reflectance at the bands used by the algorithm.

The algorithm provides one or two aerosol models and their properties in the visible and NIR wavelength domain, which allow to perform a correction of the atmosphere contribution and compute water-leaving reflectances.

The water-leaving reflectances output by the atmosphere corrections above water are normalised in order to remove dependency of the signal upon atmosphere conditions. The normalised water-leaving reflectance product ρ'_w is defined as follows:

$$\rho'_w = \frac{\pi . L_w}{E_d(0^+)}$$

where L_w is the water-leaving radiance and $E_d(0^+)$ the down-welling irradiance.

The water-leaving reflectance is used by other sub-steps and provided to the Product formatting (step 2.10). This step is applied to all pixels where ACFAIL_F is FALSE.

8.4.1.1 - Path reflectance estimate (step 2.6.9.1)

When starting the atmospheric correction, we dispose of the (measured) total glint corrected reflectance ROGC, and of $\rho_R(\lambda)$ for each wavelength. When turbid case 2 water has been detected by a previous step, we also have an estimate of the marine reflectance at TOA, $t\rho_w$_C2, otherwise this quantities is set to the reflectance of pure water.

Atmospheric corrections need an estimate of the contribution of the sky to the total reflectance, or path reflectance, at two wavelengths.

At 779 and 865 nm, for any water pixel, we subtract the water contribution so that

$\rho_{path}(\lambda) = ROGC(\lambda) - t\rho_{w}\_C2(\lambda)$

The path reflectance is then corrected for pressure variation, in order to enter the aerosol model selection at standard pressure.

8.4.1.2 – MERIS aerosol model (step 2.6.9.2)

When starting the aerosol correction, we dispose on one hand of the path reflectance ρ_{path} at two wavelengths, and of the TOA reflectance ROGC(λ) and Rayleigh reflectance $\rho_R(\lambda)$ for each wavelength, and on the other hand of tabulated relationships linking the ratio ρ_{path}/ρ_R to the aerosol optical thickness $\tau_a(\lambda)$, for N aerosol models.

The central problem is the selection, among a set of aerosol models, of the two models that most closely bracket the actual aerosol. The principle is to rely on the look-up tables, which should allow :

- To calculate the values of $\tau_a(865)$ from the $\rho_{path}(865)/\rho_R(865)$ ratio, for several aerosol models,

- To extrapolate τ_a from 865 to 775 nm, for each aerosol model,

- To obtain the ($\rho_{path}(775)$ / $\rho_R(775)$) ratios from $\tau_a(775)$, for each aerosol model. These ratios computed from aerosol model, will be noted $\zeta(\lambda)$ in the following.

- To select a couple of aerosol models, by comparing the actual ($\rho_{path}(775)$ / $\rho_R(775)$) ratio, and the various $\zeta(775)$ ratios as obtained at the previous step.

- To estimate the $\zeta(\lambda)$ ratio in the visible bands from the knowledge of the spectral behaviour of this couple of aerosol models.

The successive steps of such a correction scheme are as follows. For a given pixel, and thus for a given geometry (θ_s, θ_v, $\Delta\phi$):

(1) The ratio $\rho_{path}(\lambda)$ / $\rho_R(\lambda)$ is computed at 865 and 775 nm, $\rho_R(\lambda)$ being taken in tabulated values.

(2) A first set of N aerosol models is selected, which, in principle, is representative of clear oceanic atmospheres. For these N aerosol models, N $\tau_a(865)$ values are calculated from the ($\rho_{path}(865)$ / $\rho_R(865)$) ratio.

(3) N values of $\tau_a(775)$ are computed for the N aerosol models, from the knowledge of their spectral optical thicknesses (normalised by their values at 865 nm; tabulated values).

(4) N values of $\zeta(775)$ are computed from the N values of $\tau_a(775)$ for the N aerosol models, from the tabulated relationships between both quantities.

Copyright © 2011 ACRI S.T.

(5) The actual ($\rho_{path}(775)$ / $\rho_R(775)$) is then compared to the N individual values of $\zeta(775)$ obtained at step (4), and the 2 that most closely bracket the actual one indicate the two candidate aerosol models.

(6) 2 values of $\tau_a(\lambda)$ are calculated for bands at 510 nm and 705 nm from the normalised spectral optical thicknesses of the 2 “bracketing” aerosol models. Step (2) is now inverted, to calculate two $\zeta(\lambda)$ ratios from the two $\tau_a(\lambda)$ at 510 nm and 705 nm.

(7) The following step lies on the assumption that the actual ($\rho_{path}(\lambda)$ / $\rho_R(\lambda)$) ratio falls between the two $\zeta(\lambda)$ ratios calculated at step (6), proportionally, in the same manner as it does at 775 nm. $\rho_{path}(\lambda)$ is now estimated for bands at 510 nm and 705 nm.

(8) By making an assumption on the normalised water-leaving reflectance at 510 nm, the error in the atmospheric correction at 510 nm, $\Delta\rho_{510}$, can be assessed.

(9) A test is then made on this $\Delta\rho_{510}$ value, if a number of conditions are met. If those conditions are not met, the correction is continued at step (10). Otherwise, depending on the test result, either the correction is continued at step (10), or it is carried out once more from step (2), by selecting however a different set of N’ aerosol models. In the latter situation, the correction is actually carried out for several aerosol databases, so that steps 2-8 are carried out several times; several couples of aerosol models are then selected (one at each time steps 2-8 are done), and the one which is retained at the end is the one that leads to the lowest $\Delta\rho_{510}$.

(10) For every wavelength λ of the visible domain, 2 values of τ_a are calculated from the knowledge of the spectral scattering coefficients of the 2 “bracketing” aerosol models.

(11) Step (2) is now inverted, to calculate two $\zeta(\lambda)$ ratios from the two $\tau_a(\lambda)$ for the visible bands, and then to obtain $\rho_{path}(\lambda)$ (see step 7).

8.4.1.3 – Correction (step 2.6.9.3)

At then end of the MERIS model step, we now have an estimate of the path reflectance and aerosol optical parameters at all visible and NIR wavelengths where the atmospheric correction is required.

The water-leaving reflectance at the instrument level is then obtained as:
$t_u(\lambda).t_d(\lambda).\rho'_w(\lambda) = ROGC(\lambda) - \rho_{path}{}^*(\lambda)$

Note that we now impose $t_u(\lambda).t_d(\lambda).\rho'_w(\lambda)=t\rho_{w}\_C2(\lambda)$ at λ=779 and 865 nm, as it should theoretically be but cannot numerically because of the pressure correction, and at 885 nm as well where the signal is too small and might produce negative value.

The following step consists in calculating the diffuse transmittance, downward $t_d(\lambda)$ and upward $t_u(\lambda)$, in order to retrieve the normalised water-leaving reflectance at surface level $\rho'_w(\lambda)$.

Copyright © 2011 ACRI S.T.

8.4.2 - List of variables

An exhaustive list of variables is provided with each step /subroutine /function detailed specification in 8.4.3 below. The table below summarises the inputs, outputs of the main algorithm and auxiliary parameters of all steps /subroutines /functions.

The index entries of the LUTs described within the table below are included in the auxiliary file containing the LUTs, see AD4 for details.

The nominal wavelengths are listed in table 2.4-1. It should be noted that, in all the following tables and equations, the shorthand notation b412..b885 excludes band 11 (760 nm).

In all following sections of §8.4, μ_s, μ_v and μ are respectively equivalent to $\cos(\theta_s)$, $\cos(\theta_v)$ and $\cos(\theta_v)$ - they may be computed at the main level or within each function.

Note that the aerosol single scattering albedo and ratio of forward to total scattering are not used anymore for transmittance estimates, but still computed and stored into breakpoints.

Symbol	Descriptive name	T	U	Range /Remarks
INVALID_F (j, f)	Invalid pixel flag	i	-	Boolean, from 2.1a (§3.4)
CLOUD_F (j, f)	Cloud flag	i	-	from step 2.1c (§5.4)
LANDCONS_F(j, f)	Land flag	i	-	from step 2.1c (§5.4)
CASE2_S (j, f)	Case 2 water flag	i	-	from step 2.6.8 (§8.3)
ICE_HIGHAERO_F(j,f)	Flag for ice or high aerosol loading pixels	i	dl	from step 2.6.5 (§8.2)
MEGLINT_F(j,f)	Flag for pixels corrected for glint	i	dl	from step 2.6.5 (§8.2)
UNCGLINT_F(j,f)	Flag for pixels contaminated by glint	i	dl	from step 2.6.5 (§8.2)
WHITE_SCATT_F (j,f)	Flag identifying "white" scatter within water	i	dl	from step 2.6.8 (§8.3)
θ_s (j, f)	Sun zenith angle	i	deg	from step 2.1a (§3.4)
θ_v (j, f)	View zenith angle	i	deg	idem
$\Delta\phi$ (j, f)	Azimuth difference	i	deg	idem
P_{ECMWF} (j, f)	ECMWF pressure	i	hPa	idem
ROGC (b, j, f)	Glint corrected reflectance	i	dl	b: {b412…, b885}, from step 2.6.5 (§8.2.2)
ρ_R (b, j, f)	Rayleigh reflectance	i	dl	from step 2.1c (§5.4)
ρ_{R0} (b, j, f)	Rayleigh reflectance corrected for pressure variations	i	dl	from step 2.1c (§5.4)
τ_{R0}(b)	Rayleigh optical thickness corrected for pressure variation	i	dl	from step 2.1c (§5.4)
lat (j, f)	pixel latitude	i	deg	from step 2.1a (§3.4)
lon (j, f)	pixel longitude	i	deg	idem
z (j, f)	pixel altitude	i	m	idem
month	month of acquisition	i	dl	from L1b MPH
$t\rho_w$_C2(b, j, f)	Marine reflectance at TOA for Case 2 waters	i	dl	b: {b705, b775, b865, b885}, from step

Symbol	Descriptive name	T	U	Range /Remarks
				2.6.10 (§8.3.2)
Aerclim_Ocean_LUT [lat, lon, month]	map of aerosol climatology	s	dl	month: 1..12, lat: -90..90, lon: -180..180; coding: see table 8.4.3.5.2-2 below
$c=\{c_1,c_2,c_3, c_6\}$	constant for computation of path	s	dl	
CLIMATO_AUX	Switch to activate the use of a climatology	s	-	Boolean
CMOY	Mean value of chl. concentration	s	$mg.m^{-3}$	
BLUE_LIKE	symbolic value to signal aerosols with steep spectral dependence	s	-	See note [1]
DEPTH_LIM	Threshold on depth to set the “shallow water” flag	s	m	
DRO510_LIM	Value of $\Delta\rho$510 to set the annotation flag	s	dl	
DRO510_thres_D	threshold for the absorbing aerosol test at 510nm	s	dl	
DRO510_thresh_B	threshold for the blue aerosol test at 510nm	s	dl	
DUST_LIKE	symbolic value to signal desert dust absorbing aerosols	s	-	See note [1]
f_over_q1_LUT (b, θ_p, θ_s, $\Delta\phi$, Chl, τ_a, W_s)	LUT for the bidirectional factor f/Q	s	dl	
fatab_LUT [ia, b]	LUT for the aerosol forward scattering probability f_a	s	dl	ia: 1..N_Aer b: b412.. b885
FIRST_PASS, SECOND_PASS, THIRD_PASS, FOURTH_PASS, FIFTH_PASS	symbolic values for the number of each algorithm pass	s	-	See note [1]
FSURQ_0	Value of f/Q factor at nadir	s	dl	
LIST_Aer_01	list of aerosol models	s	dl	
LIST_Aer_02	list of aerosol models	s	dl	
LIST_Aer_03	list of aerosol models	s	dl	
LIST_Aer_04	list of aerosol models	s	dl	
LIST_Aer_05	list of aerosol models	s	dl	
LIST_Aer_06	list of aerosol models	s	dl	
LIST_Aer_07	list of aerosol models	s	dl	
LIST_Aer_08	list of aerosol models	s	dl	
$\lambda_{theo}(b)$	Theoretical wavelengths corresponding to smile corrected reflectances	s	nm	

---

[1] The value of the symbols is left to implementation as it is internal to the application and language dependant. For instance, a “C” program might use “enum” statement.

Copyright © 2011 ACRI S.T.

Symbol	Descriptive name	T	U	Range /Remarks
MAX_TAU_AER	Maximum allowed value for aerosol optical thickness	s	dl	
N_Aer	Number of aerosol models in database	s	-	
N_basic_aer	Number of aerosol models per list	s	dl	
N_co	Number of coefficients in XCtab	s	-	
N_PASSTOT	Number of passes in the algorithm	s	-	See note [1]
NOABSORBING	symbolic value to signal non-absorbing aerosols	s	-	See note [2]
PRESS_ TOLERANCE	Threshold to activate a correction for pressure	s	hPa	
P_{std}	Standard value of the surface pressure	s	hPa	
r_{ghot}_LUT (θ_p, W_s)	LUT for the ocean-atmosphere reflection factor	s	dl	
specdep(aer,τ,b)	LUT of the spectral dependence of the aerosol optical thickness	s	dl	aer: 1..N_aer; b: b412..b885
TAUA865_ threshold	Threshold for flagging the aerosol optical thickness	s	dl	
TEST_AER	Switch enabling the test for absorbing aerosol	s	-	Boolean
TETAP_ZENITH	Value of θ_p for nadir viewing	s	deg	
TETAS_limit	Threshold on Sun zenith angle for setting the SUN70 flag	s	deg	
TROW_510_MEAN	Mean value of the normalised water-leaving reflectance at 510nm	s	dl	
WAT_REF_IND	Water refraction index	s	dl	
XCtab_LUT [$k, W_S, ia, b, \theta_S, \theta_V, \Delta\phi$]	LUT for polynomial coefficients linking the ratio ρ_{path}/ρ_R to the aerosol optical thickness	s	dl	
τ_a_bl865 (aer,τ)	LUT of the optical thickness of the aerosol assemblage at 865 nm	s	dl	
τ_R(b)	Rayleigh optical thickness at standard pressure	s	dl	b: {b412..b885}
ωatab_LUT [ia, b]	LUT for the aerosol single scattering albedo ω_a	s	dl	*idem*
RWNEG_threshold(b)	LUT of negative water-leaving reflectance threshold	s	dl	b:{b412..b885}
BAD_VALUE	Product value when algorithm fails	s	dl	see § 2
μ_s	Cosine of Sun zenith angle	c	dl	
μ	Cosine of view zenith angle	c	dl	
ACFAIL_F (j, f)	Flag indicating failure of the	i	-	Boolean, to step 2.10

---

[1] The value of the symbols is left to implementation as it is internal to the application and language dependant. For instance, a "C" program might use "enum" statement.

[2] The value of the symbols is left to implementation as it is internal to the application and language dependant. For instance, a "C" program might use "enum" statement.

Symbol	Descriptive name	T	U	Range /Remarks
	atmosphere correction	/o		(§10.4), to Breakpoint
$\tau_a(b, j, f)$	Aerosol optical thickness estimate	o	dl	b: {b412..b885}: to Breakpoint; {b865}: to steps 2.9 (§8.5.4), 2.10 (§10.4)
ia1(j,f), ia2(j,f)	Index of bracketing aerosol models	o	-	to Breakpoint
aer_mix (j, f)	Mixing ratio	o	dl	to Breakpoint
$\omega_a(b)$	Aerosol single scattering albedo	o	dl	b: {b412..b885}, to Breakpoint
$f_a(b)$	Ratio of forward to total scattering	o	dl	b: {b412..b885}, to Breakpoint
$t_u(b)$	Transmittance on the target-sensor path	c	dl	b: {b412..b885}, to Breakpoint
$t_d(b)$	Transmittance on the Sun-target path	c	dl	b: {b412..b885}, to Breakpoint
ρ'_w (b, j, f)	Normalised water-leaving reflectance	o	dl	b: {b412..b885}, to step 2.10 (§10.4), to Breakpoint
α_775_865 (j, f)	Aerosol Angström exponent	o	dl	to step 2.9 (§8.5.4), 2.10 (§10.4), to Breakpoint
ANNOT (j, f)	Annotation flag for the quality of the atmospheric correction	o	-	Coding: see 8.4.4, to step 2.10 (§10.4), to Breakpoint
RWNEG (b, j, f)	Flag indicating negative water-leaving reflectance	o	-	Coding: see 8.4.4, to step 2.10 (§10.4)
ORINP0_F(j,f)	Flag indicating whether input to the clear waters atm. Corr. Is invalid	o	-	Boolean, to step 2.10 (§10.4), to Breakpoint

8.4.3 - Detailed Algorithm Specification

NOTE: For the sake of clarity, the subscripts (j,f) of parameters related to one pixel may be omitted when unambiguous, e.g. θ_s is equivalent to $\theta_s(j, f)$.

8.4.3.1 - Atmosphere corrections (step 2.6.9)

8.4.3.1.1 - Functional description

The functional block diagram in figure 8.4.3.1.1–1 below shows the operation and parameters of the Atmosphere corrections.

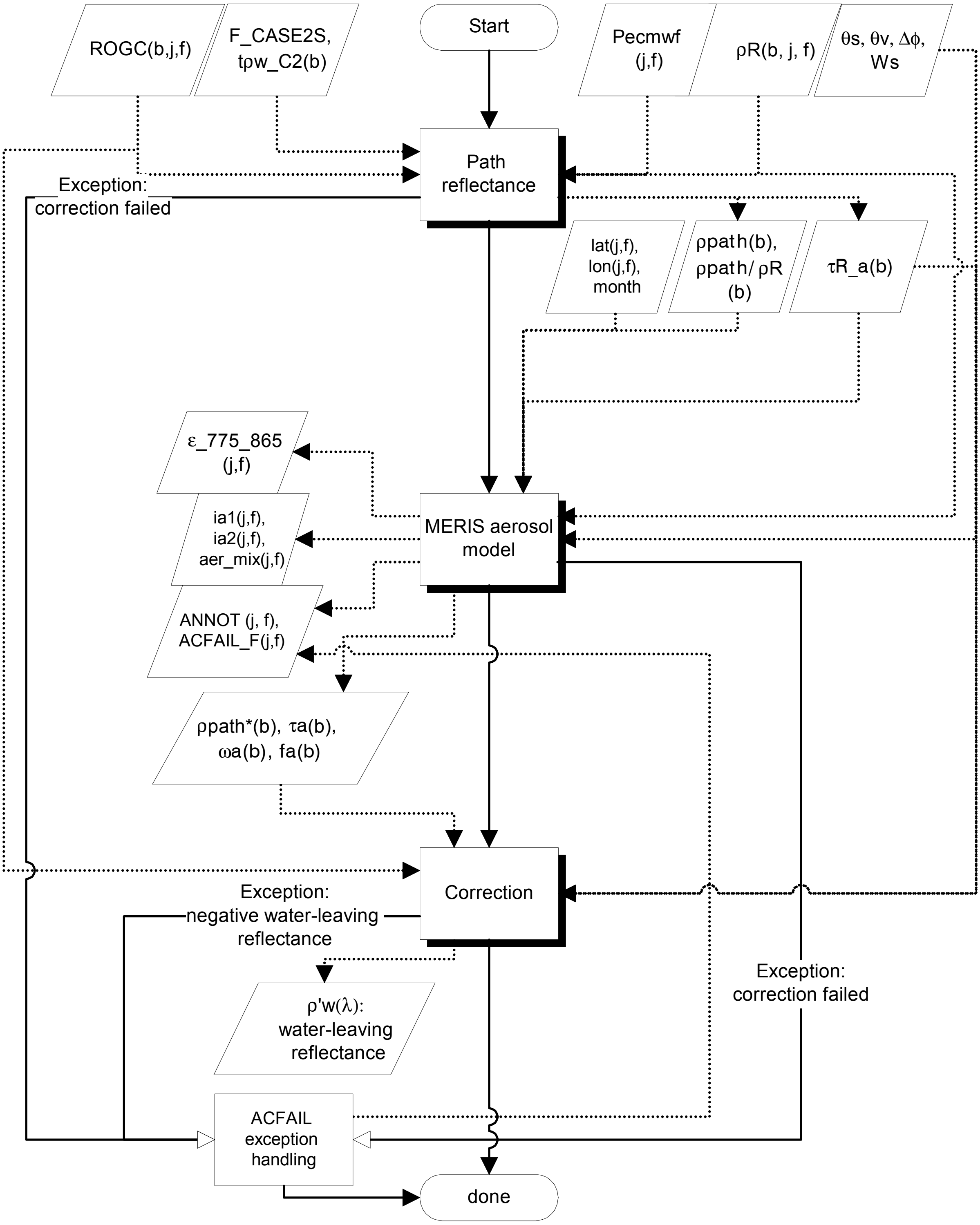


Figure 8.4.3.1.1-1: functional block diagram of Atmospheric corrections (step 2.6.9)

Notes

1) for the sake of clarity, step numbers are omitted from the figure above.

2) Process initialisation (step 2.6.9.0) is not represented in the figure above.

Copyright © 2011 ACRI S.T.

8.4.3.1.2 - Inputs /Outputs

Symbol	Descriptive name	T	U	Range /Remarks
INVALID_F (j, f)	Invalid pixel flag	i	-	Boolean
CLOUD_F (j, f)	Cloud flag	i	-	*idem*
LANDCONS_F(j, f)	Land flag	i	-	*idem*
CASE2_S (j, f)	Case 2 water flag	i	-	*idem*
θ_s (j, f)	Sun zenith angle	i	deg	
θ_v (j, f)	View zenith angle	i	deg	
$\Delta\phi$ (j, f)	Azimuth difference	i	deg	
μ_s	Cosine of Sun zenith angle	i	dl	
μ	Cosine of view zenith angle	i	dl	
P_{ECMWF} (j, f)	ECMWF pressure	i	hPa	
ROGC (b, j, f)	Glint corrected reflectance	i	dl	b: {b412..b885}
ρ_R (b, j, f)	Rayleigh reflectance	i	dl	*idem*
lat (j, f)	pixel latitude	i	deg	
lon (j, f)	pixel longitude	i	deg	
month	month of acquisition	i	dl	
ρ_{R0} (b)	Rayleigh reflectance corrected for pressure variations	i	dl	b: {b412..b885}; from step 2.1c (§5.4)
τ_{R0}(b)	Optical thickness due to Rayleigh scattering	i	dl	*idem*
$t\rho_w$_C2(b, j, f)	Marine reflectance at TOA for Case 2 waters	i	dl	b: {b510, b705, b775, b865}
taua(b)	Initial guess value for aerosol optical thickness	c	dl	b: {b510, b705, b775, b865}
ρ_{path}(b)	Path reflectance	c	dl	b: {b510, b705, b775, b865}
ρ_{path}uponρ_R (b)	Ratio of path reflectance to Rayleigh reflectance	c	dl	*idem*
ρ^*_{path}(b)	Path reflectance estimate	c	dl	b: {b412.. b885}
ω_a(b)	Aerosol single scattering albedo	c	dl	b: {b412..b885}
f_a(b)	Ratio of forward to total scattering	c	dl	b: {b412..b885}
τ_a(b, j, f)	Aerosol optical thickness estimate	o	dl	b: {b412..b885}
ia1(j,f), ia2(j,f)	Index of bracketing aerosol models	o	-	
aer_mix (j, f)	Mixing ratio	o	dl	

Table 8.4.3.1.2-1: Parameters for the atmospheric correction above water

Copyright © 2011 ACRI S.T.

Symbol	Descriptive name	T	U	Range /Remarks
ρ'_w (b, j, f)	Normalised water-leaving reflectance	o	dl	b: {b412..b885}
α_865_775 (j, f)	Aerosol Angström exponent	o	dl	
ANNOT (j, f)	Annotation flag for the quality of the atmospheric correction	o	-	Coding: see 8.4.4
RWNEG (b, j, f)	Flag indicating negative water-leaving reflectance	o	-	Coding: see 8.4.4
ACFAIL_F (j, f)	Flag indicating failure of the atmosphere correction	i /o	-	Boolean

Table 8.4.3.1.2-1: Parameters for the atmospheric correction above water (cont.)

8.4.3.1.3 - Algorithm

```
For each pixel (j, f) such that (NOT INVALID_F(j, f)) AND (NOT CLOUD_F(j, f)) AND (NOT LANDCONS_F(j, f)) AND (NOT (CASE2_S(j,f) AND ACFAIL_F(j,f)))
    activate Process initialisation (step 2.6.9.0)
    activate Path reflectance (step 2.6.9.1)
    activate MERIS aerosol model (step 2.6.9.2)
    If (NOT ACFAIL_F(j, f)) then
        activate Correction (step 2.6.9.3)
    Endif
Endfor
```

Copyright © 2011 ACRI S.T.

8.4.3.2 - Process initialisation (step 2.6.9.0)

8.4.3.2.1 - Functional description

The process initialisation step is not represented in the block diagrams. It shall be activated once, before processing any pixel.

8.4.3.2.2 - Inputs /Outputs

Symbol	Descriptive name	T	U	Range /Remarks
taua(b)	Initial guess value for aerosol optical thickness	o	dl	b: {b510, b705, b775, b865}

Table 8.4.3.2.2-1: Parameters for the process initialisation

8.4.3.2.3 - Algorithm

$$\mathrm{taua}(b775) = 0.1 \quad (2.6.9.0\text{-}1)$$

$$\mathrm{taua}(b865) = 0.1 \quad (2.6.9.0\text{-}2)$$

$$\mathrm{taua}(b510) = 0.1 \quad (2.6.9.0\text{-}3)$$

$$\mathrm{taua}(b705) = 0.1 \quad (2.6.9.0\text{-}4)$$

Copyright © 2011 ACRI S.T.

8.4.3.3 - Path reflectance (step 2.6.9.1)

8.4.3.3.1 - Functional description

This step computes the path reflectance in the infra-red and its ratio to the Rayleigh reflectance. A pressure-corrected estimate of the Rayleigh reflectance and optical thickness is also computed. Figure 8.4.3.3.1-1 below shows the operation and parameters of the step.

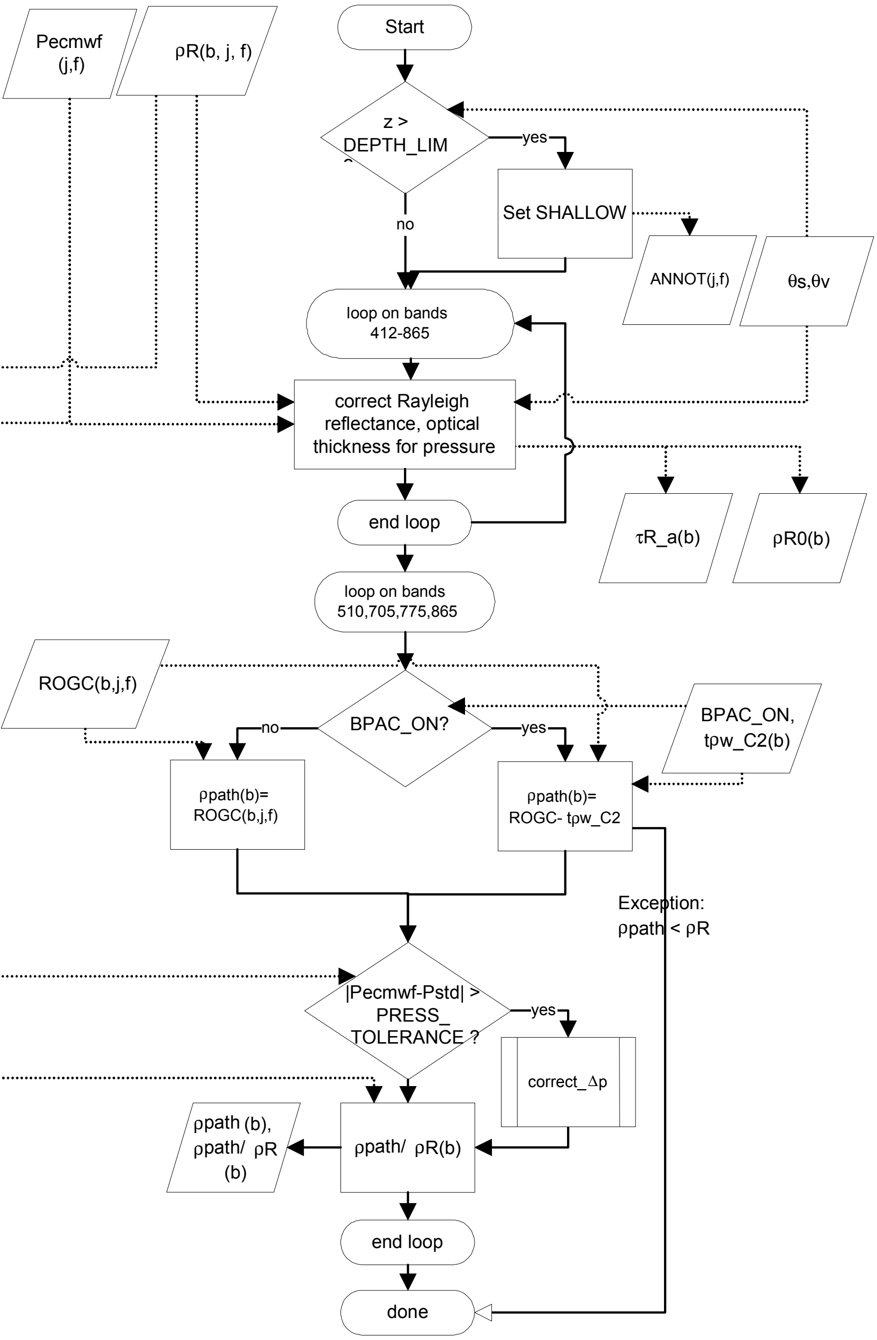


Figure 8.4.3.3.1-1: functional block diagram of step Path reflectance (2.6.9.1)

8.4.3.3.2 - Inputs /Outputs

Symbol	Descriptive name	T	U	Range /Remarks
P_{ECMWF} (j, f)	ECMWF pressure	i	hPa	
z (j, f)	Altitude	i	m	
ROGC (b, j, f)	Glint corrected reflectance	i	dl	b: {b412..b885}
ρ_R(b, j, f)	Rayleigh reflectance	i	dl	*idem*
ρ_{R0} (b, j, f)	Rayleigh reflectance corrected for pressure variations	i	dl	from step 2.1c (§5.4)
tρw_C2 (b, j, f)	Marine reflectance at TOA for Case 2 waters	i	dl	b: {b510, b705, b775, b865}
taua (b)	Initial guess value for aerosol optical thickness	i	dl	
BPAC_ON_F(j,f)	Flag triggering Bright Pixels turbid water atmosphere correction	i	-	
P_{std}	Standard value of the surface pressure	s	hPa	
PRESS_ TOLERANCE	Threshold to activate a correction for pressure	s	hPa	
ρ_{path} (b)	Path reflectance, i.e. the TOA reflectance minus eventual sediment contribution	o	dl	b: {b510, b705, b775, b865}
ρ_{path}uponρ_R (b)	Ratio ρ_{path}/ρ_R	o	dl	*idem*
ANNOT (j, f)	Annotation flag for the quality of the atmospheric correction	o	-	Coding: see 8.4.4
ACFAIL_F (j, f)	Flag indicating failure of the atmosphere correction	o	-	Boolean

Table 8.4.3.3.2-1: Parameters for the Path reflectance step

Copyright © 2011 ACRI S.T.

8.4.3.3.3 - Algorithm

```
ANNOT(j, f) = 0                                                         (2.6.9.1-1)
If ( z (j,f) > DEPTH_LIM ) then                                         (2.6.9.1-2)
    set bit SHALLOW of ANNOT(j,f))
Endif
For each b in b412..b865
    Correct Rayleigh reflectance for pressure variations
    (moved to section 8.3.3.2)                                          (2.6.9.1-3)
    Correct Rayleigh optical thickness for pressure variations
    (moved to section 8.3.3.2)                                          (2.6.9.1-4)
Endfor

Estimate $\rho_{path}$; $\rho_{path}$ is representative of the " reflectance above clear water ". At bands 775
and 865 it does include the water-leaving reflectance due to sediment.
Note: it is useless to extend the computation to 885 nm, since this band does not help
computing the atmospheric path reflectance.
For each b in {b510, b775, b865}
  If b == b510 then
    $\rho_{path}(b)$ = ROGC(b,j,f)                                      (2.6.9.1-5)
  Else
    $\rho_{path}(b)$ = ROGC(b,j,f) - tpw_C2(b,j,f)                      (2.6.9.1-6)
  Endif
exception processing: when ($\rho_{path}(b) <= \rho_{R0}(b)$ AND b in {b775, b865})
        ACFAIL_F(j,f) = TRUE
        process pixel according to "Exception: atmosphere correction failed" (§8.4.5)
end of exception processing
    If (abs($P_{ECMWF}(j,f) - P_{std}$) > PRESS_TOLERANCE) then
        $\rho_{path}(b)$ = correct_ΔP($\rho_{path}(b)$, $P_{std}$, $P_{ECMWF}(j, f)$, b, taua(b), MINUS)   (2.6.9.1-8)
The function correct_Δp is defined in 8.4.3.13.7 below.
    Endif
Form the ratio of the path reflectance upon Rayleigh reflectance
Note : the values of $\rho_R$ used in the equations below ARE NOT corrected for pressure
variations
    $\rho_{path}$upon$\rho_R(b) = \rho_{path}(b) / \rho_R(b, j, f)$      (2.6.9.1-9)
Endfor
```

Copyright © 2011 ACRI S.T.

8.4.3.4 – MERIS aerosol model (step 2.6.9.2)

8.4.3.4.1 - Functional description

This step is the core of the atmospheric correction above water for MERIS: it computes aerosol parameters for atmospheric correction taking full advantage of the MERIS spectral range and accuracy. Due to its complexity, it is further broken down in sub-steps; each sub-step is specified in a separate section below. The operation and parameters of the Aerosol model step are shown in figures 8.4.3.4.1-1 and 8.4.3.4.1-2 below.

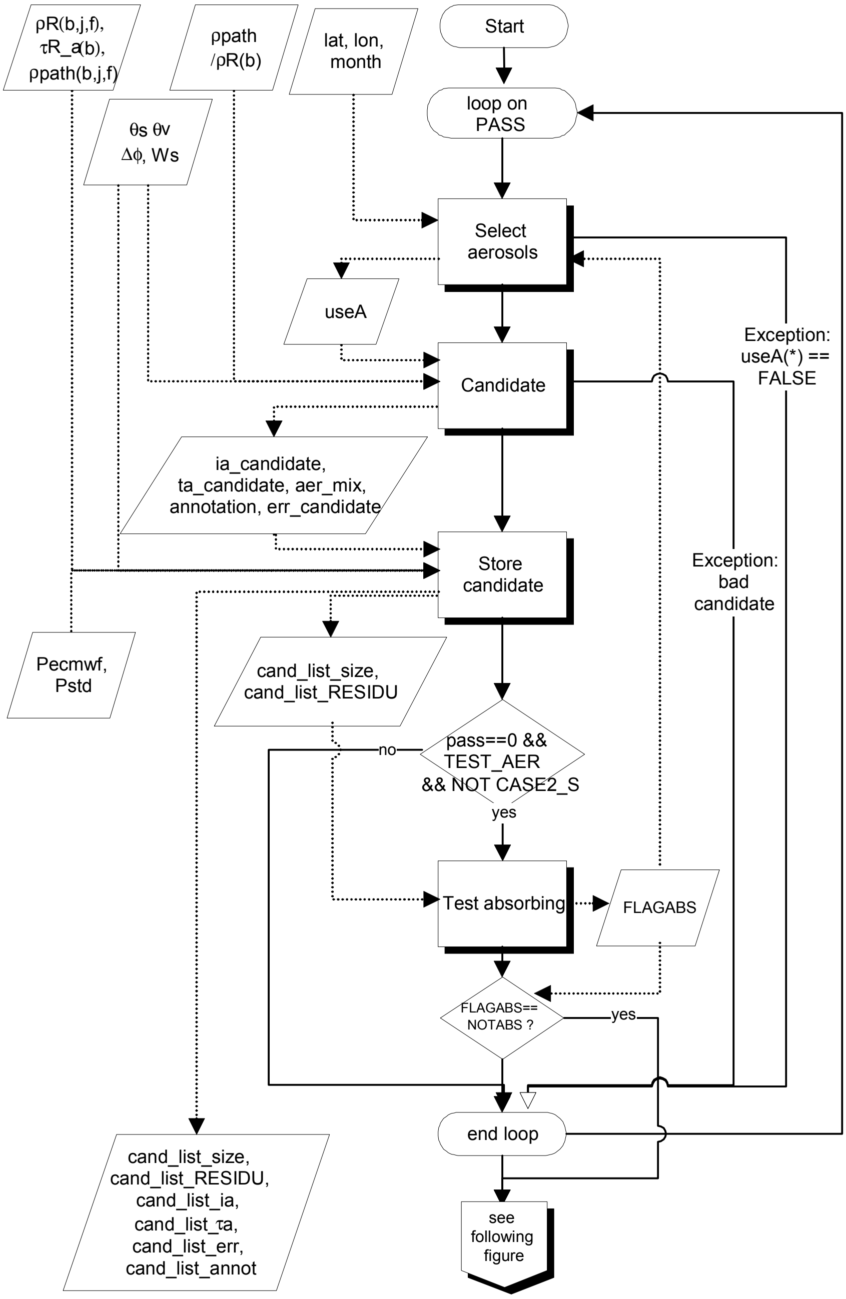


Figure 8.4.3.4.1-1: functional block diagram of MERIS aerosol model (step 2.6.9.2), part 1

Copyright © 2011 ACRI S.T.

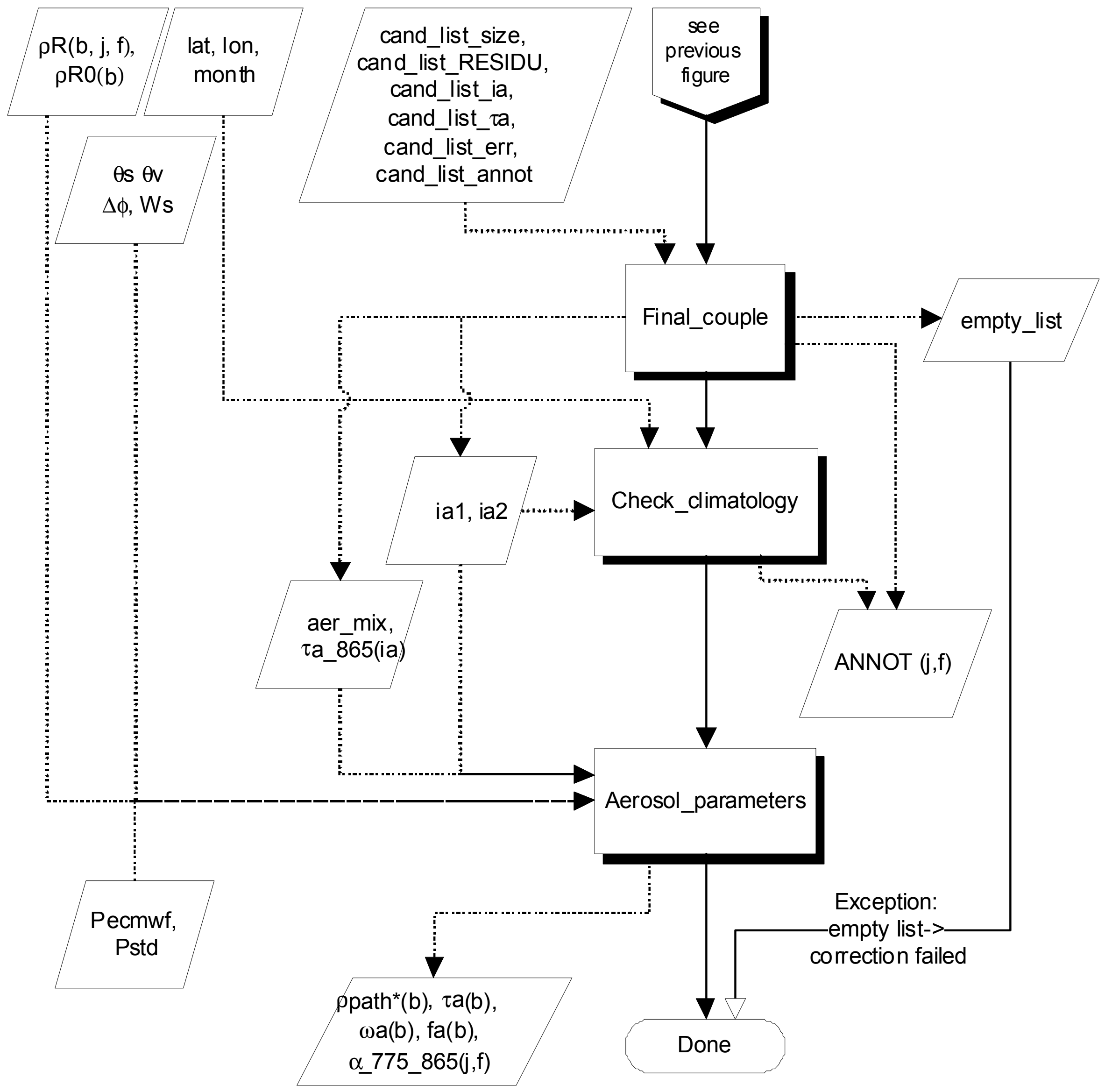


Figure 8.4.3.4.1-2: functional block diagram of MERIS aerosol model (step 2.6.9.2), part 2

Note: for the sake of clarity, step numbers are missing from the figures above. The correspondence between step identifier and number is found in §8.4.3.4.3 below.

Copyright © 2011 ACRI S.T.

8.4.3.4.2 - Input /Output

Symbol	Descriptive name	T	U	Range /Remarks
θ_S (j, f)	sun zenith angle	i	deg	
θ_V (j, f)	view zenith angle	i	deg	
$\Delta\phi$ (j, f)	azimuth difference	i	deg	
μ_S	Cosine of Sun zenith angle	i	dl	
μ	Cosine of view zenith angle	i	dl	
W_S (j, f)	wind speed modulus	i	m/s	
P_{ECMWF} (j, f)	ECMWF pressure	i	hPa	
lat (j, f)	pixel latitude	i	deg	
lon (j, f)	pixel longitude	i	deg	
CASE2_S (j, f)	Case 2 water flag	i	-	Boolean
ICE_HIGHAERO_F(j,f)	Flag for ice or high aerosol loading pixels	i	dl	Boolean
MEGLINT_F(j,f)	Flag for pixels corrected for glint	i	dl	Boolean
UNCGLINT_F(j,f)	Flag for pixels contaminated by glint	i	dl	Boolean
WHITE_SCATT_F(j,f)	Flag identifying “white” scatter within water	i	dl	Boolean
month	month of acquisition	i	dl	
$\rho_{path}upon\rho_R$ (b)	actual ratio of ρ_{path} to ρ_R	i	-	b: {b510, b705, b775, b865}
ρ_R (b, j, f)	Rayleigh reflectance	i	dl	b: {b412..b885}
ρ_{R0}(b)	Rayleigh reflectance corrected for pressure variations	i	dl	*idem*
τ_{R0}(b)	Optical thickness due to Rayleigh scattering	i	dl	*idem*
ρ_{path} (b)	Path reflectance	i	dl	b: {b510, b705, b775, b865}
FIRST_PASS, SECOND_PASS, THIRD_PASS, FOURTH_PASS, FIFTH_PASS	symbolic values for the number of each algorithm pass	s	-	See note [1]
N_PASSTOT	Number of passes in the algorithm	s	-	See note [2]
N_Aer	Number of aerosol models in database	s	-	
TEST_AER	Switch enabling the test for absorbing aerosol	s	-	Boolean
CLIMATO_AUX	Switch to activate the use of a climatology	s	-	Boolean
NOABSORBING	symbolic value to signal non-absorbing aerosols	s	-	See note [1]

Table 8.4.3.4.2-1: MERIS aerosol model parameters

---

[1] The value of the symbols is left to implementation as it is internal to the application and language dependant. For instance, a “C” program might use “enum” statement.

[2] The value of the symbols is left to implementation as it is internal to the application and language dependant. For instance, a “C” program might use “enum” statement.

Copyright © 2011 ACRI S.T.

Symbol	Descriptive name	T	U	Range /Remarks
LAST_PASS	Index of last algorithm pass	c	-	FIRST_PASS.. N_PASSTOT
ipass	Current pass within the algorithm	c	-	index of current pass
useA (ia)	flag indicating if an aerosol model is used by the atmosphere corrections	c	-	Boolean, ia: 1..N_Aer
ia_candidate (i)	candidate aerosol model pair	c	-	i: {LOW, HIGH}
τa_candidate (i)	aerosol optical thickness at 865nm for candidate aerosol model pair	c	dl	*idem*
err_candidate	error on ρ_{path}/ρ_R at 775nm	c	dl	
annotation	annotation flags	c	-	same coding as ANNOT(j, f)
bad_candidate	flag indicating failure of candidate selection	c	-	Boolean
FLAGABS	Flag indicating the presence of absorbing aerosols	c	-	
cand_list_size	current size of candidate list	c	-	
cand_list_mix(k)	list of aerosol model candidate pairs: mixing ratio	c	-	k <= N_PASSTOT
cand_list_ia(k, i)	list of aerosol model candidate pairs: aerosol model indices	c	-	k <= N_PASSTOT, i: {LOW, HIGH}
cand_list_τa(k, i)	list of aerosol model candidate pairs: aerosol optical thickness at 865nm	c	dl	*idem*
cand_list_RESIDU (k, b)	list of aerosol model candidate pairs: residual surface reflectance	c	dl	k <= N_PASSTOT, b: {b510, b705}
cand_list_annot (k)	list of aerosol model candidate pairs: annotation flags	c	-	k <= N_PASSTOT
empty_list	Flag indicating an empty list of candidates	c	-	Boolean
τa_865 (i)	Aerosol optical thickness at 865nm	c	dl	i: {LOW, HIGH}
$\rho^{*}_{path}(b)$	Path reflectance estimate	o	dl	b: {b412..b885}
$\tau_a(b, j, f)$	Aerosol optical thickness estimate	o	dl	*idem*
α_775_865(j, f)	Aerosol Angström exponent	o	dl	
$\omega_a(b)$	Aerosol single scattering albedo	o	dl	b: {b412..b885}
$f_a(b)$	Ratio of forward to total scattering	o	dl	b: {b412..b885}
ia1(j,f), ia2(j,f)	Index of bracketing aerosol models	o	-	
aer_mix (j, f)	Aerosol model mixing ratio	o	dl	
ACFAIL_F (j, f)	Flag indicating failure of the atmosphere correction	o	-	Boolean
ANNOT (j, f)	Annotation flag for the quality of the atmospheric correction	i/o	-	Coding: see 8.4.4

Table 8.4.3.4.2-1: MERIS aerosol model parameters (cont.)

Copyright © 2011 ACRI S.T.

8.4.3.4.3 - Algorithm

```
Initialisations
FLAGBS = NOABSORBING                                                  (2.6.9.2-1)
If (NOT CASE2_S(j, f)) then                                           (2.6.9.2-2)
    LAST_PASS = MIN (FIFTH_PASS, N_PASSTOT)
else
    LAST_PASS = FIRST_PASS
End if
cand_list_size = 0                                                    (2.6.9.2-3)
Multiple pass through the aerosol model data base
For ipass in FIRST_PASS..LAST_PASS
    annotation = ANNOT (j, f)                                         (2.6.9.2-4)
    activate Select aerosols (step 2.6.9.2.1)                         (2.6.9.2-5)
    exception processing: when (useA (i) == 0 for all i in 1..N_Aer): (2.6.9.2-6)
        skip the rest of this FOR loop and proceed with next ipass
    end of exception processing
    activate Candidate (step 2.6.9.2.2)                               (2.6.9.2-7)
    exception processing: when bad_candidate:                         (2.6.9.2-8)
        skip the rest of this FOR loop and proceed with next ipass
    end of exception processing
    activate Store candidate (step 2.6.9.2.3)                         (2.6.9.2-9)
    If ( (ipass == FIRST_PASS) AND TEST_AER AND NOT
       (CASE2_S(j, f) OR WHITE_SCATT_F(j,f) OR ICE_HIGHAERO_F(j,f)
      OR MEGLINT_F(j,f) OR UNCGLINT_F(j,f) )    then                  (2.6.9.2-10)
        activate Test absorbing (step 2.6.9.2.4)
    Endif
terminate loop when no absorbing aerosol has been detected
    If (FLAGABS == NOABSORBING) then                                  (2.6.9.2-11)
        break
     Endif
Endfor
Selection of optimal model
activate Final couple (step 2.6.9.2.5)                                (2.6.9.2-12)
exception processing: when empty_list == TRUE                         (2.6.9.2-13)
    ACFAIL_F (j, f) = TRUE
    skip the rest of this step;
    activate exception handling “atmosphere correction failed” (see 8.4.5 below)
end of exception processing
If (CLIMATO_AUX) then                                                 (2.6.9.2-14)
    activate Check climatology (step 2.6.9.2.6)
Endif
activate Aerosol parameters (step 2.6.9.2.7)                          (2.6.9.2-15)
For each b in {b510, b705, b775, b865} do
    taua(b) = $\tau_a$(b,j,f)                                         (2.6.9.2-16)
Endfor
```

Copyright © 2011 ACRI S.T.

8.4.3.5 - Select aerosols (step 2.6.9.2.1)

8.4.3.5.1 - Functional description

This algorithm step selects a set of aerosol models, depending on the flag “flagabs” (set at the first pass of the algorithm), on the pass of the algorithm, and (when applicable) on *a priori* climatological possibility. Its operation and parameters are shown in figure 8.4.3.5.1-1 below.

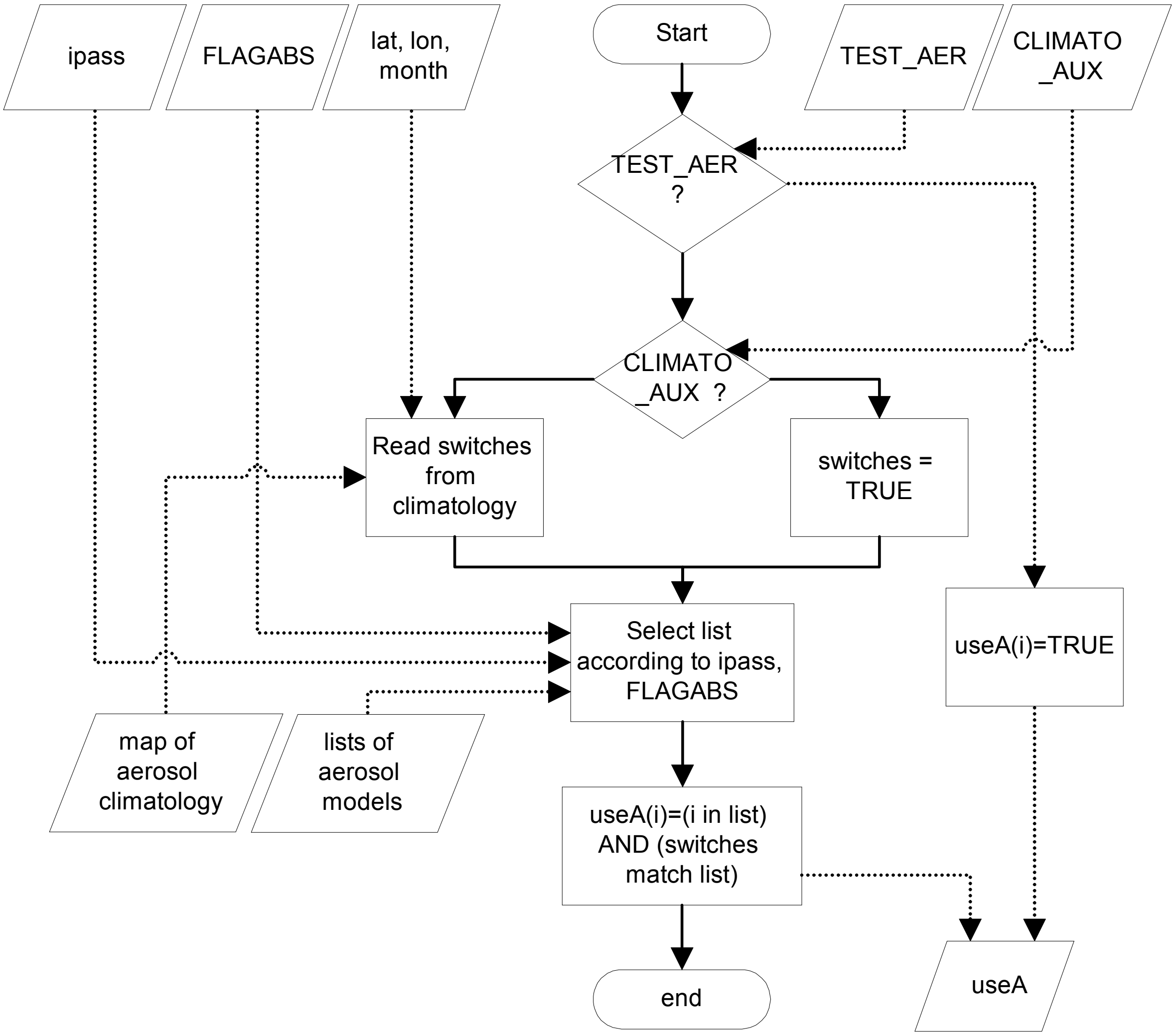


Figure 8.4.3.5.1-1: functional block diagram of step Select aerosols (2.6.9.2.1)

8.4.3.5.2 - Input /Output

Variable	Descriptive name	T	U	Range - References
ipass	Current pass within the algorithm	i	-	
FLAGABS	Flag indicating the presence of absorbing aerosols	i	-	
lat (j, f)	pixel latitude	i	deg	
lon (j, f)	pixel longitude	i	deg	
month	month of acquisition	i	dl	
CLIMATO_AUX	Switch to activate the use of a climatology	s	-	Boolean
N_aer	Number of aerosol models in database	s	dl	
N_basic_aer	Number of aerosol models per list	s	dl	
LIST_Aer_01	list of aerosol models	s	dl	
LIST_Aer_02	list of aerosol models	s	dl	
LIST_Aer_03	list of aerosol models	s	dl	
LIST_Aer_04	list of aerosol models	s	dl	
LIST_Aer_05	list of aerosol models	s	dl	
LIST_Aer_06	list of aerosol models	s	dl	
LIST_Aer_07	list of aerosol models	s	dl	
LIST_Aer_08	list of aerosol models	s	dl	
Aerclim_Ocean_LUT [lat, lon, month]	map of aerosol climatology	s	dl	month: 1..12, lat: -90..90, lon: -180..180; coding: see below
TEST_AER	Switch enabling the test for absorbing aerosol	s	-	Boolean
climato_switches	set of switches read from climatology	c	dl	
PRE_GEN	switch enabling generic aerosols	c	-	Boolean
PRE_BLUE	switch enabling blue aerosol assemblages	c	-	Boolean
PRE_DUST	switch enabling desert dust aerosol assemblages	c	-	Boolean
useA(ia)	flag indicating if an aerosol model is used by the atmosphere corrections	o	-	Boolean, ia: 1..N_Aer

Table 8.4.3.5.2-1: List of parameters for step Select Aerosols

Copyright © 2011 ACRI S.T.

Each element of Aerclim_Ocean_LUT is an array of 8 Boolean parameters, stored in one byte as follows (bit numbering follows the convention in AD5):

Description	Bit no	Used in
Switch enabling the use of generic aerosol models	0	Select Aerosols
not used	1	N/A
Switch enabling the use of "blue" aerosol assemblages	2	Select Aerosols
Switch enabling the use of desert dust-like aerosol assemblages	3	*idem*
Switch validating the detection of generic aerosol models	4	Check climatology
not used	5	N/A
Switch validating the detection of "blue" aerosol assemblages	6	Check climatology
Switch validating the detection of desert dust-like aerosol assemblages	7	*idem*

Table 8.4.3.5.2-2: Coding of Aerclim_Ocean_LUT

8.4.3.5.3 - Algorithm

```
If (NOT TEST_AER) then
    For ia=1..N_aer
        useA(ia) = TRUE                                                    (2.6.9.2.1-1)
    Endfor
Else
    If (CLIMATO_AUX) then
        climato_switches =
            Aerclim_Ocean_LUT nearest: (lat (j, f), lon (j, f), month)     (2.6.9-285)
        PRE_GEN = bit 0 of climato_switches                                (2.6.9.2.1-2)
        PRE_BLUE = bit 2 of climato_switches                               (2.6.9.2.1-3)
        deleted                        (2.6.9.2.1-4)
        PRE_DUST = bit 3 of climato_switches                               (2.6.9.2.1-5)
    Else
        PRE_GEN = TRUE                                                     (2.6.9.2.1-6)
        PRE_BLUE = TRUE                                                    (2.6.9.2.1-7)
        deleted                        (2.6.9.2.1-8)
        PRE_DUST = TRUE                                                    (2.6.9.2.1-9)
    Endif
    For (i=1..N_aer) useA(i) = FALSE endfor                                (2.6.9.2.1-10)

    If ( ipass == FIRST_PASS && FLAGABS == NOABSORBING && PRE_GEN) then
        For (1=1..N_basic_aer) useA(LISTE_aer_01(i)) = TRUE endfor         (2.6.9.2.1-11)
    Endif
    If ( FLAGABS == BLUE_LIKE) then
        If ( ipass == SECOND_PASS && PRE_BLUE) then
            For (i=1..N_basic_aer) useA(LISTE_aer_02(i))=TRUE endfor       (2.6.9.2.1-12)
```

Copyright © 2011 ACRI S.T.

```
                For (i=1..N_basic_aer) useA(LISTE_aer_01(i))=TRUE endfor       (2.6.9.2.1-13)
                deleted              (2.6.9.2.1-14) & (2.6.9.2.1-15)
            Endif
        Else if ( FLAGABS == DUST_LIKE ) then
            If ( ipass == SECOND_PASS && PRE_DUST) then
                For (i=1..N_basic_aer) useA(LISTE_aer_03(i)) = TRUE endfor     (2.6.9.2.1-16)
            Else if ( ipass == THIRD_PASS && PRE_DUST) then
                For (i=1..N_basic_aer) useA(LISTE_aer_04(i)) = TRUE endfor     (2.6.9.2.1-17)
            Else if ( ipass == FOURTH_PASS && PRE_DUST) then
                For (i=1..N_basic_aer) useA(LISTE_aer_05(i)) = TRUE endfor     (2.6.9.2.1-18)
            Endif
            deleted              (2.6.9.2.1-19)
        Endif
    Endif
```

Copyright © 2011 ACRI S.T.

8.4.3.6 - Candidate (step 2.6.9.2.2)

8.4.3.6.1 - Functional description

This algorithm step performs the selection of 2 bracketing aerosol models, on the criterion of the ratio ρ_{path} / ρ_R at 775nm. Its functional block diagram is shown in figure 8.4.3.6.1-1 below.

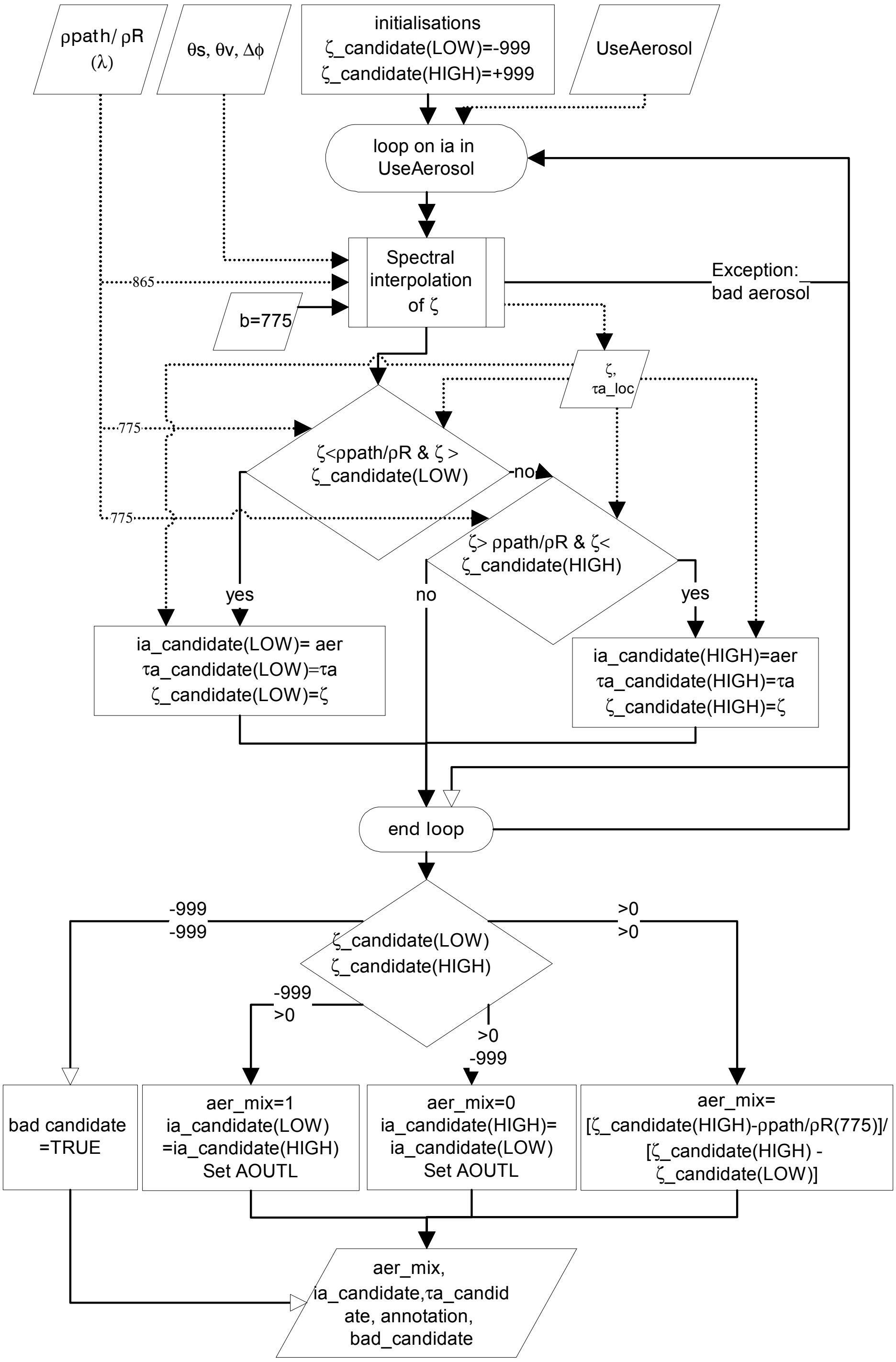


Figure 8.4.3.6.1-1: functional block diagram of step Candidate (2.6.9.2.2)

8.4.3.6.2 - Inputs /Outputs

Symbol	Descriptive name	T	U	Range /Remarks
useA (ia)	flag indicating if an aerosol model is used by the atmosphere corrections	i	-	ia: 1..N_Aer
ρ_{path}uponρ_R (b)	actual ratio of ρ_{path} to ρ_R	i	-	b: {b775, b865}
μ_s	Cosine of sun zenith angle	i	dl	
μ_v	Cosine of view zenith angle	i	dl	
$\Delta\phi$	azimuth difference	i	deg	
W_s	wind speed modulus	i	m/s	
bad_aerosol	flag indicating failure of aerosol computations	c	-	Boolean
ζ	estimate of ρ_{path} /ρ_R at 775nm	c	dl	
τ_a_865	local value of aerosol optical thickness at 865nm	c	dl	
ζ_candidate (i)	estimates of ρ_{path} /ρ_R at 775nm which bracket ρ_{path}uponρ_R(b775)	c	dl	i: {LOW, HIGH}
ia_candidate (i)	candidate aerosol model pair	o	-	i: {LOW, HIGH}
τa_candidate (i)	aerosol optical thickness at 865nm for candidate aerosol model pair	o	dl	*idem*
aer_mix	aerosol model mixing ratio	o	dl	
err_candidate	error on ρ_{path}/ρ_R at 775nm	o	dl	
annotation	annotation flags	o	-	same coding as ANNOT(j, f)
bad_candidate	flag indicating failure of candidate selection	o	-	Boolean

Table 8.4.3.6.2-1: List of variables for step Candidate

8.4.3.6.3 - Algorithm

```
Initialisations:
bad_candidate = FALSE                                              (2.6.9.2.2-1)
$\zeta$_candidate(LOW) = -999                                       (2.6.9.2.2-2)
$\zeta$_candidate(HIGH) = 999                                       (2.6.9.2.2-3)
For each ia such that (useA(ia))                                   (2.6.9.2.2-4)
    call spectral interpolation of $\zeta$ ($\rho_{path}$upon$\rho_R$(b865), ia, b775, $\mu_s$, $\mu_v$, $\Delta\phi$,
                                  $W_s$, $\zeta$ , $\tau_a$_865, bad_aerosol)
Note: the subroutine spectral interpolation of $\zeta$ is defined in 8.4.3.13.1 below.
    exception processing: when $\zeta < 0$ OR bad_aerosol            (2.6.9.2.2-5)
        skip the rest of this iteration and proceed to next aerosol
    end of exception processing
    If ($\zeta <= \rho_{path}$upon$\rho_R$(b775)) AND ($\zeta >$ $\zeta$_candidate(LOW)) then
        $\zeta$_candidate(LOW) = $\zeta$                            (2.6.9.2.2-6)
        ia_candidate (LOW) = ia                                    (2.6.9.2.2-7)
        $\tau_a$_candidate (LOW)= $\tau_a$_865
    (2.6.9.2.2-8)
    else if ($\zeta >= \rho_{path}$upon$\rho_R$(b775)) AND ($\zeta <$ $\zeta$_candidate(HIGH)) then
```

Copyright © 2011 ACRI S.T.

```
        $\zeta$_candidate(HIGH) = $\zeta$
    (2.6.9.2.2-9)
        ia_candidate (HIGH) = ia                                          (2.6.9.2.2-10)
        $\tau_a$_candidate (HIGH)= $\tau_a$_865                           (2.6.9.2.2-11)
    endif
Endfor

Calculate the aerosol mixing ratio
If ($\zeta$_candidate(LOW) > 0) AND ($\zeta$_candidate(HIGH) < 999) then
    $aer\_mix = \frac{\rho_{path} upon \rho_R (b775) - \zeta\_candidate(LOW)}{\zeta\_candidate(HIGH) - \zeta\_candidate(LOW)}$    (2.6.9.2.2-12)
    err_candidate = 0                                                     (2.6.9.2.2-13)
Else
    set bit MIXR1 of annotation                                           (2.6.9.2.2-14)
    If ($\zeta$_candidate(HIGH) < 999) then
        ia_candidate (LOW) = ia_candidate (HIGH)                          (2.6.9.2.2-15)
        $\tau$a_candidate (LOW) = $\tau$a_candidate (HIGH)                (2.6.9.2.2-16)
        aer_mix = 1                                                       (2.6.9.2.2-17)
        err_candidate = $\zeta$_candidate(HIGH) - $\rho_{path}$upon$\rho_R$(b775)    (2.6.9.2.2-18)
    else if ($\zeta$_candidate(LOW) > 0) then
        ia_candidate (HIGH) = ia_candidate (LOW)                          (2.6.9.2.2-19)
        $\tau$a_candidate (HIGH) = $\tau$a_candidate (LOW)                (2.6.9.2.2-20)
        aer_mix = 0                                                       (2.6.9.2.2-21)
        err_candidate = $\rho_{path}$upon$\rho_R$(b775) - $\zeta$_candidate(LOW)     (2.6.9.2.2-22)
    else
        bad candidate = TRUE                                              (2.6.9.2.2-23)
    Endif
Endif
```

Copyright © 2011 ACRI S.T.

8.4.3.7 - Store candidate models (step 2.6.9.2.3)

8.4.3.7.1 – Functional Description

This algorithm step stores a pair of aerosol models, with meaningful parameters, into a list from which an optimal candidate pair will be selected by a further step. The list is implemented as a counter cand_list_size and arrays cand_list_mix, cand_list_ia, cand_list_τ_a, cand_list_RESIDU, cand_list_err, cand_list_annot. Its operation and parameters are shown in figure 8.4.3.7.1-1 below.

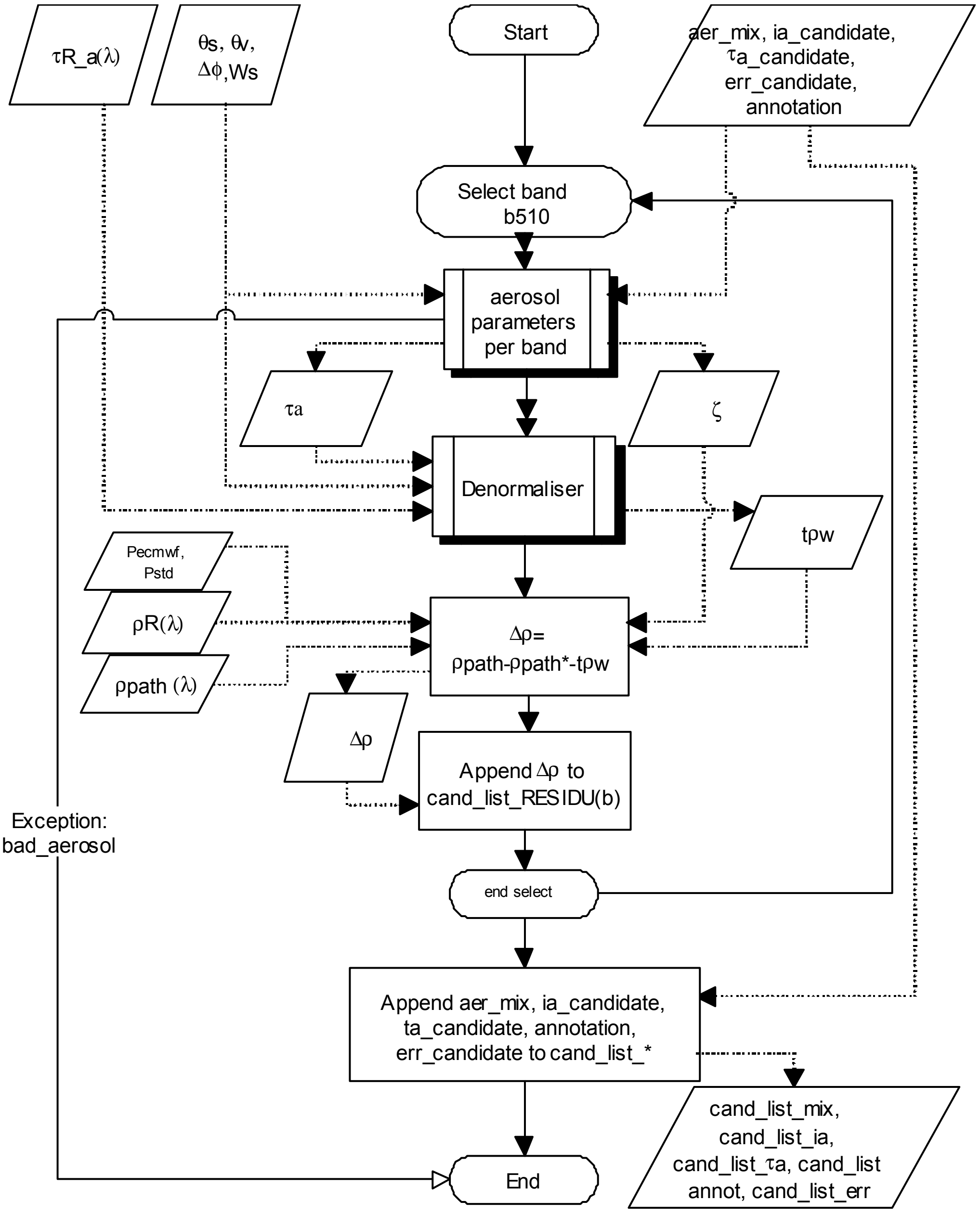


Figure 8.4.3.7.1-1: functional block diagram of step Store candidate models (2.6.9.2.3)

Copyright © 2011 ACRI S.T.

8.4.3.7.2 - Inputs /Outputs

Variable	Descriptive name	T	U	Range - References
JD1, JD2	UTC times of first and last frames in product	i	jd	From Level 1b
month	current month (counting from 1) for measurements	i	-	From Level 1b
JDCurrMonth, JDNextMonth	UTC times of mid-month day at noon for current month, and following month	i	jd	From Level 1b
lat	Latitude of current pixel	i	deg	
lon	Longitude of current pixel	i	deg	
$\Delta\phi$	Azimuth difference	i	deg	
W_s	Wind speed	i	$m.s^{-1}$	
CASE2_S	Case 2 water flag	i	-	Boolean
μ_s	Cosine of Sun zenith angle	i	dl	
μ	Cosine of view zenith angle	i	dl	
ρ_{path} (b)	Actual path reflectance	i	dl	b: {b510 }
ρ_R (b)	Rayleigh reflectance	i	dl	*idem*
τ_{R0} (b)	Optical thickness due to Rayleigh scattering	i	dl	*idem*
ia_candidate (i)	candidate aerosol model pair	i	-	i: {LOW, HIGH}
τa_candidate (i)	aerosol optical thickness at 865nm for candidate aerosol model pair	i	dl	*idem*
aer_mix	aerosol model mixing ratio	i	dl	
err_candidate	error on ρ_{path}/ρ_R at 775nm	i	dl	
annotation	annotation flags	i	-	
PRESS_ TOLERANCE	Threshold to activate a correction for pressure	s	hPa	
ROW_510_MEAN_LUT	Climatology giving mean ρ_w at 510nm	s	dl	
bad_aerosol	Flag indicating failure of aerosol computations	c	-	Boolean
DateWeight	Date weighting factor	c	-	
ζ (b)	Estimate of ρ_{path} /ρ_R	c	dl	b: {b510 }
τ_a(b)	Aerosol optical thickness	c	dl	*Idem*
t_u(b)	Transmittance on the target-sensor path	c	dl	b: {b510 }
t_d(b)	Transmittance on the sun-target path at	c	dl	*Idem*
ρ_{path}^*(b)	Estimate of ρ_{path}	c	dl	*idem*
$T\rho_w$(b)	Estimate of marine reflectance	c	dl	*idem*

Table 8.4.3.7.2-1: Parameters for the Store candidate models step

Copyright © 2011 ACRI S.T.

Variable	Descriptive name	T	U	Range - References
ROW_510_MEAN	mean ρ_w at 510nm for current pixel	c	dl	
cand_list_size	aerosol model candidate list size	i/o	-	
cand_list_mix(k)	list of aerosol model candidate pairs: mixing ratio	i/o	-	k <= N_PASSTOT
cand_list_ia(k, i)	list of aerosol model candidate pairs: aerosol model indices	i/o	-	k <= N_PASSTOT, i: {LOW, HIGH}
cand_list_τa(k, i)	list of aerosol model candidate pairs: aerosol optical thickness at 865nm	i/o	dl	*idem*
cand_list_RESIDU (k, b)	list of aerosol model candidate pairs: residual surface reflectance	i/o	dl	k <= N_PASSTOT, b: {b510, b705}
cand_list_err (k)	list of aerosol model candidate pairs: error on ρ_{path}/ρ_R at 775nm	i/o	-	k <= N_PASSTOT
cand_list_annot (k)	list of aerosol model candidate pairs: annotation flags	i/o	-	*idem*

Table 8.4.3.7.2-1: Parameters for the Store candidate models step (cont.)

8.4.3.7.3 - Algorithm

```
cand_list_size = cand_list_size + 1                                        (2.6.9.2.3-1)
Estimate the residual of the atmosphere correction at 510 nm
Let b = b510
Compute aerosol parameters at 510 nm for the bracketing aerosol models
call Aerosol parameters per band (ia_candidate(LOW),
        ia_candidate(HIGH), b, $\tau$a_candidate(*), aer_mix, $\mu_s$, $\mu_v$, $\Delta\phi$, $W_s$,
        $\zeta(b)$, $\tau_a(b)$, $\omega_a(b)$, $f_a(b)$, bad_aerosol)                        (2.6.9.2.3-2)
Note: the subroutine Aerosol parameters per band is defined in 8.4.3.13.4.
exception processing: when bad_aerosol == TRUE:                            (2.6.9.2.3-3)
    do not append candidate pair to list
    cand_list_size = cand_list_size - 1
    skip the rest of step 2.6.9.2.3
end of exception processing

$\rho_{path}^*(b) = \zeta(b) * \rho_R(b,j,f)$                                          (2.6.9.2.3-4)
If ( abs($P_{ECMWF}(j,f) - P_{std}$) > PRESS_TOLERANCE) then                     (2.6.9.2.3-5)
    $\rho_{path}^*(b)$ = correct_$\Delta$P($\rho_{path}^*(b)$, $P_{std}$, $P_{ECMWF}(j, f)$, b, $\tau_a(b)$, PLUS)
Endif
Estimate marine reflectance t$\rho$w510 according to CASE2_S
If (NOT CASE2_S(j,f)) then                                                 (2.6.9.2.3-6)
    If (cand_list_size == 1) then
        $\tau_a(b865) = \tau_{a\_candidate}(LOW)*(1-aer\_mix) + \tau_{a\_candidate}(HIGH)*aer\_mix$
```

```
        DateWeight=((JD1+JD2)/2-JDCurrMonth)(JDNextMonth-JDCurrMonth)(2.6.9.2.3-6-DateWeight)[1]
        row_510_mean =                                                  (2.6.9.2.3-6-row_510_mean)
            (1- DateWeight)   * row_510_mean_LUT interpol: (lat, lon) select: (month)+
            DateWeight        * row_510_mean_LUT interpol: (lat, lon) select: ((month+1) mod 12)

        Compute diffuse transmittance at 510 nm
        $t_u(b)$ = transmittance_up(ia_candidate(LOW), ia_candidate(HIGH),
                         b510, $\tau_{R0}$, $\tau_a$_candidate, aer_mix, $\mu$)        (2.6.9.2.3-15)
        $t_d(b)$ = transmittance_d(ia_candidate(LOW), ia_candidate(HIGH),
                         b510, $\tau_{R0}$ $\tau_a$_candidate, aer_mix, $\mu_s$, $W_s$)   (2.6.9.2.3-16)
NOTE: the functions transmittance_up is and transmittance_d are defined respectively in
8.4.3.13.5 and 8.4.3.13.10.

        $t\rho_w(b)$ = Denormaliser($t_u(b)$*$t_d(b)$*row_510_mean, $\theta_s$, $\theta_v$, $\Delta\phi$, b510, $\tau_a(b865)$, $W_s$)
Note: the function Denormaliser is defined in 8.4.3.13.6 below.
        store $t\rho_w(b)$ in memory
    Else
        retrieve $t\rho_w(b)$ from memory
    Endif
        deleted                 (2.6.9.2.3-7)
Else
    $t\rho_w(b) = 0$
Endif
$\Delta\rho(b) = \rho_{path}(b) - \rho_{path}^*(b) - t\rho_w(b)$
    (2.6.9.2.3-8)
cand_list_RESIDU (cand_list_size, b) = $\Delta\rho(b)$                      (2.6.9.2.3-9)

cand_list_mix (cand_list_size) = aer_mix                               (2.6.9.2.3-10)
cand_list_annot (cand_list_size) = annotation                          (2.6.9.2.3-11)
cand_list_err (cand_list_size) = err_candidate
    (2.6.9.2.3-12)
cand_list_ia (cand_list_size, *) = ia_candidate (*)                    (2.6.9.2.3-13)
cand_liste_$\tau$a (cand_list_size, *) = $\tau$a_candidate (*)                  (2.6.9.2.3-14)
```

---

[1] *Note that the computation of DateWeight gives the same result for all pixels of the image, and may therefore be computed only once during the processing*

8.4.3.8 - Test absorbing aerosol (step 2.6.9.2.4)

8.4.3.8.1 – Functional description

This algorithm step compares the residual reflectance at bands 510 and 705nm with thresholds to detect absorbing aerosols, whether of continental or desert dust type. Its operation and parameters are shown in figure 8.4.3.8.1-1 below.

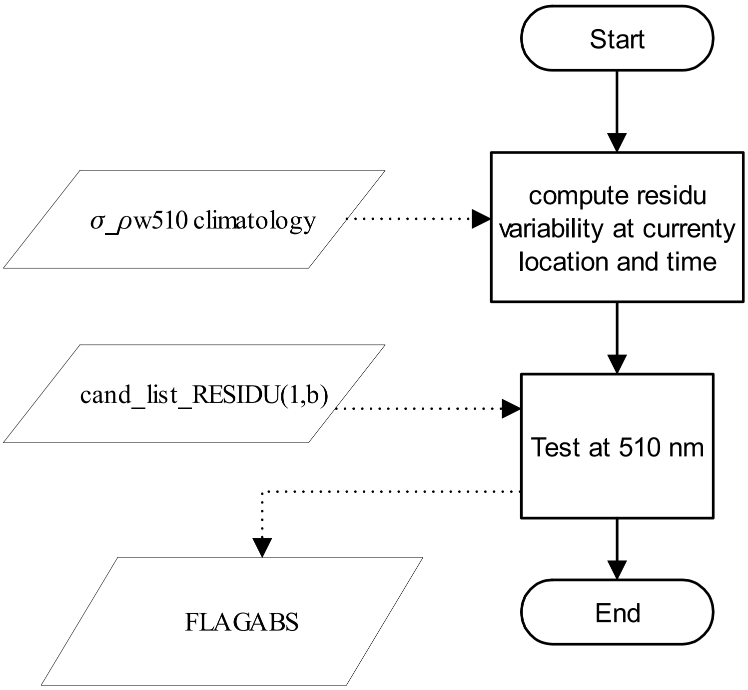


Figure 8.4.3.8.1-1: functional block diagram of step Test absorbing aerosol (2.6.9.2.4)

Copyright © 2011 ACRI S.T.

8.4.3.8.2 - Inputs /Outputs

Symbol	Descriptive name	T	U	Range /Remarks
JD1, JD2	UTC times of first and last frames in product	i	jd	From Level 1b
month	current month (counting from 1) for measurements	i	-	From Level 1b
JDCurrMonth, JDNextMonth	UTC times of mid-month day at noon for current month, and following month	i	jd	From Level 1b
lat	Latitude of current pixel	i	deg	
lon	Longitude of current pixel	i	deg	
JD1, JD2	UTC times of first and last frames in product	i	jd	From Level 1b
cand_list_size	aerosol model candidate list size	i	-	
cand_list_RESIDU (k, b)	list of aerosol model candidate pairs: residual surface reflectance	i	dl	k <= N_PASSTOT, b: {b510, b705}
cand_list_annot (k)	list of aerosol model candidate pairs: annotation flags	i/o	-	k <= N_PASSTOT
NOABSORBING	symbolic value to signal non-absorbing aerosols	s	-	See note [1]
DUST_LIKE	symbolic value to signal desert dust absorbing aerosols	s	-	See note [1]
BLUE_LIKE	symbolic value to signal “blue” aerosols	s	-	See note [1]
row_510_sigma_LUT	Climatology giving ρ_w variability at 510nm	s	dl	
DRO510_thresh_D	threshold for the absorbing aerosol test at 510nm	s	dl	
DRO510_thresh_B	threshold for the blue aerosol test at 510nm	s	dl	
DateWeight	Date weighting factor	c	-	
FLAGABS	Flag indicating the presence of absorbing aerosols	o	-	{NOABSORBING, DUST_LIKE, BLUE_LIKE}

Table 8.4.3.8.2–1: Parameters for step Test absorbing aerosol

---

[1] The value of the symbols is left to implementation as it is internal to the application and language dependant. For instance, a “C” program might use “enum” statement.

Copyright © 2011 ACRI S.T.

8.4.3.8.3 - Algorithm

Note that when this step is performed cand_list_size == 1.

Test at 510 nm

Note that the computation of DateWeight gives the same result for all pixels of the image, and may therefore be computed only once during the processing.

```
DateWeight=((JD1+JD2)/2-JDCurrMonth)(JDNextMonth-JDCurrMonth)          (2.6.9.2.4-12)
row_510_sigma    =                                                      (2.6.9.2.4-13)
    (1- DateWeight)  * row_510_sigma_LUT interpol: (lat, lon) select: (month)
  +     DateWeight   * row_510_sigma_LUT interpol: (lat, lon) select: ((month+1) mod 12)

If ( cand_list_RESIDU(cand_list_size, b510)  > + (DRO510_thresh_B+row_510_sigma) ) then
    FLAGABS = BLUE_LIKE                                                 (2.6.9.2.4-1)
Else if (cand_list_RESIDU(cand_list_size, b510) < – (DRO510_thresh_D+row_510_sigma)) then
    FLAGABS = DUST_LIKE                                                 (2.6.9.2.4-2)
Else
    FLAGABS = NOABSORBING                                               (2.6.9.2.4-3)
Endif
```

(2.6.9.2.4-4) to (2.6.9.2.4-8): deleted

deleted *(2.6.9.2.4-9)*
deleted *(2.6.9.2.4-10)*
deleted *(2.6.9.2.4-11)*

8.4.3.9 - Final couple (step 2.6.9.2.5)

8.4.3.9.1 - Functional description

This step selects among the list of aerosol model candidate pairs, that which provides the best fitting with the observed signal. Its operation is shown schematically in figure 8.4.3.9.1-1 below.

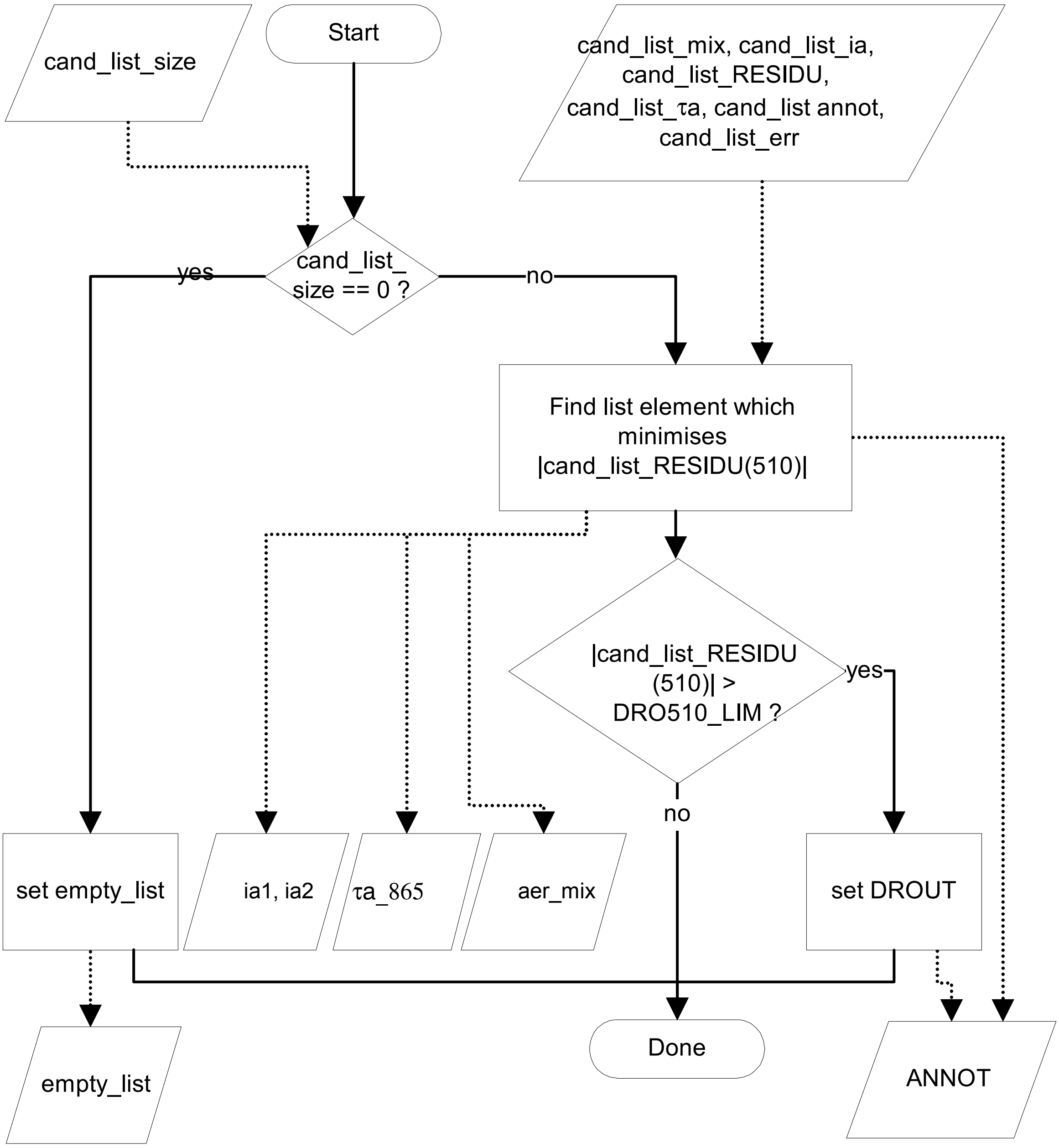


Figure 8.4.3.9.1-1: functional block diagram of step Final couple (2.6.9.2.5)

Copyright © 2011 ACRI S.T.

8.4.3.9.2 - Inputs /Outputs

Variable	Descriptive name	T	U	Range – References
cand_list_size	aerosol model candidate list size	i	-	
cand_list_mix (k)	list of aerosol model candidate pairs: mixing ratio	i	-	k <= N_PASSTOT
cand_list_ia (k, i)	list of aerosol model candidate pairs: aerosol model indices	i	-	k <= N_PASSTOT, i: {LOW, HIGH}
cand_list_τa (k, i)	list of aerosol model candidate pairs: aerosol optical thickness at 865nm	i	dl	*idem*
cand_list_RESIDU (k, b)	list of aerosol model candidate pairs: residual surface reflectance	i	dl	k <= N_PASSTOT, b: {b510}
cand_list_err (k)	list of aerosol model candidate pairs: error on ρ_{path}/ρ_R at 775nm	i	-	k <= N_PASSTOT
cand_list_annot (k)	list of aerosol model candidate pairs: annotation flags	i	-	k <= N_PASSTOT
DRO510_LIM	Value of Δρ510 to set the annotation flag	s	dl	
LIST_BLUE	list of blue aerosol models	s	-	
resmin510	Current value of smallest error at 510nm	c	dl	
Δρ510	Current value of error at 510nm	c	dl	
annot_temp	Local value of annotation flag	c	-	
empty_list	Flag indicating an empty list of candidates	o	-	Boolean
ia1 (j, f)	First bracketing aerosol model	o	-	
ia2 (j, f)	Second bracketing aerosol model	o	-	
τa_865 (i)	Aerosol optical thickness at 865nm	o	dl	i: {LOW, HIGH}
aer_mix (j, f)	Mixing ratio between bracketing aerosol models	o	dl	
ANNOT (j, f)	Annotation flag for the quality of the atmospheric correction	i/o	dl	Coding: see 8.4.4

Table 8.4.3.9.2–1: Parameters for step Final couple

8.4.3.9.3 - Algorithm

```
resmin510=99999.
empty_list = (cand_list_size == 0)                                    (2.6.9.2.5-1)
If (NOT empty_list) then
    For (candidate = 1.. cand_list_size)
        $\Delta\rho510 = |$ cand_list_RESIDU (candidate, b510) $|$          (2.6.9.2.5-3)
        deleted     (2.6.9.2.5-3bis)



        If (    $\Delta\rho510 <$ resmin510 ) then
        annot_temp = ANNOT(j, f) OR cand_list_annot (candidate)       (2.6.9.2.5-2)
```

Copyright © 2011 ACRI S.T.

```
                $resmin510 = \Delta\rho510$                                   (2.6.9.2.5-4)
                deleted                          (2.6.9.2.5-4bis)
                ia1 = cand_list_ia (candidate, LOW)                            (2.6.9.2.5-5)
                ia2 = cand_list_ia (candidate, HIGH)                           (2.6.9.2.5-6)
                $\tau a\_865(LOW) = cand\_list\_\tau a$ (candidate, LOW)       (2.6.9.2.5-7)
                $\tau a\_865(HIGH) = cand\_list\_\tau a$ (candidate, HIGH)     (2.6.9.2.5-8)
                aer_mix = cand_list_mix (candidate)                            (2.6.9.2.5-9)
            Endif
        Endfor
        ANNOT(j, f) = annot_temp                                               (2.6.9.2.5-10)
        If ( resmin510 > DRO510_LIM ) then
            set bit DROUT of ANNOT(j, f))                                      (2.6.9.2.5-11)
        Endif
        If (ia1 ∈ LIST_BLUE OR ia2 ∈ LIST_BLUE) then
          Set bit AERO_B of ANNOT(j,f)                                         (2.6.9.2.5-12)
        Endif
        If (ia1 ∈ LIST_aer_03 OR ia2 ∈ LIST_aer_03 OR
          ia1 ∈ LIST_aer_04 OR ia2 ∈ LIST_aer_04 OR
          ia1 ∈ LIST_aer_05 OR ia2 ∈ LIST_aer_05) then
          Set bit ABSO_D of ANNOT(j,f)                                         (2.6.9.2.5-13)
        Endif
Endif
```

8.4.3.10 - Check climatology (step 2.6.9.2.6)

8.4.3.10.1 - Functional description

This step verifies the compatibility of an aerosol model pair with a climatology. Its operation is shown schematically in figure 8.4.3.10.1-1 below.

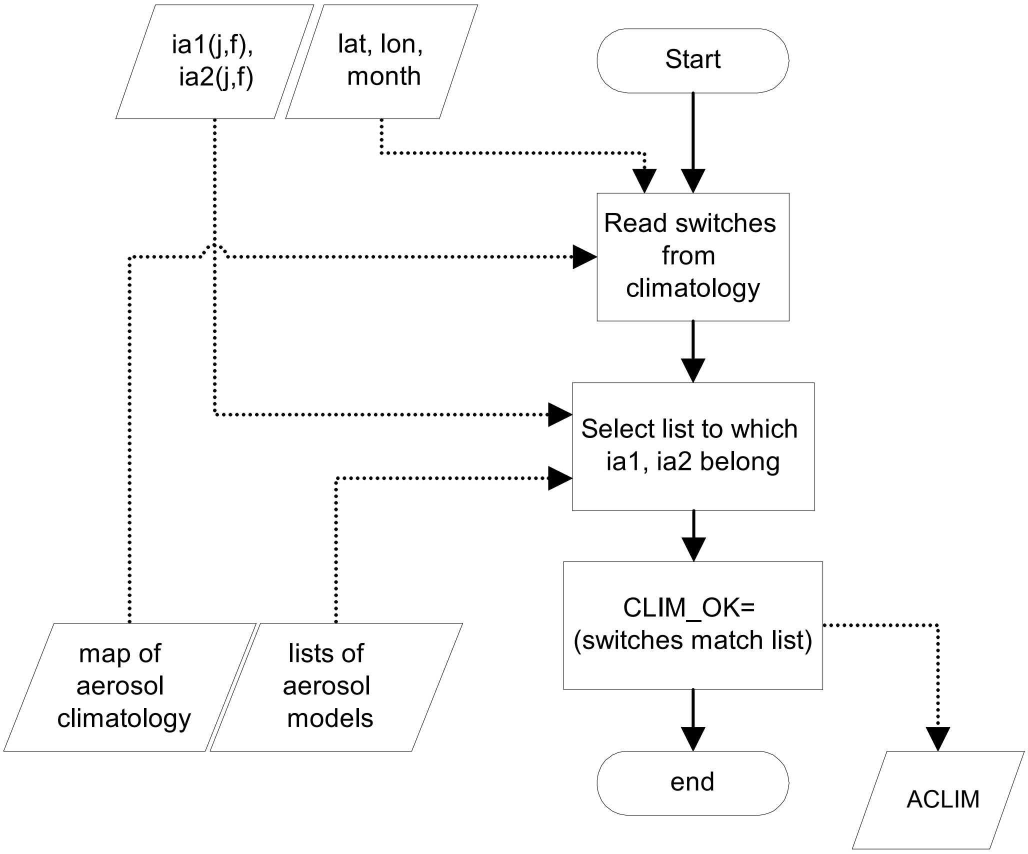


Figure 8.4.3.10.1-1: functional block diagram of step Check climatology (2.6.9.2.6)

Copyright © 2011 ACRI S.T.

8.4.3.10.2 - Inputs /Outputs

Variable	Descriptive name	T	U	Range - References
ia1 (j, f)	first bracketing aerosol model index	i	dl	
ia2 (j, f)	second bracketing aerosol model index	i	dl	
lat (j, f)	pixel latitude	i	deg	
lon (j, f)	pixel longitude	i	deg	
month	month of acquisition	i	-	
LIST_Aer_01	list of aerosol models	s	-	
LIST_Aer_03	list of aerosol models	s	-	
LIST_Aer_04	list of aerosol models	s	-	
LIST_Aer_05	list of aerosol models	s	-	
LIST_BLUE	list of blue aerosol models	s	-	
Aerclim_Ocean_LUT (lat, lon, month)	map of aerosol climatology	s	-	see Select Aerosols (step 2.6.9.2.1) above
climato_switches	set of switches read from climatology	c	-	Boolean
POST_GEN	switch enabling maritime aerosols	c	-	Boolean
POST_BLUE	switch enabling blue aerosol assemblages	c	-	Boolean
POST_DUST	switch enabling desert dust aerosol assemblages	c	-	Boolean
CLIM_OK	flag indicating agreement between ia and climatology	c	-	Boolean
ANNOT (j, f)	Annotation flag for the quality of the atmospheric correction	i/o	-	

Table 8.4.3.10.2-1: List of variables for step Check climatology

8.4.3.10.3 - Algorithm

```
CLIM_OK = TRUE                                                          (2.6.9.2.6-1)
climato_switches = Aerclim_Ocean_LUT nearest: (lat (j, f), lon (j, f), month)
POST_MAR = bit 4 of climato_switches                                    (2.6.9.2.6-2)
deleted                     (2.6.9.2.6-3)
POST_BLUE = bit 6 of climato_switches                                   (2.6.9.2.6-4)
POST_DUST = bit 7 of climato_switches                                   (2.6.9.2.6-5)
If ((ia1 in LISTE_aer_01 OR ia2 in LISTE_aer_01) AND (NOT POST_GEN)) then
    CLIM_OK = FALSE                                                     (2.6.9.2.6-6)
deleted                                 (2.6.9.2.6-7)
```

Copyright © 2011 ACRI S.T.

```
Else if ((ia1 in LIST_BLUE OR ia2 in LISTE_BLUE) AND (NOT POST_BLUE)) then
    CLIM_OK = FALSE                                                    (2.6.9.2.6-8)
Else if ((ia1 in LISTE_aer_03 OR ia1 in LISTE_aer_04 OR ia1 in LISTE_aer_05 OR
          ia2 in LISTE_aer_03 OR ia2 in LISTE_aer_04 OR ia2 in LISTE_aer_05) AND
       (NOT POST_DUST)) then
    CLIM_OK = FALSE                                                    (2.6.9.2.6-9)
Endif
If (NOT CLIM_OK) then
    set bit ACLIM of ANNOT(j, f))                                      (2.6.9.2.6-10)
Endif
```

8.4.3.11 - Aerosol parameters (step 2.6.9.2.7)

8.4.3.11.1 - Functional description

This step computes, for the finally selected aerosol model pair, the useful aerosol parameters for atmospheric correction. Its operation is shown schematically in figure 8.4.3.11.1-1 below.

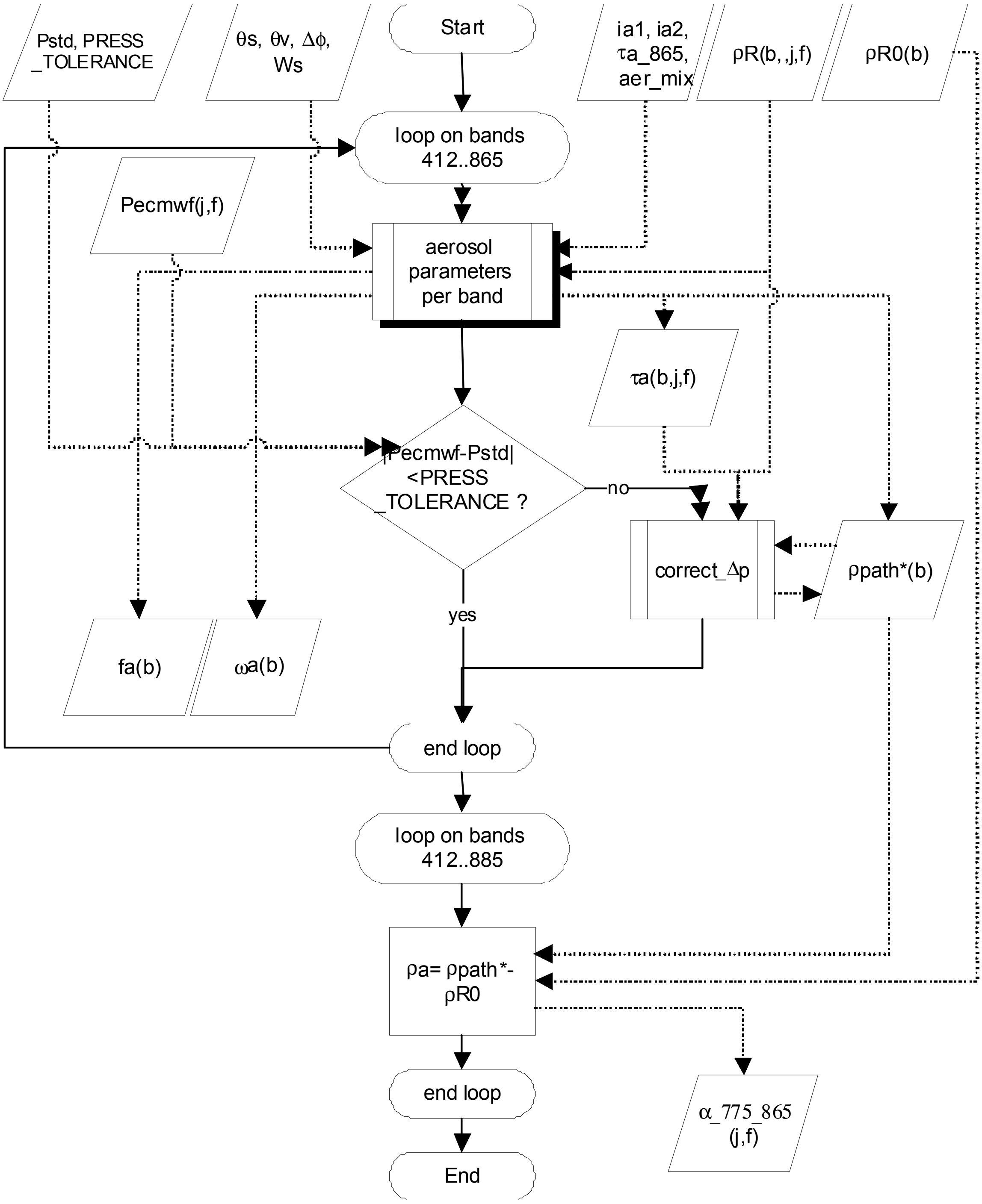


Figure 8.4.3.11.1-1: functional block diagram of step Aerosol parameters (2.6.9.2.7)

8.4.3.11.2 - Inputs /Outputs

Variable	Descriptive name	T	U	Range – References
μ_s	Cosine of sun zenith angle	i	dl	
μ_v	Cosine of view zenith angle	i	dl	
$\Delta\phi$	Azimuth difference	i	deg	
W_s	wind speed modulus	i	m/s	
ia1 (j, f), ia2 (j, f)	Index of bracketing aerosol models	i	-	
τ_a_865 (i)	Aerosol optical thickness at 865nm	i	dl	i: {LOW, HIGH}
aer_mix (j, f)	Aerosol model mixing ratio	i	dl	
ρ_R (b, j, f)	Rayleigh reflectance for all bands	i	dl	b: {b412..b885}
ρ_{R0} (b)	Rayleigh reflectance corrected for pressure variations	i	dl	*idem*
ROGC(b,j,f)	Glint corrected reflectance	i	dl	b: {b412..b885}
$t\rho w$_C2(b,j,f)	Marine reflectance at TOA for Case 2 waters	i	dl	*b: {b705, b775, b865, b885}*
P_{ECMWF} (j, f)	ECMWF pressure	i	hPa	
PRESS_ TOLERANCE	Threshold to activate a correction for pressure	s	hPa	
P_{std}	Standard value of the surface pressure	s	hPa	
TAUA865_ threshold	Threshold for flagging the aerosol optical thickness	s	dl	
λ_{theo}(b)	Theoretical wavelengths corresponding to smile corrected reflectances	s	nm	
ζ(b)	Ratio ρ_{path}/ρ_R at band b	c	dl	b: {b412..b885}
bad_aerosol	Flag indicating the failure of aerosol computations	c	-	Boolean
ρ_a(b)	Aerosol reflectance	c	dl	b: {b412..b885} to Breakpoint
ρ^*_{path}(b)	Path reflectance estimate	o	dl	*idem*
τ_a(b, j, f)	Aerosol optical thickness estimate	o	dl	*idem*
ω_a(b)	Aerosol single scattering albedo	o	dl	*idem*
f_a(b)	Ratio of forward to total scattering	o	dl	*idem*
α_775_865 (j, f)	Aerosol Angström exponent	o	dl	
ANNOT (j, f)	Annotation flag for the quality of the atmospheric correction	o	-	Coding described in 8.4.4 below

Table 8.4.3.11-1: Parameters for the Aerosol parameters step

8.4.3.11.3 - Algorithm

Compute the aerosol optical thickness at 865 nm (one of the outputs of the atmospheric corrections

```
$\tau_a$ (b865, j, f) = $\tau_a$_865(LOW) * (1 - aer_mix) + $\tau_a$_865(HIGH) * aer_mix        (2.6.9.2.7-1)
If ( $\tau_a$ (b865, j, f) > TAUA865_threshold ) then
        set bit TAU06 of ANNOT(j, f))                                                    (2.6.9.2.7-2)
Endif
Compute aerosol optical thickness at bands 412 to 885nm for the bracketing aerosol models
For each b in {b412..b885}
        call Aerosol parameters per band (ia1, ia2, b, $\tau_a$_865, aer_mix, $\mu_s$, $\mu_v$, $\Delta\phi$, $W_s$, $\zeta$ (b), $\tau_a$
(b, j, f), $\omega_a$ (b), $f_a$ (b), bad_aerosol)                                               (2.6.9.2.7-3)
Note: the subroutine Aerosol parameters per band is defined in 8.4.3.13.4.
exception processing: when bad_aerosol == TRUE:                                          (2.6.9.2.7-4)
        ACFAIL_F (j, f) = TRUE
        process pixel according to "Exception: atmosphere correction failed" (§8.4.5)
end of exception processing
        $\rho_{path}^*(b) = \zeta(b) * \rho_R(b,j,f)$                                                (2.6.9.2.7-5)
        If ( abs($P_{ECMWF}$(j,f) - $P_{std}$) > PRESS_TOLERANCE) then
                $\rho_{path}^*$(b) = correct_ΔP($\rho_{path}^*$(b), $P_{std}$, $P_{ECMWF}$(j, f), b, $\tau_a$(b, j, f), PLUS)(2.6.9.2.7-6)
        Endif
Endfor

Compute the "aerosol" reflectance (aerosol + coupling aerosol/Rayleigh), for internal
purpose (breakpoints).
Distinguish b775, b865 and b885 for being consistent with $\rho^*_w$ in step (2.6.9.3-8)
For b in { b412, … b753 }
        $\rho_a(b) = \rho_{path}^*(b) - \rho_{R0}(b)$                                                (2.6.9.2.7-7)
Endfor
For b in { b775, b865, b885 }
        $\rho_a$(b) = ROGC(b,j,f) - t$\rho$w_C2(b,j,f)- $\rho_{R0}$(b)                                 (2.6.9.2.7-11)
Endfor

Compute the aerosol Angström exponent product
If (($\tau_a$ (b775,j,f) > 0) AND ($\tau_a$(b865,j,f) > 0)) then
        $\alpha_{775\_865}(j, f) = \log(\tau_a(b775,j,f) / \tau_a(b865,j,f)) / \log(\lambda_{theo}(b865) / \lambda_{theo}(b775))$ (2.6.9.2.7-8)
else
        $\alpha_{775\_865}$(j, f) = BAD_VALUE
        (2.6.9.2.7-9)
        set bit EPSILON of ANNOT (j, f)                                                  (2.6.9.2.7-10)
endif
```

Copyright © 2011 ACRI S.T.

8.4.3.12 – Correction (step 2.6.9.3)

8.4.3.12.1 - Functional description

This step uses the estimated aerosol parameters to derive the normalised water-leaving reflectance. Its operation and parameters are shown schematically in figure 8.4.3.12.1-1 below.

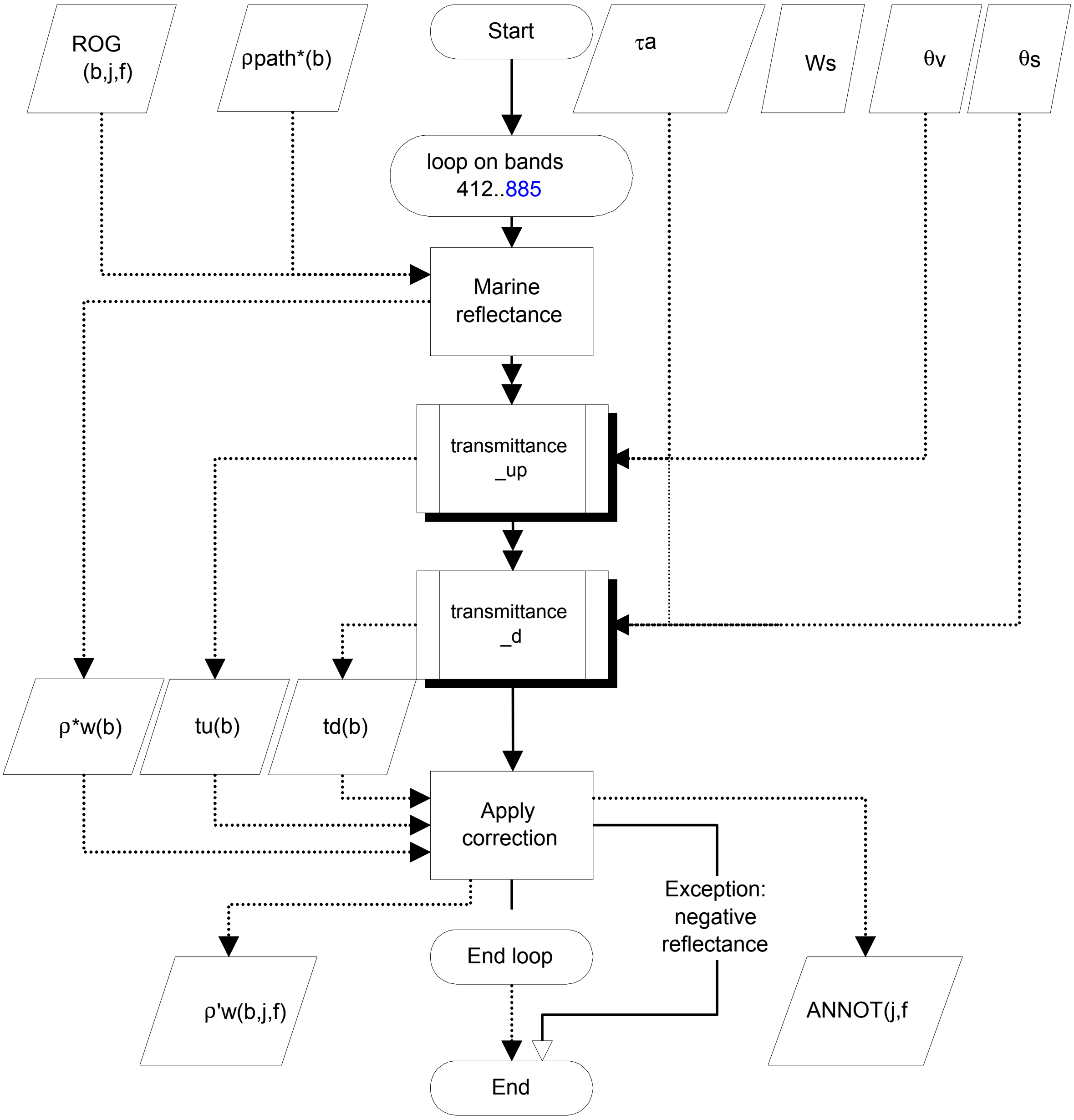


Figure 8.4.3.12.1-1: functional block diagram of step Correction (2.6.9.3)

8.4.3.12.2 - Input /Output

Variable	Descriptive name	T	U	Range – References
ROGC (b, j, f)	Glint corrected reflectance	i	dl	b: {b412..b885}
ρ_{path}^*(b)	Path reflectance estimate	i	dl	*idem*
tρw_C2(b,j,f)	Marine reflectance at TOA for Case 2 waters	i	dl	*b: {b705, b775, b865, b885}*
ia1 (j, f), ia2 (j, f)	Index of bracketing aerosol models	i	-	
τ_a_865 (i)	Aerosol optical thickness at 865nm	i	dl	i: {LOW, HIGH}
τ_{R0} (b)	Optical thickness due to Rayleigh scattering	i	dl	b: {b412..b885}
aer_mix	Aerosol mixing ratio	i	dl	
μ_s	Cosine of sun zenith angle	i	dl	
μ	Cosine of view zenith angle	i	dl	
W_s	wind speed modulus	i	m/s	
RWNEG_threshold(b)	LUT of negative water-leaving reflectance threshold	s	dl	*b:{b412..b885}*
t_u(b)	Transmittance on the target-sensor path	c	dl	b: {b412..b885}
t_d(b)	Transmittance on the Sun-target path	c	dl	*idem*
ρ^*_w (b)	Marine reflectance at TOA	c	dl	*idem*
ρ'_w (b, j, f)	Normalised water-leaving reflectance	o	dl	*Idem* + b885
ANNOT (j, f)	Annotation flag for the quality of the atmospheric correction	o	-	Coding described in 8.4.4 below
ACFAIL_F (j, f)	Flag indicating failure of the atmospheric corrections procedure	o	-	Boolean
RWNEG (b, j, f)	Flag indicating negative water-leaving reflectance	o	-	Coding: see 8.4.4

Table 8.4.3.12.2-1: Parameters for the Correction step

8.4.3.12.3 - Algorithm

```
For b = b412..b753
    $\rho^*_w(b) = ROGC(b,j,f) - \rho_{path}^*(b)$    Estimate $t\rho_w$ at bands 412 to 753 nm    (2.6.9.3-1)
exception processing :
    when ($\rho^*_w(b) <= RWNEG\_threshold(b)$):
    (2.6.9.3-2)
    RWNEG(b, j, f)  = TRUE
    continue processing
end of exception processing
Endfor
```

Force $t\rho_w$ to $t\rho_w C2$ at 775 and 865 to be consistent with the aerosol retrieval. Do it also at 885 because the signal is too low and might become negative.

```
For b=775, 865,885
    $\rho^*_w(b) = t\rho w\_C2(b,j,f)$    (2.6.9.3-8)
Endfor
```

Copyright © 2011 ACRI S.T.

```
For b=b412 … b885
    $t_u(b)$= transmittance_up(ia1, ia2, b, $\tau_{R0}(b)$, $\tau_a$_865, aer_mix, $\mu$)    (2.6.9.3-4)
    $t_d(b)$= transmittance_d(ia1, ia2, b, $\tau_{R0}(b)$, $\tau_a$_865, aer_mix, $\mu_s$, $W_s$)    (2.6.9.3-5)
NOTE: the function transmittance_up is defined in 8.4.3.13.5 below and function transmittance_d in 8.4.3.13.10.
Compute normalised water-leaving reflectance
    $\rho'_w(b,j,f) = \rho^*_w(b) / t_u(b) / t_d(b)$    (2.6.9.3-6)
Endfor    End of loop on bands
Note: b761 and b900 are ignored (see §8.1)    (2.6.9.3-7)
```

Copyright © 2011 ACRI S.T.

8.4.3.13 - Functions and Subroutines :

NOTE: *For each of the subroutines listed below, the list of variables shows the call sequence parameters as well as the auxiliary parameters*

8.4.3.13.1 - Subroutine Spectral interpolation of ζ (ζ, ia, b, μ_s, μ_v, $\Delta\phi$, W_s, ζ, τ_a_865, bad aerosol)

This function computes the ratio ζ of path reflectance to Rayleigh reflectance at a band b, from its value at 865nm, for a given aerosol model. It is called by Candidate (step 2.6.9.2.2).

8.4.3.13.1.1 - Inputs /Outputs

Variable	Descriptive name	T	U	Range - References
ζ	Ratio ρ_{path} / ρ_R at band b865	i	dl	
ia	Aerosol model index	i	-	1..N_Aer
b	Band index	i	nm	
μ_s	Cosine of sun zenith angle	i	dl	
μ_v	Cosine of view zenith angle	i	dl	
$\Delta\phi$	Azimuth difference	i	deg	
W_s	Wind speed	i	m/s	
τ_a_loc	Local estimate of aerosol optical thickness at band b	c	dl	
ζ	Ratio ρ_{path} / ρ_R at band b	o	dl	
τ_a_865	Aerosol optical thickness at band b865	o	dl	
bad aerosol	Flag indicating exception	o	-	Boolean

Table 8.4.3.13.1.1-1: List of variables for subroutine Spectral interpolation of ζ

8.4.3.13.1.2 - Algorithm

```
bad aerosol = FALSE                                                   (siz-1)
$\tau_a$_865 = $\zeta$_to_$\tau$a ($\zeta$, ia, b865, $\mu_s$, $\mu_v$, $\Delta\phi$, $W_s$)          (siz-2)
$\tau_a$_loc = $\tau_a$_865 * c_iactl (ia, $\tau_a$_865, b)                         (siz-3)
$\zeta$ = $\tau$a_to_$\zeta$ ($\tau_a$_loc, ia, b, $\mu_s$, $\mu_v$, $\Delta\phi$, $W_s$)            (siz-4)
NOTE: the functions $\zeta$_to_$\tau$a , c_iactl , $\tau$a_to_$\zeta$ are defined in 8.4.3.13.2, 8.4.3.13.8, 8.4.3.13.3, below.
exception processing:
    when the function $\zeta$_to_$\tau$a or $\tau$a_to_$\zeta$ raises the exception "bad aerosol":   (siz-5)
        bad aerosol = TRUE
    return
end of exception processing
return
```

8.4.3.13.2 - Function ζ _to_τ_a (ζ, ia, b, μ_s, μ_v, $\Delta\phi$, W_s)

At a given wavelength, this function computes the aerosol optical thickness τ from the ratio ζ of path reflectance to Rayleigh reflectance, using the tabulated quadratic relationship between these parameters. It is called by function ***Spectral interpolation of*** ζ

8.4.3.13.2.1 - Inputs /Outputs

Variable	Descriptive name	T	U	Range – References
ζ	Ratio ρ_{path} /ρ_R at band b	i	dl	
ia	Aerosol model index	i	-	
b	Band index	i	-	
μ_s	Cosine of sun zenith angle	i	dl	
μ_v	Cosine of view zenith angle	i	dl	
$\Delta\phi$	Azimuth difference	i	deg	
W_s	Wind speed modulus	i	m/s	
N_co	Number of coefficients in Xctab	s	-	
XCtab_LUT [k,W_S,ia,b,μ_S,μ_V, $\Delta\phi$]	LUT for polynomial coefficients linking the ratio ρ_{path}/ρ_R to the aerosol optical thickness	s	dl	*
XC (m)	Coefficients of the τ_a to ζ quadratic function	c	dl	m: 1..N_co
τ_a	Aerosol optical thickness at band b	o	dl	
bad aerosol	flag indicating exception	o	-	Boolean
ORINP0_F(j,f)	Flag indicating whether input to the clear waters atm. Corr. Is invalid	o	-	Boolean, to step 2.10 (§10.4), to Breakpoint

Table 8.4.3.13.2.1-1: List of variables for function ζ*_to_*τ_a

NOTES

*: the increasing order of magnitude for θ_s and θ_v indices in the LUT files, imposes a decreasing order for the corresponding μ_s and μ_v cosines.

Copyright © 2011 ACRI S.T.

8.4.3.13.2.2 Algorithm

```
Interpolate XC coefficients
For m=1…N_co
    XC(m) = XCtab_LUT interpol: ($\mu_S$, $\mu_V$, $\Delta\phi$, $W_s$) select: (m, b, ia)        (ztt-1)
Endfor

exception processing:
If error in interpolation (out of LUT index range in geometry only, not in $W_s$)
    ORINP0_F(j,f)=TRUE
    continue processing
end of exception processing


Inverse the polynomial
$\tau_a$= inversion_coef ($\zeta$, XC)        (ztt-2)

exception processing: $\tau_a < 0$:        (ztt-3)
    bad aerosol = TRUE
end of exception processing

return $\tau_a$
```

Copyright © 2011 ACRI S.T.

8.4.3.13.3 - Function τ_a_to_ζ (τ_a, ia, b, μ_s, μ_v, $\Delta\phi$, W_s)

At a given wavelength, this function computes the ratio ζ of path reflectance to Rayleigh reflectance the aerosol optical thickness τ from the aerosol optical thickness, using the tabulated quadratic relationship between these parameters. It is called by ***subroutines Spectral interpolation of ζ, Aerosol parameters per band***.

8.4.3.13.3.1 - Input /Output

Variable	Descriptive name	T	U	Range – References
τ_a	aerosol optical thickness at band b	i	dl	
ia	aerosol model index	i	-	
b	band index	i	-	
μ_s	Cosine of Sun zenith angle	i	dl	
μ_v	Cosine view zenith angle	i	dl	
$\Delta\phi$	Azimuth difference	i	deg	
W_s	Wind speed modulus	i	m/s	
N_co	Number of coefficients in Xctab	s	-	
XCtab_LUT [k,W_S,ia,b, μ_S, μ_V,$\Delta\phi$]	LUT for polynomial coefficients linking the ratio ρ_{path}/ρ_R to the aerosol optical thickness	s	dl	*
XC(m)	coefficients of the τ_a to ζ quadratic func	c	dl	m: 1..N_co
ζ	Ratio of ρ_{path}/ρ_R to τ_a	o	dl	
bad aerosol	Flag indicating exception	o	-	Boolean
ORINP0_F(j,f)	Flag indicating whether input to the clear waters atm. Corr. Is invalid	o	-	Boolean, to step 2.10 (§10.4), to Breakpoint

Table 8.4.3.13.3.1-1: List of variables for Function τ_a_to_ζ

NOTES

*: the increasing order of magnitude for θ_s and θ_v indices in the LUT files, imposes a decreasing order for the corresponding μ_s and μ_v cosines.

*8.4.3.13.3.2**Algorithm***

```
Interpolate XC coefficients
For m=1…N_co
    XC(m) = XCtab_LUT interpol: ($\mu_S$, $\mu_V$, $\Delta\phi$, $W_s$) select: (m, b, ia)          (ttz-1)
Endfor

Exception processing:
If error in interpolation (out of LUT index range in geometry only, not in $W_s$)
    ORINP0_F(j,f)=TRUE
    continue processing
end of exception processing
```

```
Compute the polynomial
$\zeta=0$                                                            (ttz-2)
For m= N_co, N_co-1, … 1
    $\zeta= \tau_a * \zeta + XC(m)$                                  (ttz-3)
Endfor

In case of concave polynomial, check tau is before zeta maximum
exception processing: $\zeta < 0$ OR $(XC(2)<0$ AND $\tau_a> -XC(1)/(2*XC(2)))$       (ttz-4)
    bad aerosol = TRUE
end of exception processing
return $\zeta$
```

Copyright © 2011 ACRI S.T.

8.4.3.13.4 - Subroutine Aerosol parameters per band (ia1, ia2, b, τ_a_865(i), aer_mix, μ_s, μ_v, $\Delta\phi$, W_s, ζ, τ_a, ω_a, f_a, bad_aerosol)

This routine interpolates all the useful optical parameters of an aerosol model pair at a given band. It is called by Store candidate models (step 2.6.9.2.3), Aerosol_parameters (step 2.6.9.2.7).

8.4.3.13.4.1 - Input /Output

Variable	Descriptive name	T	U	Range – References
ia1, ia2	aerosol candidate model indices	i	-	
b	band index	i	nm	
τ_a_865(i)	aerosol optical at 865nm	i	dl	i: {LOW, HIGH}
aer_mix	aerosol mixing ratio	i	dl	
μ_s	Cosine of Sun zenith angle	i	dl	
μ_v	Cosine of view zenith angle	i	dl	
$\Delta\phi$	Azimuth difference	i	deg	
W_s	Wind speed modulus	i	m/s	
$f_a$tab_LUT [ia, b]	LUT for the aerosol forward scattering probability f_a	s	dl	ia: 1..N_Aer b: b412.. b885
$\omega_a$tab_LUT [ia, b]	LUT for the aerosol single scattering albedo ω_a	s	dl	*idem*
τ_a_vis(ia)	Optical thickness for the two aerosol candidates at band b	c	dl	ia: {ia1, ia2}
ζ (ia)	Ratio ρ_{path}/ρ_R for the two aerosol candidates at band b	c	dl	ia: {ia1, ia2}
ζ	Ratio ρ_{path}/ρ_R at band b	o	dl	
τ_a	Aerosol optical thickness at band b	o	dl	
ω_a	Aerosol single scattering albedo at band b	o	dl	
f_a	Aerosol forward scattering probability at band b	o	dl	
bad aerosol	Flag indicating exception	o	-	Boolean

Table 8.4.3.13.4.1-1: List of variables for Subroutine Aerosol parameters per band

8.4.3.13.4.2 - Algorithm

```
bad aerosol = FALSE
For ia in {ia1, ia2}
    If ( ia == ia1 ) then
        i = LOW                                                    (apb-0a)
    Else
        i = HIGH                                                   (apb-0b)
    Endif
    $\tau_a$_vis(ia) = $\tau_a$_865(i) * c_iactl (ia, $\tau_a$_865(i), b)          (apb-1)
    Note: the function c_iactl is defined in 8.4.3.13.8.
    $\zeta$ (ia) = $\tau$a_to_$\zeta$ ($\tau_a$_vis(ia), ia, b, $\mu_s$, $\mu_v$, $\Delta\phi$, $W_s$)     (apb-2)
    Note: the function $\tau$a_to_$\zeta$ is defined in 8.4.3.13.3.
```

Copyright © 2011 ACRI S.T.

```
exception processing: $\tau_a$_vis(ia) < 0 or bad aerosol exception in $\tau a\_to\_\zeta$:      (apb-3)
        bad aerosol = TRUE
        return from subroutine
end of exception processing
Endfor
If ( ia1 != ia2 ) then
        $\zeta = \zeta(ia1) * (1 - aer\_mix) + \zeta(ia2) * aer\_mix$      (apb-4)
Else
        $\zeta = \zeta(ia1)$      (apb-5)
Endif
$\tau_a = \tau_a\_vis(ia1) * (1 - aer\_mix) + \tau_a\_vis(ia2) * aer\_mix$      (apb-6)
$\omega_a = \omega_a tab\_LUT(ia1, b) * (1 - aer\_mix) + \omega_a tab\_LUT(ia2, b) * aer\_mix$      (apb-7)
$f_a = f_a tab\_LUT(ia1, b) * (1 - aer\_mix) + f_a tab\_LUT(ia2, b) * aer\_mix$      (apb-8)
return
```

Copyright © 2011 ACRI S.T.

8.4.3.13.5 - Function transmittance_up (ia1, ia2, b, τ_{R0}, τ_a_865(i), aer_mix, μ)

This function computes the diffuse transmittance on the surface-sensor path. It is called by *Correction* (step 2.6.9.4) and by function *Denormaliser*.

8.4.3.13.5.1 - Input /Output

Variable	Descriptive name	T	U	Range – References
ia1, ia2	Index of bracketing aerosol models	i	-	
b	band index	i	nm	
τ_{R0}	Rayleigh optical thicknes corrected for actual pressure at band b	i	dl	from step 2.1.c (§5.5)
τ_a_865(i)	Aerosol optical thickness for the two aerosol candidates at 865 nm	i	dl	i:{LOW, HIGH}
aer_mix	Aerosol mixing ratio	i	dl	
μ	Cosine of view zenith angle	i	dl	
τ_R(b)	Rayleigh optical thickness at standard pressure at band b	s	dl	
t_{up}_LUT [ia, b, μ,$\tau_{a\ 865}$]	LUT for upward diffuse transmittance	s	dl	
τ_a_bl865(aer, τ)	LUT of the optical thickness of the aerosol assemblage at 865 nm	s	dl	
log_t_{up1}	Diffuse transmittance on surface-sensor path for aerosol model ia1	c	dl	
log_t_{up2}	Diffuse transmittance on surface-sensor path for aerosol model ia2	c	dl	
t_{up}	Diffuse transmittance on surface-sensor path	o	dl	

Table 8.4.3.13.5.1-1: List of variables for function transmittance_up

8.4.3.13.5.2 - Algorithm

Compute upward transmittance for the two aerosol models:

log_t_{up1}=t_{up}_LUT **interpol:**(μ, τ_a_865(LOW)) **select:**(ia1, b) *(tup-1)*

log_t_{up2}=t_{up}_LUT **interpol:**(μ, τ_a_865(HIGH)) **select:**(ia2, b) *(tup-2)*

warning: in *(tup-1)* and *(tup-2)*, the interpolation along the τ_a dimension depends on the aerosol model. Thus the interpolation indices and weight of τ_a must be taken from the interpolation of τ_a in LUT τ_a_bl865(aer, ∗), with aer selected as the current candidate model.
warning 2: the interpolation must be done on the log of t_{up}_LUT values.

Interpolate the transmittance in logscale between aerosol models and correct for pressure the Rayleigh contribution:

$$\mathrm{t_{up}} = \exp\left[\mathrm{log\_t_{up1}} * (1 - \mathrm{aer\_mix}) + \mathrm{log\_t_{up2}} * \mathrm{aer\_mix}\right] \cdot \frac{\exp\left(-\frac{\tau_{R0}}{2\mu}\right)}{\exp\left(-\frac{\tau_R(b)}{2\mu}\right)} \qquad \textit{(tup-3)}$$

Return $\mathrm{t_{up}}$

Copyright © 2011 ACRI S.T.

8.4.3.13.6 - Function Denormaliser (trhow, θ_s, θ_v, $\Delta\phi$, b, τ_{a_865}, W_s)

The function ***Denormaliser*** is called by Store candidate models (step 2.6.9.2.3).

8.4.3.13.6.1 - Input /Output

Variable	Descriptive name	T	U	Range - References
trhow	normalised marine reflectance	i	dl	
θ_s	Sun zenith angle	i	deg	
θ_v	View zenith angle	i	deg	
$\Delta\phi$	Difference of azimuth angles	i	deg	
b	band index	i	nm	
$\tau_{a\ 865}$	Aerosol optical thickness at 865nm	i	dl	
W_s	Wind speed	i	$m.s^{-1}$	
WAT_REF_IND	Water refraction index	s	dl	
CMOY	Mean value of chl. concentration	s	$mg.m^{-3}$	
f_over_q1_LUT [b, θ_p, θ_s, $\Delta\phi$, Chl, τ_a, W_s]	LUT for the bidirectional factor f/Q	s	dl	
TETAP_ZENITH	Value of θ_p for nadir viewing	s	deg	
$r_{ghot_}$LUT (θ_p, W_s)	LUT for the ocean-atmosphere reflection factor	s	dl	
FSURQ_0	Value of f/Q factor at nadir	s	dl	
θ_p	View zenith angle below water	c	deg	
fuponQ	Value of f/Q factor at pixel	c	dl	
$\Re$	Ocean-atmosphere reflection factor	c	dl	
$\Re_0$	Ocean-atmosphere reflection factor for nadir view	c	dl	
denorm	de-normalised marine reflectance	o	dl	

Table 8.4.3.13.6.1-1: List of variables for function Denormaliser

8.4.3.13.6.2- Algorithm

θ_p = **arc sin** (**sin** (θ_v) / WAT_REF_IND) *(denorm-1)*

fuponQ = f_over_q1_LUT **interpol:**(θ_p,θ_S,$\Delta\phi$,CMOY) **nearest**: (τ_{a_865},W_s) **select:**(b) *(denorm-2)*

$\Re$ = $r_{ghot_}$LUT **interpol:** (θ_p, W_S) *(denorm-3)*

$\Re_0$ = $r_{ghot_}$LUT **interpol:** (TETAP_ZENITH, W_S) *(denorm-4)*

denorm = trhow * (fuponQ /FSURQ_0) * ($\Re$ / $\Re_0$) *(denorm-7)*

Return denorm *(denorm-8)*

Copyright © 2011 ACRI S.T.

8.4.3.13.7 - Function correct_ΔP(ρ_{path}, P_{ref}, P_{cur}, b, τ_a, branch)

This function corrects the estimated path reflectance for the difference between actual pressure and reference pressure. The function ***correct_ΔP*** is called by Path reflectance (step 2.6.9.1), Aerosol_parameters (step 2.6.9.2.7).

8.4.3.13.7.1 - Input /Output

Variable	Descriptive name	T	U	Range - References
ρ_{path}	Estimated value of the path reflectance	i	dl	
P_{ref}	Reference atmosphere pressure	i	hPa	
P_{cur}	Current atmosphere pressure	i	hPa	
b	band index	i	dl	
τ_a	Estimated value of the aerosol optical thickness	i	dl	
branch	Switch for correction direction	i	-	{PLUS, MINUS}
$\tau_R(b)$	Standard values of the optical thickness due to Rayleigh scattering	s	dl	b: b412..b885
DeltaP	Pressure correction factor	c	dl	
ρ'_{path}	Pressure corrected path reflectance	o	dl	

Table 8.4.3.13.7.1-1: List of variables for function correct_ΔP

8.4.3.13.7.2- Algorithm

```
DeltaP = $(P_{cur} – P_{ref})/P_{ref}$                                  (correct-dp-1)
If (branch == PLUS)
    $\rho'_{path} = \rho_{path}$*[ 1 + DeltaP * $\tau_R(b)/(\tau_R(b)+\tau_a)$ ]       (correct-dp-2)
Else
    $\rho'_{path} = \rho_{path}$*[ 1 - DeltaP * $\tau_R(b)/(\tau_R(b)+\tau_a)$ ]       (correct-dp-3)
Return $\rho'_{path}$                                                   (correct-dp-4)
```

Copyright © 2011 ACRI S.T.

8.4.3.13.8 - Function c_iactl (ia, τa_865, b)

The function ***c_iactl*** uses a tabulated relationship to interpolate the aerosol optical thickness along wavelength, for a given aerosol model. The function ***c_iactl*** is called by function ***Spectral interpolation of*** ζ and subroutine ***Aerosol parameters per band***.

8.4.3.13.8.1 - Input /Output

Variable	Descriptive name	T	U	Range - References
ia	Aerosol model index	i	dl	
τa_865	Aerosol optical thickness at 865 nm for the aerosol model ia	i	dl	
b	band index	i	dl	
τ_a_bl865 (aer,τ)	LUT of the optical thickness of the aerosol assemblage at 865 nm	s	dl	
specdep(aer,τ,b)	LUT of the spectral dependence of the aerosol optical thickness	s	dl	aer: 1..Naer; b: b412..b885
k, weight	Optical thickness interpolation coefficients	c	dl	
specdep0	Ratio between optical thickness at band b and τa_865	o	dl	

Table 8.4.3.13.8-1: List of variables for function c_iactl

8.4.3.13.8.2- Algorithm

```
interpolation in LUT ta_bl865:
find k in {1..N_τ} such that:                                              (c_iactl-1)
$\tau_a$_bl865(ia, k) <= $\tau_a$_865 <= $\tau_a$_bl865 (ia, k+1)
weight = ($\tau_a$_865 - $\tau_a$_bl865(ia, k))/ ($\tau_a$_bl865 (ia, k+1) - $\tau_a$_bl865(ia, k))   (c_iactl-2)
exception processing: here table $\tau_a$_bl865 is supposed to be monotonously ascending   (c_iactl-3)
    when ($\tau_a$_865 <= .$\tau_a$_bl865(ia, 1))
        specdep0= specdep select: (ia, 1, b)
        return specdep0
    when ($\tau_a$_865 >= $\tau_a$_bl865(ia, N_τ)):
        specdep0= specdep select: (ia, N_τ, b)
        return specdep0
end of exception processing
specdep0 = (1. - weight).[specdep select: (ia, k, b)]                      (c_iactl-4)
    + weight. [specdep select: (ia, k+1, b)]
Return specdep0                                                            (c_iactl-5)
```

Copyright © 2011 ACRI S.T.

8.4.3.13.9 - Function inversion_coef(y, XC)

The function ***inversion_coef*** solves a second degree equation. It is called by function **ζ_to_τ_a**.

8.4.3.13.9.1 - Input /Output

Variable	Descriptive name	T	U	Range - References
y	Value of polynomial	i	dl	
XC	Coefficients of second degree polynomial	i	dl	
MAX_TAU_AER	Maximum allowed value for aerosol optical thickness	s	dl	
a, b, c	Coefficients of second degree polynomial	c	dl	
delta	Determinant of equation	c	dl	
x1, x2	Solutions of equation $y = XC(1) + XC(2) * x + XC(3) * x^2$	c	dl	
x	Selected solution or default value	o	dl	

Table 8.4.3.13.9.1-1: List of variables for function inversion_coef

8.4.3.13.9.2- Algorithm

```
Inverse the relationship y = XC(1) + XC(2) * x + XC(3) * x² in order to derive x
a = XC(3)                                                        (inversion-1)
b = XC(2)                                                        (inversion-2)
c = XC(1) – y                                                    (inversion-3)

exception handling
If (abs(a) < 10^-6) then
    a = 0                                                        (inversion-4)
Endif
If (a != 0) then
    delta = b^2 – 4*a*c                                          (inversion-5)
    If( delta >=0) then
        x1 = ( - b - sqrt(delta))/(2*a)                          (inversion-6)
        x2 = ( - b + sqrt(delta))/(2*a)                          (inversion-7)
        If (x1 >= 0 && x1 < MAX_TAU_AER) then
            x = x1                                               (inversion-8)
            If (x2 >= 0 && x2 < MAX_TAU_AER && x2 < x1) then
                x = x2                                           (inversion-10)
            Endif
        Else if (x2 >= 0 && x2 < MAX_TAU_AER) then
            x = x2                                               (inversion-11)
        Else
            x = -999                                             (inversion-12)
        Endif
    Else
        x = - 999                                                (inversion-13)
    Endif
Second degree equation degenerates
Else
```

```
        If (b > 0) then
                x = -c / b                                          (inversion-14)
                If (x > MAX_TAU_AER) then
                        x = - 999                                   (inversion-15)
                Endif
        Else
                x = - 999                                           (inversion-16)
        Endif
Endif
Return x                                                            (inversion-17)
```

8.4.3.13.10 - Function transmittance_d(ia1, ia2, b, τ_{R0}, τ_a_865(i), aer_mix, μ_s, W_s)

The function ***transmittance_d*** computes the transmittance on the Sun-pixel path for optical properties computed at a given band. It is called by Correction (step 2.6.9.4) and by function ***denormaliser***.

8.4.3.13.10.1- Input /Output

Variable	Descriptive name	T	U	Range – References
ia1, ia2	Index of bracketing aerosol models	i	-	
b	band index	i	nm	
τ_{R0}	Rayleigh optical thickness corrected for actual pressure at band b	i	dl	from step 2.1.c (§5.5)
τ_a_865(i)	Aerosol optical thickness at 865nm	i	dl	i:{LOW, HIGH}
aer_mix	Aerosol mixing ratio	i	dl	
μ_s	Cosine of Sun zenith angle	i	dl	
W_s	Wind speed modulus	i	m/s	
$\tau_R(b)$	Rayleigh optical thickness at standard pressure at band b	s	dl	
t_{down}_LUT $[ia,b,\mu_s,\tau_{a\_865},W_s]$	LUT for downward diffuse transmittance	s	dl	
τ_a_bl865(aer, τ)	LUT of the optical thickness of the aerosol assemblage at 865 nm	s	dl	
log_t_{down1}	Log of diffuse transmittance on sun-surface path for aerosol model ia1	c	dl	
log_t_{down2}	Log of diffuse transmittance on sun-surface path for aerosol model ia2	c	dl	
t_{down}	Diffuse transmittance on sun-surface path	o	dl	

Table 8.4.1.1.1.1-1: List of variables for function transmittance_d

8.4.3.13.10.2- Algorithm

Compute downward transmittance for aerosol model ia1 and ia2:

log_t_{down1}=t_{down}_LUT **interpol:**(μ_s, τ_a_865(LOW), W_s) **select:**(ia1, b) *(td-1)*

log_t_{down2}=t_{down}_LUT **interpol:**(μ_s, τ_a_865(HIGH), W_s) **select:**(ia2, b) *(td-2)*

warning: in *(td-1)* and *(td-2)*, the interpolation along the τ_a dimension depends on the aerosol model. Thus the interpolation indices and weight of τ_a must be taken from the interpolation of τ_a in LUT τ_a_bl865(aer, $*$), with aer selected as the current candidate model.
warning 2: the interpolation must be done on the log of t_{down}_LUT values.

Interpolate the transmittance in logscale between aerosol models and correct for pressure the Rayleigh contribution:

$$\mathrm{t}_{\mathrm{up}} = \exp\left[\log\_\mathrm{t}_{\mathrm{down1}} * (1-\mathrm{aer\_mix}) + \log\_\mathrm{t}_{\mathrm{down2}} * \mathrm{aer\_mix}\right]\cdot \frac{\exp\left(-\dfrac{\tau_{R0}}{2\mu_s}\right)}{\exp\left(-\dfrac{\tau_R(b)}{2\mu_s}\right)} \qquad (td\text{-}3)$$

Return $\mathrm{t}_{\mathrm{down}}$

Copyright © 2011 ACRI S.T.

8.4.4 Quality control and diagnostics

The annotation flag has been introduced within the atmospheric correction scheme, to provide the breakpoints with indications about the course of the algorithm. This flag (variable “ANNOT”) is represented by 2 bytes. Each bit implements a Boolean flag, corresponding to various, non necessarily exclusive, situations, for which the accuracy of the algorithm can be degraded :

symbol	bit pos.	Meaning
SHALLOW	0	The altitude is above DEPTH_LIM
ORLUT	1	LUT index variables out of range
EPSILON	2	The aerosol Angström exponent cannot be computed
AERO_B	3	Blue aerosols
ABSO_D	4	Desert dust absorbing aerosols
ACLIM	6	The aerosol model does not match the climatology of aerosols
ABSOA	7	Absorbing aerosols
MIXR1	8	The aerosol mixing ratio is equal to 1
DROUT	12	The minimum absolute value of $\Delta\rho 510$ is greater than DRO510_threshold
TAU06	10	The aerosol optical thickness is greater than TAU560_threshold

Table 8.4.4-1: Coding of ANNOT flag

An array of flags, RWNEG, provides a “negative water-leaving reflectance” flags for each band. It is implemented as a 16-bit word, with bit number corresponding to (band index – 1).

8.4.5 - Exception handling

The equations below specify the handling of the exception: “atmosphere correction failed”:

```
    ia1(j, f) = BAD_VALUE
    ia2(j, f) = BAD_VALUE
    $\tau_{a865}$ (j, f) = BAD_VALUE
    $\alpha$_775_865 (j, f) = BAD_VALUE
    For b in {b412..b865}
        $\rho'_w$(b, j, f) = BAD_VALUE
        $\tau_a$(b, j, f) = BAD_VALUE
    Endfor
end of exception processing
```

Other exceptions: see blocks "exception processing... end of exception processing" in 8.4.3 above.

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

8.5 - MERIS Ocean Colour Processing (step 2.9)

8.5.1- Introduction

This chapter describes the processing to be applied to surface reflectances produced by the atmospheric corrections above water (§8.4 above) in order to derive ocean bio-optical parameters (RD 8, §2.5, §2.8, §2.9, §2.10, §2.11, §2.12).

8.5.2- Algorithm Overview

Different algorithms are used as shown in flow chart 8.5.2-1.

I. A band-ratio algorithm **optimised** for open ocean clear waters (so-called "Case 1") yields a geophysical quantity :
   - Algal Pigment Index 1

II. **Robust** band-ratio algorithms valid for all water types, including yellow substance dominated (so-called Case 2 (y)) and waters with excessive back-scattering. These algorithms yield the following Product Confidence Data :
   - flag for anomalous scattering waters
   - flag for yellow substance-dominated waters

III. An algorithm to estimate the instantaneous value of the Photosynthetically Available Radiation (PAR)

Three different flags indicating the type of water sensed are used as they have an influence on processing quality:

- turbid waters (described in Section 7 of this document)
- yellow substance dominated waters
- anomalous scattering

Furthermore, range checks on input and output parameters are applied for quality control.

Copyright © 2011 ACRI S.T.

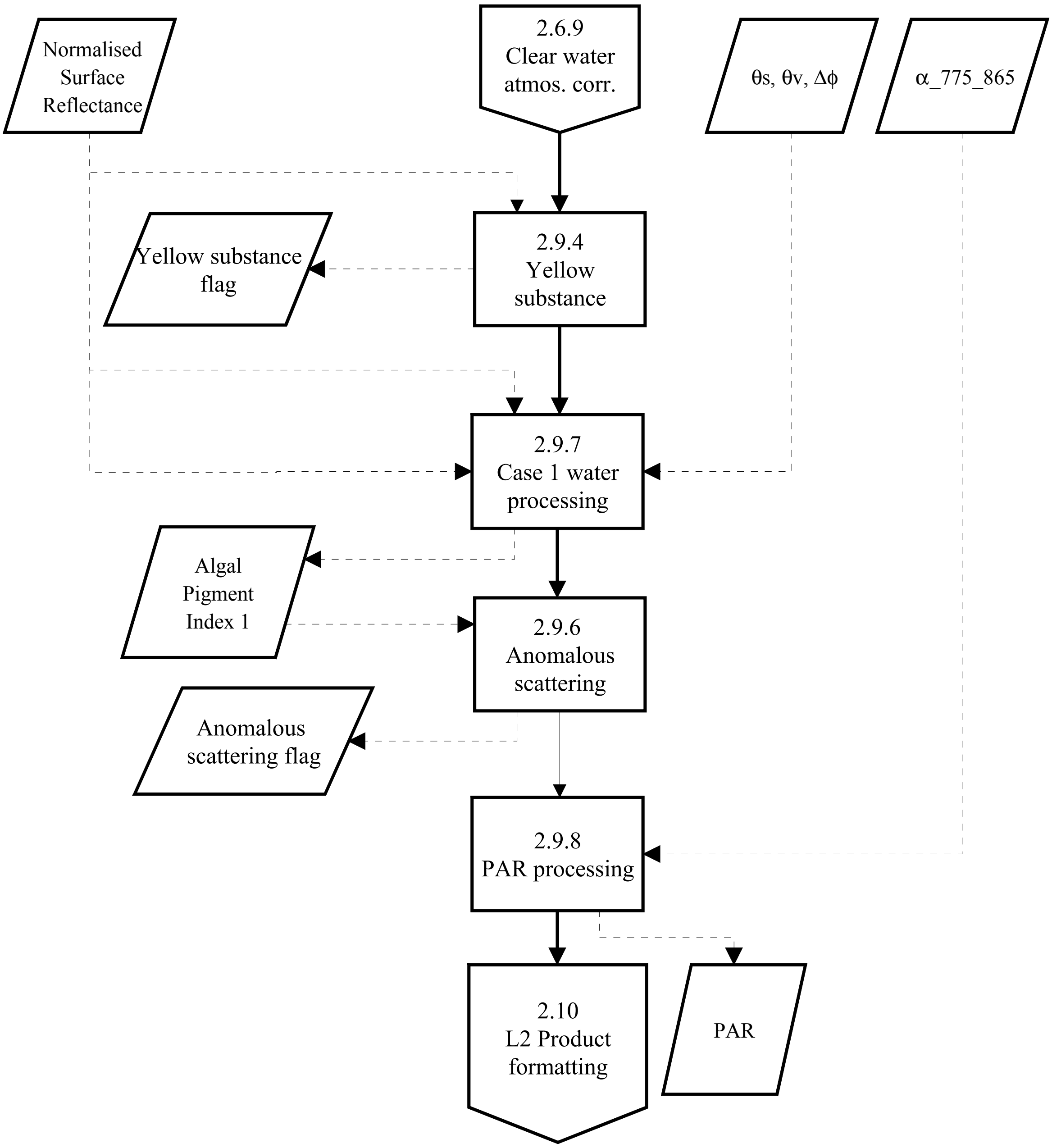


Figure 8.5.2-1 : MERIS ocean colour processing (step 2.9) functional breakdown

Copyright © 2011 ACRI S.T.

8.5.3- Mathematical Description of Algorithm (step 2.9)

8.5.3.1 - Case 2 (Yellow substance dominated) flag (step 2.9.4)

The presence of Case 2 water is flagged by Yellow substance (CASE2Y_F) flag (step 2.9.4). Input data are normalised water-leaving reflectance $\rho'_w(b, j, f)$. A LUT technique is applied for the retrieval. The procedure is described in RD8 (2.8).

8.5.3.2 - Case 1 waters processing - Algal pigment index 1 (Chl1) retrieval (step 2.9.7)

Case 1 waters processing is based on a band ratio algorithm. Inputs are normalised water-leaving reflectance $\rho'_w(b, j, f)$ and ancillary data. The theory of data processing is described in RD8 (2.9). Processing is performed in two steps:

1. a band ratio estimate of Chl1 is selected among up to three possible ones, according to ratio value
2. an iterative procedure eliminates the influence of bi-directionality (parameters f_over_q1 and f_0) on Chl1 estimate.

8.5.3.3 - Case 2 anomalous scattering water flags (step 2.9.6)

The presence of Case 2 water is flagged by the Anomalous scattering (CASE2ANOM_F) flag (step 2.9.6). Input data are normalised water-leaving reflectance $\rho'_w(b, j, f)$ and algal pigment index 1. A LUT technique is applied for the retrieval. The procedure is described in RD8 (2.8).

8.5.3.4 - Deleted

Case2 IMT products have been moved to section 8.6

8.5.3.5 - Photosynthetically Available Radiation (step 2.9.8)

Instantaneous Photosynthetically Available Radiation (PAR) is derived from the irradiance above each water pixels, under a tabulated relationship. The algorithm is RD 8, 2.18. This step is applied to all pixels where ACFAIL_F(j, f) is FALSE.

Copyright © 2011 ACRI S.T.

8.5.4- List of variables

Variable	Descriptive Name	T	U	Range-Reference
INVALID_F(j,f)	invalid flag	i	dl	from step 2.1a (§3.4)
CLOUD_F(j,f)	"cloud" flag	i	dl	from step 2.1c (§5.4)
LANDCONS_F(j,f)	consolidated "land" flag	i	dl	from step 2.1c (§5.4)
ACFAIL_F(j,f)	atmosphere corrections failed flag	i	dl	from step 2.6.10 (§8.3.2) or 2.6.9 (§8.4.2)
ICE_HIGHAERO_F(j,f)	Flag for ice or high aerosol loading pixels	i	dl	from step 2.6.5 (§8.2)
$\theta_s(j,f)$	Sun zenith angle	i	deg	from step 2.1a (§3.4)
$\theta_v(j,f)$	Viewing zenith angle	i	deg	*idem*
$\Delta\phi(j,f)$	Azimuth difference angle	i	deg	*idem*
$\rho'_w(b,j,f)$	Normalised water-leaving reflectance for pixel (j,f)	i	dl	from step 2.6.9 (§8.4.2); b in { b412..b865}
$\tau_a(b,j,f)$	Aerosol optical thickness for pixel (j,f)	i	dl	from step 2.6.9 (§8.4.2); b: b560 (step 2.9.7), b865 (step 2.9.8)
α_775_865 (j,f)	Aerosol Angström exponent	i	dl	from step 2.6.9 (§8.4.2)
w_T (j, f)	actual water vapour content	i	$g.cm^{-2}$	from step 2.3 (§ 6.4)
U_{O3} (j, f)	Actual total ozone amount	i	DU	from step 2.1a (§3.4)
$W_s(j,f)$	wind speed	i	$m.s^{-1}$	from step 2.6.5 (§8.2.2)
A_b2_b5(p)	Hyperbola coefficients for Algal Pigment Index retrieval in Case 2 waters for H(443 nm, 560 nm).	s	dl	p : order of the coefficient, in {1, 2, 3}
A_b3_b5(p)	Hyperbola coefficients for Algal Pigment Index retrieval in Case 2 waters for H(490 nm, 560 nm)	s	dl	*idem*
A_b4_b5(p)	Hyperbola coefficients for Algal Pigment Index retrieval in Case 2 waters for H(510 nm, 560 nm).	s	dl	*idem*
N1, N2, N3	Exponents used in step 2.9.4	s	dl	
B(b)	Constants for detection of yellow substance contaminated pixels	s	dl	b in { b490, b510}
$Chl1_0$	initial algal pigment index value	s	$mg.m^{-3}$	= 0.1
$Chl1_{thresh}$rangeout[2]	Chl1 validity range	s	$mg.m^{-3}$	= [0.01,30]
f_over_q1_LUT[b, θ_p, θ_s, $\Delta\phi$, Chl1, τ_a, W_s]	bidirectional factor f/Q	s	dl	
f_0_LUT[b,Chl]	factor relating irradiance reflectance to water IOPs (with Sun at zenith)	s	dl	b in {b442, b490, b510, b560} Chl: 6 values
min_R_ratio, max_R_ratio	Irradiance reflectance ratio validity range for Algal_1 computation.	s	dl	

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range-Reference
log10coeff_LUT[p]	Polynomial coefficients for Algal Pigment Index retrieval in Case 1 waters	s	dl	p:0..N_{A1}
N_{A1}	highest order of coefficients to use in log10coeff_LUT	s	dl	=5
N_{iter}	Number of iterations for Chl1 calculation	s	dl	= 3
Chl_epsilon	Convergence criterium for iterative Chl calculation	s	$mg.m^{-3}$	
PAR_LUT[angstrom,U_{O3},$\tau_{aer}(865)$,w_T]	LUT giving PAR	s	µeinstein. $m^{-2}.s^{-1}$	
AnomScatt_LUT[θ_s, θ_v, $\Delta\phi$,Chl]	LUT giving threshold on reflectance at 560 nm for anomalous scattering	s	dl	
$\rho_{w5thresh}$rangein	ρ_w(b560) threshold for controlling validity of input to Chl1 algorithm	s	dl	= 0.3
WAT_REF_IND	Water refraction index	s	dl	1.34
YS_thresh	Threshold for flagging Yellow substance dominated waters	s	$mg.m^{-3}$	5
BAD_VALUE	Output value when algorithm fails	s	dl	see § 2 above
H(b)	Intermediate guess of the Algal pigment index 2, based on hyperbolic fits	c	$mg.m^{-3}$	b in {b442, b490, b510}
chl	estimate of algal pigment index	c	$mg.m^{-3}$	
b_{Chl1}	band index used for estimate of algal pigment index	c	dl	
Chl1	estimate of algal pigment index	c	$mg.m^{-3}$	
prev_Chl1	Previous estimate of algal pigment index in iterative procedure	c	$mg.m^{-3}$	
LChl1	Logarithm of algal pigment index	c		
AnomScattValue	Anomalous scattering threshold for current geometry and Chl load	c	dl	
R(b)	Irradiance reflectance	c	dl	b in {b442, b490, b510, b560}
f_over_q1_value	variable used for storing bidirectional factor	c	dl	
θ_p	viewing angle under sea surface	c	deg	
ratio(bchl)	Reflectance ratio used for Chl retrieval	c	dl	b in {b442, b490, b510}
angström	Angström coefficient	c	dl	

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range-Reference
CASE2ANOM_F(j,f)	anomalous scattering water flag	o	dl	described in RD8 §2.8, to step 2.10 (§10.4), to Breakpoint
CASE2Y_F(j,f)	yellow substance loaded water flag	o	dl	described in RD8 §2.8, to step 2.10 (§10.4), to Breakpoint
Chl1(j,f)	Algal pigment index 1	o	$mg.m^{-3}$	Defined in RD8 §2.9 Appendix, to step 2.10 (§10.4), to Breakpoint
ORINP1_F(j,f)	Flag indicating whether input to the Chl_1 algorithm is invalid	o	dl	Boolean, to step 2.10 (§10.4), to Breakpoint
OROUT1_F(j,f)	Flag indicating whether ouput from the Chl_1 algorithm is invalid	o	dl	Boolean, to step 2.10 (§10.4), to Breakpoint
PAR(j,f)	Photosynthetically available radiation	o	$\mu.einstein.m^{-2}.s^{-1}$	RD 8, §2.18, to step 2.10 (§10.4), to Breakpoint

Copyright © 2011 ACRI S.T.

8.5.5- Equations (step 2.9)

8.5.5.1 - Case 2 Yellow substance dominated waters flagging (step 2.9.4)

For each pixel (j,f) such that ((**NOT** INVALID_F(j,f)) **AND** (**NOT** LANDCONS_F(j,f)) **AND** (NOT CLOUD_F(j,f)) **AND** (**NOT** ACFAIL_F(j, f))) *(2.9-1)*

If (ρ'_w(b442, j, f)≤ 0 **OR** ρ'_w (b490, j, f) ≤ 0 **OR** ρ'_w (b510, j, f) ≤ 0 **OR** ρ'_w (b560, j, f) ≤ 0) **then**

CASE2Y_F (j, f) = FALSE *(2.9.4-6)*

Else

$$H(b442) = \frac{\dfrac{\rho'_w(b442,j,f)}{\rho'_w(b560,j,f)} - A\_b2\_b5(1)}{A\_b2\_b5(2) - \dfrac{A\_b2\_b5(3) \times \rho'_w(b442,j,f)}{\rho'_w(b560,j,f)}} \qquad (2.9.4\text{-}1)$$

exception processing: null denominator **OR** H(b442) <= 0 in eq. 2.9.4-1 above:
CASE2Y_F (j, f) = TRUE
skip the rest of step 2.9.4
end of exception processing

$$H(b442) = [\ H(b442)\ ]^{N1} \qquad (2.9.4\text{-}8)$$

$$H(b490) = \frac{\dfrac{\rho'_w(b490,j,f)}{\rho'_w(b560,j,f)} - A\_b3\_b5(1)}{A\_b3\_b5(2) - \dfrac{A\_b3\_b5(3) \times \rho'_w(b490,j,f)}{\rho'_w(b560,j,f)}} \qquad (2.9.4\text{-}2)$$

exception processing: null denominator OR H(b490) <= 0:
CASE2Y_F (j, f) = TRUE
skip the rest of step 2.9.4
end of exception processing

$$H(b490) = [\ H(b490)\ ]^{N2} \qquad (2.9.4\text{-}9)$$

$$H(b510) = \frac{\dfrac{\rho'_w(b510,j,f)}{\rho'_w(b560,j,f)} - A\_b4\_b5(1)}{A\_b4\_b5(2) - \dfrac{A\_b4\_b5(3) \times \rho'_w(b510,j,f)}{\rho'_w(b560,j,f)}} \qquad (2.9.4\text{-}3)$$

exception processing: null denominator OR H(b510) < 0:
CASE2Y_F (j, f) = TRUE
skip the rest of step 2.9.4
end of exception processing

Copyright © 2011 ACRI S.T.

```
                    $H(b510) = [ H(b510) ]^{N3}$                    (2.9.4-10)

                    If( H(b490) < YS_thresh ) then                              (2.9.4-11)
                        If( H(b442) >= B(b490).H(b490) ) then                   (2.9.4-12)
                            CASE2Y_F(j,f)=TRUE                                  (2.9.4-4)
                        Else
                            CASE2Y_F(j,f)=FALSE                                 (2.9.4-5)
                        Endif
                    Else                                                        (2.9.4-13)
                        If( H(b490) >= B(b510).H(b510) ) then                   (2.9.4-14)
                            CASE2Y_F(j,f)=TRUE                                  (2.9.4-15)
                        Else
                            CASE2Y_F(j,f)=FALSE                                 (2.9.4-16)
                        Endif
                    Endif
                Endif
            Endfor
```

8.5.5.2 - Algal pigment index retrieval in Case 1 waters (step 2.9.7)

Note: since the overall processing structure has been revised, equation numbering of the whole step has been revised.

```
For each pixel(j,f) following the criteria in eq. (2.9-1)

  If ($\rho'_w$ (b442,j,f) $\le 0$  OR $\rho'_w$ (b490,j,f$\le 0$) OR $\rho'_w$ (b510,j,f) $\le 0$  OR
      $\rho'_w$ (b560,j,f) $\le 0$  then
    Chl1 (j,f) = BAD_VALUE                                                      (2.9.7-1)
    ORINP1_F(j,f) = TRUE                                                        (2.9.7-2)
  else

    $\theta_p = \arcsin\left(\frac{\sin\theta_v(j,f)}{WAT\_REF\_IND}\right)$                    (2.9.7-41)
    set Chlorophyll first guess
    $Chl1 = Chl1_0$                                                             (2.9.7-3)
    Start iterative procedure to retrieve chlorophyll
    For i = 1..$N_{iter}$
      Correct reference band for bi-directionality using current Chl value
      f_over_q1_value = f_over_q1_LUT
          interpol: ($\theta p$, $\theta s(j,f)$, $\Delta\phi(j,f)$, Chl1)
          nearest: ($\tau_a(b560,j,f)$,$W_s(j,f)$) select: (b560)                (2.9.7-4)
exception processing: f_over_q1_value <= 0 in equations (2.9.7-4), (2.9.7-6):
  Chl1 (j, f) = BAD_VALUE
  ORINP1_F (j,f) = TRUE
  skip the rest of step 2.9.7
end of exception processing
exception processing: $\theta_s$, $\theta_p$, $\Delta\phi$, Chl1 out of LUT range in equations (2.9.7-4), (2.9.7-6):
```

```
    continue at next equation
end of exception processing
            f0_value = f0_LUT
                interpol: (Chl1) select: (b560)                                      (2.9.7-4b)
exception processing: Chl1 out of LUT range in equations (2.9.7-4b), (2.9.7-6b):
    continue at next equation
end of exception processing
        $R(b560) = \frac{\rho'_w(b560,j,f)}{f\_over\_q1\_value} \cdot f0\_value$                        (2.9.7-5)
        Correct the other3 for bi-directionality, compute band ratios
        For bchl in {b442, b490, b510}
            f_over_q1_value = f_over_q1_LUT
                interpol: (θp, θs(j,f), Δφ(j,f), Chl1)
                nearest: ($\tau_a$(b560,j,f),$W_s$(j,f)) select: (bchl)                         (2.9.7-6)
            f0_value = f0_LUT
                interpol: (Chl1) select: (bchl)                                      (2.9.7-6b)
            $R(bchl) = \frac{\rho'_w(bchl,j,f)}{f\_over\_q1\_value} \cdot f0\_value$                     (2.9.7-7)
            $ratio(bchl) = \frac{R(bchl)}{R(b560)}$                                            (2.9.7-8)
        endfor

        store Chl value used for bi-directionnality correction
        prev_Chl1 = Chl1                                                              (2.9.7-9)

        select the maximum ratio to be used in Chl retrieval
        let rmax be the maximum value of ratio(bchl) bchl in {b442, b490, b510}      (2.9.7-10)
exception processing: rmax > max_R_ratio OR rmax < min_R_ratio:
    Chl1 (j, f) = BAD_VALUE
    ORINP1_F (j,f) = TRUE
    skip the rest of step 2.9.7
end of exception processing

        Retrieve Chl:
        $LChl1 = \sum_{p=0}^{N_{A1}} log10coeff\_LUT(p)\left[\log_{10}(r_{max})\right]^p$             (2.9.7-11)
        $Chl1 = 10^{LChl1}$                                                             (2.9.7-12)

        Check convergence
        If ( |Chl1 – prev_Chl1| < Chl_epsilon .Chl1) then break the iterative loop   (2.9.7-13)
    Endfor

    Final Chlorophyll estimate:
    Chl1 (j, f) = Chl1                                                                (2.9.7-14)
    Clip to validity range of Chl_Case 1 algorithm
```

Copyright © 2011 ACRI S.T.

```
        If (Chl1(j,f) < $Chl1_{thresh}$rangeout[0] ) then
            Chl1(j,f)= $Chl1_{thresh}$rangeout[0]                      (2.9.7-15)
            OROUT1_F(j,f)=TRUE                                          (2.9.7-16)
        Else If (Chl1(j,f) > $Chl1_{thresh}$rangeout[1] ) then
            Chl1(j,f)= $Chl1_{thresh}$rangeout[1]                      (2.9.7-17)
            OROUT1_F(j,f)=TRUE                                          (2.9.7-18)
        Endif
    Endif

    Endfor
```

Copyright © 2011 ACRI S.T.

8.5.5.3 - Anomalous scattering water flagging (step 2.9.6)

```
For each pixel (j,f) fulfilling the criteria in eq. (2.9-1) above
    If ( (Chl1 (j, f) == BAD_VALUE) OR ICE_HIGHAERO_F(j,f)) then
        CASE2ANOM_F (j, f) = FALSE                                        (2.9.6-7)
    Else
Computation of Anomalous scattering flag using LUT (step 2.9.6)
        deleted      (2.9.6-2)

        deleted      (2.9.6-3)

        AnomScattValue =
            AnomScatt_LUT interpol: ($\theta_s$(j,f),$\theta_v$(j,f), $\Delta\phi$(j,f), Chl1(j, f))      (2.9.6-4)
exception processing: out of LUT range Chl1, $\theta_s$, $\theta_v$, $\Delta\phi$ in equation (2.9.6-4) above:
    continue at next equation
end of exception processing

    If ( $\rho'_w$(b560,j,f) > AnomScattValue) then                        (2.9.6-5)
        CASE2ANOM_F(j,f)=TRUE                                             (2.9.6-6)

    Else
        CASE2ANOM_F(j,f)=FALSE                                            (2.9.6-8)

    Endif
Endfor
```

8.5.5.4 - deleted

Case2 product is now detailed in section 8.6

8.5.5.5 - PAR processing (step 2.9.8)

For each pixel(j,f) following the criteria in eq. *(2.9-1)*

```
    If ($\alpha$_775_865 (j, f) == BAD_VALUE) then
        angström = 0                                        (2.9.8-1)
    else
        angström = $\alpha$_775_865 (j, f)                  (2.9.8-3)
    Endif
    In the following equation the ozone value should be scaled to reflect the unit of the ozone
    used in the processing.
        PAR(j,f) = $\cos\theta_s(j,f)$ . PAR_LUT interpol:(angström, $\tau_a$(b865,j,f), $U_{O3}(j,f)$, $w_T(j,f)$)
                                                            (2.9.8-4)
Endfor
```

Copyright © 2011 ACRI S.T.

8.5.6- Quality Control and Diagnostics

The flag ORINP1_F summarises all out of range input conditions for the Case 1 processing algorithm step 2.9.7.
The flag OROUT1_F summarises all out of range output conditions for the Case 1 processing algorithm step 2.9.7.

8.5.7- Exception Handling

Water pixels where atmosphere corrections failed:

```
For each pixel (j,f) such that ((NOT INVALID_F(j,f)) AND (NOT LANDCONS_F(j,f))
AND (NOT CLOUD_F(j,f)) AND (ACFAIL_F(j, f)))
    Chl1(j,f) = BAD_VALUE

    PAR (j, f) = BAD_VALUE
    CASE2Y_F (j, f) = FALSE
    CASE2ANOM_F (j, f) = FALSE
    OROUT1_F (j, f) = TRUE
    Endfor
```

Other exceptions: see blocks "exception processing...end of exception processing" in section 8.5.5 above.

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

8.6 - MERIS Case2R Ocean Colour Processing (step 2.9)

8.6.1- Introduction

This chapter describes the processing to be applied to TOA reflectance corrected for gas absorption and smile effect (§5 above) in order to derive ocean bio-optical parameters in coastal waters (RD11). This processing is totally independent from the Case 1 atmospheric correction (§8.4) and Case 1 water processing (§8.5) and can be seen as a parallel branch, specifically designed for Case2 waters but applied everywhere.

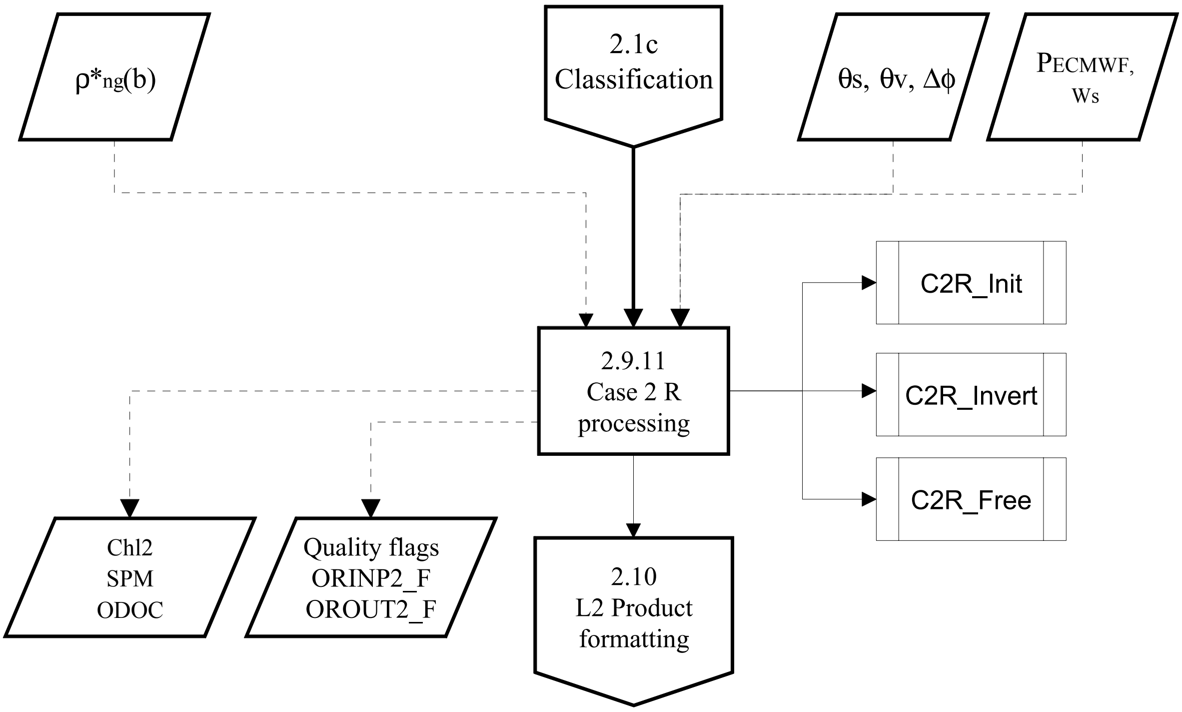
8.6.2- Algorithm Overview

The Case2R algorithm is originally a sequence of two Inverse Modelling Techniques (see RD11): first a backward Neural Network for atmospheric correction of the TOA signal, then a backward Neural Network for retrieval of marine constituents from water-leaving reflectance. Two other forward networks are also used for determining out of range inputs or outputs.

In order to simplify the C2R implementation, a CFI software has been developed, which includes all Neural Networks coefficients, calling routines and pre- and post-processing. It is applied to all atmospheric and water types and yields water inherent optical properties in turn converted into the following geophysical quantities:

- Algal Pigment Index 2 (mg.m^{-3})
- Yellow substance absorption (m^{-1})
- Sediment load (total suspended matter, g.m^{-3})

The C2R CFI software produces also two quality control flags.



Copyright © 2011 ACRI S.T.

Figure 8.5.2-1 : MERIS Case2R ocean colour processing (step 2.9.12) functional breakdown

Copyright © 2011 ACRI S.T.

8.6.3- Mathematical Description of Algorithm (step 2.9.12)

Implementation of the Case2R through a CFI software implies a minimalist mathematical description. The software starts from geometrical angles, pressure and wind speed, TOA signal corrected for gas and smile effect and returns marine inherent optical properties in logscale, to be converted into Chl2 concentration, ODOC absorption, SPM concentration, as well as two confidence flags.

The multiple non-linear regression method included in this approach leads to high reduction in computing time and is therefore fast enough for operational mass production of Level 2 products, but it requires a careful and elaborate determination of the multiple coefficients (training phase).

Copyright © 2011 ACRI S.T.

8.6.4- List of variables

Variable	Descriptive Name	T	U	Range-Reference
INVALID_F(j,f)	invalid flag	i	dl	from step 2.1a (§3.4)
CLOUD_F(j,f)	"cloud" flag	i	dl	from step 2.1c (§5.4)
LANDCONS_F(j,f)	consolidated "land" flag	i	dl	from step 2.1c (§5.4)
$\theta_s(j,f)$	Sun zenith angle	i	deg	from step 2.1a (§3.4)
$\theta_v(j,f)$	Viewing zenith angle	i	deg	*idem*
$\phi_s(j,f)$	Azimuth of sun angle	i	deg	*idem*
$\phi_v(j,f)$	Azimuth of view angle	i	deg	*idem*
$\rho^*_{ng}(b,j,f)$	TOA reflectance, corrected for stratospheric aerosol contribution and gaseous absorption for pixel (j,f)	i	dl	from step 2.6.9 (§8.4.2); b in {b412…b900}
$P_{ECMWF}(j,f)$	ECMWF pressure	i	hPa	from step 2.1a (§3.4)
$W_s(j,f)$	wind speed modulus	i	m.s^{-1}	from step 2.6.5 (§8.2.2)
$F_0(b)$	Reference Solar flux at MERIS theoretical wavelengths	s	EU	b in {b412…b900}
$\lambda_{heo}(b)$	Theoretical wavelengths corresponding to smile corrected reflectance	s	nm	b in {b412…b900}
$\tau_{O3\_norm}(b)$	Ozone optical thickness corresponding to 1cm.atm for all bands	s	dl	b in {b412…b900}
BAD_VALUE	Output value when algorithm fails	s	dl	see § 2 above
InvAbs_Chl2[4]	Conversion factors for Chl2	s	mg.m^{-3}/dl	
InvScat_SPM	Conversion factor for SPM	s	g.m^{-3}	
C2R_Input	Input vector of the C2R CFI software	c	misc	
C2R_Output	Output vector of the C2R CFI software	c	misc	
Chl2(j,f)	Algal pigment index 2	o	mg.m^{-3}	Defined in RD8 §2.10 & §2.11, to step 2.10 (§10.4), to Breakpoint
ORINP2_F(j,f)	Flag indicating whether input to the neural network is invalid	o	dl	Boolean, to step 2.10 (§10.4), to Breakpoint
OROUT2_F (j,f)	Confidence flag from neural network	o	dl	idem
ODOC(j,f)	Yellow substance absorption	o	m^{-1}	RD 8, §2.12, to step 2.10 (§10.4), to Breakpoint
SPM(j,f)	Total Suspended Matter	o	g.m^{-3}	RD 8, §2.12, to step 2.10 (§10.4), to Breakpoint

Copyright © 2011 ACRI S.T.

8.6.5- Equations (step 2.9.12)

8.6.5.1 - Process initialisation

Case2R Initialisation

```
call C2R_Init routine (see AD7)                                  (2.9.12-1)
    input: none
    return value: error status
```

exception processing: *if status in error*

- issue an error message
- Set CASE2R_INIT_FAIL to TRUE

end of exception processing

end of process initialisation section

8.6.5.2 - Pixel processing

For each pixel (j,f) such that ((**NOT** INVALID_F(j,f)) **AND** (**NOT** LANDCONS_F(j,f)) **AND** (NOT CLOUD_F(j,f))

exception processing: *if* CASE2R_INIT_FAIL == TRUE

- Apply exception handling defined in section 8.6.7
- Continue at next pixel

end of exception processing

```
Set inputs of C2R CFI
C2R_input(1) = $\theta_s(j,f)$                                   (2.9.12-2)
C2R_input(2) = $\theta_v(j,f)$                                   (2.9.12-3)
C2R_input(3) = $\phi_s(j,f)$                                     (2.9.12-4)
C2R_input(4) = $\phi_v(j,f)$                                     (2.9.12-5)
C2R_input(5) = $P_{ECMWF}(j,f)$                                  (2.9.12-6)
C2R_input(6) = Ws(j,f)                                           (2.9.12-7)
For b = b412..bb900
    C2R_input(7+b) = $\rho^{*}_{ng}(b,j,f)$                      (2.9.12-8)
    C2R_input(22+b) = $F_0(b)$                                   (2.9.12-9)
    C2R_input(37+b) = $\lambda_{theo}(b)$                        (2.9.12-10)
    C2R_input(52+b) = $\tau_{O3\_norm}(b)$                       (2.9.12-11)
Endfor

C2R CFI call:

call C2R_Invert routine (see AD7)                                (2.9.12-12)
        input: C2R_Input; number of input elements: 66;
        output: C2R_output; number of output elements: 5
```

Post-processing after C2R CFI call

Copyright © 2011 ACRI S.T.

```
    SPM(j,f) = InvScat_SPM.exp [C2R_output(1)]                                   (2.9.12-13)
    Chl2(j,f) = (InvAbs_Chl2[1] + InvAbs_Chl2[2] . C2R_output(2) ) .
                exp [InvAbs_Chl2[3] + InvAbs_Chl2[4] . C2R_output(2)]            (2.9.12-14)
    ODOC(j,f) = exp [C2R_output(3)]                                              (2.9.12-15)
    ORINP2_F (j, f) = C2R_output(4)                                              (2.9.12-16)
    OROUT2_F (j, f) = C2R_output(5)                                              (2.9.12-17)

Endfor

Case2R release

    call C2R_Free routine (see AD7)                                              (2.9.12-18)
        input: none
        return value: none
```

8.6.6- Quality Control and Diagnostics

The flag ORINP2_F provides confidence for the Inverse Model Technique algorithm included in step 2.9.12. It is based on a comparison (Chi-square difference) between the input radiance reflectance of the backward Neural Nets and reflectance reconstructed by the forward Neural Nets from the retrieved parameters.

The flag OROUT2_F checks that output marine concentrations are within the training range of the Inverse Model Technique algorithm included in step 2.9.12.

8.6.7- Exception Handling

For all non water pixels (see beginning of section 8.6.5.2) and in case Case2R initialisation fails, pixels where processing failed:

```
    Chl2(j,f) = BAD_VALUE
    SPM(j,f) = BAD_VALUE
    ODOC(j, f) = BAD_VALUE
    ORINP2_F (j, f) = TRUE
    OROUT2_F (j, f) = TRUE
```

Copyright © 2011 ACRI S.T.

9. MERIS Land Pixels Processing

9.1 - Overview

This chapter describes the algorithms to be applied to the MERIS Top Of Atmosphere reflectance in order to compute the MERIS land level 2 products:

a) quantitative

- Top Of Atmosphere Vegetation Index (TOAVI);
- Top of Aerosols reflectance in bands 412 to 885nm;
- Aerosol optical thickness and alpha above DDV
- Bottom Of Atmosphere Vegetation Index (BOAVI);

b) qualitative

- Dense Dark Vegetation (DDV) flag;

as well as flags relevant to the quality of all products.

The block diagram in figure 9.1-1 below shows the general logic of the main processing steps. These steps are detailed in sections 9.2 to 9.4 below.

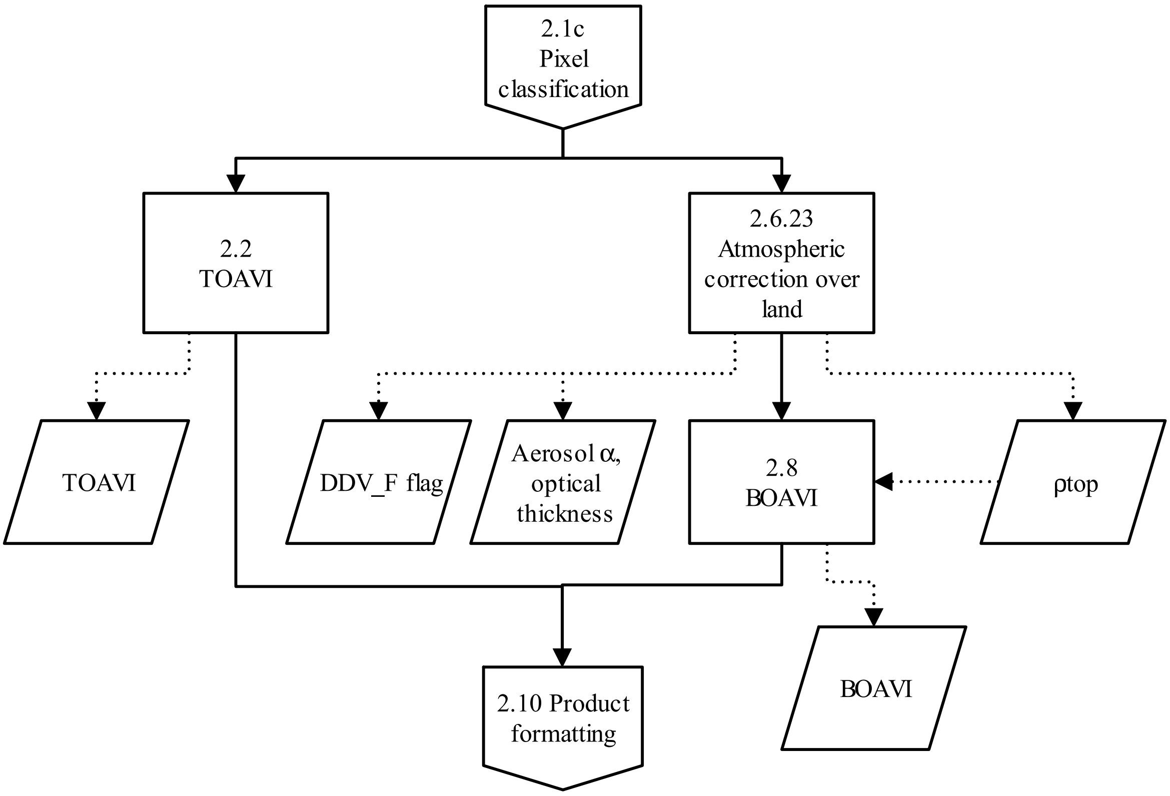


Figure 9.1-1: Land processing block diagram

Copyright © 2011 ACRI S.T.

9.2. - MERIS Top Of Atmosphere Vegetation Index (TOAVI) (step 2.2)

9.2.1. -Mathematical Description Of Algorithm

The diagram in figure 9.2.1-1 shows the logic of the TOA Vegetation Index computation. The algorithm takes as input the Top Of Atmosphere Reflectance output by step 2.1 .

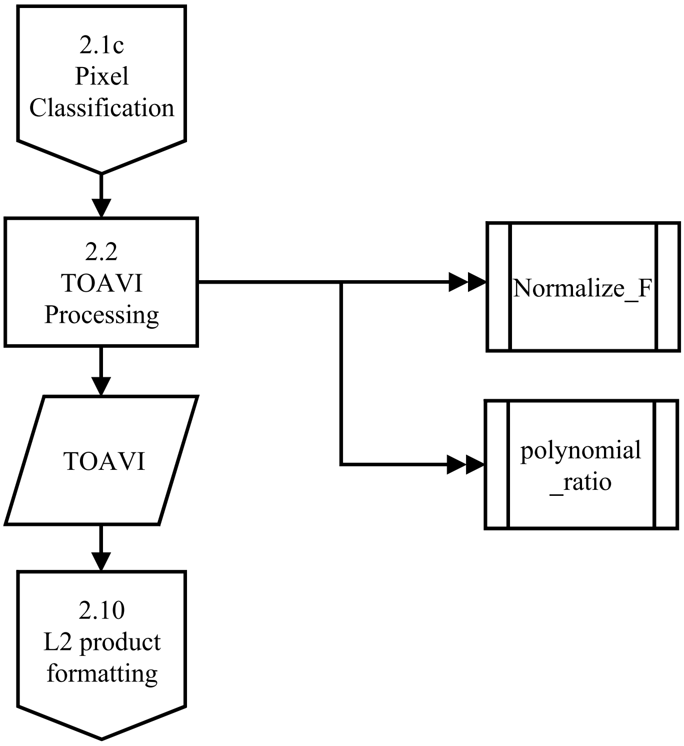


Figure 9.2.1-1 : MERIS Level 2 TOAVI computation

Before computing TOAVI, a spectral test is done on every Land pixels in order flagged any pixels that are not vegetated. Then, on the vegetated pixels, TOAVI or MERIS Global Vegetation Index (MGVI) is estimated in two steps. First, the information contained in the blue band at 442 nm is combined with that in the bands at 681 and 865 nm traditionally used to monitor vegetation, in order to generate "rectified channels" at these latter two wavelengths. The "rectification" is done in such a way as to minimise the difference between those rectified channels and the spectral reflectances that would be measured at the top of the canopy under a standard geometry of illumination and observation. The proposed algorithm assumes that ratios of polynomials are appropriate to generate both the "rectified channels" and the final spectral index, MGVI.

The MGVI has been optimised to assess the presence on the ground of healthy live green vegetation. The optimisation procedure has been constrained to provide an estimate of the Fraction of Absorbed Photosynthetically Active Radiation (FAPAR) in the plant canopy, although the index is expected to be used in a wide range of applications.

Copyright © 2011 ACRI S.T.

9.2.2. - List of Variables

Variable	Descriptive Name	T	U	Range - References
$\rho(b,j,f)$	stratospheric aerosol corrected reflectance for pixel j,f in band b	i	dl	from step 2.1c (§5.4)
LANDCONS_F(j,f)	Land/water consolidated flag	i	-	from step 2.1c (§5.4)
INVALID_F(j,f)	Invalid pixel flag	i	-	from step 2.1a (§3.4)
CLOUD_F(j,f)	Cloud pixel flag	i	-	from step 2.1c (§5.4)
$\theta_S(j,f)$	Sun zenith angle for pixel j,f	i	deg	from step 2.1a (§3.4)
$\theta_V(j,f)$	MERIS viewing angle for pixel j,f	i	deg	from step 2.1a (§3.4)
$\Delta\phi(j,f)$	Azimuth difference for pixel j,f	i	deg	from step 2.1a (§3.4)
blue_band_N	Blue band index number	s		
nir_band_N	Near infrared band index number	s		
red_band_N	Red band index number	s		
L[set][order]	Coefficients used for the polynomial ratios of reflectance rectification and TOAVI	s	dl	set: 1..5 order: 1..12
K_toavi[Class][band]	K_i toavi coefficients	s	dl	Class: VEG or BRIGHT Band: blue, red and nir
Theta_toavi[Class][band]	Theta toavi coefficients	s	dl	*idem*
Const_rho[Class][band]	Const_rho coefficients (1 coefficient for each blue, red and near infrared band)	s		*idem*
Max_rho[band]	Maximum acceptable TOA reflectances	s	dl	Band: blue, red and nir
Thresh_nir2red_ref	Near infrared to Red reflectance maximum ratio	s	dl	
BAD_VALUE	Output value when algorithm fails	s	-	see § 2
Normalized_rho_blue Normalized_rho_red Normalized_rho_nir	Normalized reflectances for the blue, red and near infrared bands	c	dl	
Rectified_rho_blue	Rectified reflectances for the blue band	c	dl	
TOAVI_CLASS_VEG(j,f)	Flag vegetated surface from TOAVI spectral tests	c	dl	
Class	Pixel class for Normalisation	c	dl	VEG or BRIGHT
ORINP1_F(j, f)	Out of range input flag # 1	o	dl	Boolean, to step 2.10 (§10.4), to Breakpoint
OROUT1_F(j, f)	Out of range output flag # 1	o	dl	idem
TOAVI (j,f)	TOA Vegetation Index for pixel j,f	o	dl	to step 2.10 (§10.4), to Breakpoint
Rectified_rho_red(j,f)	Rectified reflectance for red band	o	dl	to step 2.10 (§10.4) , to Breakpoint
Rectified_rho_nir(j,f)	Same as above for near infrared band	o	dl	to step 2.10 (§10.4) , to Breakpoint
TOAVI_CLASS_BAD(j,f)	Flag Bad data from TOAVI spectral tests	o	dl	To step 2.10 (§10.4)
TOAVI_CLASS_CSI(j,f)	Flag Cloud, snow or ice from TOAVI spectral tests	o	dl	To step 2.10 (§10.4)
TOAVI_CLASS_WS(j,f)	Flag water or deep shadow from TOAVI spectral tests	o	dl	To step 2.10 (§10.4)
TOAVI_CLASS_BRIGHT(j,f)	Flag bright from TOAVI spectral tests	o	dl	To step 2.10 (§10.4)
TOAVI_CLASS_INVAL_REC(j,f)	Flag invalid rectification	o	dl	To step 2.10 (§10.4)

Table 9.2.2-1: Parameters for TOAVI algorithm

NOTE: all calculated and output Boolean parameters shall be initialised to FALSE (0).

Copyright © 2011 ACRI S.T.

9.2.3. – Equations (step 2.2)

9.2.3.1. – TOAVI Processing (step 2.2)

For each pixel (j, f) such that (LANDCONS_F(j ,f) = **TRUE)**

The following spectral tests are used to set 5 TOAVI classes which shall be available as quality flags in the Level 2 products. TOAVI shall be calculated only if a vegeteated surface is identified.

```
Test1= ($\rho$(blue_band_N, j, f) <= 0  OR
        $\rho$(red_band_N, j, f)  <= 0  OR
        $\rho$(nir_band_N, j, f)  <= 0)                                    (2.2-1a)
Test2= ($\rho$(blue_band_N, j, f) >= Max_rho[blue]  OR
        $\rho$(red_band_N, j, f)  >= Max_rho[red] OR
        $\rho$(nir_band_N, j, f)  >= Max_rho[nir])                         (2.2-1b)
Test3= ($\rho$(nir_band_N, j, f)  >= Thresh_nir2red_ref *$\rho$(red_band_N, j, f))   (2.2-1c)
Test4= ($\rho$(blue_band_N, j, f) <= $\rho$(nir_band_N, j, f))                    (2.2-1d)


If(Test1) then set TOAVI_CLASS_BAD(j, f) = TRUE endif                     (2.2-2a)
If (Test2) then set TOAVI_CLASS_CSI(j, f) = TRUE endif                    (2.2-2b)
If NOT(Test1 OR Test2) then
   If NOT Test4 then set TOAVI_CLASS_WS(j, f) = TRUE                      (2.2-2c)
   Else if NOT Test3 then set TOAVI_CLASS_BRIGHT(j, f) = TRUE             (2.2-2d)
   Else set TOAVI_CLASS_VEG(j, f) = TRUE endif                            (2.2-2e)
   Endif
Endif

If (TOAVI_CLASS_VEG(j, f) OR TOAVI_CLASS_BRIGHT(j, f)) then
Compute normalised reflectances over both vegetated and bright land pixels
   If (TOAVI_CLASS_VEG(j, f)) then Class=VEG else Class=BRIGHT            (2.2-2f)
NOTE: the function Normalize_f is specified in 9.2.3.2 below
   Normalized_rho_blue = $\rho$(blue_band_N,j,f) /
         Normalize_F($\theta_s$(j, f), $\theta_v$ (j, f), blue, Class)              (2.2-3)
   Normalized_rho_red  = $\rho$(red_band_N,j,f)  /
         Normalize_F($\theta_s$(j, f), $\theta_v$ (j, f), red, Class)               (2.2-4)
   Normalized_rho_nir  = $\rho$(nir_band_N,j,f)  /
         Normalize_F($\theta_s$(j, f), $\theta_v$ (j, f), nir, Class)               (2.2-5)
```

```
exception processing: denominator = 0 in any of eq. (2.2-3) to (2.2-5) above:
      TOAVI (j, f) = BAD_VALUE
      Rectified_rho_red(j,f)=BAD_VALUE
      Rectified_rho_nir(j,f)=BAD_VALUE
      set ORINP_1 (j, f) = TRUE
      skip the rest of step 2.2
end of exception processing
```

```
   If ( Normalized_rho_blue    <= 0  OR
        Normalized_rho_red     <= 0  OR
```

Copyright © 2011 ACRI S.T.

```
    Normalized_rho_nir        <= 0  )
 Then
    TOAVI (j, f) = BAD_VALUE                                                    (2.2-6)
    set OROUT1_F(j, f) =TRUE                                                    (2.2-7)
 Else
    NOTE: the function Polynomial_ratio is specified in 9.2.3.3 below
    If valid normalisation, compute rectified reflectances accounting for surface type
    And compute TOAVI only for vegetated areas
    If (TOAVI_CLASS_VEG(j, f) then
       Rectified_rho_red(j,f) =
          Polynomial_ratio(Normalized_rho_blue,  Normalized_rho_red, 2)      (2.2-8)

       Rectified_rho_nir(j,f)   =
          Polynomial_ratio(Normalized_rho_blue, Normalized_rho_nir, 3)          (2.2-9)
       If (   Rectified_rho_red(j,f)   <= 0  OR
          Rectified_rho_nir(j,f)  <= 0    )
       Then
          TOAVI (j, f) = BAD_VALUE                                              (2.2-25)
          Rectified_rho_red(j,f)=BAD_VALUE                                      (2.2-26)
          Rectified_rho_nir(j,f)=BAD_VALUE                                      (2.2-27)
          set TOAVI_CLASS_INVAL_REC(j, f) = TRUE                                (2.2-28)
          set OROUT1_F(j, f) =TRUE                                              (2.2-29)
       Else
          TOAVI (j,f) =
             Polynomial_ratio(Rectified_rho_red(j,f), Rectified_rho_nir(j,f), 1)  (2.2-10)

          If TOAVI (j,f) >= 1 then TOAVI (j,f)=1                               (2.2-11)
 exception processing: TOAVI (j,f) < 0 in 2.2-10
       Force class to BRIGHT areas, set TOAVI to no data
       TOAVI (j, f) = BAD_VALUE
       TOAVI_CLASS_VEG(j, f) = FALSE
       TOAVI_CLASS_BRIGHT(j, f) = TRUE
       Re-compute normalisation and rectification according to new class
       repeat steps 2.2-2f, 2.2-3 to 2.2-5,
       continue processing at 2.2-30
 end of exception processing
       Endif
    Else if (TOAVI_CLASS_BRIGHT(j, f)) then
       TOAVI (j, f) = BAD_VALUE                                                 (2.2-30)
       Rectified_rho_red(j,f) =
          Polynomial_ratio(Normalized_rho_blue,  Normalized_rho_red, 4)      (2.2-31)

       Rectified_rho_nir(j,f)   =
          Polynomial_ratio(Normalized_rho_blue, Normalized_rho_nir, 5)          (2.2-32)
       If (Rectified_rho_red(j,f) ≤ 0 OR Rectified_rho_nir(j,f) ≤ 0 ) then
          Rectified_rho_red(j,f)=BAD_VALUE                                      (2.2-33)
          Rectified_rho_nir(j,f)=BAD_VALUE                                      (2.2-34)
```

Copyright © 2011 ACRI S.T.

```
                set TOAVI_CLASS_INVAL_REC(j, f) = TRUE                                  (2.2-35)
                set OROUT1_F(j, f) =TRUE                                                (2.2-36)
            Endif
        Endif       end of class selection for reflectance rectifcation
    Endif       end of Succesful Rectification over VEG and BRIGHT
Else      pixel is NOT identified as vegetation or bright land, set outputs to default  (2.2-20)
    TOAVI (j, f) = BAD_VALUE                                                            (2.2-21)
    Rectified_rho_red(j,f)=BAD_VALUE                                                    (2.2-22)
    Rectified_rho_nir(j,f)=BAD_VALUE                                                    (2.2-23)
    set ORINP_1 (j, f) = TRUE                                                           (2.2-24)
Endif       end of valid classes for normalisation selection
Endfor      end of loop over LANDCONS_F pixels
```

9.2.3.2 – Function Normalize_f (θ_s, θ_v, band, class)

The ***Normalize_F(θ_s, θ_v, band, class)*** *function is defined as follows :*
NOTE:
Parameters G, cosg, f1, f2, f3 are local parameters with unspecified meaning /range /unit.
Parameters band (NOT a MERIS band number) and class are indexes for arrays K_toavi, Theta_toavi.

$$G = \sqrt{\tan^2\theta_s + \tan^2\theta_v - 2.\tan\theta_s.\tan\theta_v.\cos\Delta\phi} \tag{2.2-13}$$

$$\cos\ g = \cos\theta_s * \cos\theta_v + \sin\theta_s * \sin\theta_v * \cos\Delta\phi \tag{2.2-14}$$

If

$$f_1 = \frac{(\cos\theta_s \cdot \cos\theta_v)^{(K\_toavi(Class,band)-1)}}{(\cos\theta_s + \cos\theta_v)^{(1-K\_toavi(Class,band))}} \tag{2.2-15}$$

$$f_2 = \frac{1 - Theta\_toavi(Class,band)^2}{\left(1 + 2 \cdot Theta\_toavi(Class,band) \cdot \cos g + Theta\_toavi(Class,band)^2\right)^{3/2}} \tag{2.2-16}$$

$$f_3 = 1 + \frac{1 - Const\_rho(Class,band)}{1+G} \tag{2.2-17}$$

$$\text{Normalize\_F} = f_1 * f_2 * f_3 \tag{2.2-18}$$

End of ***Normalize_F*** *function*

9.2.3.3 - Function Polynomial_ratio(ρ1,ρ2,set)

The ***Polynomial_ratio(ρ1,ρ2,band)*** *function is defined as follows :*

$$\text{Polynomial\_ratio} = \frac{L(set,1).\rho_1^2 + L(set,2).\rho_2^2 + L(set,3).\rho_1.\rho_2 + L(set,4).\rho_1 + L(set,5).\rho_2 + L(set,6)}{L(set,7).\rho_1^2 + L(set,8).\rho_2^2 + L(set,9).\rho_1.\rho_2 + L(set,10).\rho_1 + L(set,11).\rho_2 + L(set,12)} \tag{2.2-19}$$

Copyright © 2011 ACRI S.T.

```
exception processing: denominator = 0 in eq. (2.2-19) above:
    Polynomial_ratio = BAD_VALUE
end of exception processing
```

*End of **polynomial_ratio** function*

9.2.4. - Quality Control and Diagnostics.

Range checks are performed on input reflectances, normalised reflectances, rectified reflectances, see section 9.2.3 above.

9.2.5. - Exception Handling

See blocks labelled "exception processing:... end of exception processing" in section 9.2.3 above.

Copyright © 2011 ACRI S.T.

9.3. - Atmospheric correction over land (step 2.6.23)

This scheme is detailed in RD 8, sections 2.15, 2.17. The input of the algorithm are TOA reflectances, corrected for gaseous absorption and stratospheric aerosols, above all valid land pixels. The outputs of the algorithm are:

- Rayleigh corrected reflectances for all pixels;
- Dense Dark Vegetation (DDV) flag for all pixels;
- aerosol model index and Angström exponent for pixels flagged as DDV;

9.3.1. – Mathematical Description of the Algorithm

Figure 9.3.1-1 below describes the processing of the atmospheric correction over land in 13 MERIS bands (basic set of 15 bands described in section 2, minus the O_2 absorption band at 761.25 nm (band 11) and H_2O absorption band at 900 nm (band 15)).

9.3.1.1. - Rayleigh Correction Processing (step 2.6.15)

Rayleigh correction processing is organised in several steps (Fig. 9.3.1-2 below). First the Rayleigh reflectance, Rayleigh transmittance and Rayleigh spherical albedo are computed for every pixel. Then, the TOA apparent reflectance corrected for gaseous absorption ρ^*_{ng} is corrected for Rayleigh contributions for each pixel in order to derive the top of aerosol reflectance ρ_{top}.

9.3.1.2. – Dense Dark Vegetation (DDV) Screening (step 2.6.13)

DDV screening consists in flagging pixels identified as DDV (RD8, 2.17) by comparing a spectral index, the Atmosphere Robust Vegetation Index (ARVI), to a tabulated threshold which depends on Earth location and on the date of data acquisition. The ARVI is built using MERIS bands 2 (442 nm), 7 (665 nm) and 13 (865 nm).

9.3.1.3 – Aerosol above DDV (step 2.6.17)

Aerosol type and optical thickness are estimated for DDV pixels.

Aerosol optical thickness is provided at 442 nm (MERIS band 2), where the ground contribution to the signal is best accounted for. It is also derived at MERIS bands 1 and/or 7 (412 and 665 nm), allowing to characterise the aerosol Angström exponent, namely the alpha[3] product, expressing the spectral dependence of the optical thickness: $\tau(\lambda) = \tau(\lambda_0) \cdot \left(\frac{\lambda}{\lambda_0}\right)^{-\alpha}$. The nature of the aerosol models used above Land ensure that alpha remains constant over the whole MERIS spectral range.

---

[3] literature show the use of 2 definitions for alpha: with and without the minus sign. Present choice is arbitrary.

Copyright © 2011 ACRI S.T.

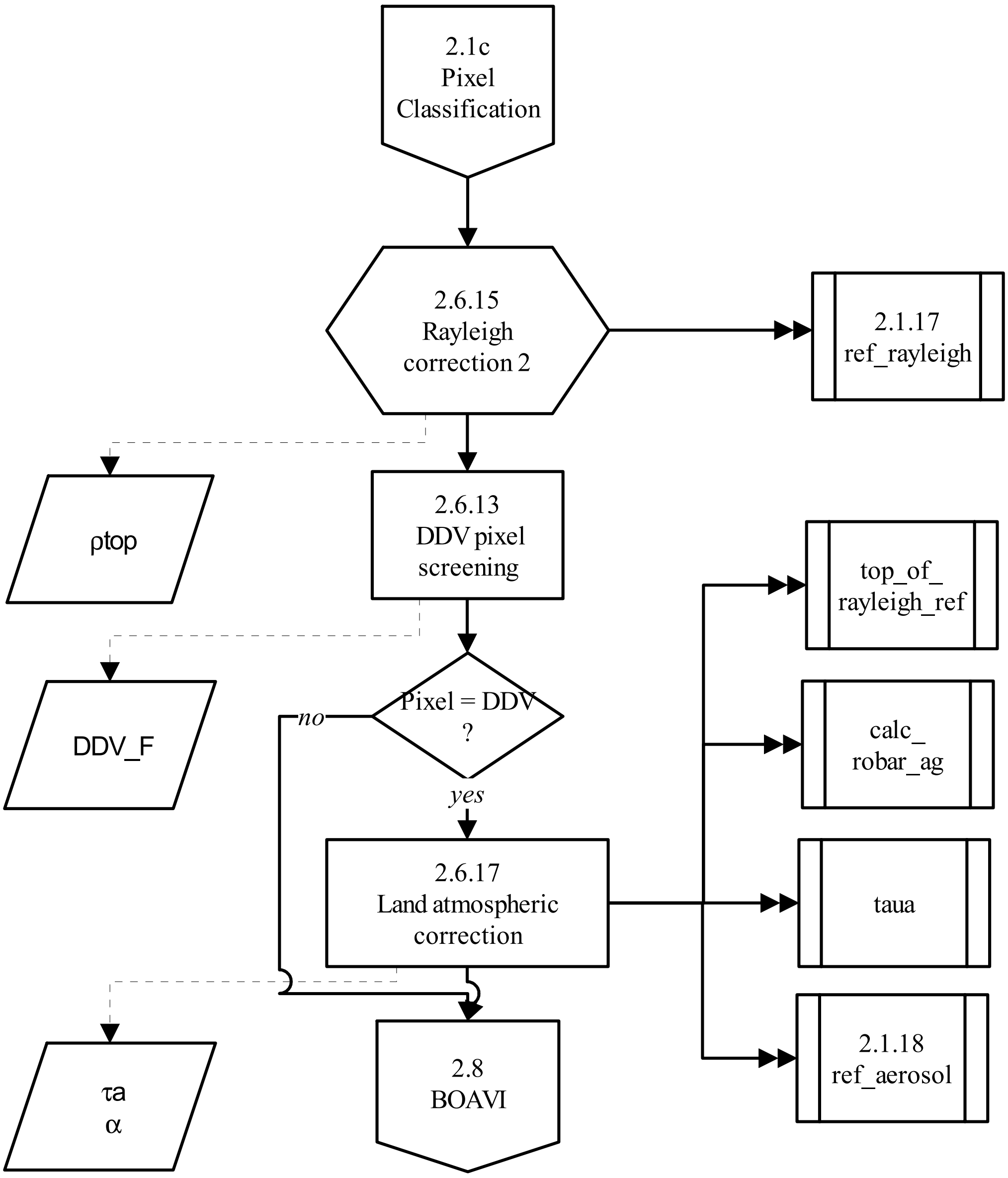


Figure 9.3.1-1: Land atmospheric corrections (step 2.6.23)

Copyright © 2011 ACRI S.T.

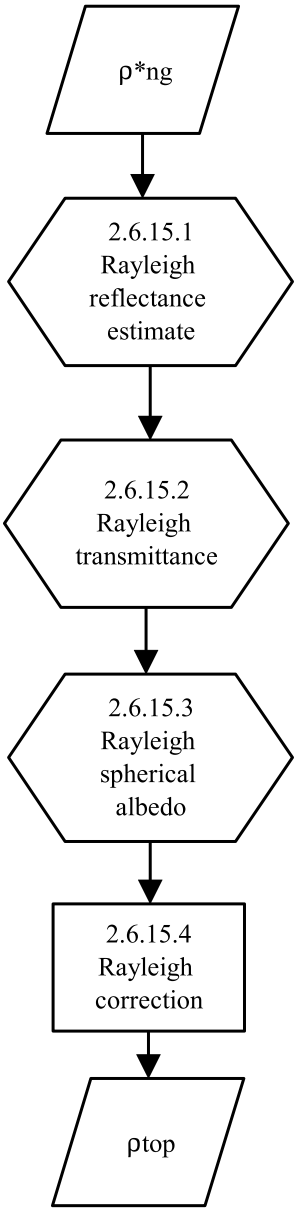


Figure 9.3.1-2: Rayleigh correction processing (step 2.6.15)

Copyright © 2011 ACRI S.T.

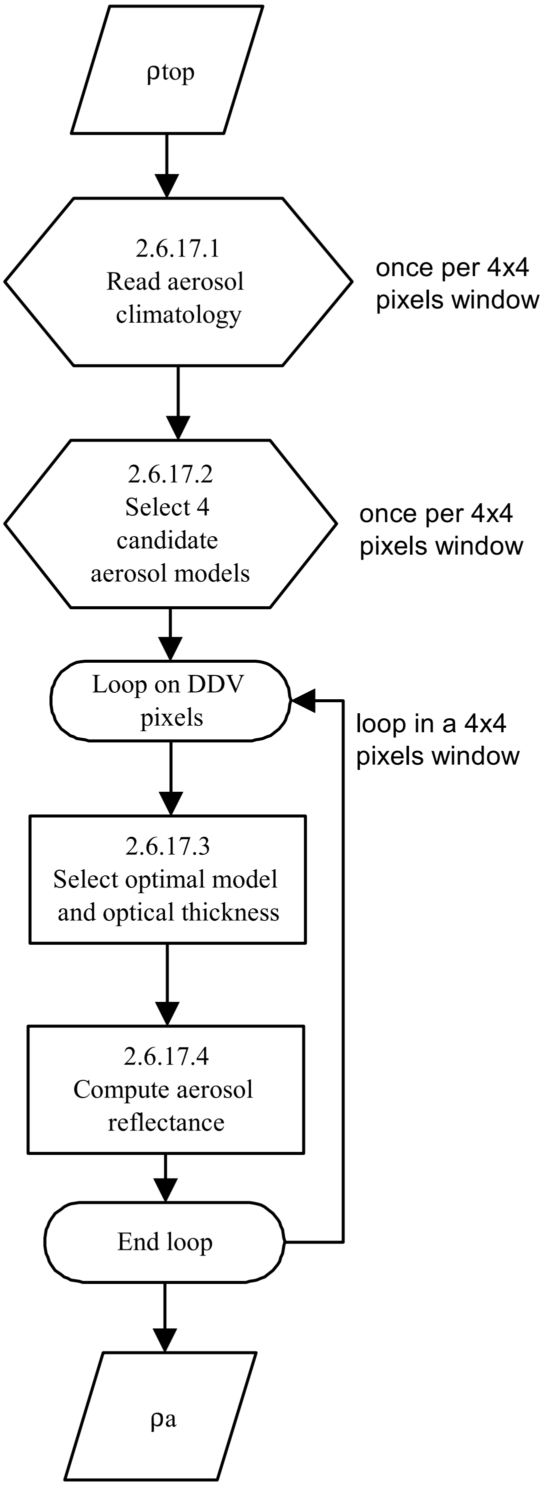


Figure 9.3.1-4: Land atmospheric corrections (step 2.6.17)

Copyright © 2011 ACRI S.T.

9.3.2. - List of Variables

Variable	Descriptive Name	T	U	Range - References
$\rho^*_{ng}(b,j,f)$	TOA reflectance corrected for gaseous absorption and smile for current pixel	i	dl	from step 2.1c
$\theta_s\,(j,f)$	Sun zenith angle	i	deg	from step 2.1a
$\theta_v\,(j,f)$	Viewing zenith angle	i	deg	from step 2.1a
$\Delta\phi(j,f)$	Azimuth angle between pixel-sensor and pixel-sun plane	i	deg	from step 2.1a
$P_{ECMWF}(j,f)$	ECMWF surface pressure	i	hPa	from step 2.1a
lat(j,f)	Latitude	i	deg	from step 2.1a
lon(j,f)	Longitude	i	deg	from step 2.1a
INVALID_F(j,f)	Invalid pixel flag	i	-	from step 2.1a
CLOUD_F(j,f)	Cloudy pixel flag	i	-	from step 2.1c
LANDCONS_F(j,f)	Land flag	i	-	from step 2.1c
month	Month of data acquisition	i	-	from L1B product MPH
$\rho_{R1}(b,j,f)$	Coarse Rayleigh reflectance	i	dl	b in {b412..b885}, from step 2.1c
$\tau_{R0}(b,j,f)$	Rayleigh optical thickness at nominal wavelengths, corrected for pressure	i	dl	b in {b412..b885}, from step 2.1c
$\lambda_{theo}(b)$	Theoretical wavelengths corresponding to smile corrected reflectances	s	nm	b in {b412..b900}
$\tau_R(b)$	Rayleigh optical thickness at standard pressure for all bands	s	dl	
P_{std}	Standard pressure	s	hPa	= 1013.25
t_0,t_1,t_2	Rayleigh transmittance coefficients	s	dl	
Rayalb_LUT[τ]	Rayleigh spherical albedo as a function of optical thickness	s	dl	τ : 0.02 to 0.32 by 0.02
DDV_clim [lat, lon]	Climatological table to select biome according to location	s	dl	
DDV_LUT [biome, month]	Climatological table to select DDV model according to biome and season	s	dl	biome: 1..11
DDV_ARVI_LUT[$\theta_s,\theta_v,\Delta\phi$, DDV_model]	ARVI threshold used for DDV screening	s	dl	78 (θ_s,θ_v) couples $\Delta\phi$ = 0 to 180° by 10° 20 DDV models
γ	Gamma coefficient used in ARVI computation	s	dl	$\gamma = 1.3$
Aerclim_LUT[lat,lon,month,k]	Aerosol climatology	s	dl	Lat : 0..180 by 1deg lon : 0..360 by 1 deg 78 aerosol models 12 optical thicknesses k=1 for τa, 2 for iaer,
Aerpha_LUT [$\cos\Theta_{sc}$, iaer]	Aerosol phase function times single scattering albedo	s	dl	$\cos\Theta_{sc}$: 83 values 78 aerosol models
Aerosol_refindex [iaer]	Aerosol refraction index m corresponding to each of the aerosol models.	s	dl	m in {1.33,1.44,1.55] 78 aerosol models

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
DDV_Bands[3]	List of band indices to be used for step 2.6.17-3	s	dl	Indices of 2 or 3 bands among {bb412, bb442, bb490, bb665}
DDV_THR_LUT [θ_s,θ_v,$\Delta\phi$, DDV_model, b]	DDV reflectances at 4 bands as function of geometry and DDV model	s	dl	b: b412, b442, b490, b665 78 (θ_s,θ_v) couples $\Delta\phi$ = 0 to 180° by 10° 20 DDV models
Aerosol_angstrom [iaer]	Aerosol Angström exponent α corresponding to each of the aerosol models.	s	dl	α = 0.,0.5,1.,1.5 78 aerosol models
$\Delta\tau_a$	Initial increment in optical thickness used within the iterative procedure	s	dl	0.1
Max_τ_a	Maximal optical thickness allowed within the iterative procedure	s	dl	1.5
TA_LUT[θ,τ_a,iaer]	Look-up table of aerosol transmittance	s	dl	τ_a : 15 values θ : 12 values iaer : 78 models
SA_LUT[τ_a,iaer]	Look-up table of aerosol spherical albedo	s	dl	τ_a : 15 values iaer : 78 models
robar_ra_LUT[k,iaer,θ]	Look-up table of the four polynomial order terms for the BRDF Rayleigh aerosol coupling term	s	dl	k : 0,1,2,3 iaer : 78 models θ : 12 values
robar_rg_LUT[b, DDV_model, θ]	Look-up table of the BRDF Rayleigh ground coupling term	s	dl	b: b412, b442, b490, b665 20 DDV models θ : 12 values
robar_ag_LUT[s,b,iaer, DDV_model,θ_s,θ_v]	Look-up table of the five Fourier series terms for the BRDF aerosol ground coupling term	s	dl	s : 0,1,2,3,4 b: b412, b442, b490, b665 iaer : 78 models 20 DDV models 78 (θs,θv) couples
albedo_g[DDV_model,b]	Ground albedo for a DDV model at band b	s	dl	20 DDV models b: b412, b442, b490, b665
ΔARVI$_{min}$_LUT[month, lat, lon]	Minimum acceptable value for ΔARVI	s	dl	1 deg. x 1 deg. lat, lon grid, 12 months
ΔARVI$_{max}$_LUT[month, lat, lon]	Maximum acceptable value for ΔARVI	s	dl	1 deg. x 1 deg. lat, lon grid, 12 months
Cnorm_LUT[month, lat, lon, band]	DDV reflectance monthly adjustment factors	s	dl	1 deg. x 1 deg. lat, lon grid, 12 months b: b412, b442, b490, b665
DDV_Slope_LUT[month, lat, lon, band]	DDV reflectance linear correction factors	s	dl	1 deg. x 1 deg. lat, lon grid, 12 months b: b412, b442, b490, b665
R$_{865}$_CS	Reflectance threshold to screen out cloud shadow from DDV pixels	s	dl	
ρ_{Ground}665_Threshold	Threshold on ground reflectance at 665 above which iteration on aerosol models shall be disabled	s	dl	

Variable	Descriptive Name	T	U	Range - References
j0, f0	column, line co-ordinates of North-east corner of 4x4 window	c	dl	
θ_{s_4x4}	Sun zenith angle for 4x4 sub-window	c	deg	
θ_{v_4x4}	View zenith angle for 4x4 sub-window	c	deg	
$\Delta\phi_{\_4x4}$	Azimuth difference for 4x4 sub-window	c	deg	
$M_{\_4x4}$	Air mass for 4x4 sub-window	c	dl	
$\tau_{R0}(b)$	Rayleigh optical thickness corrected for pressure for 4x4 sub-window	c	dl	
$T_R(\theta_s ,b)$	Transmittance used in Rayleigh transmittance computation	c	dl	
$T_{R\_\theta s}(b)$	Rayleigh tansmittance on Sun-surface path	c	dl	
$T_R(\theta_v ,b)$	Transmittance used in Rayleigh transmittance computation	c	dl	
$T_{R\_\theta v}(b)$	Rayleigh tansmittance on surface-sensor path	c	dl	
$S_R(b)$	Rayleigh spherical albedo for 4x4 sub-window	c	dl	
$\rho_{top}^{C}(b,j,f)$	TOA reflectance corrected for gaseous correction and Rayleigh scattering for current pixel	c	dl	RD 8, 2.15
$S_R(b,j,f)$	Rayleigh spherical albedo for current pixel	c	dl	*idem*
$T_{R\_\theta s}(b,j,f)$	Rayleigh transmittance on sun-target path for current pixel	c	dl	*idem*
$T_{R\_\theta v}(b,j,f)$	Rayleigh transmittance on target-sensor path for current pixel	c	dl	*idem*
biome	biome index	c	dl	
DDV_model	DDV model number	c	dl	to Breakpoint
ARVI_thres(j,f)	Threshold for ARVI index	c	dl	to Breakpoint
$\rho_{RB}(j,f)$	Reflectance used in ARVI estimation	c	dl	
ARVI(j,f)	ARVI index for current pixel	c	dl	to Breakpoint
ΔARVI(j,f)	Difference between ARVI of current pixel and ARVI threshold	c	dl	
lon_0	Longitude at centre of 32x64 window	c	deg	
lat_0	Latitude at centre of 32x64 window	c	deg	
AerModels[0..N-1]	Selection of aerosol models	c	dl	to Breakpoint
m	Aerosol refractive index	c	dl	
$\rho_{DDV}(b,j,f)$	Mean DDV reflectance in band b	c	dl	b: DDV_Bands
C_Corr(b)	correction factor for DDV reflectance and coupling terms	c	dl	b: DDV_Bands
$\rho_{Ground}(b)$	Corrected ground reflectance	c	dl	b: DDV_Bands, to Breakpoint
iaer	Aerosol model number	c	dl	
iaer1	Index of selected aerosol model	c	dl	

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
$\alpha 1$	Angström exponent for model iaer1	c	dl	
$\alpha_{calc}(k)$	Angström exponent retrieved for the aerosol models assuming a known refractive index	c	dl	k: iaer0,…,iaer25
B(k)	intercept of the function reg	c	dl	k: iaer0,…,iaer25
$\tau_{ak}(b,k)$	Interpolated aerosol optical thickness for current band and current model	c	dl	b: b in DDV_Bands k: iaer0,…,iaer25
τ_{a550}	Aerosol optical thickness at 550nm	c	dl	
τ_{atmp}	Temporary aerosol optical thickness	c	dl	
stock_taua	stored aerosol optical thickness for current band	c	dl	
$\rho^{*}_{ag}(b,j,f)$	Estimate reflectance above Rayleigh aerosol and DDV surface	c	dl	b: DDV_Bands
stock_rhoag	Stored estimate reflectance above Rayleigh, aerosol and DDV surface	c	dl	
ag_FOU(s)	Fourier series term for the aerosol ground BRDF coupling term	c	dl	s : 0,1,2,3,4
ra_POL(k)	Polynomial coefficient for Rayleigh aerosol BRDF coupling term	c	dl	k : 0,1,2,3
$Rob\_ag(\theta_s,\theta_v)$	BRDF aerosol ground coupling term for a given geometry	c	dl	to Breakpoint
$Rob\_ra(\theta)$	BRDF Rayleigh aerosol coupling term for a given zenith angle	c	dl	to Breakpoint
$Rob\_rg(\theta)$	BRDF Rayleigh ground coupling term for a given zenith angle	c	dl	to Breakpoint
S_g	ground albedo	c	dl	
ttrs	direct downward Rayleigh transmittance	c	dl	
ttrv	direct upward Rayleigh transmittance	c	dl	
ttas	direct downward aerosol transmittance	c	dl	
ttav	direct upward aerosol transmittance	c	dl	
tdrs	diffused downward Rayleigh transmittance	c	dl	
tdrv	diffused upward Rayleigh transmittance	c	dl	
tdas	diffused downward aerosol transmittance	c	dl	
tdav	diffused upward aerosol transmittance	c	dl	
$T_a(\theta)$	Aerosol transmittance for a given zenith angle	c	dl	
S_a	Aerosol spherical albedo	c	dl	
ia (j, f)	Aerosol model index	c	dl	for DDV pixels, to Breakpoint
$\rho_{top}(b,j,f)$	Top of aerosol reflectance for land pixel	o	dl	b: all except b761, b900, to step 2.10, to Breakpoint
DDV_F(j,f)	DDV flag for pixel (j,f)	o	dl	to Breakpoint

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range - References
RWNEG (b, j, f)	Flag indicating negative top of aerosol reflectance	o	-	Coding: see § 8.4.4, to step 2.10, to Breakpoint
τ_{a442} (j, f)	Aerosol optical thickness at 442 nm	o	dl	for DDV pixels, to step 2.10
α (j, f)	Aerosol Angström exponent	o	dl	for DDV pixels, to step 2.10, to Breakpoint

NOTE: all calculated and output Boolean parameters shall be initialised to FALSE.

9.3.3. - Equations

For each pixel (j, f) such that LANDCONS_F (j, f) == TRUE *(2.6.23-1)*

deleted *(2.6.23-2), (2.6.23-3), (2.6.23-4), (2.6.23-5)*

9.3.3.1. - Rayleigh correction (step 2.6.15)

1. Estimation of Rayleigh reflectance (step 2.6.15.1)

deleted (2.6.15.1-3), (2.6.15.1-4), (2.6.15.1-5).

For each band b in {b412..b753,b775..b885}

2. Estimation of Rayleigh transmittance T_R (step 2.6.15.2)

Compute Rayleigh transmittance on sun-surface path

$$T_R(\theta_s,b)=\frac{\left(2/3+\cos(\theta_s)\right)+\left(2/3-\cos(\theta_s)\right).e^{-\frac{\tau_{R0}(b,j,f)}{\cos(\theta_s)}}}{4/3+\tau_{R0}(b,j,f)} \qquad (2.6.15.2\text{-}1)$$

$$T_{R\_\theta s}(b,j,f)=t_0+t_1T_R\left(\theta_s,b\right)+t_2T_R^2(\theta_s,b) \qquad (2.6.15.2\text{-}2)$$

Compute Rayleigh transmittance on surface-sensor path

$$T_R(\theta_v,b)=\frac{\left(2/3+\cos(\theta_v)\right)+\left(2/3-\cos(\theta_v)\right).e^{-\frac{\tau_{R0}(b,j,f)}{\cos(\theta_v)}}}{4/3+\tau_{R0}(b,j,f)} \qquad (2.6.15.2\text{-}3)$$

$$T_{R\_\theta v}(b,j,f)=t_0+t_1T_R\left(\theta_v,b\right)+t_2T_R^2(\theta_v,b) \qquad (2.6.15.2\text{-}4)$$

3. Estimation of Rayleigh spherical albedo S_R (step 2.6.15.3)

S_R(b,j,f) = Rayalb_LUT **interpol :** (τ_{R0}(b,j,f)) *(2.6.15.3-1)*

4. Estimation of reflectance corrected for Rayleigh scattering (step 2.6.15.4)

deleted *(2.6.15.4-1) , (2.6.15.4-2), (2.6.15.4-3), (2.6.15.4-4)*

Copyright © 2011 ACRI S.T.

$$\rho^{C}_{top}(b,j,f)=\frac{\rho^{*}_{ng}(b,j,f)-\rho_{R1}(b,j,f)}{T_{R\_\theta s}(b,j,f)\cdot T_{R\_\theta v}(b,j,f)} \qquad (2.6.15.4\text{-}5)$$

exception processing: ρ^{C}_{top} (b,j,f) <= 0 in equation (2.6.15.4-5) above:
RWNEG(b, j, f) = TRUE
DDV_F (j,f) = FALSE
continue processing at 2.6.15.4-6
skip steps 2.6.13 and 2.6.17 for pixel (j,f)
end of exception processing

$$\rho_{top}(b,j,f)=\frac{\rho^{C}_{top}(b,j,f)}{1+S_R(b,j,f).\rho^{C}_{top}(b,j,f)} \qquad (2.6.15.4\text{-}6)$$

Endfor *End of loop over bands*
Endfor *End of loop over pixels*

9.3.3.2. - DDV Screening (step 2.6.13)

For each 4x4 pixels sub-window containing at least one pixel such that (LANDCONS_F (j,f) == TRUE)

latitude and longitude of window North-East corner

Let j0=north-east corner column index, f0= north-east corner line index,
$lat_0 = lat\ (j_0, f_0)$ *(2.6.13-7)*
$lon_0 = lon\ (j_0, f_0)$ *(2.6.13-8)*

select DDV model as a function of latitude, longitude and month
biome = DDV_clim **nearest**: (lat_0, lon_0) *(2.6.13-1)*
DDV_model = DDV_LUT **select**: (biome, month) *(2.6.13-9)*

For each pixel (j,f) in 4 x 4 window such that (LANDCONS_F (j,f) == TRUE)

Interpolate ARVI threshold as a function of geometry and DDV model
ARVI_thres(j,f)=DDV_ARVI_LUT **interpol:**(θ_s(j,f), θ_v(j,f),$\Delta\phi$ (j,f))
select:(DDV_model) *(2.6.13-2)*

Compute ARVI index from MERIS measurements corrected for gaseous absorption and Rayleigh scattering

$\rho_{RB}(j,f) = \rho_{top}(b665,j,f) - \gamma[\rho_{top}(b442,j,f) - \rho_{top}(b665,j,f)]$ *(2.6.13-3)*

exception processing: ρ_{RB} (j,f) <= 0 in (2.6.13-3):
DDV_F(j,f) = FALSE
skip equations *(2.6.13-4) to (2.6.13-6)*
continue processing
end of exception processing

$$ARVI(j,f)=\frac{\rho_{top}(b865,j,f)-\rho_{RB}(j,f)}{\rho_{top}(b865,j,f)+\rho_{RB}(j,f)} \qquad (2.6.13\text{-}4)$$

Copyright © 2011 ACRI S.T.

```
Compare computed ARVI with ARVI threshold and flag pixels identified as DDV
    $\Delta ARVI_{min} = \Delta ARVI_{min}\_LUT$ nearest: (lat(j,f), lon(j,f)), select: (month)     (2.6.13-10)
    $\Delta ARVI(j,f) = ARVI(j,f) - ARVI\_thres(j,f)$     (2.6.13-11)
    $DDV\_F(j,f) = (\Delta ARVI(j,f) > \Delta ARVI_{min})$ AND $(\rho^*_{ng}(b865, j, f) > R_{865}\_CS)$     (2.6.13-5)
        deleted     (2.6.13-6)
    Endfor End of loop over pixels
Endfor End of loop over 4x4 sub-windows
```

9.3.3.3. – Aerosols above DDV (step 2.6.17)

0. Define the band set to be used for aerosol identification (once for all at initialisation) (step 2.6.17.0)

```
Define the band set to be used for aerosol identification (once for all at initialisation)
DDV_BandSet ={}
nb_DDV=0
For b in {b412,b442,b490,b665} /* scan allowed bands in increasing wavelength order */
    If ( (b ∈ DDV_bands) AND (b ∉ DDV_BandSet) ) then  /* add found ones, but only once */
        DDV_BandSet = DDV_BandSet ∪ {b}
        nb_DDV = nb_DDV+1
    endif
Endfor
exception processing: nb_DDV ∉ {2,3}:
    issue error message
    Stop processing
end of exception processing
```

For each 4x4 pixels sub-window containing at least one pixel such that (LANDCONS_F (j,f) == TRUE)

1. Read climatology and retrieve aerosol model as a first guess for optical thickness at 550nm (step 2.6.17.1)

```
    Let j0=north-east corner column index, f0= north-east corner line index,
    $lat_0 = lat(j_0, f_0)$
    $lon_0 = lon(j_0, f_0)$

    iaer0 = Aerclim_LUT nearest: ($lat_0$, $lon_0$, month) select: (k=2)     (2.6.17.1-1)
    deleted     (2.6.17.1-2)
```

2. Select refractive index corresponding to the aerosol model found in climatology and the N-1 additional aerosol models having the same refractive index (step 2.6.17.2)

m = Aerosol_refindex (iaer0) *(2.6.17.2-1)*

Scan the table Aerosol_refindex to find additional aerosol models AerModels[0..N-1] that give the same refractive index; in other words that satisfy for all i (i between 0 and N-1):

```
Aerosol_refindex (AerModels[i]) == m                                    (2.6.17.2-2)
 deleted    (2.6.17.2-3)
 deleted    (2.6.17.2-4)
```

3. Derive optimal aerosol model within the set of N models, and its optical thickness, by iterative procedure (2.6.17.3)

```
Define geometry quantities at North-East corner of 4x4 window           (2.6.17.3-0)
$\theta_{s\_4x4} = \theta_s (j0, f0)$
$\theta_{v\_4x4} = \theta_{v\ s} (j0, f0)$
$\Delta\phi_{\_4x4} = \Delta\phi_{\ s} (j0, f0)$
$M_{\_4x4} = 1./\mathbf{cos} (\theta_{s\_4x4}) + 1. /\mathbf{cos} (\theta_{v\_4x4})$
For each pixel (j,f) in 4 x 4 window such that DDV_F(j,f) = TRUE
Interpolate reflectances for DDV pixels for selected bands and selected DDV model
    UsedAerModels = AerModels
    For each band b in DDV_bandSet
        $\rho_{DDV}(b,j,f)$ = DDV_THR_LUT interpol:($\theta_{s\_4x4}, \theta_{v\_4x4}, \Delta\phi_{\_4x4}$)
                                         select:(DDV_model,band = b)    (2.6.17.3-1)
        Compute surface reflectance at every band
        C_Corr(b) = RefCorr(lat, lon, month, b, $\rho_{DDV}(b,j,f)$, ΔARVI)  (2.6.17.3-51)
        $\rho_{Ground}(b) = \rho_{DDV}(b,j,f) \cdot C\_Corr$             (2.6.17.3-52)
        Compare surface reflectance at 665 nm to threshold
        If b == b665 AND $\rho_{Ground}(b) > \rho_{Ground}665\_Threshold$ then
            Too high: no chance to find good match, disable iterative search and select aerosol from climatology.
            UsedAerModels = {iaer0}
        endif
    End for End of loop over bands
```

Derive aerosol model and optical thickness by iterative procedure
(Iterative procedure for aerosol optical thickness (at all bands in DDV_BandSet) estimate: top of Rayleigh reflectances at each band are estimated; we vary optical thickness τ_a *in order to match measured top of Rayleigh reflectance* $\rho^*_{ng}(b,j,f)$*.*

NOTES:
The procedure ***top_of_rayleigh_ref*** , computing the reflectance above Rayleigh and aerosols layers for a given aerosol model, optical thickness, surface reflectance, geometry, is specified in section 9.3.3.6 below.
The function ***taua*** to derive aerosol optical thickness at one band b': $\tau_a(b')$, from its value at another band b: $\tau_a(b)$, for aerosol model **iaer**, is specified in section 9.3.3.7 below.

exception processing: $\rho_{Ground}(b) > \rho_{top}(b,j,f)$ for any $b \in$ DDV_BandSet:

Copyright © 2011 ACRI S.T.

```
        $\alpha(j, f)$ = BAD_VALUE
        $\tau_{a442}(j, f)$ = BAD_VALUE
        skip the rest of 2.6.17 and process next pixel
    end of exception processing

  For each aerosol model iaer in UsedAerModels
    Initial estimate of optical thickness using Angström exponent corresponding to selected aerosol model :
      $\alpha(iaer)$ = Aerosol_angstrom select : (iaer)                    (2.6.17.3-18)
      deleted (2.6.17.3-19)

      For each band b in DDV_BandSet
        $\tau_{a550} = 0$                                                  (2.6.17.3-39)
        $\tau_{atmp} = 0$                                                  (2.6.17.3-40)
        deleted (2.6.17.3-41)
        Initialise reflectance for an aerosol optical thickness equal to 0
        $\rho^*_{ag}(b, j, f)$= top_of_rayleigh_ref($\theta_{s\_4x4}$, $\theta_{v\_4x4}$, $\Delta\phi_{\_4x4}$, $M_{\_4x4}$,
                    iaer, $\tau_{atmp}$, $\rho_{R1}(b,j,f)$, $T_{R\_\theta s}(b,j,f)$ ,$T_{R\_\theta v}(b,j,f)$, $S_R(b,j,f)$,
                    $\rho_{Ground}(b)$, C_Corr(b), DDV_model, $\tau_{R0}(b)$, b)      (2.6.17.3-49)
        deleted (2.6.17.3-20)
        Repeat
          stock_rhoag = $\rho^*_{ag}(b, j,f)$                              (2.6.17.3-42)
          stock_taua = $\tau_{atmp}$                                       (2.6.17.3-43)
          $\tau_{a550} = \tau_{a550} + \Delta\tau_a$                       (2.6.17.3-44)
          $\tau_{atmp}$ = taua ($\tau_{a550}$, 550, $\lambda_{theo}(b)$, $\alpha(iaer)$)  /* optical thickness at band b */ (2.6.17.3-7)
          deleted (2.6.17.3-8)
          $\rho^*_{ag}(b,j,f)$= top_of_rayleigh_ref($\theta_{s\_4x4}$, $\theta_{v\_4x4}$, $\Delta\phi_{\_4x4}$, $M_{\_4x4}$,
                    iaer, $\tau_{atmp}$, $\rho_{R1}(b,j,f)$, $T_{R\_\theta s}(b,j,f)$, $T_{R\_\theta v}(b,j,f)$, $S_R(b,j,f)$,
                    $\rho_{Ground}(b)$, C_Corr(b), DDV_model, $\tau_{R0}(b)$, b)      (2.6.17.3-50)

          deleted (2.6.17.3-9), (2.6.17.3-21), (2.6.17.3-22), (2.6.17.3-10),
          (2.6.17.3-23), (2.6.17.3-24), (2.6.17.3-11)
          if ($\tau_{a550} > Max\_\tau_a$) break                          (2.6.17.3-25)
        until ($\rho^*_{ag}(b,j,f) \geq \rho^*_{ng}(b,j,f)$ ) convergence criterion    (2.6.17.3-12)
          deleted (2.6.17.3-26), (2.6.17.3-13), (2.6.17.3-14), (2.6.17.3-27),
          (2.6.17.3-15), (2.6.17.3-16), (2.6.17.3-34)

    Interpolate aerosol optical thickness at band b as a function of $\rho^*_{ag}(b,iaer,j,f)$
        $\tau_{ak}(b,iaer) = (\tau_{atmp} - stock\_taua).\frac{\rho^*_{ng}(b,j,f) - stock\_rhoag}{\rho^*_{ag}(b,j,f) - stock\_rhoag} + stock\_taua$    (2.6.17.3-45)

    exception processing: $\rho^*_{ag}(b,j,f)$ - stock_rhoag == 0 :
        $\tau_{ak}(b, iaer)$ = max(stock_taua, $10^{-6}$)
        skip (2.6.17.3-45) and continue loop
    end of exception processing
```

Copyright © 2011 ACRI S.T.

```
exception processing: $\tau_{a550} > Max\_\tau_a$ OR $\tau_{ak}(b, iaer) \leq 0$ :
    $\alpha_{calc}(iaer) = BAD\_VALUE$
    skip (2.6.17.3-46/-53) and continue loop on aerosol models
end of exception processing

  End for End loop on band

  Determine Angström exponent from retrieved optical thicknesses
  If (nb_DDV) == 2 then
    Let b1 and b2 be each of the 2 bands of DDV_BandSet
    $\alpha_{calc}(iaer) = \frac{\log(\tau_{ak}(b_2, iaer)) - \log(\tau_{ak}(b_1, iaer))}{\log(\lambda_{theo}(b_2)) - \log(\lambda_{theo}(b_1))}$        (2.6.17.3-53)
  Else
    Linear regression of log ($\tau_{ak}(b,k)$) as a function of log ($\lambda(b)$)
    NOTE: The linear regression reg function can be found in the Numerical Recipes (RD 5).
      call reg (log ($\lambda_{theo}(b)$), log ($\tau_{ak}(b,iaer)$), b in DDV_bandSet,
        slope = $\alpha_{calc}(iaer)$, intercept = B(iaer))        (2.6.17.3-46)
  Endif
End for End loop on aerosol model

Aerosol model selection
deleted      (2.6.17.3-31), (2.6.17.3-32)
  Select iaer1 within AerModels[0..N-1] such that :        (2.6.17.3-47)
    | $\alpha calc(iaer1)+\alpha(iaer1)$ | is minimal and $\alpha calc(iaer1) \neq BAD\_VALUE$
```

4. Compute aerosol parameters over DDV pixels (step 2.6.17.4)

```
Calculate aerosol optical thickness at 442 nm
  If b442 $\in$ DDV_BandSet then $\tau_{a442}(j, f) = \tau_{ak}(b442, iaer0)$        (2.6.17.4-1a)
  Else $\tau_{a442}(j, f) = \tau_{ak}(b_1, iaer0) \cdot \left(\frac{\lambda_{theo}(b442)}{\lambda_{theo}(b_1)}\right)^{\alpha calc(iaer1)}$        (2.6.17.4-1b)
  Where b1 is the lowest band in DDV_BandSet.

    deleted      (2.6.17.4-3, 2.6.17.4-5 to 2.6.17.4-7)

  $\alpha(j,f) = - \alpha_{calc}(iaer1)$        (2.6.17.4-9)
  ia (j, f) = iaer1        (2.6.17.4-10)

    deleted      (2.6.17.4-11 to 2.6.17.4-16)
```

Copyright © 2011 ACRI S.T.

```
exception processing:
    let $\alpha_{min}$ = min(Aerosol_angstrom select: (* in AerModels[0..N-1]))
    let $\alpha_{max}$ = max(Aerosol_angstrom select: (* in AerModels[0..N-1]))
    $\alpha_{step} = (\alpha_{max} - \alpha_{min}) / (N-1)$
    if $-\alpha_{calc}(iaer1) < \alpha_{min} - 2* \alpha_{step}$ OR $-\alpha_{calc}(iaer1) > \alpha_{max} + 2* \alpha_{step}$ then
        $\alpha(j, f)$ = BAD_VALUE
        $\tau_{a442}(j, f)$ = BAD_VALUE
    endif
end of exception processing
Endfor End of loop over pixels
Endfor End of loop over 4 x 4 windows
```

Copyright © 2011 ACRI S.T.

9.3.3.4 - Procedure *calc_robar_ag* to calculate aerosol ground BRDF coupling term

The List of variables below identifies the dummy input and output variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, whether read from a database or computed locally, are listed in table 9.3.2 above.

Variable	Descriptive Name	T	U
$\Delta\phi$	Azimuth angle between pixel-sensor and pixel-sun plane	i	deg
θ_s	Sun zenith angle	i	deg
θ_v	Viewing zenith angle	i	deg
Iaer	aerosol model index	i	dl
DDV_model	DDV model index	i	dl
b	MERIS band index	i	dl
Rob_ag(θ_s,θ_v)	Aerosol ground BRDF coupling term	o	dl

*The **calc_robar_ag** procedure is called by procedure top_of_rayleigh_ref (§ 9.3.3.6).*
*The **calc_robar_ag** procedure is defined as follows:*

```
Compute Fourier series terms
For each Fourier series order s = 0...4
        $ag\_FOU^{(s)}$=robar_ag_LUT interpol: ($\theta_s$, $\theta_v$)select: (s, b, iaer, DDV_model)
                (2.6.17.3.1-1)
Endfor End of loop over index s

Compute aerosol ground coupling term as a Fourier sum
/* care of the azimuth*/
$Rob\_ag(\theta_s,\theta_v)=ag\_FOU^{(0)}+2ag\_FOU^{(1)}\cos(\Delta\phi)+2ag\_FOU^{(2)}\cos(2(\Delta\phi))$
        $+2ag\_FOU^{(3)}\cos(3(\Delta\phi))+2ag\_FOU^{(4)}\cos(4(\Delta\phi))$        (2.6.17.3.1-2)
return Rob_ag

End of calc_robar_ag procedure
```

Copyright © 2011 ACRI S.T.

9.3.3.5 - Procedure *calc_robar_ra* to calculate Rayleigh aerosol BRDF coupling term

The List of variables below identifies the dummy input and output variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, whether read from a database or computed locally, are listed in table 9.3.2 above.

Variable	Descriptive Name	T	U
θ	zenith angle	i	deg
iaer	aerosol model index	i	dl
τ_a	aerosol optical thickness	i	dl
Rob_ra	Rayleigh aerosol BRDF coupling term	o	dl

*The **calc_robar_ra** procedure is called by procedure top_of_rayleigh_ref (§ 9.3.3.6).*
*The **calc_robar_ra** procedure is defined as follows:*

```
For each polynomial order k = 0…3
    $ra\_POL^{(k)}$=robar_ra_LUT interpol: ($\theta$) select: (k,iaer)                    (2.6.17.3.2-1)
Endfor End of loop over polynomial order

Compute Rayleigh aerosol BRDF coupling term
Rob_ra = $ra\_POL^{(0)}+(\tau_a)\ ra\_POL^{(1)\,)}+(\tau_a)^2\ ra\_POL^{(2)\,)}+(\tau_a)^3\ ra\_POL^{(3)}$    (2.6.17.3.2-2)
return Rob_ra

End of calc_robar_ra procedure
```

Copyright © 2011 ACRI S.T.

9.3.3.6 - Procedure *top_of_rayleigh_ref* to derive reflectance above Rayleigh and aerosols

9.3.3.6.1 – Input /Output

The List of Variables below identifies the dummy input and output variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, whether read from a database or computed locally, are listed in table 9.3.2 above.

Variable	Descriptive Name	T	U
θ_s	Sun zenith angle	i	deg
θ_v	Viewing zenith angle	i	deg
$\Delta\phi$	Azimuth angle between pixel-sensor and pixel-sun plane	i	deg
M	air mass	i	dl
iaer	aerosol model index	i	dl
τ_a	aerosol optical thickness	i	dl
ρ_{R1}	Rayleigh reflectance	i	dl
$T_{R\_\theta s}$, $T_{R\_\theta v}$	Rayleigh transmittance	i	dl
S_R	Rayleigh spherical albedo	i	dl
ρ_{Ground}	Corrected surface reflectance	i	dl
C_Corr	correction factor for DDV reflectance and coupling terms	i	i
DDV_model	DDV model index	i	dl
τ_{R0}	Rayleigh optical thickness	i	dl
band	band index	i	dl
ρ_{ag}	reflectance at top of Rayleigh and aerosol	o	dl

*The **top_of_rayleigh_ref** procedure is called by step 2.6.17.3.*

9.3.3.6.2 - Equations

NOTE: The procedure **ref_aerosol** computing the aerosols reflectance for a given aerosol model, optical thickness, geometry, is specified in section 5.5.7.

*The **top_of_rayleigh_ref** function is defined as follows :*

$$\cos\Theta_{scat} = -\sqrt{1-\mathbf{cos}^2\,\theta_s}\,.\sqrt{1-\mathbf{cos}^2\,\theta_v}\,\mathbf{.cos}\,\Delta\phi - \mathbf{cos}\,\theta_s\,\mathbf{.cos}\,\theta_v \qquad (2.6.17.3.3\text{-}1)$$

$Px\omega_0$ = Aerpha_LUT **interpol:** ($\cos\Theta_{scat}$) **select:** (iaer) *(2.6.17.3.3-2)*

ρ_a = **ref_aerosol** (θ_s, θ_v, $\Delta\phi$, M, iaer, τ_a, $Px\omega_0$) *(2.6.17.3.3-3)*

$$T_a(\theta_s) = \mathrm{TA\_LUT}\ \mathbf{interpol:}(\theta_s, \tau_a)\ \mathbf{select:}(\mathrm{iaer}) \qquad (2.6.17.3.3\text{-}4)$$

$$T_a(\theta_v) = \mathrm{TA\_LUT}\ \mathbf{interpol:}(\theta_v, \tau_a)\ \mathbf{select:}(\mathrm{iaer}) \qquad (2.6.17.3.3\text{-}5)$$

Copyright © 2011 ACRI S.T.

$S_a = SA\_LUT\ \mathbf{interpol}:(\tau_a)\ \mathbf{select}:(iaer)$ *(2.6.17.3.3-6)*

$Rob\_ag(\theta_s, \theta_v) = \mathbf{calc\_robar\_ag}(\theta_s, \theta_v, \Delta\phi, iaer, DDV\_model, band)$ *(2.6.17.3.3-7)*

$Rob\_ag(\theta_v, \theta_s) = \mathbf{calc\_robar\_ag}(\theta_v, \theta_s, \Delta\phi, iaer, DDV\_model, band)$ *(2.6.17.3.3-8)*

$Rob\_rg(\theta_s) = robar\_rg\_LUT\ \mathbf{interpol:}(\theta_s)\ \mathbf{select:}(DDV\_model, band)$ *(2.6.17.3.3-9)*

$Rob\_rg(\theta_v) = robar\_rg\_LUT\ \mathbf{interpol:}(\theta_v)\ \mathbf{select:}(DDV\_model, band)$ *(2.6.17.3.3-10)*

$Rob\_ra(\theta_s) = \mathbf{calc\_robar\_ra}(\theta_s, iaer, \tau_a)$ *(2.6.17.3.3-11)*

$Rob\_ra(\theta_v) = \mathbf{calc\_robar\_ra}(\theta_v, iaer, \tau_a)$ *(2.6.17.3.3-12)*

$S\_g = albedo\_g\ \mathbf{select:}(DDV\_model, band)$ *(2.6.17.3.3-13)*

deleted *(2.6.17.3.3-14)*

deleted (2.6.17.3.3-15), (2.6.17.3.3-16), (2.6.17.3.3-17), (2.6.17.3.3-18), (2.6.17.3.3-19)

deleted *(2.6.17.3.3-20)*

$Rob\_ag(\theta_s, \theta_v) = Rob\_ag(\theta_s, \theta_v) * C\_Corr$ *(2.6.17.3.3-21)*

$Rob\_ag(\theta_v, \theta_s) = Rob\_ag(\theta_v, \theta_s) * C\_Corr$ *(2.6.17.3.3-22)*

$Rob\_rg(\theta_s) = Rob\_rg(\theta_s) * C\_Corr$ *(2.6.17.3.3-23)*

$Rob\_rg(\theta_v) = Rob\_rg(\theta_v) * C\_Corr$ *(2.6.17.3.3-24)*

$S\_g = S\_g * C\_Corr$ *(2.6.17.3.3-25)*

$ttrs = e^{-\frac{\tau_{RO}}{\cos\theta_s}}$ *(2.6.17.3.3-26)*

$ttrv = e^{-\frac{\tau_{RO}}{\cos\theta_v}}$ *(2.6.17.3.3-27)*

$ttas = e^{-\frac{\tau_a}{\cos\theta_s}}$ *(2.6.17.3.3-28)*

$ttav = e^{-\frac{\tau_a}{\cos\theta_v}}$ *(2.6.17.3.3-29)*

$tdrs = T_{R\_\theta_s} - ttrs$ *(2.6.17.3.3-30)*

$tdrv = T_{R\_\theta_v} - ttrv$ *(2.6.17.3.3-31)*

$tdas = T_a(\theta_s) - ttas$ *(2.6.17.3.3-32)*

$tdav = T_a(\theta_v) - ttav$ *(2.6.17.3.3-33)*

reflectance above a DDV surface

$\rho_{ag} = \rho_{R1}$ +
$[(1./(1.-S_a*S\_g))*(\rho_{Ground}*ttas*ttav + Rob\_ag(\theta_v, \theta_s)*tdas*ttav + Rob\_ag(\theta_s, \theta_v)*tdav*ttas$
$+ S\_g*tdav*tdas)$ *
$(1./(1.-S_R*S\_g))*(ttrs*ttrv + (Rob\_rg(\theta_v)/\rho_{Ground})*tdrs*ttrv + (Rob\_rg(\theta_s)/\rho_{Ground})*tdrv*ttrs$
$+(S\_g/\rho_{Ground})*tdrv*tdrs)]$ +
$(1./(1.-S_a*S_R))*(\rho_a*ttrs*ttrv + Rob\_ra(\theta_v)*tdrs*ttrv + Rob\_ra(\theta_s)*tdrv*ttrs$
$+ S_a*tdrv*tdrs)$

(2.6.17.3.3-34)

return ρ_{ag}

End of top_of_rayleigh_ref procedure

Copyright © 2011 ACRI S.T.

9.3.3.7 - Procedure *taua* to compute aerosol optical thickness at band b

9.3.3.7.1 – Input /Output

The List of variables below identifies the dummy input and output variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, whether read from a database or computed locally, are listed in table 9.3.2 above.

Variable	Descriptive Name	T	U
τ_{a0}	aerosol optical thickness at reference wavelength	i	dl
λ_0	Reference wavelength	i	nm
$\lambda(b)$	Wavelength at band b	i	nm
α	Angström exponent	i	dl
τ_a (b)	aerosol optical thickness at band b	o	dl

The procedure taua is called by step 2.6.17.3 and by procedure top_of_rayleigh_ref (§ 9.3.3.6).

9.3.3.7.2 - Equations

The procedure taua is defined as follows:

Compute aerosol optical thickness

$$\tau_a(b) = \tau_{a0} \cdot \left(\frac{\lambda(b)}{\lambda_0}\right)^{-\alpha} \qquad (2.6.17.4\text{-}4)$$

return $\tau_a(b)$

9.3.3.7 - Procedure *RefCorr* to compute the correction factor required to derive surface reflectance from DDV reflectance

9.3.3.7.1 – Input /Output

The List of variables below identifies the dummy input and output variables of the procedure. Actual inputs and outputs should be traced to whichever algorithm step the procedure is called from. Other variables of the procedure, whether read from a database or computed locally, are listed in table 9.3.2 above.

Variable	Descriptive Name	T	U
Lat, Lon	Latitude and longitude	i	
month	Month of measurement	i	-
band	band index	i	dl
ρ_{DDV}	surface reflectance of DDV	i	dl
ΔARVI	Difference between ARVI of current pixel and ARVI threshold	i	dl

Copyright © 2011 ACRI S.T.

The procedure RefCorr is called by step 2.6.17.3 and by procedure top_of_rayleigh_ref (§ 9.3.3.6).

9.3.3.7.2 - Equations

The procedure RefCorr is defined as follows:

```
    C_norm=Cnorm_LUT nearest: (lat, lon), select(month, band)                 (2.6.17.3.3-14)
    slope=DDV_Slope_LUT nearest: (lat, lon), select(month, band)              (2.6.17.3.3-15)
    $\rho_{Ground}=\rho_{DDV}$* C_norm                                         (2.6.17.3.3-16)

    $\Delta ARVI_{max} = \Delta ARVI_{max}$_LUT nearest: (lat, lon), select: (month)   (2.6.17.3.3-17)
    If $\Delta ARVI < \Delta ARVI_{max}$ then
        C_ext=$(slope*\Delta ARVI+\rho_{Ground})/\rho_{Ground}$                (2.6.17.3.3-18)
    Else
        C_ext= $(slope*\Delta ARVI_{max}+\rho_{Ground})/\rho_{Ground}$         (2.6.17.3.3-19)
    Endif

    CorrFactor = C_norm * C_ext                                               (2.6.17.3.3-20)
return CorrFactor
```

9.3.4. - Confidence checks and diagnostics

The algorithm is able to estimate a valid ρ_{top} above any valid land pixel.

9.3.5. - Exception Handling

Any pixel such that INVALID_F = TRUE shall not be processed *(2.6.23-6)*

See blocks "exception processing:... end of exception processing" in section 9.3.3 above.

Copyright © 2011 ACRI S.T.

9.4. - MERIS Bottom Of Atmosphere Vegetation Index (BOAVI) (step 2.8)

9.4.1. – Mathematical Description of the Algorithm

The diagram in figure 9.4.1-1 shows the logic of the BOA Vegetation Index computation. The products delivered by the atmospheric corrections processing are used as input to the BOAVI algorithm.

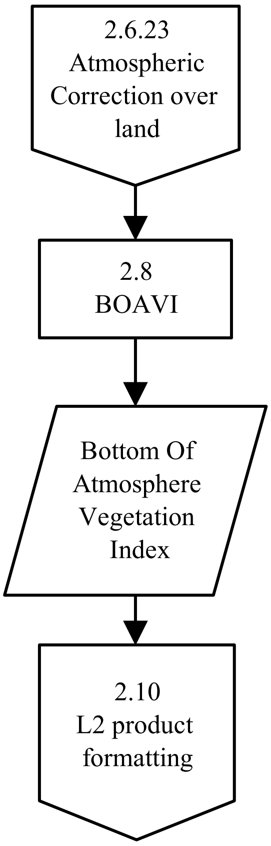


Figure 9.4.1-1 : MERIS Level 2 BOAVI computation (step 2.8)

The processing is only applied to land pixels. The algorithm is the MERIS Terrestrial Chlorophyll Index (MTCI).

Copyright © 2011 ACRI S.T.

9.4.2. - List of Variables

Variable	Descriptive Name	T	U	Range - References
$\rho_{top}(b,j,f)$	Top of aerosol reflectance for land pixels	i	dl	b: 3 bands specified through external data ; from step 2.6.23.
INVALID_F(j,f)	Invalid pixel flag	i	-	from step 2.1a
LANDCONS_F(j,f)	Land/water consolidated flag	i	-	from step 2.1c
BOAVI_RANGE(0..1)	Range limits for BOAVI	s	dl	0: min. valid value, 1: max. valid value
boavi_red_band	red band number for BOAVI	s	dl	Nominally b681
boavi_nir1_band	near infrared band #1 for BOAVI	s	dl	Nominally b705
boavi_nir2_band	near infrared band #2 for BOAVI	s	dl	Nominally b753
boavi_nir3_band	near infrared band #3 for BOAVI	s	dl	Nominally b865
ρ_red_max	Maximum value of ρ_{top} in red band to allow MTCI computation	s	dl	Nominally 0.3
ρ_nir2_min	Minimum value of ρ_{top} in nir2 band to allow MTCI computation	s	dl	Nominally 0.1
ρ_diff_min1	Minimum value of the reflectance difference between nir1 and red to allow MTCI computation	s	dl	Nominally $1.0\ 10^{-6}$
ρ_diff_min2	Minimum value of the reflectance difference between nir3 and red to allow MTCI computation	s	dl	Nominally 0.05
BAD_VALUE	Output value when algorithm fails	s	-	see § 2
BOAVI (j,f)	Vegetation index	o	dl	to step 2.10, to Breakpoint
ORINP2_F(j,f)	Out of range input flag for BOAVI	i/o	-	default: FALSE, to step 2.10, to Breakpoint
OROUT2_F(j,f)	Out of range output flag for BOAVI	i/o	-	default: FALSE, to step 2.10, to Breakpoint

Table 9.4.2-1: List of variables

9.4.3. - Equations (step 2.8)

For each pixel (j,f) such that **(INVALID_F(j,f) == FALSE) AND (LANDCONS_F(j,f) = TRUE)**

```
exception processing: when ($\rho_{top}$(boavi_red_band,j,f) <= 0) OR
    ($\rho_{top}$(boavi_red_band,j,f) >= ρ_red_max) OR
    ($\rho_{top}$(boavi_nir2_band,j,f) <= ρ_nir2_min) OR
    (|$\rho_{top}$(boavi_nir1_band,j,f)- $\rho_{top}$(boavi_red_band,j,f)| < ρ_diff_min1) OR
    ($\rho_{top}$(boavi_nir3_band,j,f)- $\rho_{top}$(boavi_red_band,j,f) < ρ_diff_min2) :
        ORINP2_F (j,f) = TRUE                                   (2.8-1)
        BOAVI(j,f) = BAD_VALUE                                  (2.8-2)
        skip the rest of step 2.8
end of exception processing
```

```
    $BOAVI(j,f) = \frac{\rho_{top}(boavi\_nir2\_band, j, f) - \rho_{top}(boavi\_nir1\_band, j, f)}{\rho_{top}(boavi\_nir1\_band, j, f) - \rho_{top}(boavi\_red\_band, j, f)}$    (2.8-3)
    If (BOAVI(j,f) < BOAVI_RANGE(0)) OR (BOAVI(j,f) > BOAVI_RANGE(1)) then
        OROUT2_F(j,f) = TRUE    (2.8-4)
        BOAVI(j,f) = BAD_VALUE    (2.8-5)
    Endif
Endfor
```

9.4.4. - Quality Control and Diagnostics

See equation (2.8-4) above.

9.4.5. - Exception Handling

See equations (2.8-1 to 2.8-2) above.

Copyright © 2011 ACRI S.T.

10. - MERIS Level 2 Product Formatting Algorithm

10.1 - Introduction

This chapter describes the processing to be applied to parameters used or created during the MERIS Level 2 processing, to generate the MERIS Level 2 products: Reduced Resolution and Full Resolution geophysical products.

10.2 - Algorithm Overview

MERIS processed data samples corresponding annotations and flags are collected from previous steps and formatted according to Level 2 product description in AD4.

10.3- Algorithm Description

10.3.1 - Theoretical Description

10.3.1.1 - Physics of The Problem

Each MERIS Level 2 geo-physical product is derived from a MERIS Level 1B product (herein after called "parent L1B product") and auxiliary parameter files specific of the MERIS Level 2 processing.

The MERIS Level 2 product is composed of : the Main Product Header (MPH), the Specific Product Header (SPH), one Summary Quality Annotation Data Sets (SQ ADS)), one Global Annotation Data Sets (GADS), one Annotation Data Sets and twenty 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. A large amount of SPH contents can be directly derived from the parent L1B product SPH.

The first ADS (SQ ADS) contains information on the quality of the product.

The GADS contains all the data scaling factors.

The second ADS contains information on geo-location, measurement viewing and illumination geometry and auxiliary environment parameters for a subset of the product pixels: the tie-points. One ADSR includes the set of tie points corresponding to a given satellite location. It is the same as in the parent Level 1B product.

The Measurement Data Sets (MDS) contain geo-physical parameters derived by the L2 processing. The products are distributed in order to obtain

- maximum homogeneity of the information: the "reflectance" bands, for instance, contain reflectance whatever the underlying surface is;
- maximum storage efficiency: the bytes allocated for a given pixel will be used to store different parameters, relevant to the surface observed.

The Flags MDS (20) contains all information needed to decode and check for quality the distributed pixel information.

Copyright © 2011 ACRI S.T.

One MDSR includes the parameters for all pixels corresponding to a given time sample of MERIS. The term "product line" will be used hereafter to name the MDSRs of the different MDS for the same time sample, i.e. with the same MDSR index.

Information coming either from parent Level 1B product, from external data sources, or generated by any processing step are gathered, organised, scaled and coded according to AD4 specifications to build the Level 2 product file.

10.3.1.2 - Mathematical Description of Algorithm

10.3.1.2.1 - Main Product Header

Main product header is formatted as described in AD4. The Error Message MPH field summarises the errors encountered in processing.

10.3.1.2.2 - Specific Product Header

Specific product header is formatted as described in AD4.

10.3.1.2.3 - Annotation Data Set "Summary Product Quality"

The annotation data set is composed of one Annotation Data Set Records (ADSR) for every 8 tie frames, i.e. every 128 (Reduced Resolution) or 512 (Full Resolution) product lines.

Each ADSR, following AD4, is composed of :

- Start time of the measurement or MJD, modified Julian Day of time sample
- Attachment Flag
- % of water pixels having absorbing aerosols (wrt water pixels)
- % of water, % of DDV land, % of land, % of cloud pixels (wrt valid pixels);
- % of pixels w/ bad surface pressure (wrt valid Land pixels);
- % of pixels w/ bad cloud top pressure (wrt valid Cloud pixels);
- % of pixels w/ out of range inputs for water vapour processing (wrt valid pixels);
- % of pixels w/ out of range outputs for water vapour processing (wrt valid pixels);
- % of pixels w/ out of range inputs for Cloud processing (wrt cloud pixels);;
- % of pixels w/ out of range outputs for Cloud processing (wrt cloud pixels);
- % of pixels w/ out of range inputs for Land processing (wrt land pixels);
- % of pixels w/ out of range outputs for Land processing (wrt land pixels);
- % of pixels w/ out of range inputs for Water processing (wrt water pixels);
- % of pixels w/ out of range outputs for Water processing (wrt water pixels);
- % of pixels w/ out of range inputs for Case 1 processing (wrt water pixels);
- % of pixels w/ out of range outputs for Case 1 processing (wrt water pixels);
- % of pixels w/. out of range inputs for Case 2 processing (wrt water pixels);
- % of pixels w/. out of range outputs for Case 2 processing (wrt water pixels);

The counters are accumulated according to every pixel in the time interval between a Q-ADSR (included) and the following one (excluded) and dumped to the Q-ADSR. The last Q-ADSR of the product may relate to a smaller number of product lines than the others.

10.3.1.2.4 - Global Annotation Data Set - Scaling Factors

Global Annotation Data Set is formatted as described in AD4. Scaling factors and offsets are read from an auxiliary data product.

Copyright © 2011 ACRI S.T.

10.3.1.2.5 - Annotation Data Set "Tie Points Location and corresponding Auxiliary Data"

Annotation Data Set "Tie Points Location and corresponding Auxiliary Data" is the same as found in the parent L1B product.

10.3.1.2.6 - Measurement Data Sets

There are 20 MDS:

- MDS 1 to 13 for the Normalised Reflectance for any valid pixel, at those MERIS bands not dedicated to gaseous absorption measurements: b412, b442, b490, b510, b560, b620, b665, b681, b705, b753, b775, b865, b885;
- MDS-14 for total water vapour for any valid pixel;
- MDS-15 for Algal Pigment Index I (water pixels) or TOAVI (land pixels) or Cloud Top Pressure (cloud pixels);
- MDS-16 for Yellow Substance and Total Suspended Matter (water pixels);
- MDS-17 for Algal Index II (water pixels) or BOAVI (land pixels);
- MDS-18 for PAR (water pixels) or Cloud Albedo (cloud pixels) or surface pressure (land and bright pixels);
- MDS-19 for Aerosols Angström exponent and optical thickness (water, land pixels) or cloud type and Optical Thickness (cloud pixels);
- MDS-20 for the associated flags for any pixel;

with the same record structure : an MDS is composed of one Measurement Data Set Record (MDSR) by product time sample. The structures are specified in AD4.

The normalised surface reflectance MDSR contains, according to AD4 :

- start time of sample in MJD2000 format;
- quality indicator (0 if nominal, -1 if no data are available; in such a case the data field of the MDSR is filled with zeroes);
- one (scaled) normalised surface reflectance value per pixel (1121 in RR, 2241 in FR, 1153 in FR imagette, 4481 in FR FullSwath).

Geo-physical parameters are expressed in counts using the scaling factor and offset stored in the GADS. Each value is stored in one or two bytes.

The flag MDSR contains :

- start time of sample in MJD2000 format;
- quality indicator
- one flag set (three bytes) per pixel (1121 in RR, 2241 in FR, 1153 in FR imagette, 4481 in FR FullSwath).

The coding of flags is specified in AD4.

10.4 - List of variables

Indexing convention :

- subscript b stands for the band index
- subscript j for the product pixel index
- subscript f for the product line index

Variable	Descriptive Name	T	U	Range-Reference
Product flags				
LAND_F(j,f)	Land/water pixel flag	i	dl	from L1B product flags MDS
COASTLINE_F(j,f)	Coastline pixel flag	i	dl	*idem*
DUPLICATED_F(j,f)	Duplicated pixel flag	i	dl	*idem*
COSMETIC_F(j,f)	Cosmetic pixel flag	i	dl	*idem*
SUSPECT_F(j,f)	Suspect pixel flag	i	dl	*idem*
INVALID_F(j,f)	Invalid pixel flag	i	dl	*idem*
PCD_NN_F(j,f)	Out of range input or output for NN pressure estimate	i	dl	from step 2.1b, §4.4
CLOUD_F(j,f)	Cloud flag	i	dl	from step 2.1c, §5.4
SNOW_ICE_F(j,f)	Snow or Ice flag	i	dl	From step 2.1c, § 5.4
ORINPWV_F(j,f)	Out of range input for Water Vapour processing	i	dl	from step 2.3, §6.4
OROUTWV_F(j,f)	Out of range output for Water Vapour processing	i	dl	from step 2.3, §6.4
ICE_HIGHAERO_F(j,f)	Ice or high aerosol loading flag	i	dl	from step 2.6.5, §8.2.2
LANDCONS_F(j,f)	Consolidated Land flag	i	dl	from step 2.1c, §5.4
MEGLINT_F(j,f)	Medium Glint flag	i	dl	from step 2.6.5, §8.2.2
UNCGLINT_F(j,f)	Flag for pixels non corrected for glint	i	dl	from step 2.6.5, §8.2.2
HIINLD_F(j,f)	Flag for low pressure water	i	dl	from step 2.6.5, §8.2.2
WHITECAPS_F(j,f)	Whitecaps flag	i	dl	from step 2.6.5, §8.2.2
DDV_F(j,f)	Dark Dense Vegetation flag	i	dl	from step 2.6.23, §9.3.2
CASE2_S(j,f)	Turbid water flag	i	dl	from step 2.6.8, §8.3.2
BPAC_ON_F(j,f)	Bright Pixel Atmosphere Correction turned ON flag	i	dl	from step 2.6.8, §8.3.2
WHITE_SCATT_F	Flag identifying “white” scatter within water	o	-	from step 2.6.8, §8.3.2
ACFAIL_F(j,f)	Atmosphere correction failed flag	i	dl	from step 2.6.10 §8.3.2, 2.6.9 §8.4.2
ORINP0_F(j,f)	Out of range input for atmosphere corrections	i	dl	from steps 2.6.12 §5.4 (cloud and land) and 2.6.10 §8.3.2 (water)
OROUT0_F(j,f)	Out of range output for atmosphere corrections	i	dl	from step 2.6.12 §5.4
ORINP1_F(j,f)	Out of range input TOAVI or Case 1 or Cloud albedo	i	dl	from steps 2.2, 2.4 or 2.9 (§ 9.2.2, §7.4 or 8.5.4)
OROUT1_F(j,f)	Out of range output TOAVI or Case 1	i	dl	*Idem*
ANNOT(j,f)	Annotation flag for the quality of the atmosphere correction	i	dl	from step 2.6.9 (§8.4.2); coding in table 8.4.4-1
RWNEG (b, j, f)	Annotation flag for negative corrected reflectance	i	dl	from step 2.6.9 (§8.4.2) or 2.6.23 (§9.3.2); coding in §8.4.4.
CASE2ANOM_F(j,f)	Anomalous Scattering flag	i	dl	from step 2.9, §8.5.4
CASE2Y_F(j,f)	Case 2 (y) flag	i	dl	from step 2.9, §8.5.4

Table 10.4-1 - Parameters used in the Formatting algorithm

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range-Reference
ORINP2_F(j,f)	Out of range input BOAVI or Case 2 or cloud optical thickness	i	dl	from steps 2.4, 2.8 or 2.9 (§7.4, 9.4.2 or 8.5.4)
OROUT2_F(j,f)	Out of range output BOAVI or Case 2	i	dl	from steps 2.8 or 2.9 (§9.4.2 or 8.5.4)
TOAVI_CLASS_BAD(j,f)	Flag Bad data from TOAVI spectral tests	i	dl	From step 2.2
TOAVI_CLASS_CSI(j,f)	Flag Cloud, snow or ice from TOAVI spectral tests	i	dl	From step 2.2
TOAVI_CLASS_WS(j,f)	Flag water or deep shadow from TOAVI spectral tests	i	dl	From step 2.2
TOAVI_CLASS_BRIGHT(j,f)	Flag bright from TOAVI spectral tests	i	dl	From step 2.2
TOAVI_CLASS_INVAL_REC(j,f)	Flag invalid rectification from TOAVI spectral tests	i	dl	From step 2.2
Product parameters				
P(j,f)	Surface Pressure	i	hPa	from step 2.1b, §4.4
P_{top}(j,f)	Cloud Top Pressure	i	hPa	from step 2.1b, §4.4
TOAVI(j,f)	Top of Atmosphere Vegetation Index	i	dl	from step 2.2, §9.2.2
W_T(j,f)	Water vapour content	i	g.cm^{-2}	from step 2.3, §6.4
α_c(j,f)	Cloud Albedo	i	dl	from step 2.4, §7.4
τ_c(j,f)	Cloud Optical Thickness	i	dl	from step 2.4, §7.4
Ctype (j,f)	Cloud type	i	dl	from step 2.4, §7.4
TOAR(b,j,f)	TOA Radiances	i	LU	All bands; from L1b MDS
ρ(b,j,f)	stratospheric aerosol corrected reflectance	i	dl	bands b412..b753, b775..b885 from step 2.1c, §5.4 (cloud)
ρ'_w(b,j,f)	Normalised water-leaving reflectance	i	dl	bands b412..b753, b775..b885 from step 2.6.9, §8.4.2 or 2.10, §8.3.2 (water)
ρ_{top}(b,j,f)	Top Of Aerosol Reflectance	i	dl	bands b412..b753, b775..b885 from step 2.6.23, §9.3.2 (land)
Rectified_rho_red(j,f)	Rectified reflect. for red band	i	dl	from step 2.2-1 §9.2.2
Rectified_rho_nir(j,f)	Same as above for near infrared band	i	dl	from step 2.2-1 §9.2.2
τ_{a865}(j,f)	Aerosol Optical Thickness at 865nm (water pixels)	i	dl	from step 2.6.9, §8.4.2 (water)
τ_{a442} (j, f)	Aerosol Optical Thickness at 442nm (land pixels)	i	dl	from step 2.6.23, §9.3.2 (land)
α_775_865(j,f)	Aerosol Angström exponent	i	dl	from step 2.6.9, §8.4.2 or 2.10, §8.3.2 (water pixels only)
α(j,f)	Aerosol Angström exponent	i	dl	from step 2.6.23, §9.3.2 (land only)
BOAVI(j,f)	Bottom of Atmosphere Vegetation Index	i	dl	from step 2.8, §9.4.2
Chl1 (j,f)	Algal pigment index 1	i	mg.m^{-3}	from step 2.9, §8.5.4
SPM (j,f)	Suspended Particulate Matter	i	g.m^{-3}	from step 2.9, §8.5.4
ODOC (j,f)	Yellow Substance	i	m^{-1}	from step 2.9, §8.5.4
Chl2 (j,f)	Algal pigment index 2	i	mg.m^{-3}	from step 2.9, §8.5.4
PAR (j,f)	Photosynthetically Available Radiation	i	µEinstein .m^{-2}.s^{-1}	from step 2.9, §8.5.4
NC	Number of samples per output line	i	-	from parent L1B product SPH
DF	Number of lines between along track tie points	i	-	from parent L1B product SPH
NF	Number of frames in product	i	dl	from parent L1B product SPH
SATURATED_F(b,j,f)	Saturated pixel flag	i	-	from 2.1.4 (§3.5)

Table 10.4-1 - Parameters used in the Formatting algorithm (cont.)

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range-Reference
reflect_scale	scaling factor - reflectances	s	dl	
Chl_scale	scaling factor - Algal pigment index	s	dl	
ODOC_scale	scaling factor - Yellow substance	s	dl	
SPM_scale	scaling factor - Suspended particulate matter	s	dl	
aot_scale	scaling factor - Aerosol optical thickn.	s	dl	
cot_scale	scaling factor - Cloud optical thickness	s	dl	
press_scale	scaling factor - Surface pressure	s	dl	
wv_scale	scaling factor - Water vapour	s	dl	
TOAVI _scale	scaling factor - TOA Vegetation index	s	dl	
BOAVI_scale	scaling factor - BOA Vegetation index	s	dl	
ca_scale	scaling factor - cloud albedo	s	dl	
ctp_scale	scaling factor - cloud top pressure	s	dl	
alpha_scale	scaling factor - alpha	s	dl	
reflect_offset	offset - reflectances	s	dl	
TETAS_LIMIT	Value of the sun zenith angle above which the annotation flag is modified	s	deg	
Chl_offset	offset - Algal pigment index	s	$\log_{10}$ (mg.m^{-3})	
ODOC_offset	offset - Yellow substance	s	m^{-1}	
SPM_offset	offset - Suspended particulate matter	s	$\log_{10}$(g.m^{-3})	
aot_offset	offset - Aerosol optical thickness	s	dl	
cot_offset	offset - Cloud optical thickness	s	dl	
press_offset	offset - Surface pressure	s	hPa	
wv_offset	offset - Water vapour	s	g.cm^{-2}	
PAR_offset	offset - PAR	s	μEinstein. m^{-2}	
TOAVI _offset	offset - TOA Vegetation index	s	dl	
BOAVI_offset	offset - BOA Vegetation index	s	dl	
ca_offset	offset - cloud albedo	s	dl	
ctp_offset	offset - cloud top pressure	s	hPa	
alpha_offset	offset - alpha	s	dl	
Rect_rho_scale[b]	Scaling factor for rectified reflectances in red and near infrared bands	s	dl	b:{0,1}
Rect_rho_offset[b]	Same as above for offsets	s	dl	b:{0,1}
DFSQ	Number of tie frames between two Q-ADSR	s	dl	See note [1]
INV_CLASS	Symbol for the class of invalid pixels	s	-	See note [1]
CLOUD_CLASS	Symbol for the class of cloud pixels	s	-	
LAND_CLASS	Symbol for the class of land pixels	s	-	
WATER_CLASS	Symbol for the class of water pixels	s	-	
boavi_nir_band	near infrared band number for BOAVI	s	-	
boavi_red_band	red band number for BOAVI	s	-	
blue_band_N	blue band for TOAVI	s	-	
red_band_N	red band for TOAVI	s	-	
nir_band_N	near infrared band for TOAVI	s	-	

Table 10.4-1 - Parameters used in the Formatting algorithm (cont.)

---

[1] The value of the symbols is left to implementation as it is internal to the application and language-dependent. For instance, a "C" program might use "enum" statement.

Copyright © 2011 ACRI S.T.

Variable	Descriptive Name	T	U	Range-Reference
PixClass (j, f)	Surface type of pixel (j, f)	c	-	
pc_aa_water	% of water pixels w/ absorbing aerosols	c	dl	
pc_water	% of water pixels	c	dl	
pc_ddv_land	% of DDV land pixels	c	dl	
pc_land	% of land pixels	c	dl	
pc_cloud	% of cloud pixels	c	dl	
pc_bad_surf_press	% of "bad" surface pressure	c	dl	
pc_bad_cloud_press	% of "bad" cloud top pressure	c	dl	
pc_inp_wv	pixels % w/ water vapour proc. out of range inputs	c	dl	
pc_out_wv	pixels % w/ water vapour proc. out of range outputs	c	dl	
pc_inp_cloud	pixels % w/ Cloud proc. out of range inputs	c	dl	
pc_out_cloud	pixels % w/ Cloud proc. out of range outputs	c	dl	
pc_inp_land	pixels % w/ Land proc. out of range inputs	c	dl	
pc_out_land	pixels % w/ Land proc. out of range outputs	c	dl	
pc_inp_water	pixels % w/ Water proc. out of range inputs	c	dl	
pc_out_water	pixels % w/ Water proc. out of range outputs	c	dl	
pc_inp_case1	pixels % w/ Case 1 proc. out of range inputs	c	dl	
pc_out_case1	pixels % w/ Case 1 proc. out of range outputs	c	dl	
pc_inp_case2	pixels % w/. Case 2 proc. out of range inputs	c	dl	
pc_out_case2	pixels % w/. Case 2 proc. out of range outputs	c	dl	
npix_land	number of land pixels	c	dl	
npix_water	number of water pixels	c	dl	
npix_cloud	number of cloud pixels	c	dl	
npix_image	number of pixels in the image (all pixels between two consecutive Tie-frames of the SQADS)	c	dl	
X (M, f, j)(k)	Coded geo-physical product	c	dl	See note 1

Table 10.4-1 - Parameters used in the Formatting algorithm (cont.)

Note 1: X is a data structure reflecting the values stored in the MDS,
X(M,f,j) is the parameter of MDS number M at product frame f and column j
X(M,f,j)(k) is the parameter of MDS number M at product frame f and column j and byte k for multi-byte parameters, e.g. yellow substance /total suspended matter in MDS 16.

Copyright © 2011 ACRI S.T.

10.5 - Equations

Note: data set structures are described precisely in AD4, in a way that allows to avoid a redundant description here.

10.5.1 - Step 2.10.1 Build MPH

format MPH according to AD4; ALL MPH fields are identical to the parent L1B MPH fields except for Product ID, Processing Center ID, UTC time of processing, Errors, Total size of product, Number of Data Sets attached *(2.10.1-1)*

write MPH

10.5.2 - Step 2.10.2 Build SPH

Format SPH according to AD4; ALL SPH fields are identical to the parent L1B SPH fields except for the Data Set Descriptors *(2.10.2-1)*

write SPH

10.5.3 - Step 2.10.3 Build GADS "Scaling Factors and Offsets"

Copy scaling factors for altitude, roughness, zonal wind, meridional wind, atmospheric pressure, ozone, relative humidity; gain setting, sampling rate, Sun spectral flux from the parent L1B GADS *(2.10.3-1)*

deleted (2.10.3-2) to (2.10.3-7)

scaling factor - reflectances = reflect_scale *(2.10.3-8)*
scaling factor - Algal pigment index = Chl_scale *(2.10.3-9)*
scaling factor - Yellow substance = ODOC_scale *(2.10.3-10)*
scaling factor - Suspended particulate matter = SPM_scale *(2.10.3-11)*
scaling factor - Aerosol optical thickness = aot_scale *(2.10.3-12)*
scaling factor - Cloud optical thickness = cot_scale *(2.10.3-13)*
scaling factor - Surface pressure = press_scale *(2.10.3-14)*
scaling factor - Water vapour = wv_scale *(2.10.3-15)*
scaling factor - PAR = PAR_scale *(2.10.3-16)*
scaling factor - TOA Vegetation index = TOAVI _scale *(2.10.3-17)*
scaling factor - BOA Vegetation index = BOAVI_scale *(2.10.3-18)*
scaling factor - Cloud albedo = ca_scale *(2.10.3-19)*
scaling factor - cloud top pressure = ctp_scale *(2.10.3-20)*
scaling factor – alpha = alpha_scale *(2.10.3-20b)*
offset- reflectances = reflect_offset *(2.10.3-21)*
offset- Algal pigment index = Chl_offset *(2.10.3-22)*
offset- Yellow substance = ODOC_offset *(2.10.3-23)*
offset- Suspended particulate matter = SPM_offset *(2.10.3-24)*
offset- Aerosol optical thickness = aot_offset *(2.10.3-25)*

Copyright © 2011 ACRI S.T.

```
offset- Cloud optical thickness          = cot_offset       (2.10.3-26)
offset- Surface pressure                 = press_offset     (2.10.3-27)
offset- Water vapour                     = wv_offset        (2.10.3-28)
offset- PAR                              = PAR_offset       (2.10.3-29)
offset- TOA Vegetation index             = TOAVI _offset    (2.10.3-30)
offset- BOA Vegetation index             = BOAVI_offset     (2.10.3-31)
offset- Cloud albedo                     = ca_offset        (2.10.3-32)
offset- cloud top pressure               = ctp_offset       (2.10.3-33)
offset- alpha                            = alpha_offset     (2.10.3-34)

write GADS
```

10.5.4 - Step 2.10.4 Build ADS "Summary Quality"

```
Build Annotation Data Set
Loop on tie points sub-grid lines
for each tie point grid line L with step of DF * DFSQ product lines
reset counters
   pc_aa_water     = 0;                                        (2.10.4-1)
   pc_water        = 0;                                        (2.10.4-2)
   pc_ddv_land     = 0;                                        (2.10.4-3)
   pc_land         = 0;                                        (2.10.4-4)
   pc_cloud        = 0;                                        (2.10.4-5)
   pc_bad_surf_press= 0;                                       (2.10.4-5.b)
   pc_bad_cloud_press= 0;                                      (2.10.4-5.c)
   pc_inp_wv       = 0;                                        (2.10.4-6)
   pc_out_wv       = 0;                                        (2.10.4-7)
   pc_inp_cloud    = 0;                                        (2.10.4-8)
   pc_out_cloud    = 0;                                        (2.10.4-9)
   pc_inp_land     = 0;                                        (2.10.4-10)
   pc_out_land     = 0;                                        (2.10.4-11)
   pc_inp_water    = 0;                                        (2.10.4-12)
   pc_out_water    = 0;                                        (2.10.4-13)
   pc_inp_case1    = 0;                                        (2.10.4-14)
   pc_out_case1    = 0;                                        (2.10.4-15)
   pc_inp_case2    = 0;                                        (2.10.4-16)
   pc_out_case2    = 0;                                        (2.10.4-17)
   npix_land       = 0;                                        (2.10.4-18)
   npix_water      = 0;                                        (2.10.4-19)
   npix_cloud      = 0;                                        (2.10.4-20)
   npix_image      = 0;                                        (2.10.4-21)

loop on tie points grid lines between two sub-grid lines
   for each product line f in L..L+DF.DFSQ-1
```

loop on all samples in image zone

Note: the following assumes that the logical value TRUE of flags is equivalent to the integer 1

```
      for each pixel j in 1..NC
Code pixel surface class for convenient use
         if (INVALID_F (j, f)) then
            PixClass (j, f) = INV_CLASS                                  (2.10.4-66)
         else if (CLOUD_F (j, f)) then
            PixClass (j, f) = CLOUD_CLASS                                (2.10.4-67)
         else if (LANDCONS_F (j, f)) then
            PixClass (j, f) = LAND_CLASS                                 (2.10.4-68)
         else
            PixClass (j, f) = WATER_CLASS                                (2.10.4-70)
         end if

         switch (PixClass (j ,f))
            case LAND_CLASS:
               npix_land   = npix_land + 1                               (2.10.4-22)
               break;
            case WATER_CLASS:
               npix_water  = npix_water + 1;                             (2.10.4-23)
               break;
            case CLOUD_CLASS:
               npix_cloud  = npix_cloud  + 1;                            (2.10.4-24)
               break;
         end switch
         If (NOT INVALID_F (j, f)) then
            npix_image  = npix_image     +1 ;                            (2.10.4-25)
         end if
         if (PixClass (j, f) == WATER_CLASS) then                        (2.10.4-26)
                    pc_aa_water = pc_aa_water + (bit ABSOA of ANNOT(j,f) == 1)
         end if
         deleted  (2.10.4-27)
         if (PixClass (j, f) == LAND_CLASS) then
            pc_ddv_land = pc_ddv_land+ DDV_F(j,f);                       (2.10.4-28)
         end if
         If (NOT INVALID_F (j, f)) then
            pc_bad_surf_press = pc_bad_surf_press +
               (PCD_NN_F(j,f) AND (PixClass(j,f) == LAND_CLASS))         (2.10.4-29)
            pc_bad_cloud_press  = pc_bad_cloud_press  +
               (PCD_NN__F(j,f) AND (PixClass(j,f) == CLOUD_CLASS))(2.10.4-30)
            pc_inp_wv   = pc_inp_wv + ORINPWV_F(j,f);                    (2.10.4-31)
            pc_out_wv   = pc_out_wv + OROUTWV_F(j,f);                    (2.10.4-32)
         end if
         deleted                                                         (2.10.4-33)
         deleted                                                         (2.10.4-34)
         if (PixClass (j, f) == LAND_CLASS) then
```

Copyright © 2011 ACRI S.T.

```
            pc_inp_land = pc_inp_land + ORINP0_F(j,f)                     (2.10.4-35)
            pc_out_land = pc_out_land + OROUT0_F(j,f)                     (2.10.4-36)
         else if (PixClass (j, f) == WATER_CLASS) then
            pc_inp_water = pc_inp_water + ORINP0_F(j,f)                   (2.10.4-37)
            pc_out_water = pc_out_water + OROUT0_F(j,f)                   (2.10.4-38)
            if ((NOT CASE2_S(j,f)) AND (NOT CASE2_Y(j,f))) then
               pc_inp_case1 = pc_inp_case1 + ORINP1_F(j,f)                (2.10.4-39)
               pc_out_case1 = pc_out_case1 + OROUT1_F(j,f)                (2.10.4-40)
            else
               pc_inp_case2 = pc_inp_case2 + ORINP2_F(j,f)                (2.10.4-41)
               pc_out_case2 = pc_out_case2 + OROUT2_F(j,f)                (2.10.4-42)
            end if
         else if (PixClass (j, f) == CLOUD_CLASS) then
            pc_inp_cloud = pc_inp_cloud + ( ORINP1_F(j,f) OR
               ORINP2_F (j,f) )                                           (2.10.4-64)
            pc_out_cloud = pc_out_cloud + ( OROUT1_F(j,f) OR
               OROUT2_F (j, f) )                                          (2.10.4-65)
         end if    /* PCD updated for valid pixels */
      end for      /* end loop on pixels */
if end of product reached, break the loop on product line
      if (f == NF) then
         break;                                                           (2.10.4-43)
      end if
   end for /* end of loop on product lines */
   pc_aa_water       = 100 * pc_aa_water   /  npix_ water ;               (2.10.4-44)
   pc_water          = 100 * npix_water    /  npix_image  ;               (2.10.4-45)
   pc_ddv_land       = 100 * pc_ddv_land   /  npix_ land    ;             (2.10.4-46)
   pc_land           = 100 * npix_land     /  npix_image  ;               (2.10.4-47)
   pc_cloud          = 100 * npix_cloud    /  npix_image  ;               (2.10.4-48)
   pc_bad_surf_press = 100 * pc_bad_surf_press  /  npix_land   ;          (2.10.4-48.b)
   pc_bad_cloud_press= 100  * pc_bad_cloud_press /  npix_cloud   ;        (2.10.4-48.c)
   pc_inp_wv         = 100 * pc_inp_wv     /  npix_ image ;               (2.10.4-49)
   pc_out_wv         = 100 * pc_out_wv     /  npix_ image ;               (2.10.4-50)
   pc_inp_cloud      = 100 * pc_inp_cloud /  npix_cloud    ;              (2.10.4-51)
   pc_out_cloud      = 100 * pc_out_cloud /  npix_cloud    ;              (2.10.4-52)
   pc_inp_land       = 100 * pc_inp_land   /  npix_land      ;            (2.10.4-53)
   pc_out_land       = 100 * pc_out_land   /  npix_land      ;            (2.10.4-54)
   pc_inp_water      = 100 * pc_inp_water /  npix_water    ;              (2.10.4-55)
   pc_out_water      = 100 * pc_out_water /  npix_water    ;              (2.10.4-56)
   pc_inp_case1      = 100 * pc_inp_case1 /  npix_water    ;              (2.10.4-57)
   pc_out_case1      = 100 * pc_out_case1 /  npix_water    ;              (2.10.4-58)
   pc_inp_case2      = 100 * pc_inp_case2 /  npix_water    ;              (2.10.4-59)
   pc_out_case2      = 100 * pc_out_case2 /  npix_water    ;              (2.10.4-60)
```

Copyright © 2011 ACRI S.T.

```
build Q-ADSR with MJD and flags registers
   start time field of QADSR = start time field of corresponding Q-ADSR in parent L1B product        (2.10.4-61)
   attachment flag of Q-ADSR = attachment flag of corresponding Q-ADSR in parent L1B product;        (2.10.4-62)

write Q-ADSR into L2 product
   write Q-ADSR;                                                                                     (2.10.4-63)
end for
```

10.5.5 - Step 2.10.5 Build ADS "Tie Points Annotations and corresponding Auxiliary Data"

```
Copy Annotation Data Set values from parent Level 1 product to Level 2 product ADS        (2.10.5 - 1)
```

10.5.6 - Step 2.10.6 Build Normalised Surface Reflectance MDS 1 to 13

Note: all MDS shall be initialised at the value BAD_PRODUCT for all pixels.

Build Measurements Data Sets
Data Sets 1 to 13 : Normalised Surface Reflectance

```
for each product line f in 1..NF /* loop to create all MDS of Level 2 product, line by line */

Time stamp and quality flag of each MDSR follow those in the parent L1B product

   for each MDS M of L2 product (M in 1..20)
      start time field of MDSR f in mds M = start time field of MDSR f in MDS(1)
      of parent L1B product                                                    (2.10.6-1)

      quality flag field of MDSR f in MDS M = quality flag field of MDSR f in
      MDS(1) of parent L1B product                                             (2.10.6-2)
   end for   /* end loop on MDS index */

   for each product column j
      deleted                                          (2.10.6-7, 2.10.6-7a, 2.10.6-7b)
Normalised reflectance in visible bands
      for b in { b412, b442, b490, b510, b560, b620, b665, b682, b705, b753,
      b775, b865, b885 }
         let M be the MDS index corresponding to band b
         If (SATURATED_F (b, j, f)) then
            X(M,f,j) = BAD_PRODUCT;                                            (2.10.6-7c)
         else

         switch (PixClass (j, f))
            case WATER_CLASS:
               if (NOT ACFAIL_F (j,f) then
```

```
                  if ($\rho'_w$(b,j,f) != BAD_VALUE) then
                     X(M,f,j)= int(($\rho'_w$(b,j,f)-reflect_offset(b))/
                                    reflect_scale(b));            (2.10.6-4)
```

exception processing: when X(M,f,j) < 1 OR X (M, f, j) > 65535:
X shall be clipped to the range extrema (65535 if the numerical count is above 65535, 1 if the numerical count is below 1); *(2.10.6-5)*
if (b <= b705) **then** PCD_1_13(f, j) = TRUE; *(2.10.6-6)*
end of exception processing

```
                  endif
               else
```

Atmosphere correction failed: Set field to 0 according to the convention in AD5

```
                  X(M,f,j) = BAD_PRODUCT;                          (2.10.6-8)
               end if;
               break;
            case LAND_CLASS:
               if ($\rho_{top}$(b,j,f) != BAD_VALUE) then
                  X(M,f,j) = int(($\rho_{top}$(b,j,f)-reflect_offset(b))/
                                    reflect_scale(b))];           (2.10.6-10)
```

exception processing: see 2.10.5, 2.10-6 above

```
               endif
               break;
            case CLOUD_CLASS:
               if ($\rho$(b,j,f) != BAD_VALUE) then
                  X(M,f,j) = MAX [1, int(($\rho$(b,j,f)-reflect_offset(b))/
                                           reflect_scale(b))];    (2.10.6-9)
```

exception processing: see 2.10.5, 2.10-6 above

```
                  endif
               break;
         end switch
         end if
      end for   /* end loop of reflectance spectral bands */
```

10.5.7 - Step 2.10.7 Build Total water vapour MDS 14

NOTE on exception processing: in this section and the following (up to § 10.5.12), when the parameter to code is different from BAD_VALUE and the encoded value X(M,f,j) falls outside the range [1..255], then the corresponding PCD flag shall be raised in X(19,f,j), and X shall be clipped to the range extrema (255 if the numerical count is above 255, 1 if the numerical count is below 1), and processing shall continue.

```
      if (PixClass (j, f) != INV_CLASS) then
         if (Wt(j,f) != BAD_VALUE) then
            X(14, f,j) = int((Wt(j,f) - wv_offset)/wv_scale)];     (2.10.7-1)
         end if
      end if
```

10.5.8 - Step 2.10.9 Build Algal index I or Top of Atmosphere Vegetation Index MDS 15

Copyright © 2011 ACRI S.T.

```
      switch (PixClass (j, f))
         case WATER_CLASS:
algal pigment index I in water
            if (NOT ACFAIL_F(j, f) AND Chl1(j,f) != BAD_VALUE) then
               X(15,f,j) = int((log10(Chl1(j,f)) - Chl_offset)/
                                Chl_scale);                    (2.10.9-1)
            else
               X(15,f,j) = BAD_PRODUCT;
            endif;
            break;
         case CLOUD_CLASS:
cloud top pressure
            if (Ptop(j,f) != BAD_VALUE) then
               X(15,f,j) = int((Ptop(j,f) - ctp_offset)/
                             ctp_scale);                       (2.10.9-2)
            endif;
            break;
         case LAND_CLASS:
TOAVI in land
            if (TOAVI(j,f) != BAD_VALUE) then
               X(15,f,j) = int( (TOAVI(j,f) - TOAVI_offset)/
                                TOAVI_scale)];                 (2.10.9-3)
            endif;
            break;
      end switch
```

10.5.9 - Step 2.10.10 Build Yellow Substance and Total Suspended Matter MDS 16.

```
      if (PixClass (j, f) == WATER_CLASS) AND (NOT ACFAIL_F (j,f)) then
Yellow substance and suspended matter interleaved in water
         if (ODOC(j,f) != BAD_VALUE) then
            X(16,f,j)(0) = int(log10((ODOC(j,f)) - ODOC_offset)/
                             ODOC_scale)];                     (2.10.10-1)
         endif;
         if (SPM(j,f) != BAD_VALUE) then
            X(16,f,j)(1) = int((log10(SPM(j,f)) - SPM_offset)/
                                SPM_scale)];                   (2.10.10-2)
         endif;
      else if (PixClass (j, f) == LAND_CLASS) then
         if (Rectified_rho_red(j, f) != BAD_VALUE) then
          X(16,f,j)(0)=int((Rectified_rho_red(j,f) - rect_rho_offset[0])/
                             rect_rho_scale[0])];              (2.10.10-4)
         endif
         if (Rectified_rho_nir(j, f) != BAD_VALUE) then
            X(16,f,j)(1)=int((Rectified_rho_nir(j,f) – rect_rho_offset[1])/
                             rect_rho_scale[1])];              (2.10.10-5)
         endif
      else
```

Copyright © 2011 ACRI S.T.

```
      X(16, f, j)(0..1) = BAD_PRODUCT                    (2.10.10-3)
   end if
```

10.5.10 - Step 2.10.11 Build Algal index II or Bottom of Atmosphere Vegetation Index MDS 17

```
   switch (PixClass (j, f))
      case WATER_CLASS:
         if (NOT ACFAIL_F (j,f) AND Chl2(j,f) != BAD_VALUE) then
algal pigment index II in water
            X(17,f,j) = int((log10(Chl2(j,f)) - Chl_offset)/
                               Chl_scale);               (2.10.11-1)
         else
            X(17,f,j) = BAD_PRODUCT;
         endif
         break;
      case LAND_CLASS:
BOAVI in land
         if (BOAVI(j,f) != BAD_VALUE) then
            X(17,f,j) = int((BOAVI(j,f) - BOAVI_offset)/
                               BOAVI_scale)];       (2.10.11-3)
         endif
         break;
   end switch
```

10.5.11 - Step 2.10.12 Build Pressure or PAR or Cloud Albedo MDS 18.

```
   switch (PixClass (j, f))
PAR in water
      case WATER_CLASS:
         if (NOT ACFAIL_F(j,f) AND PAR(j,f) != BAD_VALUE) then
            X(18,f,j) = int((PAR(j,f)) - PAR_offset)/ PAR_scale);
                                                         (2.10.12-1)
         else
            X(18,f,j) = BAD_PRODUCT;
         endif
         break;
cloud albedo
      case CLOUD_CLASS:
         if (αc(j,f) != BAD_VALUE) then
            X(18,f,j) = int((αc(j,f) - ca_offset)/ ca_scale);
                                                         (2.10.12-2)
         endif
         break;
Pressure in land
      case LAND_CLASS:
         if (P(j,f) != BAD_VALUE) then
```

Copyright © 2011 ACRI S.T.

```
            X(18,f,j) = int((P(j,f) - press_offset)/ press_scale);
                                                              (2.10.12-3)
         endif
         break;
   end switch
```

10.5.12 - Step 2.10.13 Build Aerosol alpha or Cloud type and optical thickness MDS 19

```
   switch (PixClass (j, f))
cloud optical thickness
      case CLOUD_CLASS:
         X(19,f,j)(0) = Ctype (j, f);                         (2.10.13-1)
         if ($\tau_c$(j,f) != BAD_VALUE) then
            X(19,f,j)(1) = int(($\tau_c$(j,f) - cot_offset)/ cot_scale);
                                                              (2.10.13-2)
         endif
         break;
aerosol type and optical thickness
      case LAND_CLASS:
         if ($\tau_{a442}$(j,f) != BAD_VALUE) then
            X(19,f,j)(1) = int(($\tau_{a442}$(j,f) - aot_offset)/
                                  aot_scale);                 (2.10.13-3)
         endif
         if ($\alpha$(j,f) != BAD_VALUE) then
            X(19,f,j)(0) = int(($\alpha$(j,f) - alpha_offset)/
                               alpha_scale)                   (2.10.13-4)
         endif
         break;
      case WATER_CLASS:
         if ($\tau_{a865}$(j,f) != BAD_VALUE) then
            X(19,f,j)(1) = int(($\tau_{a865}$(j,f) - aot_offset)/
                                  aot_scale);                 (2.10.13-5)
         endif
         if ($\alpha$_775_865(j,f) != BAD_VALUE) then
            X(19,f,j)(0) = int(($\alpha$_775_865(j,f) - alpha_offset)/
                               alpha_scale)                   (2.10.13-6)
         endif
         break;
   end switch
```

10.5.13 - Step 2.10.14 Build flags MDS

The logic of the combination of Internal flags (as defined in this document) to form Product Flags (as defined in AD4) is expressed in table below as a logical expression. Definitions of internal flags involved are listed together with their relevance with respect to pixel type. The column “Symbol” refers to the flags MDS, following AD4 table 5.4.1.8.8c. Symbols in the column “Equations” refer

Copyright © 2011 ACRI S.T.

to pixel flags (see table 10.4-1 above) or to the ANNOT (j, f) set of flags of the atmospheric correction above water, as described in table 8.4.4-1 above.

Symbol	**Equation**
LAND	LANDCONS_F
CLOUD	CLOUD_F
WATER	**NOT** (INVALID_F **OR** LANDCONS_F **OR** CLOUD_F)
PCD_1_13	INVALID_F **OR** [(WATER_CLASS **OR** LAND_CLASS) **AND** (ORINP0_F **OR** OROUT0_F **OR** RWNEG (b412..b705)**]** **OR** [WATER_CLASS **AND** (UNCGLINT_F **OR** HIINLD_F **OR** WHITECAPS_F **OR** ICE_HIGHAERO_F **OR** (**NOT**(CASE2_S **OR** CASE2Y_F OR CASE2_ANOM) AND DROUT) **OR** ACFAIL_F **OR** TAU06)**]**
PCD_14	INVALID_F **OR** ORINPWV_F **OR** OROUTWV_F **OR** **OR** SATURATED_F (b885, b900) **OR (**WATER_CLASS **AND** SATURATED_F (b775, b865)) **OR** (LAND_CLASS **AND** SATURATED_F (b753))
PCD_15	INVALID_F **OR** [(WATER_CLASS **OR** LAND_CLASS) **AND** (ORINP0_F **OR** OROUT0_F**]** **OR** [WATER_CLASS **AND** (UNCGLINT_F **OR** HIINLD_F **OR** WHITECAPS_F **OR** ICE_HIGHAERO_F **OR** ACFAIL_F **OR** TAU06 **OR** DROUT **OR** CASE2ANOM _F **OR** CASE2Y_F **OR** ORINP1_F) **OR** (SATURATED_F(b442, b490, b510, b560, b775, b865))**]** **OR** [CLOUD_CLASS **AND** (PCD_NN_F OR SATURATED_F(b753, b760))**]** **OR** [LAND_CLASS **AND** SATURATED_F(blue_band_N, red_band_N, nir_band_N)**]**
PCD_16	INVALID_F **OR** [WATER_CLASS **AND** (ORINP2_F **OR** OROUT2_F **OR** SATURATED_F(b412..b705, b775, b865)**]** **OR** [LAND_CLASS **AND** (ORINP1_F **OR** OROUT1_F **OR** SATURATED_F(blue_band_N, red_band_N, nir_band_N))**]**
PCD_17	INVALID_F **OR** [WATER_CLASS **AND** (ORINP2_F **OR** OROUT2_F **OR** SATURATED_F(b412..b705, b775, b865)**]** **OR** [LAND_CLASS **AND** (ORINP2_F **OR** OROUT2_F OR SATURATED_F(boavi_red_band, boavi_nir_band1, boavi_nir_band2))**]**

Copyright © 2011 ACRI S.T.

Symbol	Equation
PCD_18	INVALID_F **OR** [WATER_CLASS **AND** (ORINP0_F **OR** OROUT0_F **OR** ACFAIL_F **OR** UNCGLINT_F **OR** ICE_HIGHAERO_F **OR** HIINLD_F **OR** SATURATED_F(b775, b865))] **OR** [LAND_CLASS **AND** (PCD_NN_F **OR** SATURATED_F(b753, b760))] **OR** [CLOUD **AND** (ORINP1_F **OR** OROUT1_F **OR** SATURATED_F(b753))]
PCD_19	INVALID_F **OR** [(WATER_CLASS **OR** LAND_CLASS) **AND** (ORINP0_F **OR** OROUT0_F **OR** ACFAIL_F] **OR** [WATER_CLASS **AND** (WHITECAPS_F **OR** UNCGLINT_F **OR** ICE_HIGHAERO_F **OR** HIINLD_F **OR** ACLIM **OR** SATURATED_F(b775, b865) **OR** (**NOT**(CASE2_S **OR** CASE2Y_F **OR** CASE2_ANOM) **AND** DROUT))] **OR** [CLOUD **AND** (ORINP2_F **OR** SATURATED_F(b753, b760))] **OR** [LAND_CLASS **AND** DDV_F **AND** SATURATED_F(b412, b442, b490, b665, b865)]
COASTLINE	COASTLINE_F
COSMETIC	COSMETIC_F
SUSPECT	SUSPECT_F
OADB	WATER_CLASS **AND** MIXR1
ABSOA_DUST	WATER_CLASS **AND** ABSO_D
CASE2_S OR SNOW_ICE	[WATER_CLASS **AND** CASE2_S] **OR** [LAND_CLASS **AND** SNOW_ICE_F]
CASE2_ANOM **OR** TOAVI_BRIGHT	[WATER_CLASS **AND** CASE2ANOM_F] **OR** [LAND_CLASS **AND** TOAVI_CLASS_BRIGHT]
CASE2_Y **OR** TOAVI_BAD	[WATER_CLASS **AND** CASE2Y_F] **OR** [LAND_CLASS **AND** TOAVI_CLASS_BAD]
ICE_HAZE **OR** TOAVI_CSI	[WATER_CLASS **AND** ICE_HIGHAERO_F] **OR** [LAND_CLASS **AND** TOAVI_CLASS_CSI]
MEDIUM_GLINT **OR** TOAVI_WS	[WATER_CLASS **AND** MEGLINT_F] **OR** [LAND_CLASS **AND** TOAVI_CLASS_WS]
DDV OR BPAC_ON	[WATER_CLASS **AND** BPAC_ON_F] **OR** [LAND_CLASS **AND** DDV_F]
HIGH_GLINT **OR** TOAVI_INVAL_REC	[WATER_CLASS **AND** UNCGLINT_F] **OR** [LAND_CLASS **AND** TOAVI_CLASS_INVAL_REC]
LOW_SUN	(θ_s>TETAS_LIMIT)
WHITE_SCATTERER	[WATER_CLASS **AND** WHITE_SCATT_F]

Table 10.5.13 : MERIS Level 2 Internal Flags to Product Flags Mapping

The bit positions for the three bytes that are reserved for the flags are listed in AD4 Table 5.4.1.8.8.c .

deleted (2.10.14-4), (2.10.14-5), (2.10.14-8) (2.10.14-6)

```
   end for                    /* end of loop on product pixels */
```

10.5.14 – Write L2 MDS

```
   for each MDS index M
      write X(M,f,*)(*) to L2 product                          (2.10.14-9)
   end for
end for                        /*  end of loop on product lines */
```

Copyright © 2011 ACRI S.T.

10.6 - Accuracy Requirements

Start time field of Q-ADSR, ADSR, MDSR records shall be an exact copy of the start time field of the corresponding records of the parent L1B product.

Formatted values for all MDS fields shall be computed with an accuracy of 1 Least Significant Digit.

All tie point annotation fields shall be an exact copy of the tie point annotation fields of the parent L1B product.

All fields in Q-ADSR shall be computed with an accuracy of 1 Least Significant Digit.

10.7 - Product Confidence Data Summary

Product Formatting raises no PCD of its own. Confidence data computed in the previous steps are processed as follows:

10.7.1 - Flags obtained from the Level 1 processing

The quality flags obtained from Level 1 processing : SUSPECT_F, COSMETIC_F, are copied without modification to the corresponding bit in level 2 flags.

10.7.2 - Flags obtained from the Level 2 processing

Internal level 2 flags are combined according to Table 10.5.13 to yield Level 2 product flags.

10.7.3 - Summary quality ADS

Product metrics are provided by three counters: water pixels, land pixels, cloud pixels.

In addition to individual pixels flagging, the quality flags obtained from Level 2 processing : ORINPWV_F, OROUTWV_F, ORINP0_F, OROUT0_F, ORINP1_F, OROUT1_F, ORINP2_F, OROUT2_F, LOW_POL_F, LOW_NN_F are accumulated in a percentage per class (cloud, land, water, case 1 water, case 2 water, polynomial and NN pressure).

Attributes influencing the processing quality and Quality flags from the atmosphere corrections process are accumulated in a percentage per class:

1. water pixels having absorbing aerosols ;
2. Dense Dark Vegetation (DDV) land pixels, where the atmospheric corrections have been performed.

Copyright © 2011 ACRI S.T.

Copyright © 2011 ACRI S.T.

11 - BREAKPOINTS

The following data shall be used as breakpoints in the testing of the Level 2 processing. Subscripts (b, j, f) refer to band, product column, product frame. Unless explicitly stated, all bands, columns and frames are meaningful. Parameters are listed in the order of DPM appearance.

The column "cloud /land /water" indicates to which categories of pixel a breakpoint parameter is applicable. When not applicable, parameters should have a default value as specified in §2 above.

Symbol	Description	Unit	Cloud / Land / Water	Origin step no.	Required Accuracy
Numerical parameters					
lat (j, f)	Latitude	deg	clw	2.1.0	10^{-5}
lon (j, f)	Longitude	deg	clw	2.1.0	10^{-5}
θ_s (j, f)	Sun zenith angle	deg	clw	2.1.0	10^{-5}
θ_v (j, f)	View zenith angle	deg	clw	2.1.0	10^{-5}
ϕ_s (j, f)	Sun azimuth angle	deg	clw	2.1.0	10^{-5}
$\Delta\phi$ (j, f)	Azimuth difference angle	deg	clw	2.1.0	10^{-5}
P_{ECMWF} (j,f)	Surface pressure	hPa	clw	2.1.0	10^{-5*}
ρ_{TOA}(b,j,f)	TOA reflectance	dl	clw	2.1.4	10^{-5*}
iaer_sa (j, f)	Stratospheric aerosol index	-	clw	2.1.12	exact
ρ_{R1}	Coarse Rayleigh reflectance	dl	clw	2.1.7	10^{-5}
Papp(j,f)	Apparent surface pressure	hPa	clw	2.1.7	10^{-5*}
MDSI(j,f)	MERIS differential snow index	dl	clw	2.1.7	10^{-5}
α_{surf}(j,f)	Surface albedo	dl	clw	2.1.5	10^{-5*}
P_{top}(j,f)	Cloud Top pressure	hPa	c	2.1.5	10^{-5*}
P_S(j,f)	Surface pressure	hPa	l	2.1.12	10^{-5}
errcode (j,f)	Cloud top pressure error flags	-	clw	2.1.5	exact
L_T(b,j,f)	TOA radiance, corrected for stratospheric aerosol, b: b775, b865, b885, b900	dl	clw	2.3	10^{-5*}
ρ(b,j,f)	TOA reflectance, corrected for stratospheric aerosol	dl	clw	2.1.9	10^{-5*}
ρ_{ng}(b,j,f)	Reflectance corrected for stratospheric aerosol and gaseous absorption; b: all except {b760, b900}	dl	lw	2.6.12	10^{-5*}
ρ^{*}_{ng}(b,j,f)	TOA reflectance, corrected for stratospheric aerosol contribution, and gaseous absorption and smile effect; b: all except {b760, b900}	dl	lw	2.1.6	10^{-5*}
w_T(j,f)	Total water vapour content	g.cm^{-2}	clw	2.3	10^{-5*}
α_c(j,f)	Cloud albedo	dl	c	2.4.1	10^{-5*}

---

[*] Relative accuracy

Copyright © 2011 ACRI S.T.

Symbol	Description	Unit	Cloud / Land / Water	Origin step no.	Required Accuracy
τ_c(j,f)	Cloud Optical thickness	dl	c	2.4.3	10^{-5*}
Ctype (j,f)	Cloud type	-	c	2.4.8	exact
W_s(j,f)	wind speed modulus	$m.s^{-1}$	w	2.6.26	10^{-5*}
ROG (j, f)	Glint reflectance	dl	w	2.6.26	10^{-5*}
TOAROG(b,j,f)	Glint reflectance at TOA; b: {b412..b705; b775; b865}	dl	w	2.6.5	10^{-5*}
ROGC(b,j,f)	Reflectance corrected for sunglint; b: {b412..b705; b775; b865}	dl	w	2.6.5	10^{-5*}
ρ_R(b,j,f)	Rayleigh reflectance; b: all except {b760, b900}	dl	w	2.1.6	10^{-5*}
ρ_R(b,j,f)	Rayleigh reflectance; b: all except {b760, b900}	dl	l	2.6.15	10^{-5*}
bbp_775_fe(bs)	Final bbp estimate at band 775 retrieved by the BPAC; bs: {bsLOW, bsHIGH}	dl	w	2.6.8.6	10^{-5*}
ρ_w_fe(b, bs)	Final reflectance retrieved by the BPAC; b:{b775}, bs:{bsLOW, bsHIGH}	dl	w	2.6.8.6	10^{-5*}
ang_exp(bs)	Final Angrstrom exponent retrieved by the BPAC, bs:{bsLOW, bsHIGH}	dl	w	2.6.8.6	10^{-5*}
SPM_{br}(j,f)	Sediment load retrieved by the bright pixel method	$g.m^{-3}$	w†	2.6.8.8	10^{-5*}
tρ_w_C2 (b, j, f)	Marine reflectance; b: { b705, b775, b865, b885}	dl	w†	2.6.8.8	10^{-5*}
ia1(j,f), ia2(j,f) (water) ; iaer0(j,f), iaer1(j,f) (land)	Bracketing aerosol models	dl	lw	2.6.10, 2.6.9, 2.6.17,	Exact
aer_mix(j,f)	Mixing ratio	dl	w	2.6.9	10^{-5*}
f_a(b,j,f)	Aerosol forward scattering probability; b: {b412..b705, b775, b865}	dl	w	2.6.9	10^{-5*}
ω_a(b,j,f)	Aerosol single scattering albedo; b: {b412..b705, b775, b865}	dl	w	2.6.9	10^{-5*}
τ_a(b,j,f)	Aerosol optical thickness; b: {b412..b705, b775, b865}	dl	w	2.6.10, 2.6.9	10^{-5*}
t_u(b,j,f)	Transmittance on the target-sensor path	dl	w	2.6.9	10^{-5*}
t_d(b,j,f)	Transmittance on the Sun-target path	dl	w	2.6.9	10^{-5*}
ρ_a(b,j,f)	Aerosol reflectance; b : {b412…b705, b775, b865}	dl	w	2.6.9	10^{-5*}
α_775_865 (j, f) (water), α(j,f) (land)	Aerosol alpha	dl	lw	2.6.10, 2.6.9, 2.6.17	10^{-5*}
ρ'_w(b,j,f)	Normalised water-leaving Reflectance; b : {b412, b442, b665, b775, b865}	dl	w	2.6.10, 2.6.9	10^{-5*}
Chl1(j,f)	Algal pigment index 1	$mg.m^{-3}$	w	2.9.7	10^{-5*}
Chl2(j,f)	Algal pigment index 2	$mg.m^{-3}$	w	2.9.11	10^{-5*}
ODOC(j,f)	Yellow substance absorption	m^{-1}	w	2.9.11	10^{-5*}
SPM(j,f)	Suspended particulate matter	$g.m^{-3}$	w	2.9.11	10^{-5*}
PAR (j ,f)	PAR	$\mu Einstein.m^{-2}.s^{-1}$	w	2.9.8	10^{-5*}
TOAVI (j,f)	TOA Vegetation Index	dl	l	2.2	10^{-5*}
ρ_{top}(b,j,f)	Rayleigh corrected reflectance; b: all except b760, b900	dl	l	2.6.15	10^{-5*}

---

† only when BPAC_ON (j, f) is set

Copyright © 2011 ACRI S.T.

Symbol	Description	Unit	Cloud / Land / Water	Origin step no.	Required Accuracy
DDV_model (j, f)	DDV model index	-	l	2.6.13	Exact
ARVI (j,f)	ARVI	-	l	2.6.13	10^{-5*}
AerModels (j, f)	aerosol model indices	-	l	2.6.17	exact
BOAVI (j,f)	BOAVI	dl	l	2.8	10^{-5*}
Rectified_rho_red(j,f)	Rectified reflectance for red band	dl	l	2.2	10^{-5*}
Rectified_rho_nir(j,f)	Same as above for near infrared band	dl	l	2.2	10^{-5*}
Flags					
INVALID_F (j, f)	Invalid flag	-	all	2.1.0	exact
SATURATED_F(b,j,f)	Saturated pixel flag	-	clw	2.1.4	exact
LAND_F (j, f)	Land flag from L1B	-	all	2.1.0	exact
errcode (j,f)	Cloud top pressure error flags	-	clw	2.1.5	exact
PCD_NN_F(j,f)	Flag for out of range input or output in cloud top pressure	-	clw	2.1.5	exact
BRIGHT_F(j,f)	Flag for bright pixel	-	clw	2.1.7	exact
BRIGHT_RC_F(j,f)	Flag for bright pixel (from Rayleigh corrected ρ)	-	clw	2.1.7	exact
BRIGHT_TOA_F(j,f)	Flag for bright pixel for TOA ρ	-	clw	2.1.7	exact
SLOPE_1_F (j, f)	Spectral slope test 1 flag	-	clw	2.1.7	exact
SLOPE_2_F (j, f)	Spectral slope test 2 flag	-	clw	2.1.7	exact
CLOUD_F(j,f)	Flag for cloud pixel	-	clw	2.1.8	exact
LowP_F(j,f)	Flag on low apparent pressure	-	clw	2.1.7	exact
SNOW_ICE_F(j,f)	Snow or Ice flag		clw	2.1.8	exact
ORINP0_F(j,f)	Out of range input flag from the gaseous absorption corrections	-	lw	2.6.12, 2.6.9	exact
OROUT0_F(j,f)	Out of range output flag from the gaseous absorption corrections	-	lw	2.6.12, 2.6.9	exact
UNCERTAIN_F (j, f)	Uncertain Surface Type flag	-	lw	2.6.26	exact
ISLAND_F(j,f)	Flag for land in water	-	l	2.6.26	exact
LOINLD_F(j,f)	Flag for Inland water	-	w	2.6.26	exact
LANDCONS_F (j, f)	Land /water consolidated flag	-	lw	2.6.26	exact
ORINPWV_F(j,f)	Flag for out of range input for water vapour processing	-	clw	2.3	exact
OROUTWV_F(j,f)	Flag for out of range output for water vapour processing	-	clw	2.3	exact
ORINP1_F(j,f)	Out of range input flag from the Cloud, Chl_1 or TOAVI algorithm	-	clw	2.2, 2.4, 2.9.7	exact
OROUT1_F(j,f)	Out of range ouput flag from the Cloud, Chl_1 or TOAVI algorithm	-	clw	2.2, 2.4, 2.9.7	exact
ORINP2_F(j,f)	Out of range input flag for Cloud or Case 2 IMT or BOAVI	-	clw	2.4, 2.8, 2.9.11	exact
OROUT2_F(j,f)	Out of range output flag for Cloud or Case 2 IMT or BOAVI	-	clw	2.8, 2.9.11	Exact
HIINLD_F(j,f)	Flag for low pressure water	-	w	2.6.5	Exact
ICE_HIGHAERO_F(j,f)	Flag for ice or high aerosol loading pixels	-	w	2.6.5	Exact
MEGLINT_F(j,f)	Flag for pixels with medium glint reflectance	-	w	2.6.5	Exact

Copyright © 2011 ACRI S.T.

Symbol	Description	Unit	Cloud / Land / Water	Origin step no.	Required Accuracy
UNCGLINT_F(j,f)	Flag for pixels non corrected for glint	-	w	2.6.5	Exact
WHITECAPS_F(j,f)	Whitecaps flag	-	w	2.6.5	Exact
CASE2_S(j,f)	Turbid water flag for pixel (j,f)	-	w	2.6.8	Exact
BPAC_ON_F(j,f)	Bright Pixel Atmosphere Correction ON flag	-	w	2.6.8	Exact
ACFAIL_F(j,f)	Flag indicating failure of the atmosphere correction	-	w	2.6.8, 2.6.9	Exact
WHITE_SCATT_F(j,f)	Flag identifying "white" scatter within water	-	w	2.6.8	Exact
ANNOT(j,f)	Annotation flags for the quality of the atmospheric corrections	-	w	2.6.9	exact
RWNEG (b, j, f)	Negative corrected reflectance flag; b: all except b760, b900	-	lw	2.6.9, 2.6.15	exact
CASE2ANOM_F(j,f)	Anomalous scattering water flag for pixel (j,f)	-	w	2.9.6	exact
CASE2Y_F(j,f)	Yellow substance loaded water flag for pixel (j,f)	-	w	2.9.4	exact
DDV_F(j,f)	DDV flag for pixel (j,f)	-	l	2.6.23	exact
ANNOT_BPAC(j,f)	Annotation flags for the quality of the BPAC	-	w	2.6.8.1	Exact

Copyright © 2011 ACRI S.T.

12 - LOW RESOLUTION PRODUCT EXTRACTION

The following section provides a specification of the algorithm to apply to Level 2 Reduced Resolution product pixels, to derive the Low Resolution Cloud and Water Vapour product pixels (see [A-4]).

The total water vapour (resp. cloud optical thickness or cloud top pressure) field of any LR pixel shall be the average of the water vapour field (resp. cloud optical thickness or cloud top pressure) of the corresponding 4x4 pixels in the RR grid. Correspondence is with respect to the product grids described in [A-4].

The following precautions shall be taken when performing the average:

1. Invalid pixels, i.e. those MER_RR_2P product pixels which have their product set to BAD_PRODUCT in the corresponding MDS, should be discarded when computing the 4x4 pixels average, as well for cloud optical thickness, cloud top pressure as for total water vapour.
2. Cloud pixels shall NOT be taken into account in the Water Vapour averaging process, even if the water vapour content is valid, only Land and Water pixels shall be considered.
3. Cloud optical thickness or cloud top pressure are only applicable to cloudy pixels, i.e. those which bear the "cloud" class flag in MDS(20). For other valid pixels, the same field of the MER_RR_2P product stores the aerosol optical thickness. When averaging, non-cloudy RR pixels shall be discarded. It should be noted that this also applies to the extraction of MER_RRC_2P product.
4. image boundaries shall be handled as exceptions, 4x4 pixel blocks not being available near the last frame and last column.
5. spatial averaging cannot be applied to flags. Therefore a different algorithm needs to be applied to the MDS(20) Flags of MER_RR_2P, also to be sub-sampled, as follows:

Copyright © 2011 ACRI S.T.

flag in MER_LRC_2P pixel	Conditions on MER_LRC_2P pixel type flags	Conditions on the flags of the averaged MER_RR_2P pixels
LAND		>50% (1)
CLOUD		>=50% (1)
WATER		>50% (1)
PCD_1_13		never (not relevant)
PCD_14	LAND OR WATER	>=50%
PCD_15	CLOUD	>=50% of the CLOUD pixels
PCD_16		never (not relevant)
PCD_17		never (not relevant)
PCD_18		never (not relevant)
PCD_19	CLOUD	>=50% of the CLOUD pixels
COASTLINE		1 or more
COSMETIC		>=50%
SUSPECT		>=50%
OADB		never (not relevant)
ABSOA_DUST		never (not relevant)
CASE2_S		never (not relevant)
CASE2_ANOM **OR** TOAVI_BRIGHT		never (not relevant)
CASE2_Y **OR** TOAVI_BAD		never (not relevant)
ICE_HAZE **OR** TOAVI_CSI		never (not relevant)
MEDIUM_GLINT **OR** TOAVI_WS		never (not relevant)
DDV OR BPAC_ON		never (not relevant)
HIGH_GLINT **OR** TOAVI_INVAL_REC		never (not relevant)
LOW_SUN		>=50%
LOW_PRESSURE **OR** WHITE_SCATTERER		never (not relevant)

(1) when the MER_LRC_2P pixel is composed of exactly 50% of one pixel type and 50% of another, the following priority rules must apply:

- 50% LAND & 50% WATER = LAND,
- 50% LAND & 50% CLOUD = CLOUD
- 50% WATER & 50% CLOUD = CLOUD

Copyright © 2011 ACRI S.T.

ANNEX A

Parameters Data List

Copyright © 2011 ACRI S.T.

Variable	**Descriptive Name**	**IODD section**	**IODD table**	**Param #**	**Product**	**ADS**	**DPM section**	**Algorithm step**
	Water Vapour over Land Neural Network table	6.12	5	all	Water vapour par.	Neural Network for Water Vapour retrieval over Land	6.5	2.3
	Case 2 neural network table	6.16	8	all	Ocean II par.	Case 2 neural network	8.5	2.9.11
	Cloud top pressure neural network for not null surface albedo	6.17	8	all	Cloud measurement par.	Cloud neural network for not null surface albedo	4	2.1.5
	Cloud top pressure neural network for null surface albedo	6.17	9	all	Cloud measurement par.	Cloud neural network for null surface albedo	4	2.1.5
	Surface Pressure Neural network table	6.17	11	all	Cloud measurement par.	Surface Pressure Neural Network	4.5	2.1b
{A,B}	Coefficients to correct for molecule anisotropy	6.11	4	9	Atmosphere parameters	General	5	2.1
α_15_14_LUT [lat,lon,month]	Surface albedo slope LUT	6.12	9	all	Water vapour par.	Surface Albedo Slope between 900 and 885 nm	6.5	2.3
A_b2_b5(p)	Hyperbola coefficients for Algal Pigment Index retrieval in Case 2 waters for H(443 nm, 560 nm) ; p is the order of the coefficient.	6.16	5	2	CASE 2	Case II yellow substance detection coeffs.	8.5	2.9
A_b3_b5(p)	Hyperbola coefficients for Algal Pigment Index retrieval in Case 2 waters for H(490 nm, 560 nm) ; p is the order of the coefficient.	6.16	5	3	Ocean II par.	Case II yellow substance detection coeffs.	8.5	2.9
A_b4_b5(p)	Hyperbola coefficients for Algal Pigment Index retrieval in Case 2 waters for H(510 nm, 560 nm) ; p is the order of the coefficient.	6.16	5	4	Ocean II par.	Case II yellow substance detection coeffs.	8.5	2.9
α_bad	Bad data value for surface albedo ratio b15/b14	6.12	4	16	Water vapour par.	General		
α_Scatt_Threshold	Threshold on marine basckscatter spectral slope estimate	6.16	4	32	Ocean II par.	General	8.3	2.6.8
a_to_bb_c(b)	specific absorption, case of coccoliths	6.16	4	21	Ocean II par.	General	8.3	2.6.8
a_to_bb_p(b)	specific absorption, case of particulate	6.16	4	22	Ocean II par.	General	8.3	2.6.8
Aerclim_LUT [lat, lon, month, k]	Aerosol climatology above land	6.7	5	all	Aerosol Climatology	Aerosol climatology	9.3	2.6.17
Aerclim_Ocean_LUT[lat,lon, month]	Aerosol climatology above ocean	6.7	7	all	Aerosol Climatology	Aerosol climatology above ocean	8.4	2.6.9

Variable	Descriptive Name	IODD section	IODD table	Param #	Product	ADS	DPM section	Algorithm step
Aermult_LUT [θ_s,θ_v,iaer,s,k]	Polynomial coefficients for each of the Fourier terms used to compute the correcting factor for aerosol multiple scattering	6.14	10	all	Land aerosols par.	Multiplicative function to account for aerosol multiple scattering effects	5	2.1
Aerosol_angstrom [iaer]	Aerosol Angström exponent α corresponding to each of the N aerosol models.	6.14	4	8	Land aerosols par.	General	9.3	2.6.23, 2.6.17
Aerosol_refindex[iaer]	Aerosol refraction index m corresponding to each of the N aerosol models.	6.14	4	8	Land aerosols par.	General	9.3	2.6.23, 2.6.17
Aerosol_wv_LUT [μ_s, μ_v, $\Delta\phi$, k]	LUTs of polynomial coefficients for aerosol correction over water	6.12	10	all	Water vapour par.	Aerosol corrections	6	2.3.2
Aerpha_LUT[cosΘ_{scat}, iaer]	Aerosol phase function times single scattering albedo values as a function of cosine of scattering angle and aerosol model	6.14	11	all	Land aerosols par.	Aerosol phase function times single scattering albedo	9.3	2.6.23, 2.6.17
$a_h(b)$,$b_h(b)$,$c_h(b)$,$d_h(b)$	polynomial coefficients for H2O correction	6.11	6	3	Atmosphere parameters	H2O Transmission	5	2.6.12.3
albedo_g[DDV_model,b]	Ground albedo for a DDV model at a given band	6.14	16	3	Land aerosols par.	DDV parameters for bi-directionality correction	9.3	2.6.23
AnomScatt_LUT[θ_s, θ_v, $\Delta\phi$,Chl]	LUT giving threshold on reflectance at 560 nm for anomalous scattering	6.16	6	all	Ocean II par.	Anomalous Scattering Detection	8.5	2.9.6
aot_offset	offset - Aerosol optical thickn.	6.10	4	24	Level 2 control par.	General	10	2.10
aot_scale	scaling factor - Aerosol optical thickn.	6.10	4	10	Level 2 control par.	General	10	2.10
AOT_P	Default aerosol optical depth	6.17	4	29	Cloud measurement par.	General	4.5	2.1b
APF_Junge_LUT[i]	APF of the Junge aerosol model nb 10	6.11	4	31	Atmosphere parameters	General	5.5	2.1c
$\alpha_{thresh}(b)$	Constant applying to threshold value derived from LUT. Allows to take into account environment and bathymetric effects	6.14	5	1,3	Land aerosols par.	Thresholds for inland waters processing	5	7.3.1
aw(b)	Absorption of pure water	6.16	4	12	Ocean II par.	General	8.3	2.6.8
B(b)	Constants for detection of yellow substance contaminated pixels	6.16	5	1	Ocean II par.	Case II yellow substance detection coeffs.	8.5	2.9
b_bright	Index of band for reflectance test	6.10	7	12	Level 2 control par.	Classification parameters	5	2.1

Variable	**Descriptive Name**	**IODD section**	**IODD table**	**Param #**	**Product**	**ADS**	**DPM section**	**Algorithm step**
b_bright2	Index of band for test on TOA reflectance	6.10	7	10	Level 2 control par.	Classification parameters	5.5	2.1c
b_slope1_d	Index of denominator band for test 1	6.10	7	3	Level 2 control par.	Classification parameters	5	2.1
b_slope1_n	Index of numerator band for test 1	6.10	7	2	Level 2 control par.	Classification parameters	5	2.1
b_slope2_d	Index of denominator band for test 2	6.10	7	7	Level 2 control par.	Classification parameters	5	2.1
b_slope2_n	Index of numerator band for test 2	6.10	7	6	Level 2 control par.	Classification parameters	5	2.1
bb_775_ie(bs)	Initial estimate of backscatter at 775 for LOW and HIGH band estimate	6.16	4	23	Ocean II par.	General	8.3	2.6.8
bbp_init	Initial value of bbp to initialise rhow_bb_routine	6.16	4	28	Ocean II par.	General	8.3	2.6.8
bbp_star_c(b)	Specific back scattering of coccoliths	6.16	4	14	Ocean II par.	General	8.3	2.6.8
bbp_star_p(b)	Specific back scattering of particulates	6.16	4	15	Ocean II par.	General	8.3	2.6.8
bbp_tol	Convergence criteria on bbp in the BPAC iterations	6.16	4	18	Ocean II par.	General	8.3	2.6.8
bbw(b)	Back scattering of pure water	6.16	4	13	Ocean II par.	General	8.3	2.6.8
β_L	Threshold for in-land waters screening spectral slope test	6.14	4	6	Land aerosols par.	General	5	2.6.26
blue_band_N	Blue band index number	6.18	4	1	Land Vegetation Index par.	General	9.2	2.2
boavi_nir1_band	near infrared band #1 for BOAVI	6.18	4	11	Land Vegetation Index par.	General	9.4	2.8
boavi_nir2_band	near infrared band #2 for BOAVI	6.18	4	12	Land Vegetation Index par.	General	9.4	2.8
BOAVI_offset	offset - BOA Vegetation index	6.10	4	30	Level 2 control par.	General	10	2.10
BOAVI_RANGE[0..1]	Range limits for BOAVI	6.18	4	13	Land Vegetation Index par.	General	9.4	2.8
boavi_red_band	red band number for BOAVI	6.18	4	10	Land Vegetation Index par.	General	9.4	2.8
BOAVI_scale	scaling factor - BOA Vegetation index	6.10	4	16	Level 2 control par.	General	10	2.10

Variable	**Descriptive Name**	**IODD section**	**IODD table**	**Param #**	**Product**	**ADS**	**DPM section**	**Algorithm step**
$b_{thresh}(b)$	Indices of bands to be used for comparison with threshold within the island and in-land water screening	6.14	4	5	Land aerosols par.	General	5	2.6.16
β_W	Threshold on spectral slope used in island screening over waters	6.14	4	7	Land aerosols par.	General	5	2.6.26
ca_offset	offset - cloud albedo	6.10	4	31	Level 2 control par.	General	10	2.10
ca_scale	scaling factor - cloud albedo	6.10	4	17	Level 2 control par.	General	10	2.10
Calb_LUT [μ_s, μ_v, $\Delta\phi$, α_{surf},k]	LUTs of polynomial coefficients for estimating cloud albedo as a function of geometry and surface albedo	6.17	6	all	Cloud measurement par.	Polynomial coefficients for cloud albedo retrieval	7	2.4
Chl_epsilon	Convergence criterium for iterative Chl calculation	6.15	7	2	Ocean 1 par.	log10 polynomial coefficients	8.5	2.9.7
Chl_offset	offset - Algal pigment index	6.10	4	20	Level 2 control par.	General	10	2.10
Chl_scale	scaling factor - Algal pigment index	6.10	4	6	Level 2 control par.	General	10	2.10
$Chl1_0$	initial algal pigment index value	6.15	4	13	Ocean 1 par.	General	8.5	2.9.7
$Chl1_{thresh}$rangeout	Chl1 range thresholds for validity of output	6.15	6	2	Ocean 1 par.	Thresholds	8.5	2.9.7
CLIMATO_AUX	Switch to activate the use of a climatology	6.10	6	22	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
Cloud_wv_LUT[μ_s, μ_v, $\Delta\phi$, δ, α, k]	LUTs of polynomial coefficients water vapour retrieval over cloud	6.12	8	all	Water vapour par.	Polynomial coefficients for water vapoure retrieval over clouds	6	2.3
CMOY	Mean value of chlorophyll	6.15	4	21	Ocean 1 par.	General	8.4	2.6.9
Cnorm_LUT[month, lat, lon, band]	DDV reflectance monthly adjustment factors	6.14	18	1	Land aerosols par.	DDV Reflectance Correction Parameters	9.3	2.6.17.3
Const_rho[Class][band]	rho_i normalisation parameters for blue, red, and NIR channels and for Vegetated and Bright soils	6.18	4	6	Land Vegetation Index par.	General	9.2	2.2

Variable	Descriptive Name	IODD section	IODD table	Param #	Product	ADS	DPM section	Algorithm step
cot_offset	offset - Cloud optical thickness	6.10	4	25	Level 2 control par.	General	10	2.10
cot_scale	scaling factor - Cloud optical thickness	6.10	4	11	Level 2 control par.	General	10	2.10
Cthick_LUT [μ_s, μ_v, $\Delta\phi$, α_{surf}, k]	LUTs of polynomial coefficients for estimating cloud optical thickness as a function of geometry and surface albedo	6.17	7	all	Cloud measurement par.	Polynomial coefficients for cloud optical thickness retrieval	7	2.4
ctp_offset	offset - cloud top pressure	6.10	4	32	Level 2 control par.	General	10	2.10
ctp_scale	scaling factor - cloud top pressure	6.10	4	18	Level 2 control par.	General	10	2.10
Ctype_δ_c_range [1..Ctype_n_δ_c]	range of optical thickness values for cloud type classification	6.17	4	21	Cloud measurement par.	General	7	2.4
Ctype_LUT [δ_c,P_{top}]	LUT of cloud type index	6.17	10	all	Cloud measurement par.	Cloud type index	7	2.4
Ctype_n_δ_c	number of optical thickness values for cloud type classification	6.17	4	23	Cloud measurement par.	General	7	2.4
Ctype_n_P	number of pressure values for cloud type classification	6.17	4	22	Cloud measurement par.	General	7	2.4
Ctype_P_range [1..Ctype_n_P]	range of pressure values for cloud type classification	6.17	4	20	Cloud measurement par.	General	7	2.4
ΔARVI$_{max}$_LUT[month, lat, lon]	Maximum acceptable value for ΔARVI	6.14	18	3	Land aerosols par.	DDV Reflectance Correction Parameters	9.3	2.6.17.3
ΔARVI$_{min}$_LUT[month, lat, lon]	Minimum acceptable value for ΔARVI	6.14	18	3	Land aerosols par.	DDV Reflectance Correction Parameters	9.3	2.6.13
DDV_ARVI_LUT[θ_s,θ_v,$\Delta\phi$, DDV_model]	ARVI threshold used for DDV screening	6.14	6	all	Land aerosols par.	ARVI thresholds for DDV models	9.3	2.6.23
DDV_Bands[3]	List of band indices to be used for step 2.6.17-3	6.14	4	20	Land aerosols par.	General	9.3.3	2.6.17
DDV_clim [lat, lon]	Climatological table to select biome according to location	6.14	15	all	Land aerosols par.	DDV climatology	9.3	2.6.23

Variable	Descriptive Name	IODD section	IODD table	Param #	Product	ADS	DPM section	Algorithm step
DDV_LUT [biome, month]	Climatological table to select DDV model according to biome and season	6.14	4	12	Land aerosols par.	General	9.3	2.6.23
DDV_Slope_LUT[month, lat, lon, band]	DDV reflectance linear correction factors	6.14	18	2	Land aerosols par.	DDV Reflectance Correction Parameters	9.3	2.6.17.3
DDV_THR_LUT[θ_s,θ_v,$\Delta\phi$, DDV_model,b]	DDV reflectances at 4 bands as function of geometry and DDV model	6.14	7	all	Land aerosols par.	Standard surface reflectance ranges for DDV models	9.3	2.6.23
DEPTH_LIM	threshold on depth to signal the "shallow water flag"	6.10	6	12	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
DRO510_LIM	Value of $\Delta\rho510$ to set the annotation flag	6.15	4	23	Ocean 1 par.	General	8.4	2.6.9
DRO510_threshold_D	Threshold for absorbing aerosol test at 510 nm	6.10	6	2	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
DRO705_threshold_B	Threshold for blue aerosol test at 510 nm	6.10	6	3	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
$Dsun_0^2$	Square of Sun-Earth distance at reference date	6.10	5	1	Level 2 control par.	Smile Effect Correction	3	2.1.4
$\Delta\tau_a$	Increment in optical thickness used within the iterative procedure	6.14	4	11	Land aerosols par.	General	9.3	2.6.23, 2.6.17
$E^P{}_{CTP}$	Solar flux reference value at b753 consistent with CTP NN	6.17	4	8 (1st element)	Cloud measurement par.	General	4.5	2.1b
$E^P{}_{ratio}$	Solar flux ratio, consistent with CTP NN, to convert reflectance ration into normalised radiance ratio	6.17	4	8 (2nd element)	Cloud measurement par.	General	4.5	2.1b
epsilon_offset	offset - epsilon	6.10	4	23	Level 2 control par.	General	10	2.10
epsilon_scale	scaling factor - epsilon	6.10	4	9	Level 2 control par.	General	10	2.10
f_over_q1_LUT[b,θ_p,θ_s,$\Delta\phi$,Chl,τ_a,W_s]	bidirectional factor f/Q	6.15	8	all	Ocean 1 par.	f/Q factor	8.4, 8.5	2.6.9, 2.9
f0_LUT[b, chl]	f0 factor	6.15	11	all	Ocean 1 par.	f0 factor	8.5	
F_0(b)	Solar irradiance spectrum at reference wavelength, bandwith and date	6.10	5	7	Level 2 control par.	Smile Effect Correction	5	2.1.9

Variable	Descriptive Name	IODD section	IODD table	Param #	Product	ADS	DPM section	Algorithm step
F_0^C(b753)	Solar flux at 753.75nm for cloud LUTs	6.17	4	24	Cloud measurement par.	General	4	2.1
F_0^{FR}(b,k)	Extra-terrestrial Sun irradiance at reference date for all MERIS FR detectors and band	6.10	5	11	Level 2 control par.	Smile Effect Correction	3	2.1.4
F_0^{RR}(b,k)	Extra-terrestrial Sun irradiance at reference date for all MERIS RR detectors and band	6.10	5	9	Level 2 control par.	Smile Effect Correction	3	2.1.4
F_0^{WV}(b)	Solar flux consistent with Cloud LUTs	6.12	4	7	Water vapour par.	General	6.5	2.3
fatab_LUT [ia, b]	Lookup Table for the aerosol forward scattering probability	6.13	8	all	Ocean aerosols par.	Aerosols scattering probability	8.4	2.6.9
Fp_LUT [θs, θv, Δϕ, Ws, coeff, b]	LUT of coefficients of F' to IOPs relation	6.16	7	all	Ocean II par.	Coefficients of F' to IOPs relation	8.3	2.6.8
fresnel_Coeff_LUT[i]	Fresnel coefficients	6.11	4	32	Atmosphere parameters	General	5.5	2.1c
f_{SL}	Correction factor for residual stray-light in band 11	6.17	4	16 (FR) & 17 (RR)	Cloud measurement par.	General	4.5	2.1b
FSURQ_0	Value of F/Q factor at 510 nm for nadir angle	6.15	4	20	Ocean 1 par.	General	8.4	2.6.9
γ	Gamma coefficient used in ARVI computation	6.14	4	10	Land aerosols par.	General	9.3	2.6.23, 2.6.13
gain_vicarious(b)	Vicarious adjustment gains	6.13	4	7	Ocean aerosols par.	General	8.2	2.6.5.5
H_2O_{705}Corr_Poly_LUT[$\Delta\lambda$, k]	Polynomial coefficients for H_2O transmission correction at 709nm (b705)	6.11	6	2	Atmosphere parameters	H2O Transmission	5	2.6.12.3
Hp	pressure scale height	6.11	4	28	Atmosphere parameters	General	3	2.1.0
init_ang	Initial estimate of the Angström exponent	6.16	4	24	Ocean II par.	General	8.3	2.6.8
INV_WV	Threshold value on radiance at 885 nm for marking a pixel as invalid for water vapour processing	6.12	4	4	Water vapour par.	General	6	2.3
InvAbs_Chl2[4]	Conversion factors for Chl2	6.16	4	10	Ocean II par.	GADS General	8.5	2.9.11
InvScat_SPM	Conversion factors for SPM	6.16	4	11	Ocean II par.	GADS General	8.5	2.9.11
K_toavi[Class][band]	K_i normalisation parameters for blue, red, and NIR channels and for Vegetated and Bright soils	6.18	4	4	Land Vegetation Index par.	General	9.2	2.2

Variable	**Descriptive Name**	**IODD section**	**IODD table**	**Param #**	**Product**	**ADS**	**DPM section**	**Algorithm step**
λ(b)	Wavelength for a given band number	5.3.1.4	4	31	Level 1 SPH	SPH	9.3	2.6.23, 2.6.17
L[set][order]	polynomial coefficients used for the polynomial ratios of reflectance rectification and TOAVI	6.18	4	8	Land Vegetation Index par.	General	9.2	2.2
LandRefCorr_b(b,i)	array of pairs of band indices for estimation of reflectance spectral derivative (Land pixels)	6.10	5	4	Level 2 control par.	Smile Effect Correction	7	2.1.6
LandRefCorr_sw(b)	array of per band switches enabling Smile Effect Correction for Land pixels reflectance	6.10	5	2	Level 2 control par.	Smile Effect Correction	6	2.1.6
λ^{C}_{761}(detector)	band 11 wavelength optimised for pressure retrievals	6.17	4	14 (FR) & 15 (RR)	Cloud measurement par.	General	4.5	2.1b
LIST_BLUE	list of blue aerosol models	6.10	6	8	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9.2.5
LISTE_aer_01(5)	List of aerosol models indices	6.10	6	4	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
LISTE_aer_02(5)	List of aerosol models indices	6.10	6	4	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
LISTE_aer_03(5)	List of aerosol models indices	6.10	6	4	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
LISTE_aer_04(5)	List of aerosol models indices	6.10	6	4	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
LISTE_aer_05(5)	List of aerosol models indices	6.10	6	4	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
Ln_min_705	Minimum normalised radiance measurable by MERIS at b705	6.16	4	27	Ocean II par.	General	8.3	2.6.8
log10coeff_LUT [p]	Polynomial coefficients for Algal Pigment Index retrieval in Case 1 waters	6.15	7	1	Ocean 1 par.	log10 polynomial coefficients	8.5	2.9.7
λpix^{FR}(b,k)	Central wavelengths for each MERIS FR detector and band	6.10	5	10	Level 2 control par.	Smile Effect Correction	4, 5	2.1.3, 2.1.5, 2.1.7, 2.6.12, 2.1.6
λpix^{RR}(b,k)	Central wavelengths for each MERIS RR detector and band	6.10	5	8	Level 2 control par.	Smile Effect Correction	4, 5	2.1.3, 2.1.5, 2.1.7, 2.6.12, 2.1.7
$\lambda ref(\Delta\lambda)$	Spectral shift reference wavelength grid	6.11	4	26	Atmosphere parameters	General	4	2.1.3

Variable	**Descriptive Name**	**IODD section**	**IODD table**	**Param #**	**Product**	**ADS**	**DPM section**	**Algorithm step**
$\lambda_{ref}[b,\Delta\lambda]$	Reference wavelengths wavelength grids for correction coefficients of H20 at 709nm and O2 at 779nm	6.11	4	27 for b705, 26 for b761	Atmosphere parameters	General	5	2.6.12.3
λ_{ref}_O2	Reference wavelength values for the Pscatt and P1 LUTs	6.11	4	26	Atmosphere parameters	General	5.5	2.1c
$\lambda_{theo}(b)$	Theoretical wavelengths corresponding to smile corrected reflectances	6.10	5	6	Level 2 control par.	Smile Effect Correction	5, 8.3, 8.4	2.1.6, 2.1.7, 2.6.8, 2.6.9, 2.6.10
MAX_PRESSURE	Maximum acceptable value for pressure	6.11	4	21	Atmosphere parameters	General	4	2.1
Max_rho[3]	Maximum acceptable TOA reflectances (1 coefficient for each blue, infrared and near infrared band)	6.18	4	7	Land Vegetation Index par.	General	9.2	2.2
Max_τ_a	Maximal optical thickness allowed within the iterative procedure	6.10	4	4	Level 2 control par.	General	9.3	2.6.23, 2.6.17
MAX_TAU_AER	maximum value for aerosol optical thickness	6.10	6	11	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
max_TOARb753	Maximum acceptable value for TOAR(b753)	6.17	4	10	Cloud measurement par.	General	4	2.1
max_TOARb761	Maximum acceptable value for TOAR(b761)	6.17	4	12	Cloud measurement par.	General	4	2.1
MDSI_Thresh	Threshold on MDSI	6.10	7	1	Level 2 control par.	Classification parameters	5.5	2.1c
min_R_ratio, max_R_ratio	Irradiance reflectance ratio validity range for Algal_1 computation using log10 polynomial	6.15	7	3	Ocean 1 par.	log10 polynomial coefficients	8.5	2.9.7
min_TOARb753	Minimum acceptable value for TOAR(b753)	6.17	4	9	Cloud measurement par.	General	4	2.1
min_TOARb761	Minimum acceptable value for TOAR(b761)	6.17	4	11	Cloud measurement par.	General	4	2.1
N_aer	Number of aerosol models in database	6.10	6	5	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
N_basic_aer	Number of aerosol models per lists	6.10	6	21	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9

Variable	**Descriptive Name**	**IODD section**	**IODD table**	**Param #**	**Product**	**ADS**	**DPM section**	**Algorithm step**
N_co	Number of coefficients in XCtab	6.10	6	7	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
N_PASSTOT	Number of passes within aerosol database	6.10	6	6	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
N1, N2, N3	Exponents used in step 2.9.4	6.16	5	5	Ocean II par.	Case II yellow substance detection coefficients	8.5	2.9.4
N_{A1}	highest order of coefficients to use in log10coeff_LUT	6.15	7	4	Ocean 1 par.	log10 polynomial coefficients	8.5	2.9.7
N_Fp_coeff	Number of coefficient for fitting the F' function	6.16	4	2	Ocean II par.	General	8.3	2.6.8
nir_band_N	Near infrared band index number	6.18	4	3	Land Vegetation Index par.	General	9.2	2.2
Niter	Number of iterations for Chl1 calculation	6.10	4	1	Level 2 control par.	General	8.5	2.9.7
Niter_rhow_bb	Number of iterations in the rhow_to_bb routine	6.16	4	20	Ocean II par.	General	8.3	2.6.8
NIterBPAC(bs)	Number of iterations in BPAC for band set LOW and band set HIGH	6.16	4	16	Ocean II par.	General	8.3	2.6.8
NN_Log_Switch	Switch enabling the reflectance log scaling at NN input	6.16	8	2	Ocean II par.	Case 2 neural network	8.5	2.9.11
NN_min_rho, NN_min_log_rho	Floor values for NN inputs [reflectance threshold, floor NN input]	6.16	4	31	Ocean II par.	General	8.5	2.9.11
NN_USE_SHIFT	Switch to use spectral shift as NN input	6.17	4	13	Cloud measurement par.	General	4	2.1
ODOC_offset	offset - Yellow substance	6.10	4	21	Level 2 control par.	General	10	2.10
ODOC_scale	scaling factor - Yellow substance	6.10	4	7	Level 2 control par.	General	10	2.10
OUT_MAX	Maximum acceptable output value	6.12	4	6	Water vapour par.	General	6	2.3
OUT_MIN	Minimum acceptable output value	6.12	4	5	Water vapour par.	General	6	2.3
P_1_Thresh	Apparent pressure threshold over Land	6.10	7	16	Level 2 control par.	Classification parameters	5.5	2.1c
par_LUT [angstrom, U_{O3}, $\tau_{aer}(865)$, W_T]	LUT giving PAR	6.11	13	all	Atmosphere parameters	Photosynthetically available radiation	8.5	2.9

Variable	**Descriptive Name**	**IODD section**	**IODD table**	**Param #**	**Product**	**ADS**	**DPM section**	**Algorithm step**
PAR_offset	offset - PAR	6.10	4	28	Level 2 control par.	General	10	2.10
PAR_scale	scaling factor - PAR	6.10	4	14	Level 2 control par.	General	10	2.10
press_offset	offset - Surface pressure	6.10	4	26	Level 2 control par.	General	10	2.10
press_scale	scaling factor - Surface pressure	6.10	4	12	Level 2 control par.	General	10	2.10
PRESS_TOLERANCE	Threshold to activate a correction for pressure	6.10	6	19	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
pressLevel	Reference pressure levels for TO2_Atm_LUT	6.11	4	30	Atmosphere parameters	General	5.5	2.1c
P_{scatt}_Thresh	Apparent pressure threshold over Water	6.10	7	17	Level 2 control par.	Classification parameters	5.5	2.1c
P_{Smin}, P_{Smax}	Validity range for surface pressure NN output	6.17	4	27 & 28	Cloud measurement par.	General	4.5	2.1b
Pstd	Standard pressure	6.11	4	19	Atmosphere parameters	General	5, 8.2, 8.3, 8.4, 9.3	2.1, 2.6.5.1, 2.6.23, 2.6.9, 2.6.10, 2.9
P_{thresh}	Threshold value for low pressure water	6.11	4	16	Atmosphere parameters	General	8.2	7.3.1
θ_{ref}_O2	Reference zenith angle values for the Pscatt and P1 LUTs	6.11	4	29	Atmosphere parameters	General	5.5	2.1c
ρ_diff_min1	Minimum value of the top of aerosol reflectance difference between near infrared #1 and red bands to allow MTCI computation	6.18	4	17	Land Vegetation Index par.	General	9.4	2.8
ρ_diff_min2	Minimum value of the top of aerosol reflectance difference between near infrared #3 and red bands to allow MTCI computation	6.18	4	18	Land Vegetation Index par.	General	9.4	2.8
ρ_nir2_min	Minimum value of top of aerosol reflectance in near infrared band #2 to allow MTCI computation	6.18	4	15	Land Vegetation Index par.	General	9.4	2.8
ρ_nir2_min	Minimum value of top of aerosol reflectance in near infrared band #2 to allow MTCI computation	6.18	4	16	Land Vegetation Index par.	General	9.4	2.8

Variable	**Descriptive Name**	**IODD section**	**IODD table**	**Param #**	**Product**	**ADS**	**DPM section**	**Algorithm step**
ρ_red_max	Maximum value of top of aerosol reflectance in red band to allow MTCI computation	6.18	4	14	Land Vegetation Index par.	General	9.4	2.8
R_{10_12}_thresh	Minimum b10-b12 spectral slope value to consider apparent pressure over water	6.10	7	19	Level 2 control par.	Classification parameters	5.5	2.1c
R_{865}_CS	Reflectance threshold at 865 nm to screen out cloud shadow from DDV pixels	6.14	4	18	Land aerosols par.	General	9.3.3	2.6.13
R_{865}_CS	Reflectance threshold to screen out cloud shadow from DDV pixels	6.14	4	21	Land aerosols par.	General	9.3	2.6.13
Rayalb_LUT[τ]	Rayleigh spherical albedo as a function of optical thickness	6.11	8	all	Atmosphere parameters	Rayleigh spherical albedo	9.3	2.6.23
Rayscatt_coef_LUT [θ_s, θ_v, s, k]	LUT of polynomial coefficients for the 3 Fourier series terms used to compute the correction factor for Rayleigh multiple scattering	6.11	7	all	Atmosphere parameters	Rayleigh scattering function	5	2.1
Rect_rho_offset[b]	Offset for rectified reflectances in infrared and near infrared bands	6.10	4	34,36	Level 2 control par.	GADS General	10.5.9	2.10.10
Rect_rho_scale[b]	Scaling factor for rectified reflectances in red and near infrared bands	6.10	4	33,35	Level 2 control par.	GADS General	10.5.9	2.10.10
red_band_N	Red band index number	6.18	4	2	Land Vegetation Index par.	General	9.2	2.2
reflect_offset	offset - reflectances	6.10	4	19	Level 2 control par.	General	10	2.10
reflect_scale	scaling factor - reflectances	6.10	4	5	Level 2 control par.	General	10	2.10
ρ_{Ground}665_Threshold	665nm ground reflectance threshold for iterative aerosol identification	6.14	4	19	Land aerosols par.	General	9.3.3	2.6.17
r_{ghot}_LUT [θp, Ws]	LUT for the ocean-atmosphere reflection factor	6.15	5	all	Ocean 1 par.	Geometrical factor R	8.3, 8.4.3.13.6	2.6.10, 2.6.9
Rho_rc_LUT[k,θ_s, θ_v, Δϕ]	LUT of thresholds on Rayleigh corrected reflectance at 442nm	6.10	8	all	Level 2 control par.	Reflectance thresholds	5	2.1
rho$_{753}$_thresh	Minimum b10 reflectance value to consider apparent pressure over land	6.10	7	18	Level 2 control par.	Classification parameters	5.5	2.1c
rho$_{TOA}$_thresh	Threshold on TOA reflectance at band b_bright2	6.10	7	11	Level 2 control par.	Classification parameters	5.5	2.1c
rhow_bb_tol	Convergence criteria on bb in the rhow_to_bb routine	6.16	4	19	Ocean II par.	General	8.3	2.6.8

Variable	Descriptive Name	IODD section	IODD table	Param #	Product	ADS	DPM section	Algorithm step
robar_ag_LUT[s, b, iaer, DDV_model, θs, θv]	Look-up table of the five Fourier serie terms for the BRDF aerosol ground coupling term	6.14	16	2	Land aerosols par.	DDV parameters for bi-directionality correction	9.3	2.6.17
robar_ra_LUT[k,iaer, θ]	Look-up table of the four polynomial order terms for the BRDF rayleigh aerosol coupling term	6.14	17	1	Land aerosols par.	Aerosol parameters for bi-directionality correction	9.3	2.6.17
robar_rg_LUT [b, DDV_model, θ]	Look-up table of the BRDF rayleigh ground coupling term	6.14	16	1	Land aerosols par.	DDV parameters for bi-directionality correction	9.3	2.6.17
ROG_LUT[θ_s, θ_v, $\Delta\phi$, W_s, chiw]	LUT containing glint reflectance as a function of geometry, wind speed and direction	6.15	9	all	Ocean 1 par.	Glint reflectance	8.2	2.6.5.1
ROW_510_MEAN_LUT	Climatology giving mean ρ_w at 510nm	6.15	10	1	Ocean 1 par.	Mean ρ_w at 510nm	8.4	2.6.9.2.3
row_510_sigma_LUT	Climatology giving ρ_w variability at 510nm	6.15	10	2	Ocean 1 par.	Mean ρ_w at 510nm	8.4	2.6.9.2.4
row_775_do_both_th	Threshold on rhow at 775 to activate the HIGH bandset	6.16	4	25	Ocean II par.	General	8.3	2.6.8
row_775_do_high_th	Threshold on rhow at 775 to desactivate the LOW bandset	6.16	4	26	Ocean II par.	General	8.3	2.6.8
ROW9_LUT [θ_s,θ_v,$\Delta\phi$]	LUT for threshold value on TOA ρw in channel 9	6.15	6	8	Ocean 1 par.	Thresholds	8.3	2.6.8
$\rho_R$tab_LUT [W_s,b,θ_s, θ_v, $\Delta\phi$]	LUT for the Rayleigh reflectance	6.11	12	all	Atmosphere parameters	Rayleigh reflectance over ocean	8.3	2.6.8
ρ_{thresh}_LUT[b,θ_s, θ_v, $\Delta\phi$]	LUT containing threshold values for island and in-land waters screening	6.14	5	2,4	Land aerosols par.	Thresholds for inland waters processing	5	2.6.11, 2.6.30
$\rho_{w5thresh}$rangein	ρ_w(b5) threshold for controlling validity of input to Chl1 algorithm	6.15	6	1	Ocean 1 par.	Thresholds	8.5	2.9.6
RWNEG_threshold(b)	Reflectance thresholds to set the negative reflectanceflag	6.10	6	1	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
SA_LUT[τ_a,iaer]	Look-up table of aerosol spherical albedo	6.14	8	all	Land aerosols par.	Aerosol spherical albedo	9.3	2.6.23, 2.6.17
SATURATION_L[b]	Default radiance for saturated pixels	6.10	4	3	Level 2 control par.	GADS General	5.5.6, 10.5.6	2.1.7, 2.10.6
Slope_1_high	Upper limit of slope range for test 1	6.10	7	5	Level 2 control par.	Classification parameters	5	2.1
Slope_1_low	Lower limit of slope range for test 1	6.10	7	4	Level 2 control par.	Classification parameters	5	2.1

Variable	Descriptive Name	IODD section	IODD table	Param #	Product	ADS	DPM section	Algorithm step
Slope_2_high	Upper limit of slope range for test 2	6.10	7	9	Level 2 control par.	Classification parameters	5	2.1
Slope_2_low	Lower limit of slope range for test 2	6.10	7	8	Level 2 control par.	Classification parameters	5	2.1
specdep [aer,τ,b]	LUT of the spectral dependence of the aerosol optical thickness for the various aerosol assemblages	6.13	5	all	Ocean aerosols par.	Spectral optical thickness	8.4	2.6.9
SPM_Case2_Thresh	Threshold on SPM concentration to identify sediment dominated waters	6.16	4	17	Ocean II par.	General	8.3	2.6.8
SPM_offset	offset - Suspended particulate matter	6.10	4	22	Level 2 control par.	General	10	2.10
SPM_scale	scaling factor - Suspended particulate matter	6.10	4	8	Level 2 control par.	General	10	2.10
SP_{NN}_min, SP_{NN}_max	Validity ranges for surface pressure NN inputs	6.17	4	25 & 26	Cloud measurement par.	General	4.5	2.1b
STRAT_CORR	Switch to perform stratospheric aerosol correction	6.10	4	2	Level 2 control par.	General	4, 5	2.1.9
Strato_aerpha_LUT [cosΘscat, i_eff_radius, band]	Aerosol phase function times single scattering albedo values as a function of cosine of scattering angle and effective radius index	6.14	14	1	Land aerosols par.	Stratospheric aerosol reflectance parameters	5	2.1.9
Strato_multi [iaer_sa]	Table of index in multiple scattering LUT for strato. aerosol	6.14	4	16	Land aerosols par.	General	5	2.1.9
Strato_rad [iaer_sa]	Table of effective radius index for strato. aerosol	6.14	4	15	Land aerosols par.	General	5	2.1.9
Strato_spectr [i_eff_radius, band]	Table of spectral dependency of optical thickness as a function of stratospheric aerosol effective radius	6.14	14	2	Land aerosols par.	Stratospheric aerosol reflectance parameters	5	2.1.9
Strato_sphalb [iaer_sa, b]	Look-up table of stratospheric aerosol spherical albedo	6.14	12	all	Land aerosols par.	Stratospheric aerosol spherical albedo	5	2.1.9
Strato_tau [iaer_sa]	Table of optical thickness at a reference band for strato. aerosol	6.14	4	17	Land aerosols par.	General	5	2.1.9
Stratospheric_LUT [lat,lon]	LUT of stratospheric aerosol model as a function of latitude, longitude	6.7	7	all	Aerosol Climatology	Stratospheric aerosol	4	2.1.9
Surface_Confidence_Map[lat, lon]	Atlas map for confidence on a priori surface type (land/water) knowledge	6.19	all	all	Surface Confidence Map par.	all	5	2.6.26

Variable	Descriptive Name	IODD section	IODD table	Param #	Product	ADS	DPM section	Algorithm step
Surfalb_b11_LUT [lat,lon,month]	LUT of surface albedo at b761 as a function of latitude, longitude and month of year	6.17	5	all	Cloud measurement par.	Surface Albedo at 761 nm	7.5	2.4
Surfalb_b14_LUT [lat,lon,month]	Surface albedo at b885 LUT	6.12	11	all	Water vapour par.	Surface Albedo at 885 nm	6.5	2.3
t_0,t_1,t_2	Rayleigh transmittance coefficients	6.11	4	1	Atmosphere parameters	General	9.3	2.6.23
τa_bl865 [aer,τ]	Aerosol optical thickness at 865 (boundary layer) for the N_aer aerosol assemblages and N_τ values	6.13	6	all	Ocean aerosols par.	Aerosol optical thickness of the boundary layer	8.4	2.6.9
TA_LUT[θ,τ_a,iaer]	Look-up table of aerosol transmittance	6.14	9	all	Land aerosols par.	Aerosol transmittance	9.3	2.6.23, 2.6.17
TA_Strato_LUT [iaer_sa, band, θ]	Look-up table of stratospheric aerosol transmittance	6.14	13	all	Land aerosols par.	Stratospheric aerosol transmittance	5	2.1.9
TAUA865_threshold	Threshold for the aerosol optical thickness	6.10	6	15	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
t_{down}_LUT[ia, b, μs,τa_865, Ws]	LUT for downward diffuse transmittance	6.13	11	all	Ocean aerosols par.	Downward Transmittance	8.4	2.6.9
TEST_AER	Switch indicating if the tests concerning the possible presence of absorbing aerosols are carried out (switch = TRUE) or not (switch = FALSE)	6.10	6	16	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4	2.6.9
TETAP_ZENITH	Value of θp for nadir viewing	6.15	4	24	Ocean 1 par.	General	8.4	2.6.9
TETAS_limit	Value of the sun zenith angle above which the annotation flag is modified	6.10	6	17	Level 2 control par.	Atmospheric corrections for Case 1 waters	8.4 , 10	2.6.9, 2.10
Theta_toavi[Class][band]	theta_i normalisation parameters for blue, red, and NIR channels and for Vegetated and Bright soils	6.18	4	5	Land Vegetation Index par.	General	9.2	2.2
thres_lowg	Threshold value for low glint	6.15	6	5	Ocean 1 par.	Thresholds	8.2	2.6.5.1
thres_medg	Upper threshold value for ratio between glint and TOA reflectance	6.15	6	6	Ocean 1 par.	Thresholds	8.2	2.6.5.1
Thres_WVhg	Water Vapour high glint threshold	6.15	6	9	Ocean 1 par.	Thresholds	8.2	2.6.5
Thresh_BlueROGC_LUT [θ_s, θ_v, Δϕ]	Bright threshold on Glint corrected reflectance at 412 nm	6.13	9	all	Ocean aerosols par.	Blue ROGC Threshold	8.2	2.6.5.4
Thresh_nir2red_ref	Infrared to near infrared reflectance maximum ratio	6.18	4	9	Land Vegetation Index par.	General	9.2	2.2

Variable	Descriptive Name	IODD section	IODD table	Param #	Product	ADS	DPM section	Algorithm step
TO2_Atm_Aer_LUT[$\lambda,\theta_s,\theta_v$]	O2 aerosol atmospheric transmittance for Ha=2km	6.11	10	2	Atmosphere parameters	Apparent Pressure Parameters	5.5	2.1c
TO2_Atm_LUT[λ_{761}, layer,θs,θv]	O2 atmospheric transmittance LUT	6.11	10	4	Atmosphere parameters	Apparent Pressure Parameters	5.5	2.1c
TO2_Fresnel_LUT[$\lambda,\theta_s,\theta_v$]	O2 Aerosol Fresnel transmittance	6.11	10	3	Atmosphere parameters	Apparent Pressure Parameters	5.5	2.1c
TO2_LUT[$\lambda^{775},L_N^{775},\theta_s,\theta_v,\Delta\phi$]	LUT for O2 correctionat 779	6.11	9	all	Atmosphere parameters	O2 transmission around 779	4	2.1
TO2_Ray_LUT[$\lambda,\theta_s,\theta_v$]	O2 Rayleigh transmittance	6.11	10	1	Atmosphere parameters	Apparent Pressure Parameters	5.5	2.1c
$\tau_{O3\_norm}(b)$	Ozone optical thickness corresponding to 1cm.atm for all bands	6.11	5	5	Atmosphere parameters	Optical thicknesses	5	2.6.12.1
TOAVI _offset	offset - TOA Vegetation index	6.10	4	29	Level 2 control par.	General	10	2.10
TOAVI _scale	scaling factor - TOA Vegetation index	6.10	4	15	Level 2 control par.	General	10	2.10
$\tau_R(b)$	Rayleigh optical thickness at standard pressure for all bands	6.11	5	4	Atmosphere parameters	Optical thicknesses	5, 8.2, 8.3, 8.4,9.3	2.1, 2.6.5.1, 2.6.8, 2.6.10, 2.6.9, 2.6.23, 2.9
tup_LUT [ia, b, μv, τa_865]	LUT for upward diffuse transmittance	6.13	12	all	Ocean aerosols par.	Upward Transmittance	8.4	2.6.9
WAT_REF_IND	Water refraction index	6.15	4	22	Ocean 1 par.	General	8.3, 8.4, 8.5	2.6.10, 2.6.9, 2.9
$\omega_a$tab_LUT [ia, b]	LUT for the aerosol single scattering albedo for the N_aer aerosol assemblages and the N_wl bands	6.13	7	all	Ocean aerosols par.	Aerosol single scattering albedo	8.4	2.6.9
Water_noglint_wv_LUT [μ_s, μ_v, $\Delta\phi$, δ, w, k]	LUTs of polynomial coefficients for water vapour retrieval over water without glint	6.12	6	all	Water vapour par.	Polynomial coefficients for water vapour retrieval over ocean when there is no glint	6	2.3.2
WaterRefCorr_b(b,i)	array of pairs of band indices for estimation of reflectance spectral derivative (Water pixels)	6.10	5	5	Level 2 control par.	Smile Effect Correction	9	2.1.6
WaterRefCorr_sw(b)	array of per band switches enabling Smile Effect Correction for Water pixels reflectance	6.10	5	3	Level 2 control par.	Smile Effect Correction	8	2.1.6

Variable	Descriptive Name	IODD section	IODD table	Param #	Product	ADS	DPM section	Algorithm step
WHITECAP_THR	wind speed threshold for whitecaps	6.15	6	7	Ocean 1 par.	Thresholds	8.2	7.3.1
wv_offset	offset - Water vapour	6.10	4	27	Level 2 control par.	General	10	2.10
wv_scale	scaling factor - Water vapour	6.10	4	13	Level 2 control par.	General	10	2.10
WVNN_INmax	Maximum valid values for Neural Net inputs	6.12	4	18	Water vapour par.	General		
WVNN_INmin	Minimum valid values for Neural Net inputs	6.12	4	17	Water vapour par.	General		
WVNN_OUTmax	Maximum valid value for Neural Net output	6.12	4	20	Water vapour par.	General		
WVNN_OUTmin	Minimum valid value for Neural Net output	6.12	4	19	Water vapour par.	General		
XCtab_LUT [k, W_s, ia, b, θ_s, θ_v, $\Delta\phi$]	LUT for polynomial coefficients linking the ratio ρ_{path}/ρ_R to the aerosol optical thickness	6.13	10	all	Ocean aerosols par.	Coefficients of ($\rho T/\rho R$) to τa relation	8.4	2.6.9
YS_thresh	Threshold for flagging YS dominated waters	6.16	4	29	Ocean II par.	General	8.5	2.9
Z_{max}_INLAND	Altitude threshold above which inland water screening is disabled	6.14	5	5	Land aerosols par.	Thresholds for inland waters processing		