EUMETSAT AC SAF
ATMOSPHERIC COMPOSITION MONITORING

PRODUCT USER MANUAL
IASI reprocessed L2 CO CDR (O3M-517) Metop-A&B

Prepared by:

R. Astoreca, D. Hurtmans, P. Coheur	Université libre de Bruxelles, Belgium
J. Hadji-Lazaro, S. Safieddine, M. George, C. Clerbaux	LATMOS, France
M. Doutriaux-Boucher, C. Vicente, M. Crapeau	EUMETSAT

DOCUMENT STATUS SHEET

Issue	Date	Modified items / Reason for change
1.0	27/10/2022	First version of the IASI CO Level 2 CDR PUM.
1.1	28/02/2023	New version of the IASI CO CDR PUM after discussion with validation teams.
1.2	25/01/2024	New version of the IASI CO CDR PUM after DRR reviewer's comments.
1.3	08/04/2024	Section 4.2: the Python reading routine has been updated.

RELATED PRODUCT LIST

Product ID	Product name
O3M-80	IASI NRT total column CO Metop-A&B

TABLE OF CONTENTS

1. INTRODUCTION 4
1.1 Purpose and scope 4
1.2 Acronyms 4
1.3 Applicable and reference documents 5
2. IASI Reprocessed CO CDR 6
2.1 NetCDF CDR file name convention 6
2.2 CDR file size estimate 6
2.3 Description of the content of the netCDF file 6
3. THE FORLI-CO product 13
3.1 Product description 13
3.2 How to get the FORLI-CO products I need? 13
3.3 Using the product 17
3.4 Accuracy of the product 18
3.5 Validation of the product 18
3.6 Product dissemination and archiving 19
3.7 Product ordering 19
4. Appendices 20
4.1 A priori variance-covariance matrix S_a used in the FORLI-CO algorithm 20
4.2 Reading routines in Python to reconstruct H, S and A 21
4.3 Introduction to EUMETSAT Satellite Application Facility on Atmospheric Composition monitoring (AC SAF) 26
4.4 IASI-FORLI retrieval algorithm 27

1. INTRODUCTION

1.1 Purpose and scope

This document is the Product User Manual for reprocessed IASI Level 2 carbon monoxide climate data record (L2 CO CDR) produced within the context of the Satellite Application Facility on Atmospheric Composition Monitoring (AC SAF) fourth Continuous Development and Operations Phase (CDOP-4). This document gives a brief overview on the IASI CO retrieval algorithm and explains how to use and interpret the reprocessed IASI CO CDR.

The IASI CO CDR (DOI: 10.15770/EUM_SAF_AC_0047) is produced at EUMETSAT using the Metop-A&B NRT IASI CO v20151001.

The reprocessing for Metop A and B covers the following dates:

Start date: IASI-A: 10/07/2007, IASI-B: 20/02/2013

End date: IASI-A: 15/10/2021, IASI-B: 31/12/2022

1.2 Acronyms

AC SAF: Atmospheric Composition Monitoring Satellite Application Facility
BIRA: Belgian Institute for Space Aeronomy
CDOP-4: fourth Continuous Development and Operations Phase
CDR: Climate data record
CNES: Centre National d'Études Spatiales
CP: Partial Column
DOFS: Degrees of Freedom for Signal
FORLI: Fast Optimal Retrievals on Layers for IASI
EUMETSAT: European Organisation for the Exploitation of Meteorological Satellites
GTS: Global Telecommunication System
IASI: Infrared Atmospheric Sounding Interferometer
LATMOS: Laboratoire Atmosphères, Observations Spatiales
NetCDF: Network Common Data Form
NRT: Near Real Time
OEM: Optimal Estimation Method
TOA: Top Of the Atmosphere
ULB: Université Libre de Bruxelles
VMR: Volume Mixing Ratio
WMO: World Meteorological Office

1.3 Applicable and reference documents

1.3.1 Applicable documents

[AD1] FORLI Algorithm Theoretical Basis Document SAF/O3M/ULB/FORLI_ATBD Issue 1, 20/02/2014

[AD2] IASI CO NRT PUM SAF/AC/ULB/PUM/001 Issue 1.5, 08/11/2021

[AD3] Product Requirements Document SAF/AC/FMI/RQ/PRD/001 Issue 1.9.1, 03/02/2022

[AD4] IASI CO CDR L2 validation report SAF/AC/IASI/VR/IASI_CO_CDR_L2 Issue 1, 31/10/2022.

1.3.2 Reference documents

[RD1] Hilton, F.; August, T.; Barnet, C.; Bouchard, A.; Camy-Peyret, C.; Clarisse, L.; Clerbaux, C.; Coheur, P.-F.; Collard, A.; Crevoisier, C.; Dufour, G.; Edwards, D.; Faijan, F.; Fourrié, N.; Gambacorta, A.; Gauguin, S.; Guidard, V.; Hurtmans, D.; Illingworth, S.; Jacquinet-Husson, N.; Kerzenmacher, T.; Klaes, D.; Lavanant, L.; Masiello, G.; Matricardi, M.; McNally, T.; Newman, S.; Pavelin, E.; Péquignot, E.; Phulpin, T.; Remedios, J.; Schlüssel, P.; Serio, C.; Strow, L.; Taylor, J.; Tobin, D.; Uspensky, A. & Zhou, D. Hyperspectral Earth Observation with IASI. *Bull. Am. Meteorol. Soc., 93(3)*, 347-370, doi: 10.1175/BAMS-D-11-00027.1, 2012.

[RD2] Camy-Peyret, C. & Eyre, J. The IASI Science Plan. Technical report, A Report From The IASI Sounding Science Working Group, 1998.

[RD3] Clerbaux, C.; Boynard, A.; Clarisse, L.; George, M.; Hadji-Lazaro, J.; Herbin, H.; Hurtmans, D.; Pommier, M.; Razavi, A.; Turquety, S.; Wespes, C. & Coheur, P. F. Monitoring of atmospheric composition using the thermal infrared IASI/MetOp sounder. *Atmos. Chem. Phys*., 9(16):6041-6054, 2009.

[RD4] Rodgers, C.D. Inverse methods for atmospheric sounding: Theory and Practice, Series on Atmospheric, Oceanic and Planetary Physics - Vol. 2. World Scientific, Singapore, New Jersey, London, Hong Kong, 2000.

[RD5] Hurtmans, D.; Coheur, P.; Wespes, C.; Clarisse, L.; Scharf, O.; Clerbaux, C.; Hadji-Lazaro, J.; George, M. & Turquety, S. FORLI radiative transfer and retrieval code for IASI. *J. Quant. Spectrosc. Radiat. Transfer, 113*, 1391-1408, 2012.

[RD6] EUMETSAT Data Store https://www.eumetsat.int/eumetsat-data-store

2. IASI Reprocessed CO CDR

2.1 NetCDF climate data record (CDR) file name convention

The naming of the IASI Level 2 CO CDR products distributed on EUMETSAT data store follow the convention:

W_XX-EUMETSAT-Darmstadt,HYPERSPECT+SOUNDING,METOPX+CO+IASI_C_EUMP_ DateAndTimeStart _ DateAndTimeEnd _eps_r_l2_version.nc

where:

- W = For all products
- XX-EUMETSAT-Darmstadt = Location Identifier
- HYPERSPECT+SOUNDING = Data designator
- METOPX+CO+IASI = Free description: satellite (with X = A or B)+product+instrument
- C = For all products
- EUMP = Originator code for Operational
- DateAndTimeStart and DateAndTimeEnd are written in the form: YYYYMMDDhhmmssZ

with YYYYMMDD = the UTC year, month and day of the data start and end sensing times.
hhmmss = the UTC hour, minute, second of the data start and end sensing times.

- eps = Native format
- r = reprocessing
- l2 = Level of data
- version = release version

2.2 CDR file size estimate

The size of the CO CDR files is on average 30 MB with ~14 orbits per day per instrument, with a total size of around 450 MB/day/instrument or 164 GB/year/instrument or 328 GB/year for the two instruments

2.3 Description of the content of the netCDF file

The IASI L2 CO CDR netCDF files include FORLI-CO variables, and auxiliary variables (mainly meteorological) used to build atmospheric pressure profiles associated with CO observations. First we give an example of the file's global attributes. Table 1 details dimensions of FORLI-CO and auxiliary variables. In Tables 2 and 4 we list FORLI-CO variables and auxiliary variables, respectively. Table 3 describes the retrieval error flags combined in co_bdiv.

// global attributes:
:creator_name = "EUMETSAT" ;
:creator_url = "http://www.eumetsat.int" ;
:creator_email = "ops@eumetsat.int" ;
:institution = "EUMETSAT" ;
:Conventions = "CF-1.7" ;
:Metadata_Conventions = "Unidata Dataset Discovery v1.0" ;
:data_format_type = "NetCDF-4 classic model" ;
:producer_agency = "EUMETSAT" ;
:platform = "M01" ;
:platform_type = "spacecraft" ;
:sensor = "IASI" ;
:processing_level = "02" ;
:spacecraft_id = "M01" ;
:processor_major_version = "6" ;
:product_minor_version = "5" ;
:format_major_version = "11" ;
:format_minor_version = "0" ;
:orbit_start = "48195" ;
:orbit_end = "48196" ;
:start_orbit_number = 48195 ;
:end_orbit_number = 48196 ;
:orbit_semi_major_axis = 7.204494e+09f ;
:orbit_eccentricity = 0.00121f ;
:orbit_inclination = 98.686f ;
:orbit_perigee_argument = 75.809f ;
:orbit_right_ascension = 63.334f ;
:orbit_mean_anomaly = 284.378f ;
:x_position = 5123897.f ;
:y_position = -5061554.f ;
:z_position = 6550.077f ;
:x_velocity = -1169.753f ;
:y_velocity = -1162.231f ;
:z_velocity = 7355.053f ;
:rev_orbit_period = 6081.7f ;
:equator_crossing_longitude = -5061554.f ;
:summary = "This file contains the Release 1 IASI L2 reprocessed for one orbit. The reprocessing was done on Linux using an adapted version of the EUMETSAT operational algorithm V6.6. Until december 2016, the reprocessed Metop-A IASI L1C input data were used (doi:10.15770/EUM_SEC_CLM_0014), after this date and for Metop-B the operational IASI L1c were used; this is to produce an homogeneous CDR. ERA5 were used as auxiliary model data";
:processing_mode = "R" ;
:start_sensing_data_time = "2022-01-01T00:56:53Z" ;
:end_sensing_data_time = "2022-01-01T02:41:57Z" ;
:start_sensing_data_time_theoretical = "2022-01-01T00:57:00Z" ;
:end_sensing_data_time_theoretical = "2022-01-01T02:42:00Z" ;
:production_date_time_start = "2023-02-22T18:44:32Z" ;
:production_date_time_end = "2023-02-22T18:44:33Z" ;
:equator_crossing_date_time = "2022-01-01T00:29:19Z" ;
:parent_name = "IASI_xxx_1C_M01_20220101005653Z_20220101024157Z_N_O_20220101014546Z" ;
:processing_centre = "EUM" ;
:source = "Metop-B IASI" ;
:platform_long_name = "Metop-B" ;
:project = "This data is provided as part of the EUMETSAT Satellite Application Facility on Atmospheric

Composition Monitoring service, AC-SAF" ;
:id = "DOI: 10.15770/EUM_SAF_AC_0047" ;
:history = "Release 1" ;
:title = "IASI CO CDR Release 1" ;
:title_short_name = "IASI CO CDR" ;
:keywords = "IASI CO CDR Climate" ;

Table 1: Dimensions of FORLI-CO and auxiliary variables

Dimension name	Dimension value	Dimension description
along_track	depends on the files	number of IASI scanlines included in the files
across_track	120	number of IASI pixels per scanline
nlt	101	number of vertical levels in temperature profiles
nlq	101	number of vertical levels in humidity profiles
nl_co	19	number of vertical layers in CO profiles
neva_co	10	number of main eigenvalues of the sensitivity matrix
neve_co	190	number of main eigenvectors of the sensitivity matrix

Table 2: Description of FORLI-CO variables in the netCDF file.

Variable name in CDR CO product	Dimensions	Variable description and unit	Variable name in NRT CO product	Units
co_cp_co_a	along_track, across _track, nl_co	A-priori partial columns for CO on each retrieved layer between CO partial layer heights given in forli_layer_heights_co variable	CO_CP_CO_A	molecules/ cm^2
co_x_co	along_track, across _track, nl_co	Scaling vector multiplying the a-priori CO vector in order to define the retrieved CO vector on each retrieved layer between CO partial layer heights given in forli_layer_heights_co variable	CO_X_CO	N/A

co_nfitlayers	along_track, across _track	Number of layers actually retrieved	CO_NFITLAYERS	N/A
co_nbr_values	along_track	Number of CO profiles retrieved in scanline	CO_NBR	N/A
co_npca	along_track, across _track	Number of vectors describing the characterization matrices	CO_NPCA	N/A
co_h_eigenvalues	along_track, across _track, neva_co	Main eigenvalues of the sensitivity matrix	CO_H_EIGENVALUES	N/A
co_h_eigenvectors	along_track, across _track, neve_co	Main eigenvectors of the sensitivity matrix	CO_H_EIGENVECTORS	N/A
co_qflag	along_track, across _track	General quality flag of air partial columns on each retrieved layer (= 2 for the most reliable pixels; = 1 for the valuable pixels, to use with caution; = 0 for the remaining pixels that we recommend not to use)	CO_QFLAG	N/A
co_bdiv	along_track, across _track	Retrieval flags	CO_BFIV_LO and CO_BDIV_HI	N/A
co_cp_air	along_track, across_track, nl_co	Air partial columns on each retrieved layer between CO partial layer heights given in forli_layer_heights_co variable	CO_CP_AIR	molecules/cm^2
forli_layer_heights_co	nl_co	CO partial layer heights: the bottom height of the layer; 0 is understood as the altitude of the surface and the top of the atmosphere is at 60 km		m

In Table 2, co_qflag is a single code assessing the quality of the retrieved profiles following the retrieval error flags co_bdiv, as described in Table 3 below. co_qflag is a FORLI-CO output, see Section 5.3.1.

Please note that the units of the variables a priori and air partial columns are molecules/cm^2, this is different from the IASI CO NRT product where the units of these variables are moles/cm^2.

Table 3: Potential processing and inputs errors and diagnostics on the retrieval of FORLI-CO (combined in co_bdiv). Note that bits have been reassigned with new WMO descriptor bits.

Name	Value	Bit	WMO bit	Description	Comment
General					
AMP_ERROR	1	0	1	An error has been detected	
Origin					
AMP_L1	2	1	2	Message from L1	
AMP_L2	4	2	3	Message from L2	
AMP_ANC	8	3	4	Message from ancillary data	
AMP_FIT	16	4	5	Message from fitting procedure	
Input content					
AMP_QUALFLAG	256	8	7	Quality flag	Either bad L1 (qFlag) or L2 (F_IASI_Bad) flag raised
AMP_LINREG_L2	512	9	8	Level 2 “from linear regression” (F_Qual), report a pixel where L2 are not fully trusted	
AMP_EMPTY	1024	10	9	Empty field or data	Indicate missing T or humidity level(s) in the vertical profile
AMP_INCOMPLETE	2048	11	10	Missing surface pressure value	
Filtering					
AMP_RADFILTER	4096	12	11	Radiance filtering	Not used in this context
AMP_POLES	8192	13	12	Polar regions	Not used in this context
AMP_NIGHT	16384	14	13	Location in the night	Not used in this context
AMP_NEGZO	32768	15	14	Negative altitude	Surface below m.s.l.
AMP_COVERAGE	65536	16	15	Cloud covered scene	
AMP_SEA	131072	17	16	Scene above the sea	Not used so far
AMP_DESERT	262144	18	17	Scene above desert	
AMP_TSKIN	524288	19	18	Skin temperature	Missing skin temperature, start from BT
AMP_TDIFF	1048576	20	19	Skin temperature differential	Retrieved skin T too different from model
AMP_CONTRAST	2097152	21	20	Spectral line contrast too weak	No lines seen on spectrum (polar regions)
Fitting					
AMP_ITERATIONS	4194304	22	21	Maximum number of iterations exceeded	
AMP_NEGPC	8388608	23	22	Negative partial columns	
AMP_CONDITION	16777216	24	23	Matrix ill conditioned	
AMP_DIVERGED	33554432	25	24	Fit diverged	
AMP_GSL	67108864	26	25	Error in gsl usage	
AMP_BIAS	134217728	27	26	Residuals “biased”	
AMP_SLOPE	268435456	28	27	Residuals “sloped”	
AMP_RMS	536870912	29	28	Residuals rms large	
AMP_AVK	1073741824	30	29	Weird averaging kernels	
AMP_ICE	2147483648	31	30	Ice presence detected	

co_bdiv is built by adding quality indicators associated to each IASI observation and to CO retrieval. As L1C and L2 input data were reprocessed, FORLI-CO results included in CDR netCDF files could be perceptibly different to FORLI-CO results included in NRT BUFR files. These differences can also affect co_bdiv values.

Table 4: Description of auxiliary variables

Variable name in CDR CO product	Dimensions	Variable description and unit
atmospheric_temperature	along_track, across_track, nlt	Atmospheric temperature (for 120 IFOV with up to 101 vertical levels, in K)
atmospheric_water_vapor	along_track, across_track, nlq	Atmospheric water vapour (for 120 IFOV with up to 101 vertical levels, in kg/kg)
fg_atmospheric_temperature	along_track, across_track, nlt	A-priori atmospheric temperature profile (for 120 FOV with up to 101 vertical levels, in K)
fg_atmospheric_water_vapor	along_track, across_track, nlq	A-priori water vapour (for 120 FOV with up to 101 vertical levels, in kg/kg)
fg_surface_temperature	along_track, across_track	A-priori surface temperature (for 120 FOV, in K)
flag_daynit	along_track, across_track	Discrimination between day and night (0: day, 1: night, 2: twilight)
flag_landsea	along_track, across_track	Specifies surface type (0: water, 1: land low, 2: land high, 3: land water low, 4: land water high)
lat	along_track, across_track	Latitude
lon	along_track, across_track	Longitude
pressure_levels_humidity	nlq	Pressure levels on which atmospheric humidity profiles are retrieved (Pa)

pressure_levels_temp	nlt	Pressure levels on which atmospheric temperature profiles are retrieved (Pa)
record_start_time	along_track	Observation time at the start of the scanline (in seconds since 01/01/2000 at 00:00:00)
record_stop_time	along_track	Observation time at the end of the scanline (in seconds since 01/01/2000 at 00:00:00)
satellite_azimuth	along_track, across_track	angular relation for the earth view: satellite azimuth in degrees
satellite_zenith	along_track, across_track	angular relation for the earth view: satellite zenith in degrees
solar_azimuth	along_track, across_track	angular relation for the earth view: solar azimuth in degrees
solar_zenith	along_track, across_track	angular relation for the earth view: solar zenith in degrees
surface_pressure	along_track, across_track	Surface pressure (Pa)
surface_temperature	along_track, across_track	Surface temperature (for 120 IFOV, in K)
surface_z	along_track, across_track	Altitude of surface (m)

Usual FORLI-CO products have to be reconstructed/calculated from the fields mentioned in Table 2, see Section 5.

3. THE FORLI-CO product

3.1 Product description

The product FORLI-CO includes several variables, described above in Table 2. The principal product is a vertical profile of CO provided on 18 layers, from the ground to 18 km with an additional layer from 18 km to top of the atmosphere (TOA). In order to allow a rational use each retrieved profile is associated with averaging kernels and posterior error covariance matrices following the characterization of the optimal estimation. **For saving space, the matrices are compressed and have to be reconstructed using eigenvectors and eigenvalues of the sensitivity matrix** (‘co_h_eigenvalues' and 'co_h_eigenvectors'). Each retrieval has also associated retrieval flags.

Note that the retrievals are performed on the basis of the *a priori* partial columns (co_cp_co_a), which are scaled individually using a *multiplication factor* (co_x_co). The multiplication factor equals therefore 1 at initial stage, is unitless, and should remain close to unity in normal circumstances. This ensures homogeneity of the retrieved values all along the altitudes, even when molecular amounts span several decades. The content of co_cp_co_a is computed using ray tracing methods described in the ATBD [AD1], while co_x_co is retrieved using OEM method in a logarithmic space in order to avoid nonphysical negative values.

3.2 How to get the FORLI-CO products I need?

Table 5: FORLI-CO products that can be obtained from the CO CDR netCDF files.

FORLI-CO products	Notation	How to get/calculate it?
CO profile (molecules.cm^{-2})	co_cp_co	see Section 3.2.1, Eq 1
CO profile (VMR)	co_vmr_co	see Section 3.2.1, Eq 2
CO total column (molecules.cm^{-2})	co_tc	see Section 3.2.1, Eq. 3
A priori profile (molecules.cm^{-2})	co_cp_co_a	Variable in netCDF file.
A priori profile (VMR)	co_vmr_co_a	See Section 3.2.3.2, Eq 10
Averaging Kernel matrix in scaling factor	A	see Section 3.2.2, Eq.6
Averaging Kernel matrix in partial column	A_{PC}	see Section 3.2.3.1, Eq. 9
Averaging Kernel matrix in VMR	A_{VMR}	see Section 3.2.3.2, Eq. 12
Total column averaging kernel vector	k	See Section 3.2.3.3, Eq. 13

Degrees Of Freedom of the Signal	DOFS	trace(A) = trace(A_{PC}) = trace(A_{VMR})
Absolute total retrieval error on the total column	σ_{TC}	See Section 3.2.3.3, Eq. 14
Relative error profile (relative to the retrieved CO profile in molecules.cm^{-2} or VMR)	σ	See Section 3.2.2, Eq. 7
Pressure levels (Pa)	p	See Section 5.2.3.4, Eq. 16 to 18

3.2.1 Reconstruction of the CO profile and calculation of the total column

The final **partial column profile** (molecules.cm^{-2}) is to be reconstructed by multiplying element-wise the two vectors defined earlier e.g.:

$$co\_cp\_co_i = co\_cp\_co\_a_i \times co\_x\_co_i \quad \forall i \tag{1}$$

Profile spans *co_nfitlayers* layers, sampled on a 1 km grid, except the first one which starts from surface altitude and hence could be thinner, and the last one which extends up to TOA.

To convert this profile in VMR:

$$co\_vmr\_co_i = co\_cp\_co_i \div co\_cp\_air_i \quad \forall i \tag{2}$$

The **CO total column** (molecules.cm^{-2}) is obtained by summing the partial columns defined in Eq (1) on all retrieved layers:

$$cp\_tc = \sum_i co\_cp\_co_i \tag{3}$$

The total (or partial columns) can be similarly expressed in moles.cm^{-2} by dividing the values in molecules.cm^{-2} by 6.02214076x10^{+23}, the Avogadro constant.

3.2.2 Reconstruction of the characterisation matrices

Averaging kernel, which is normally an asymmetric matrix (co_nfitlayers × co_nfitlayers), is compressed by using a principal component decomposition representation. A reduced subset of principal vectors (co_npca out of co_nfitlayers) of the sensitivity matrix, *H*, is retained in order to achieve a meaningful compression. Typical compression rates are of about 4 for CO. The averaging kernel matrix, A, is then reconstructed. The posterior variance-covariance matrix is also rebuilt during this procedure.
Reconstruction is done using the following formulation:

$$H = \nu \, \text{diag}(\lambda) \, \nu^T \quad (4)$$
$$\hat{S} = (H + S_a^{-1})^{-1} \quad (5)$$
$$A = \hat{S}H \quad (6)$$

where:

ν is the principal eigenvectors matrix (co_nfitlayers × co_npca);
λ, the principal eigenvalues vector (co_npca);
S_a, the *a priori* variance-covariance matrix;
$\hat{S}$, the posterior variance-covariance matrix;
A, the averaging kernels matrix;
and diag constructs a diagonal matrix the elements of which are given by the parameter vector.

When the surface altitude > 1 km (i.e. co_nfitlayers < 19), users have to be careful and reduce S_a accordingly by decimating the first rows/columns corresponding to the unused altitude layers.

Eigenvectors matrix v is the co_h_eigenvectors linear entry properly reshaped, and eigenvalues vector λ is the co_h_eigenvalues entry.

The *a priori* variance-covariance matrix S_a needed for the reconstruction is provided in Section 4.1. S_a is also provided online on the EUMETSAT website (https://navigator.eumetsat.int/product/EO:EUM:DAT:METOP:IASIL2COX/print, Ressources).

Then the relative error profile can be calculated:

$$\sigma_i = \frac{\sqrt{\hat{S}_{i,i}}}{co\_X\_co_i} \quad \forall i \quad (7)$$

As it is relative to the retrieval, the relative error profile is the same for the retrieved CO profile in molecules.cm^{-2} or VMR. It has therefore not to be recalculated.

A python reading routine that reconstruct H, $\hat{S}$ and A is given in Section 6.2.

3.2.3 Unit conversions

All computations made in Section 3.2.2 were done in the unitless space of the multiplication factor. Users wishing to change the unit space should apply the following conversion rules:

3.2.3.1 Partial columns

Partial column being defined by equation 1, it is easy to demonstrate that:

$$\hat{S}_{PC} = diag(co\_cp\_co\_a) \, \hat{S} \, diag(co\_cp\_co\_a) \quad (8)$$
$$A_{PC} = diag(co\_cp\_co\_a) A \, diag(co\_cp\_co\_a)^{-1} \quad (9)$$

3.2.3.2 Volume mixing ratios

Average volume mixing ratios (VMR) of the layers are computed as the ratios of the partial columns by the corresponding air partial columns. These latter are provided as co_cp_air. Hence conversions are given by:

$$co\_vmr\_co\_a_i = co\_cp\_co\_a_i / co\_cp\_air_i \quad \forall i \tag{10}$$
$$\hat{S}_{VMR} = diag(co\_vmr\_co\_a)\ \hat{S}\ diag(co\_vmr\_co\_a) \tag{11}$$
$$A_{VMR} = diag(co\_vmr\_co\_a) A\ diag(co\_vmr\_co\_a)^{-1} \tag{12}$$

3.2.3.3 Total columns

The total column averaging kernel vector (*k*) is obtained by summing the rows of the averaging kernel matrix *A*:

$$k = (k_1\ \ k_2\ \ \ldots\ \ k_{CO\_NFITLAYERS}),$$
with $k_i = A_{1,i} + A_{2,i} + \ldots + A_{CO\_NFITLAYERS,i}$, i=1 to co_nfitlayers, (13)

The absolute total retrieval error on the total column is then calculated as σ_{TC}

$$\sigma_{TC} = \sqrt{\sum_{i,j} \hat{S}_{PC_{i,j}}} \tag{14}$$

3.2.3.4 Building of vertical pressure profiles associated with FORLI-CO products

Temperature and humidity vertical profiles extracted from the IASI L2 CO CDR product are given on 101 pressure levels (in Pa). To calculate the pressure levels corresponding to the altitude levels of the FORLI-CO retrievals, one should first calculate the altitude levels corresponding to the IASI L2 CO CDR product.

From temperature and humidity vertical profiles, available in the CDR netcdf file, the correspondence between altitude and pressure could be calculated by iterating from the surface to the top of the atmosphere. When “atmospheric_temperature” profile is set to filling values, we replace “atmospheric_temperature” and “atmospheric_water_vapor” profiles by “fg_atmospheric_temperature” and “fg_atmospheric_water_vapor” profiles. The assumptions on the surface characteristics are:

surface altitude = z_0 (“surface_z”, from CDR netCDF files)
surface pressure = p_0 (“surface_pressure” from CDR netCDF files)
surface temperature = T_o, extrapolation of the temperature profile T at the surface pressure (“atmospheric_temperature" or “fg_atmospheric_temperature” from CDR netCDF files)
surface humidity = q_o, first level of the humidity profile q (“atmospheric_water_vapor" or “fg_atmospheric_water_vapor” from CDR netCDF files)

The acceleration due to the gravity is function of the geographic latitude ϕ and of the altitude z_i:

$$g(z_i, \phi) = g_\phi - (3.085462 \times 10^{-6} + 2.27 \times 10^{-9} \cos(2\phi)) z_i + (7.254 \times 10^{-13} + 1.0 \times 10^{-20} \cos(2\phi)) {z_i}^2 - (1.517 \times 10^{-19} + 6 \times 10^{-22} \cos(2\phi)) {z_i}^3 \quad (15)$$

where

$$g_\phi = 9.806160(1 - 0.0026373 \cos(2\phi) + 0.0000059 \cos^2(2\phi)) \text{ ms}^{-2} \quad (16)$$

The mean virtual temperature between two pressure levels p_i and p_{i+1} (just above level i) is then:

$$\overline{Tv_{i,i+1}} = \frac{T_i(1+0.608\, q_i) + T_{i+1}(1+0.608\, q_{i+1})}{2} \quad (17)$$

with T_i and q_i, the temperature and humidity at p_i, respectively, and T_{i+1} and q_{i+1}, the temperature and humidity at p_{i+1}, respectively.

Then the altitude of the pressure level p_{i+1} can be estimated from the pressure level p_i (just below level i+1):

$$z_{i+1} = z_i + \frac{R \times \overline{Tv_{i,i+1}}}{g(z_i, \phi)} \times \ln \frac{p_i}{p_{i+1}} \quad (18)$$

with R = 287.06 $JK^{-1}kg^{-1}$, the gas constant for dry air

We obtain the altitude profile corresponding to the CDR "(fg_)atmospheric_temperature" and "(fg_)atmospheric_water_vapor" products.
Then we can extract the pressure levels associated to FORLI-CO product from the pressure vertical profile by using a cubic spline interpolation.

The conversion between pressure and altitude is done as in the "IASI Level2 Product Generation specification" document.

3.3 Using the product

3.3.1 Quality Flags for the retrieved profile

co_qflag is a quality assessment associated with the quality of FORLI retrieved CO total column. co_qflag is a FORLI-CO output. It can be 2 (best quality), 1 (acceptable quality) or 0 (the rest).

co_qflag=2 for the most reliable pixels, in other words the best quality pixels, *i.e.* when:

- DOFS > 0.5376,
- CO total column < 20 x 10^{18} molecules.cm^{-2},
- the flag AMP_NEGPC (negative retrieval for H_2O) is null
- **1.** flags AMP_NEGZ0, AMP_TSKIN, AMP_TDIFF, AMP_DESERT, AMP_ITERATIONS, AMP_SLOPE, AMP_CONTRAST, AMP_AVK, AMP_BIAS and AMP_RMS are null

  or

2. total cloud cover ≤12% and flags AMP_NEGZ0, AMP_TDIFF, AMP_DESERT, AMP_ITERATIONS, AMP_SLOPE, AMP_CONTRAST, AMP_AVK, AMP_BIAS and AMP_RMS are null.

co_qflag=1 for the valuable pixels, to use with caution, *.i.e.* when:

- DOFS > 0.5376,
- CO total column < 20 x 10^{18} molecules.cm^{-2},

co_qflag=0 for the remaining pixels. We recommend not using these pixels.

For data validation or assimilation purposes, we recommend using the data with co_qflag equal to 2. For specific studies, if more pixels are needed, co_qflag equal to 1 can be used but analysis must consider the non-optimal quality of these pixels. Depending on the use case (with or without AVK) a user can refine the quality filtering, for example under high latitude conditions.

In addition to the previous filtering, users should avoid using the CO CDR data for which:

- nfitlayers is -1 or
- co_x_co vertical scaling profile above surface altitude is constant or
- co_x_co vertical scaling profile above surface altitude includes values higher than 6.5e+17 or
- co_x_co vertical scaling profile above surface altitude includes NaN, Inf or 0 besides to the entries filled with the _FillValue or missing_value attribute.

3.4 Accuracy of the product

The accuracy of the product is given in terms of threshold, target and optimal values in Table 6 below. This information is taken from the Product Requirements Document [AD3]. The product performance is given in the Validation Report of this product.

Table 6: Accuracy of the FORLI-CO product.

		Accuracy (%)			Spatial resolution	Spatial coverage	Cloud fraction
		Threshold	Target	Optimal			
Total column	*Standard*	25	12	5	IASI pixel	Global	< 25 %
	Unusual	50	20	10			

3.5 Validation of the product

The validation of the IASI CO CDR product is performed at BIRA. The 'IASI CDR CO validation report' [AD4] gives verification work performed using ground-based CO profile measurements from the NDACC network.

3.6 Product dissemination and archiving

The IASI CDR Level 2 products are disseminated to users through EUMETSAT Data Store [RD6]. The data are disseminated in netCDF format. A full description of the IASI CO CDR Level 2 netCDF content is given in Section 4.

3.7 Product ordering

Access to this data record is granted to all users without charge but accepting the EUMETSAT Data Policy provided in the corresponding EUMETSAT webpage:
https://www.eumetsat.int/legal-framework/data-policy
To access data, you need to register with the EUMETSAT Data Centre. When registered, you can order the data through a written request send to EUMETSAT's helpdesk.

3.7.1 *Register with the Data Centre*

Do this to register with the EUMETSAT Data Centre:
- Register in the EUMETSAT EO-Portal (https://eoportal.eumetsat.int) by clicking on the New User – Create New Account tab;
- After finalisation of the registration process, an e-mail is sent to the e-mail address entered in the registration. Click the confirmation link in the e-mail to activate your account;
- Login and subscribe to the Data Centre Service by going to the Service Subscription Tab and selecting Data Centre Service. Follow instructions issued from the web page to add needed information.

3.7.2 *Order Data*

The data record described in this product user guide can also be ordered via the EUMETSAT User Service Helpdesk in Darmstadt, Germany. Please send a written request to the helpdesk, email ops@eumetsat.int, indicating the data record that you want to order including its Digital Object Identifier (DOI) number: 10.15770/EUM_SAF_AC_0047.

4. Appendices

4.1 A priori variance-covariance matrix S_a used in the FORLI-CO algorithm

```
Sa['CO']=matrix([[+3.9650531E-01,+2.5280535E-01,+1.9817827E-01,+1.6509750E-
01,+1.4575962E-01,+1.3394338E-01,+1.2549382E-01,+1.1861376E-01,+1.0993634E-
01,+9.8773297E-02,+8.5791044E-02,+7.0830532E-02,+5.9056920E-02,+4.6892112E-
02,+3.6040621E-02,+2.7196075E-02,+2.0282477E-02,+1.6480811E-02,+1.2681867E-02],

[+2.5280535E-01,+2.1312735E-01,+1.7760667E-01,+1.5098094E-01,+1.3355103E-
01,+1.2313758E-01,+1.1514362E-01,+1.0878214E-01,+1.0160108E-01,+9.2449041E-
02,+8.1896901E-02,+6.8933915E-02,+5.7637855E-02,+4.5622891E-02,+3.5272658E-
02,+2.7180976E-02,+2.0831196E-02,+1.7531716E-02,+1.4114632E-02],

[+1.9817827E-01,+1.7760667E-01,+1.6324326E-01,+1.4154247E-01,+1.2598168E-
01,+1.1645555E-01,+1.0885072E-01,+1.0253117E-01,+9.5641950E-02,+8.7176116E-
02,+7.7261109E-02,+6.4852199E-02,+5.3756843E-02,+4.2089189E-02,+3.2444706E-
02,+2.5153707E-02,+1.9418228E-02,+1.6626404E-02,+1.3696698E-02],

[+1.6509750E-01,+1.5098094E-01,+1.4154247E-01,+1.3055180E-01,+1.1882531E-
01,+1.1036372E-01,+1.0351414E-01,+9.7197261E-02,+9.0595067E-02,+8.2420077E-
02,+7.3115010E-02,+6.1147391E-02,+5.0348259E-02,+3.9108593E-02,+2.9978290E-
02,+2.3214689E-02,+1.7877493E-02,+1.5401913E-02,+1.2795437E-02],

[+1.4575962E-01,+1.3355103E-01,+1.2598168E-01,+1.1882531E-01,+1.1245899E-
01,+1.0530277E-01,+9.9349451E-02,+9.3305760E-02,+8.6987187E-02,+7.9124308E-
02,+7.0364465E-02,+5.8925228E-02,+4.8502213E-02,+3.7738231E-02,+2.8997367E-
02,+2.2558374E-02,+1.7419529E-02,+1.5036864E-02,+1.2515015E-02],

[+1.3394338E-01,+1.2313758E-01,+1.1645555E-01,+1.1036372E-01,+1.0530277E-
01,+1.2375655E-01,+9.9453555E-02,+9.4562261E-02,+8.7151744E-02,+7.8447455E-
02,+6.9426830E-02,+5.8140592E-02,+4.8070988E-02,+3.7877306E-02,+2.9504837E-
02,+2.3012359E-02,+1.7715941E-02,+1.5061397E-02,+1.2219129E-02],

[+1.2549382E-01,+1.1514362E-01,+1.0885072E-01,+1.0351414E-01,+9.9349451E-
02,+9.9453555E-02,+1.0267673E-01,+9.4113078E-02,+8.8361885E-02,+8.0365382E-
02,+7.1873010E-02,+6.1172664E-02,+5.1596352E-02,+4.2155666E-02,+3.4158262E-
02,+2.7668672E-02,+2.2077410E-02,+1.8895864E-02,+1.5391697E-02],

[+1.1861376E-01,+1.0878214E-01,+1.0253117E-01,+9.7197261E-02,+9.3305760E-
02,+9.4562261E-02,+9.4113078E-02,+9.7282060E-02,+8.9883701E-02,+8.3131170E-
02,+7.5309408E-02,+6.5400956E-02,+5.6697692E-02,+4.8167228E-02,+4.0518766E-
02,+3.3880813E-02,+2.7860870E-02,+2.3913686E-02,+1.9532473E-02],

[+1.0993634E-01,+1.0160108E-01,+9.5641950E-02,+9.0595067E-02,+8.6987187E-
02,+8.7151744E-02,+8.8361885E-02,+8.9883701E-02,+9.4594165E-02,+8.9762907E-
02,+8.4797569E-02,+7.6342695E-02,+6.8837247E-02,+6.1585573E-02,+5.4380604E-
02,+4.7603575E-02,+4.1043694E-02,+3.6081571E-02,+3.0268036E-02],

[+9.8773297E-02,+9.2449041E-02,+8.7176116E-02,+8.2420077E-02,+7.9124308E-
02,+7.8447455E-02,+8.0365382E-02,+8.3131170E-02,+8.9762907E-02,+9.7114357E-
02,+9.5932433E-02,+9.1083251E-02,+8.5912764E-02,+8.0501089E-02,+7.3983506E-
02,+6.7178677E-02,+5.9944514E-02,+5.3498915E-02,+4.5703244E-02],

[+8.5791044E-02,+8.1896901E-02,+7.7261109E-02,+7.3115010E-02,+7.0364465E-
02,+6.9426830E-02,+7.1873010E-02,+7.5309408E-02,+8.4797569E-02,+9.5932433E-
02,+1.0800795E-01,+1.0777174E-01,+1.0584537E-01,+1.0307407E-01,+9.7606563E-
02,+9.0974472E-02,+8.2977977E-02,+7.4879065E-02,+6.4795076E-02],
```

```
[+7.0830532E-02,+6.8933915E-02,+6.4852199E-02,+6.1147391E-02,+5.8925228E-
02,+5.8140592E-02,+6.1172664E-02,+6.5400956E-02,+7.6342695E-02,+9.1083251E-
02,+1.0777174E-01,+1.2298416E-01,+1.2690542E-01,+1.2939287E-01,+1.2645510E-
01,+1.2039518E-01,+1.1180304E-01,+1.0143799E-01,+8.8351673E-02],

[+5.9056920E-02,+5.7637855E-02,+5.3756843E-02,+5.0348259E-02,+4.8502213E-
02,+4.8070988E-02,+5.1596352E-02,+5.6697692E-02,+6.8837247E-02,+8.5912764E-
02,+1.0584537E-01,+1.2690542E-01,+1.4342173E-01,+1.5204045E-01,+1.5290874E-
01,+1.4846492E-01,+1.3973979E-01,+1.2720509E-01,+1.1121698E-01],

[+4.6892112E-02,+4.5622891E-02,+4.2089189E-02,+3.9108593E-02,+3.7738231E-
02,+3.7877306E-02,+4.2155666E-02,+4.8167228E-02,+6.1585573E-02,+8.0501089E-
02,+1.0307407E-01,+1.2939287E-01,+1.5204045E-01,+1.7100608E-01,+1.7663051E-
01,+1.7495438E-01,+1.6700804E-01,+1.5236096E-01,+1.3359072E-01],

[+3.6040621E-02,+3.5272658E-02,+3.2444706E-02,+2.9978290E-02,+2.8997367E-
02,+2.9504837E-02,+3.4158262E-02,+4.0518766E-02,+5.4380604E-02,+7.3983506E-
02,+9.7606563E-02,+1.2645510E-01,+1.5290874E-01,+1.7663051E-01,+1.8878089E-
01,+1.9045242E-01,+1.8416965E-01,+1.6861073E-01,+1.4847936E-01],

[+2.7196075E-02,+2.7180976E-02,+2.5153707E-02,+2.3214689E-02,+2.2558374E-
02,+2.3012359E-02,+2.7668672E-02,+3.3880813E-02,+4.7603575E-02,+6.7178677E-
02,+9.0974472E-02,+1.2039518E-01,+1.4846492E-01,+1.7495438E-01,+1.9045242E-
01,+1.9869960E-01,+1.9552225E-01,+1.8040823E-01,+1.6050618E-01],

[+2.0282477E-02,+2.0831196E-02,+1.9418228E-02,+1.7877493E-02,+1.7419529E-
02,+1.7715941E-02,+2.2077410E-02,+2.7860870E-02,+4.1043694E-02,+5.9944514E-
02,+8.2977977E-02,+1.1180304E-01,+1.3973979E-01,+1.6700804E-01,+1.8416965E-
01,+1.9552225E-01,+1.9637976E-01,+1.8289759E-01,+1.6484417E-01],

[+1.6480811E-02,+1.7531716E-02,+1.6626404E-02,+1.5401913E-02,+1.5036864E-
02,+1.5061397E-02,+1.8895864E-02,+2.3913686E-02,+3.6081571E-02,+5.3498915E-
02,+7.4879065E-02,+1.0143799E-01,+1.2720509E-01,+1.5236096E-01,+1.6861073E-
01,+1.8040823E-01,+1.8289759E-01,+1.7365334E-01,+1.5951999E-01],

[+1.2681867E-02,+1.4114632E-02,+1.3696698E-02,+1.2795437E-02,+1.2515015E-
02,+1.2219129E-02,+1.5391697E-02,+1.9532473E-02,+3.0268036E-02,+4.5703244E-
02,+6.4795076E-02,+8.8351673E-02,+1.1121698E-01,+1.3359072E-01,+1.4847936E-
01,+1.6050618E-01,+1.6484417E-01,+1.5951999E-01,+1.5063304E-01]])
```

4.2 Reading routines in Python to reconstruct H, S and A

```
--------------------------------------------------------------------------------
################################################################################
#         Reading routine in Python to reconstruct H, S and A         #
################################################################################

"""
From one single L2 CO CDR netcdf file, this script reads variables, deletes observations
with latitude and longitude out of normal range, and, for valuable observations,
reconstructs matrices and estimates retrieval errors.

Usage : python -Wignore reconstruct_matrices_PUM.py

"""
```

```
# metadata information
__date__ = "29 March 2024"
__author__ = "J. Hadji-Lazaro"
__copyright__ = "Copyright 2024, LATMOS"
__organization__ = "LATMOS/CNRS/UVSQ/SU"
__license__ = "GPL"
__maintainer__ = "J. Hadji-Lazaro"
__email__ = "juliette.hadji-lazaro@latmos.ipsl.fr"

import sys
import numpy as np
import pandas as pd
from netCDF4 import Dataset
import csv

# read a priori variance-covariance matrix, provided in section 6.1
Sa_CO_init=pd.read_csv('Sa_CO',header=None,index_col=False)
Sa_CO=Sa_CO_init.values
del(Sa_CO_init)

# read one single L2 CO CDR netcdf file and delete observations for which latitudes and longitudes do not
have valid values
netcdf_inputfile='PATH_TO_CO_CDR_FILES/W_XX-EUMETSAT-
Darmstadt,HYPERSPECT+SOUNDING,METOPX+CO+IASI_C_EUMY_YYYYMMDDhhmmssZ_YYY
YMMDDhhmmssZ_eps_r_l2_0100.nc'
print(netcdf_inputfile)
data = Dataset(netcdf_inputfile, 'r', format='NETCDF4')
along_track=data.dimensions['along_track'].size
print(along_track)
across_track=data.dimensions['across_track'].size
print(across_track)
nlco=data.dimensions['nl_co'].size
print(nlco)
nevaco=data.dimensions['neva_co'].size
print(nevaco)
lat=data.variables['lat']
size_lat=list(np.shape(lat))
lat=np.reshape(lat,size_lat[0]*size_lat[1])
lon=data.variables['lon']
lon=np.reshape(lon,size_lat[0]*size_lat[1])
lon=np.delete(lon,np.where(abs(lat)>90.))
gqf=data.variables['co_qflag']
gqf=np.reshape(gqf,size_lat[0]*size_lat[1])
gqf=np.delete(gqf,np.where(abs(lat)>90.))
prior=data.variables['co_cp_co_a']
prior=np.reshape(prior,(size_lat[0]*size_lat[1],nlco))
prior=np.delete(prior,np.where(abs(lat)>90.),axis=0)
air=data.variables['co_cp_air']
air=np.reshape(air,(size_lat[0]*size_lat[1],nlco))
air=np.delete(air,np.where(abs(lat)>90.),axis=0)
scalfac=data.variables['co_x_co']
scalfac=np.reshape(scalfac,(size_lat[0]*size_lat[1],nlco))
scalfac=np.delete(scalfac,np.where(abs(lat)>90.),axis=0)
```

```
nfitlay=data.variables['co_nfitlayers']
nfitlay=np.reshape(nfitlay,size_lat[0]*size_lat[1])
nfitlay=np.delete(nfitlay,np.where(abs(lat)>90.))
npca=data.variables['co_npca']
npca=np.reshape(npca,size_lat[0]*size_lat[1])
npca=np.delete(npca,np.where(abs(lat)>90.))
eve=data.variables['co_h_eigenvectors']
eve=np.reshape(eve,(size_lat[0]*size_lat[1],nevaco*nlco))
eve=np.delete(eve,np.where(abs(lat)>90.),axis=0)
eva=data.variables['co_h_eigenvalues']
eva=np.reshape(eva,(size_lat[0]*size_lat[1],nevaco))
eva=np.delete(eva,np.where(abs(lat)>90.),axis=0)
lat=np.delete(lat,np.where(abs(lat)>90.))
data.close()

# for each observation, calculation of the ratio between the minimum and maximum values of the scale
factor profile (scalfac) above sea surface level
scaltemp=np.tile(np.nan,(len(lat),19))
for i in range(len(lat)) :
    if nfitlay[i]>-1 :
        scaltemp[i,0+(19-nfitlay[i]):19]=scalfac[i,0+(19-nfitlay[i]):19]
scalmin=np.nanmin(scaltemp,1) # NaN if all elements of scaltemp are NaN
scalmax=np.nanmax(scaltemp,1) # NaN if all elements of scaltemp are NaN
ratio=np.divide(scalmax,scalmin) # NaN if scalmin and/or scalmax is NaN

ratio_nan=np.isnan(ratio)
ratio_inf=np.isinf(ratio)
# detection of NaN in scalfac
nan_in_scal_day=np.isnan(scalfac)
depist_nan=np.tile(0,(len(lat)))
depist_nan=np.where(nan_in_scal_day.sum(axis=1) > 0, 1, depist_nan)
# detection of Inf in scalfac
inf_in_scal_day=np.isinf(scalfac)
depist_inf=np.tile(0,(len(lat)))
depist_inf=np.where(inf_in_scal_day.sum(axis=1) > 0, 1, depist_inf)
# detection of zeros in scalfac
zeros_in_scalfac=np.tile(0,(len(lat),19))
zeros_in_scalfac[scalfac==0]=1
depist_zeros_scalfac=np.tile(0,(len(lat)))
depist_zeros_scalfac=np.where(zeros_in_scalfac.sum(axis=1) > 0, 1, depist_zeros_scalfac)
# detection of values between 650000 and 660000 in scalfac
outliers_650k_in_scal_day=np.tile(0,(len(lat),19))
outliers_650k_in_scal_day[(scalfac>650000.)&(scalfac<660000.)]=1
depist_outliers_650k=np.tile(0,(len(lat)))
depist_outliers_650k=np.where(outliers_650k_in_scal_day.sum(axis=1) > 0, 1, depist_outliers_650k)
# detection of filling values above surface level in scalfac (values higher than 9.96e36 in scaltemp, scalfac
above sea surface level)
outliers_in_scal_day=np.tile(0,(len(lat),19))
outliers_in_scal_day[scaltemp>9.96e36]=1
depist_outliers=np.tile(0,(len(lat)))
depist_outliers=np.where(outliers_in_scal_day.sum(axis=1) > 0, 1, depist_outliers)
# detection of scalfac profiles lower than 1e-5
depist_profils_inf1=np.tile(0,(len(lat)))
depist_profils_inf1=np.where(scalmin <= 1.e-5, 1, depist_profils_inf1)
```

```
# detection of zeros in prior
zeros_ds_prior=np.tile(0,(len(lat),19))
zeros_ds_prior[prior==0]=1
depist_zeros_prior=np.tile(0,(len(lat)))
depist_zeros_prior=np.where(zeros_ds_prior.sum(axis=1) > 0, 1, depist_zeros_prior)

# conversion in moles/m2 and replacement of filling values by NaN
prior=np.where(prior > 65535., prior/6.02214086e+19, np.nan)
air=np.where(air > 65535., air/6.02214086e+19, np.nan)
scalfac=np.where(scalfac < 9.96e+36, scalfac, np.nan)
fit=prior*scalfac
coltot=np.nansum(fit,axis=1)

# detection of prior profiles including less "not NaN" values than the number of layers actually retrieved
(nfitlay)
prior_aberrant=np.tile(0,(len(lat),19))
prior_aberrant[~np.isnan(prior)]=1
depist_prior_aberrant=np.tile(0,(len(lat)))
depist_prior_aberrant=np.where((nfitlay > -1) & (prior_aberrant.sum(axis=1) < nfitlay), 1,
depist_prior_aberrant)

print('Errors estimation and matrices reconstruction')
#   initialisation of errors, averaging kernel matrix and dofs
fit_out=np.tile(np.nan,(len(lat),19))
coltot_out=np.tile(np.nan,(len(lat)))
ERROR_PROF=np.tile(np.nan,(len(lat),19))
ERROR_COLTOT=np.tile(np.nan,(len(lat)))
AVK=np.tile(np.nan,(len(lat),19,19))
dofs=np.tile(np.nan,(len(lat)))
for i in range(len(lat)) :
    i=int(i)
# tests to avoid processing outliers
    if nfitlay[i] > -1 and gqf[i] > -1 and ratio[i] != 1 and ratio_nan[i] == 0 and ratio_inf[i] == 0 and
depist_nan[i] == 0 and depist_inf[i] == 0 and depist_zeros_scalfac[i] == 0 and depist_zeros_prior[i] == 0
and depist_outliers_650k[i] == 0 and depist_outliers[i] == 0 and depist_prior_aberrant[i] == 0 and
depist_profils_inf1[i] == 0 and np.sum(eva[i,range(npca[i])]) == npca[i]:
        fit_out[i,:]=fit[i,:]
        coltot_out[i]=coltot[i]
#   matrices reconstruction + errors estimation
        tmp_eve=eve.take(i,axis=0)
        tmp_eve2=tmp_eve.take(range(0,int(nfitlay[i])*int(npca[i])))
        transp_nu=np.reshape(tmp_eve2,(int(npca[i]),int(nfitlay[i])))
        del(tmp_eve)
        del(tmp_eve2)
        nu=np.transpose(transp_nu)
        tmp_eva=eva.take(i,axis=0)
        tmp_eva2=tmp_eva.take(range(0,int(npca[i])))
        dia_lambda=np.diag(tmp_eva2)
        del(tmp_eva)
        del(tmp_eva2)
        H1=nu.dot(dia_lambda)
        H=H1.dot(transp_nu)
        del(H1)
        Sa=Sa_CO.take(range(19-int(nfitlay[i]),19),axis=1)
```

```
        Sa=Sa.take(range(19-int(nfitlay[i]),19),axis=0)
        inv_Sa=np.linalg.inv(Sa)
        S=np.linalg.inv(H+inv_Sa)
        ERROR_PROF1=np.divide(np.sqrt(np.diag(S)),scalfac[i,range(19-int(nfitlay[i]),19)])
        ERROR_PROF[i,range(19-int(nfitlay[i]),19)]=ERROR_PROF1
        S_CP1=np.diag(prior[i,range(19-int(nfitlay[i]),19)]).dot(S)
        S_CP=S_CP1.dot(np.diag(prior[i,range(19-int(nfitlay[i]),19)]))
        del(S_CP1)
        ERROR_COLTOT[i]=np.sqrt(np.sum(np.sum(S_CP)))/np.sum(fit[i,range(19-int(nfitlay[i]),19)])
        del(S_CP)
        AVK_SF=S.dot(H)
        del(H)
        del(S)
        AVK_CP1=np.diag(prior[i,range(19-int(nfitlay[i]),19)]).dot(AVK_SF)
        AVK_CP=AVK_CP1.dot(np.linalg.inv(np.diag(prior[i,range(19-int(nfitlay[i]),19)])))
        del(AVK_CP1)
        dofs[i]=np.sum(np.diag(AVK_CP))
        AVK[i,19-int(nfitlay[i,]):,19-int(nfitlay[i]):]=AVK_CP
        del(AVK_CP)

print('Writing in output files')

mat_zip=np.hstack((lat[:,None],lon[:,None],gqf[:,None],nfitlay[:,None],6.02214086E+19*fit_out[:,:],ERROR_PROF[:,:],6.02214086E+19*coltot_out[:,None],ERROR_COLTOT[:,None],dofs[:,None]))

outputfile="profiles_and_column.out"
output_id=open(outputfile,'w')
output_writer = csv.writer(output_id,delimiter=' ')
output_writer.writerows(mat_zip)
output_id.close()

AVK_zip=np.vstack(AVK[:,:,:])
AVKfile="AVK.out"
AVK_id=open(AVKfile,'w')
AVK_writer = csv.writer(AVK_id,delimiter=' ')
AVK_writer.writerows(AVK_zip)
AVK_id.close()
```

The script source (reconstruct_matrices_PUM.py) and examples of output files are available at the address https://owncloud.latmos.ipsl.fr/index.php/s/25pNKWFfCJ5Fwwb. The *_outliers.out files are output files associated with W_XX-EUMETSAT-Darmstadt,HYPERSPECT+SOUNDING,METOPB+CO+IASI_C_EUMP_20220222121200Z_20220222130559Z_eps_r_l2_0100.nc CO CDR input file which includes only 63 valuable observations on 24120 and many outliers. Profiles_and_column.out and AVK.out are output files associated with W_XX-EUMETSAT-Darmstadt,HYPERSPECT+SOUNDING,METOPB+CO+IASI_C_EUMP_20220222145400Z_20220222163255Z_eps_r_l2_0100.nc which is a classical L2 CO CDR file.

4.3 Introduction to EUMETSAT Satellite Application Facility on Atmospheric Composition monitoring (AC SAF)

Background

The monitoring of atmospheric chemistry is essential due to several human caused changes in the atmosphere, like global warming, loss of stratospheric ozone, increasing UV radiation, and pollution. Furthermore, the monitoring is used to react to the threats caused by the natural hazards as well as follow the effects of the international protocols.

Therefore, monitoring the chemical composition and radiation of the atmosphere is a very important duty for EUMETSAT and the target is to provide information for policy makers, scientists and general public.

Objectives

The main objectives of the AC SAF is to process, archive, validate and disseminate atmospheric composition products (O_3, NO_2, SO_2, BrO, HCHO, H_2O, OClO, CO, NH3), aerosol products and surface ultraviolet radiation products utilising the satellites of EUMETSAT. The majority of the AC SAF products are based on data from the GOME-2 and IASI instruments onboard Metop satellites.

Another important task besides the near real-time (NRT) and offline data dissemination is the provision of long-term, high-quality atmospheric composition products resulting from reprocessing activities.

Product categories, timeliness and dissemination

NRT products are available in less than three hours after measurement. These products are disseminated via EUMETCast, WMO GTS or internet.

- Near real-time trace gas columns (total and tropospheric O_3 and NO_2, total SO_2, total HCHO, CO) and high-resolution ozone profiles
- Near real-time absorbing aerosol indexes from main science channels and polarization measurement detectors
- Near real-time UV indexes, clear-sky and cloud-corrected

Offline products are available within two weeks after measurement and disseminated via dedicated web services at EUMETSAT and AC SAF.

- Offline trace gas columns (total and tropospheric O_3 and NO_2, total SO_2, total BrO, total HCHO, total H_2O) and high-resolution ozone profiles
- Offline absorbing aerosol indexes from main science channels and polarization measurement detectors
- Offline surface UV, daily doses and daily maximum values with several weighting functions

Climate Data records are available after reprocessing activities from the EUMETSAT Data Centre and/or the AC SAF archives.

- Data records generated in reprocessing
- Lambertian-equivalent reflectivity
- Total OClO

Users can access the AC SAF offline products and data records (free of charge) by registering at the AC SAF web site.

More information about the AC SAF project, products and services: https://acsaf.org/

AC SAF Helpdesk: helpdesk@acsaf.org

Twitter: https://twitter.com/Atmospheric_SAF

4.4 IASI-FORLI retrieval algorithm

4.4.1 IASI instrument

IASI is an infrared Fourier transform spectrometer developed jointly by CNES (the French spatial agency) with support of the scientific community (for a review see [RD1]), and by EUMETSAT. IASI is mounted on-board the European polar-orbiting Metop satellite with the primary objective to improve numerical weather predictions, by measuring tropospheric temperature and humidity with high horizontal resolution and sampling, with 1 km vertical resolution, and with respectively 1 K and 10% accuracy [RD2]. IASI also contributes to atmospheric composition measurements for climate and chemistry applications [RD3]. To reach these two objectives, IASI measures the infrared radiation of the Earth's surface and of the atmosphere between 645 and 2760 cm^{-1} at nadir and along a 2200 km swath perpendicular to the satellite track. A total of 120 views are collected over the swath, divided as 30 arrays of 4 individual Field-of-views (FOVs) varying in size from 36 $\times\pi$ km^2 at nadir (circular 12 km diameter pixel) to 10 x 20 x π km^2 at the larger viewing angle (ellipse-shaped FOV at the end of the swath). IASI offers in this standard observing mode global coverage twice daily, with overpass times at around 9:30 and 21:30 mean local solar time. The very good spatial and temporal sampling of IASI is complemented by fairly high spectral and radiometric performances: the calibrated level 1C radiances are at 0.5 cm^{-1} apodized spectral resolution (the instrument achieves a 2 cm optical path difference), with an apodized noise that ranges below 2500 cm^{-1} between 0.1 and 0.2 K for a reference blackbody at 280 K [RD1].

4.4.2 FORLI overview

FORLI (Fast Optimal/Operational Retrievals on Layers for IASI) is a radiative transfer model based on precalculated look-up tables (LUTs) capable of processing in near-real-time the numerous radiance measurements made by the high-spatial and high-spectral resolution IASI, with the objective to provide global concentration distributions of atmospheric trace gases. For the inversion step, it relies on a scheme based on the widely used Optimal Estimation theory [RD4]. Three versions of the software have been set-up to process IASI level 1C radiances in near-real-time, for vertical profile retrievals of CO, O_3 and HNO_3.

The algorithm description with the methods used for forward and inverse modelling is given in the FORLI ATBD [AD1] and in [RD5].

Information on how to use the NRT CO product can be found in the NRT CO PUM [AD2].