Reference Documentation

Software Version 4AOP2012 v1.0

March 2012

Parc Technologique du Canal
2, Avenue de l'Europe
31520 Ramonville-Saint-Agne
France

Centre National d'Etudes Spatiales
2, place Maurice Quentin
75039 Paris Cedex 01
France

Ecole Polytechnique
91128 Palaiseau Cedex
France

Table of contents

PART I. USER'S GUIDE 7

1. INTRODUCTION 7

1.1. PURPOSE 7
1.2. HOW TO USE THIS DOCUMENT 8
1.3. DEFINITIONS, ACRONYMS AND ABBREVIATIONS 8
1.4. PEOPLE AND ACKNOWLEDGMENTS 9
1.5. REFERENCING 4A/OP IN PUBLICATIONS 9

2. GENERAL FEATURES OF 4A/OP 10

2.1. PHILOSOPHY 10
2.2. ATTRIBUTES 11

3. STRUCTURE OF 4A/OP 12

3.1. FLOW CHART OF THE SOFTWARE 12
3.2. DIRECTORY DESCRIPTION 13
3.3. DATA 14
3.3.1. Input data 14
3.3.1.1. Algorithm fixed input data 14
3.3.1.2. User-defined input data 14
3.3.2. Output data 14
3.3.2.1. "Pseudo-infinite" case 14
3.3.2.2. "Instrument" case 14
3.4. DESCRIPTION OF THE SOFTWARE PROCESS MANAGEMENT 14
3.4.1. Use Make 15
3.4.1.1. Make description 15
3.4.1.2. The file makefile 15

4. INSTALLATION OF 4A/OP 17

4.1. WHAT IS REQUIRED ? 17
4.1.1. Compilers 17
4.1.2. Tcl/Tk 17
4.1.3. Gnuplot 18
4.1.4. Resources 18
4.2. INSTALLATION PROCEDURE 18
4.2.1. System installation 18
4.2.1.1. Summary 18
4.2.1.2. Details 18
4.2.1.3. Contents of the system installation 19
4.2.2. User installation 19
4.3. WAS THE INSTALLATION SUCCESSFUL ? 20

5. HOW TO USE 4A/OP? 22

5.1. GENERAL 22
5.1.1. Execution 22
5.1.2. CPU runtime 23
5.2. USER INPUT 23
5.2.1. Atmospheric profiles 23
5.2.1.1. Content 23

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

5.2.1.2. Format 24
5.2.2. Spectral emissivity 25
5.2.2.1. Content 25
5.2.2.2. Format 25
5.2.3. Instrument Spectral Response Function (ISRF) 26
5.2.3.1. Content 26
5.2.3.2. Format 26
5.2.3.3. Creation example 27
5.2.4. Aerosol/ice cloud physical parameters 28
5.2.4.1. Content 28
5.2.4.2. Format 29
5.2.5. Simulation definition parameters 30
5.2.5.1. Global variables 30
5.2.5.2. Run characterization parameters 32
5.2.5.3. Aerosol/ice cloud run characterization parameters 44
5.3. OUTPUT 46
5.3.1. Output file description 46
5.3.1.1. Content 46
5.3.1.2. Format 47
5.3.2. Examples 51
5.3.2.1. Example 1 51
5.3.2.2. Example 2 53
5.3.3. Result visualization 55
5.3.3.1. Jacobian unit recommendation 55
5.3.4. Statistics tool 56
5.3.4.1. File of the spectra difference 56
5.3.4.2. File of the statistical characteristics per class 57

6. GRAPHICAL USER INTERFACE (GUI) 58

6.1. PRODUCE OR OPEN A GUI PARAMETER FILE 58
6.2. PRODUCE A SCRIPT FILE 58
6.3. USER'S CHOICES 58
6.4. SAVE 65
6.5. RUN 65
6.6. VISUALIZATION 66
6.6.1. Zoom 66
6.6.2. User specific plots 66
6.6.3. Examples 67
6.7. FUTURE IMPROVEMENTS 70
6.7.1. Run 70
6.7.2. Visualization 70

7. REVISION HISTORY 71

PART II. REFERENCE MANUAL 73

1. ALGORITHM DESCRIPTION 73

1.1. DESCRIPTION OF THE PROGRAMS AND DEPENDENCIES 73
1.1.1. Routines used for the radiative transfer computation 74
1.1.2. Routines used to format the atmospheric profiles 75
1.1.3. Routines used to format the instrument functions 75
1.1.4. Routines used for the formatted writing of the 4A outputs concerning high-resolution spectra 75
1.1.5. Routines used for the formatted writing of the 4A outputs concerning convolved spectra and Jacobians 76
1.1.6. Routines used for the computation of the difference between two spectra and the corresponding statistics 76

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2. COMPONENT DESIGN SPECIFICATIONS 77
1.2.1. Radiative transfer computation with 4AOP 78
1.2.1.1. Flow chart 78
1.2.1.2. aaaa 79
1.2.1.3. para4a 86
1.2.1.4. atmref 92
1.2.1.5. gascon 94
1.2.1.6. lireemis 95
1.2.1.7. convini 96
1.2.1.8. atmsph 99
1.2.1.9. calcsphtraj 103
1.2.1.10. init_contin 109
1.2.1.11. peicom 111
1.2.1.12. liread 113
1.2.1.13. readatlas_header 116
1.2.1.14. readatlas 117
1.2.1.15. inouttools 118
1.2.1.16. calcsig 119
1.2.1.17. continua 127
1.2.1.18. contO2 130
1.2.1.19. contN2 132
1.2.1.20. calcrad 133
1.2.1.21. contract 138
1.2.1.22. echantini 141
1.2.1.23. echantillon 143
1.2.1.24. cnvspe 144
1.2.1.25. cnvspehole 151
1.2.1.26. calcisrf 156
1.2.1.27. lireisrf 159
1.2.1.28. parascat 161
1.2.1.29. readaerosols 164
1.2.1.30. contract_sig 166
1.2.1.31. pre_disort 168
1.2.1.32. pre_sos 173
1.2.2. External tools for radiative transfer 178
1.2.3. Mathematical tools 179
1.2.3.1. inttab 179
1.2.3.2. inttab2 180
1.2.3.3. intspline 181
1.2.3.4. getcubicsplinecoeff 182
1.2.4. Physical tools 182
1.2.4.1. plkdir 182
1.2.4.2. plkder 183
1.2.4.3. plkderdble 184
1.2.4.4. tbrillans 184
1.2.5. Useful tools 184
1.2.5.1. dichotom 184
1.2.5.2. dichotomd 185
1.2.6. Atmospheric profile formatting 187
1.2.6.1. Flow chart 187
1.2.6.2. atmformat 187
1.2.6.3. lecatmdsf 190
1.2.7. Instrument function formatting 193
1.2.7.1. Flow chart 193
1.2.7.2. ifct 193
1.2.7.3. lecifctdsf - ifct$(INS) 194
1.2.8. Formatted writing of the 4A output concerning high-resolution spectra 197
1.2.8.1. Flow chart 197
1.2.8.2. lirespi4a 197

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.9. Formatted writing of the 4A output concerning convolved spectra and Jacobians 199
1.2.9.1. Flow chart 199
1.2.9.2. lirespc4a 200
1.2.9.3. readconvdata 202
1.2.9.4. converjacob 205
1.2.10. Statistics tool 208
1.2.10.1. statanalys 208
1.2.10.2. readconvsp 210
1.2.10.3. statchar 211
1.2.10.4. statcharClass 212
1.2.10.5. calcStat 213
1.2.10.6. calcStatJ 214
1.2.11. Constant and common modules 215
1.2.11.1. model 215
1.2.11.2. constants 216
1.2.11.3. instruments 216
1.2.11.4. disort_runparams 216
1.2.11.5. sos_runparams 217
1.2.11.6. error 218
1.2.11.7. spectumio 219
1.2.11.8. convolution 220
1.2.11.9. modatlas 220
1.2.11.10. modinouttools 220
1.2.11.11. spstats 221
1.2.11.12. precision 221

2. SOFTWARE MAINTENANCE 222

2.1. VERSION CONTROL 222
2.2. ERROR CONTROL 222

3. MODEL PHYSICS 223

3.1. RADIATIVE TRANSFER EQUATION 223
3.1.1. General approach 223
3.1.1.1. Discrete equations for the radiative transfer (terrestrial) 224
3.1.1.2. Discrete equations for Jacobians (terrestrial) 225
3.1.1.3. Jacobians for surface temperature and emissivity (terrestrial) 226

PART III. APPENDICES 227

1. APPENDIX A.: DETAILED DESCRIPTION OF REVISIONS 227

1.1. FORTRAN 90 PORTING 227
1.2. SPECTRAL SURFACE EMISSIVITY 227
1.2.1. Spectral surface emissivity 228
1.3. SPHERICAL ATMOSPHERE APPROXIMATION 229
1.4. HOLE CONVOLUTION 231
1.4.1. Introduction 231
1.4.2. Algorithm description 231
1.5. SOLAR CONTRIBUTION 234
1.5.1. Introduction 234
1.5.2. Mathematical description 234
1.5.2.1. Reminder of the general approach 234
1.5.2.2. The solar component 238
1.5.3. Modifications 240
1.5.3.1. Modified software routines 240
1.5.3.2. Modified input/output files 240
1.5.3.3. Run time evaluation 240
1.5.4. Validation 241

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.6. IMPLEMENTATION OF AEROSOL EXTINCTION 247
1.6.1. General 248
1.6.2. Aerosol layer definition and options on spectral variation 249
1.6.3. Comparison with reference software without aerosol 251
1.6.4. Impact of the user parameterised spectral sampling 253
1.6.5. Validation of variable viewing geometry 254
1.6.6. Validation of the simulation of solar radiation 255
1.6.7. Run time tests and trade-off speed vs. accuracy 257
1.6.7.1. Spectral sampling step 257
1.6.7.2. Number of streams 258
1.6.7.3. Accuracy parameter in DISORT simulations including solar radiation 260
1.7. CIRRUS EXTINCTION 262
1.8. IMPLEMENTATION OF LIMB GEOMETRY 262
1.8.1. Introduction 262
1.8.2. Theoretical Basis 263
1.8.3. Validation 268

2. APPENDIX B.: SPECTROSCOPIC DATABASE GEISA 274

3. APPENDIX C.: TEST CASES 275

3.1. DESCRIPTION OF THE 4A SIMULATION TESTS AVAILABLE IN THE FILE CASTEST 286
3.1.1. Simulation definition parameters: global variables 287
3.1.2. Simulation definition parameters: run characterization parameters 288
3.1.3. Atmospheric profiles 289
3.1.4. Surface spectral emissivity 291
3.1.5. Instrument functions 291

4. APPENDIX D.: PERCENTILES 294

5. APPENDIX E.: ON-LINE HELP OF THE GUI 296

5.1. OVERVIEW 296
5.2. GRAPHICAL USER INTERFACE DESCRIPTION 296
5.2.1. File menu 296
5.2.2. Run menu 296
5.2.3. Visualization menu 297
5.3. GUIDE 297
5.4. SPECIFIC HELP FOR SOME PARAMETERS 298
5.4.1. Viewing configuration 298
5.4.2. Atmospheric profile indexes 301
5.4.3. Emissivity indexes 301
5.5. RESOURCES 302
5.6. BUG REPORTS 302
5.7. ANNEX: ACRONYMS 302

6. APPENDIX F.: ERROR MANAGEMENT 303

PART IV. REFERENCES 308

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Part I. User's Guide

1. Introduction

1.1. Purpose

The software 4A/OP is provided for an enhancement by NOVELTIS, in accordance with the convention signed between CNES, LMD/CNRS and NOVELTIS.

The objective of the software is to allow the fast simulation of the radiative transfer in particular over the infrared range either with a "pseudo-infinite" (high) resolution or with a spectral resolution of the simulated instruments (low resolution).

The 4A model is a line-by-line model. It allows a rapid computation of the radiative transfer without accuracy loss thanks to a prior creation of an optical thickness database. This database named Atlas contains the monochromatic optical thicknesses for the various atmospheric constituents (Scott and Chédin, 1981: [Ref. 2]). The atlases are created once and for all by using the line-by-line and layer-by-layer model, STRANSAC (Scott, 1974: [Ref. 1]) in its latest 2000 version with up to date spectroscopy from the GEISA spectral line catalogue (Jacquinet-Husson *et al.*, 1999, 2003, 2008, 2011: [Ref. 5], [Ref. 6], [Ref. 7], [Ref. 8]). This concept has been developed and maintained at the Laboratoire de Météorologie Dynamique (LMD) under the abbreviation 4A for Automatized Atmospheric Absorption Atlas.

4A output is the radiance spectrum in a user-defined spectral domain in the infrared region; the usual spectral domain is between 600 and 3,000 cm^{-1}. 4A can be used for a wide variety of surface and earth atmospheric conditions; its use could be extended to extra-terrestrial atmospheric conditions. Spectra are computed at high spectral resolution (the nominal spectral resolution is 5.10^{-4} cm^{-1}, but it can be changed by users). They can also be convolved with various types of instrument function. Partial derivatives of the radiance with respect to the temperature and gas mixing ratio can be also computed. They allow the model coupling with an inversion algorithm for the atmospheric constituent retrieval from infra-red radiance measurements.

4A heritage is from 4A2000 with its initial development at LMD supported under CNRS/ENS and CNES programs. CNES and CNRS/ENS support the continued advancements of 4A through a subcontract with NOVELTIS. NOVELTIS is in charge of the industrialization and distribution of the 4A software. The aim is to complete an easy-to-use product for several research group users that is also easy to integrate in operational processing chains. 4A/OP software is a version of 4A for distribution to registered users.

Numerous works have been achieved in order to improve the readability of the software code and to favour its portability and its maintenance. The following main tasks have been carried out:

- Inventory of fixtures: description and analysis of the full 4A/OP source code in order to estimate its quality; setup of test cases (and non-regression tests) in order to check the conformity of the consolidated code results with the original code results;
- Addition of scientific functions:
  - Spectral emissivity,
  - The plan-parallel atmosphere assumption has been replaced by a spherical atmosphere approximation,
  - Hole convolution option for computer time saving in some particular cases,

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Implementation of user option for simulation of the solar contribution to spectral radiance and Jacobians,
- Implementation of user option for radiative transfer simulation in limb geometry including refraction,
- Implementation of scattering for aerosol and cloud (cirrus,…) contribution;

- Code industrialization:
  - Setup of a revision control and error management,
  - Addition of comments,
  - Fortran 90 porting,
  - Implementation of the compliance with the Fortran Standards;
- Software promotion:
  - Reference documentation,
  - Graphical User Interface (GUI),
  - Web site http://www.noveltis.fr/4AOP/ including an on-line registration form,
  - Distribution with maintenance and assistance; User licence conditions.

The purpose of the Reference Documentation for 4A/OP is to provide instructions for using the model 4A. It also contains information on the internal data, parameterisations, procedures and the physics used. It should both educate users about what the software does, and instruct them how to use it. The role of this guide is to support the software by explaining as clear as possible the process management and the parameters that are playing a major part.

1.2. How to use this document

The Reference Documentation is divided into three parts: the user's guide, the reference manual and appendices. This document starts with an overview of the model general features (section 2). The next section describes the software structure. It concerns at first the description of the software flow chart (section 3.1) and the directory (section 3.2). Then required input to run the code and output of the software are presented (section 3.3). The following section explains the processing organisation. It aspires to illustrate the principle of the process management (section 3.4). The next section describes the installation of the model (section 4).The following one is a description of the 4A run mode that characterizes the software execution. Section 6 displays the graphical user interface (GUI) and how it has been developed from the existing code in order to make the software use easier. A revision history is also presented in the last section of the user's guide (section 7). The second part is a reference manual that describes in detail the code sources (Part II.1). The programs and sub-programs are presented through dependency trees. The next section presents the software maintenance set up by NOVELTIS (section Part II.2). The last section of the second part describes the physics applied in the model. The third part contains appendices that present more details of the software evolution.

1.3. Definitions, acronyms and abbreviations

4A	Automatized Atmospheric Absorptions Atlas
4A/OP	4A version « opérationnelle »
CNES	Centre National d'Etudes Spatiales
CNRS	Centre National de la Recherche Scientifique
cpu	central processing unit
GEISA	Gestion et Etude des Informations Spectroscopiques Atmosphériques: Management and Study of Atmospheric Spectroscopic Information [Ref. 5], [Ref. 6]

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

IASI	Infrared Atmospheric Sounding Interferometer
IGBP	International Geosphere-Biosphere Program
LMD	Laboratoire de Météorologie Dynamique
MODIS	Moderate Resolution Imaging Spectroradiometer
NedT	Noise equivalent temperature difference
RAM	Random Access Memory
RTM	Radiative Transfer Model
TIGR	Thermodynamic Initial Guess Retrieval: Climatological library of about 2000 representative atmospheric situations from radiosonde reports [Ref. 9]

1.4. People and acknowledgments

The following persons were involved in building and improving the 4A model, as well as in writing the current Reference Documentation.

Authors from LMD: Raymond Armante
Noëlle A. Scott

Authors from NOVELTIS: Laure Chaumat
Carsten Standfuss
Bernard Tournier

4A/OP was developed under funding by CNRS and from CNES programs. Further development is supported by CNES.

1.5. Referencing 4A/OP in publications

If a user is presenting results obtained using 4A/OP, the most complete way to reference it would be to reference the following:

- Scott, N.A. and A. Chedin, 1981: A fast line-by-line method for atmospheric absorption computations: The Automatized Atmospheric Absorption Atlas. *J. Appl. Meteor.*, 20, 802-812;
- L. Chaumat, C. Standfuss, B. Tournier, E. Bernard, R. Armante and N. A. Scott, 2012: 4A/OP Reference Documentation, NOV-3049-NT-1178-v4.3, NOVELTIS, LMD/CNRS, CNES, 311 pp.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

2. General features of 4A/OP

This section gives an overview about the general features of 4A/OP. Details can be found in the second part (section 3 page 223) of this document.

2.1. Philosophy

The radiative transfer model 4A has been designed to allow fast and accurate simulation of radiance spectra as expected to be measured from space with the remote sensing infrared interferometer IASI ([Ref. 3], [Ref. 1]). The spectral range of IASI is 645-2760 cm^{-1}. The direct simulation of the radiative transfer is required for the simulation of the instrument measurement in order to define the instrument characteristics related to the scientific mission requirements. The radiative transfer modelling is also useful for the development of the algorithms used for the atmospheric constituent retrieval from remote radiance measurements.

The simulation of such high-resolution instruments is subjected to three requirements:

- The need to increase the computation accuracy in order to increase the spectral resolution;
- The need to decrease the computation time for the modelling or data processing of these instruments;
- Improvement of the accuracy for the physical process simulation apart from the model structure.

4A is a concept that allows meeting all requirements thanks to the prior building of a database named Atlas. These atlases contain accurate optical thicknesses due to gaseous absorption. They are computed for an ad hoc sampling of temperatures and atmospheric gas mixing ratios at reference pressure levels. Then the model can reconstruct the transmittance profile for any given atmospheric condition by interpolating in the atlases. Afterwards, the integration of the radiative transfer equation is performed.

In more detailed way, the principle of the model computation is as follows.

STRANSAC - the LMD genuine line-by-line and layer-by-layer model (Scott, 1974: [Ref. 1] and later on Tournier *et al.*, 1995: [Ref. 3]) - is used to compute the atlases of optical depths:

- for 12 nominal atmospheres (12 temperature profiles 7K distant);
- for each absorbing gas (reference mixing-ratio profiles);
- for a set of 44 pressure levels (between surface and top of the atmosphere, see Table 1 below; 44 is the new current default but not mandatory);
- at, at least, a fraction of half width spacing (adaptable, through atmospheric layers "spectral representation step").

4A is separated into 15 cm^{-1} blocks, leading to several thousands (15 cm^{-1} / representation step) by 528 (44 layers times 12 temperatures) matrices for each gas and each 15 cm^{-1} region. Each matrix is compressed in wavenumber/layer/temperature and wavenumber/layer/gas space and only the significant values are kept. Values (in transmittance units) too close to zero are treated separately in order to be reintroduced and taken into account in case of high values of viewing angles or absorber amounts. The resulting compressed matrices are stored. The 2-bytes numerical encoding (integer 2) is now replaced by a 4-bytes numerical encoding (real 4) that allows a better computation precision.

To do a calculation, the compressed matrix is:

1. uncompressed ("zero" is reintroduced);
2. interpolated:
   - to the correct temperature for each layer (interpolation between two adjacent temperature profiles),

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- to the correct pressure levels (the nominal 44 pressure levels *are not mandatory*) scaled to the correct absorber amount and secant *(all* gases may be considered as «variable» with respect to the altitude).

Absorption from each individual gas is added up and a radiance calculation, as well as transmittance and jacobian calculations (these latter are optional), are performed.

Starting from these high spectral resolution optical depths, transmittance profiles, Jacobian profiles, brightness temperatures are generated using an appropriate *"spectral integration step"* combined with a relevant convolution step to take into account the various instrument functions.

Level	Pressure (hPa)	Level	Pressure (hPa)	Level	Pressure (hPa)	Level	Pressure (hPa)
1	0.0026	12	3.33	23	106.27	34	471.86
2	0.0089	13	4.98	24	131.20	35	525.00
3	0.0240	14	7.43	25	161.99	36	584.80
4	0.0500	15	11.11	26	200.00	37	651.04
5	0.0900	16	16.60	27	222.65	38	724.78
6	0.1700	17	24.79	28	247.87	39	800.00
7	0.3000	18	37.04	29	275.95	40	848.69
8	0.5500	19	45.73	30	307.20	41	900.33
9	1.0000	20	56.46	31	341.99	42	955.12
10	1.5000	21	69.71	32	380.73	43	1013.25
11	2.2300	22	86.07	33	423.85	44	1050.00

Table 1: New current 44-level vertical grid of the 4AOP optical thickness atlases

2.2. Attributes

Some important 4A/OP physical attributes are as follows.

- χ-correction of Voigt profiles (Perrin *et al*;, 1989: [Ref. 10]);
- Continuum updates:
  - H_2O (CKD2.1 Clough *et al.*, 1989: [Ref. 11] and [Ref. 12]),
  - N_2 (Lafferty's *et al.* values, 1996: [Ref. 13]),
  - O_2 (Thibault's *et al* values, 1997: [Ref. 14]) ;
- Update of GEISA spectroscopic parameters (Jacquinet-Husson *et al.*, 1999, 2003, 2008, 2009: [Ref. 5], [Ref. 6], [Ref. 7] and [Ref. 8]);
- Pressure shift for H_2O, CO2 and N2O;
- Partition Function used for the temperature dependence of the line intensities (updated version of TIPS-DS97, Gamache *et al.*, 2000: [Ref. 15]);
- Effects of CO_2 line mixing (Rodrigues, 1998: [Ref. 16]; Niro *et al.*, 2004, 2005: [Ref. 17] [Ref. 18]): new algorithm from Lamouroux *et al.*, 2010 [Ref. 19] including pressure shift for CO_2 (Hartmann, 2009 [Ref. 20]).

4A is maintained at LMD which includes introducing all the newly derived parameters for spectroscopy, for line-mixing or for continua, for aerosols, cirrus and CFCs, ..., as soon as they have been validated.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

3. Structure of 4A/OP

3.1. Flow chart of the software

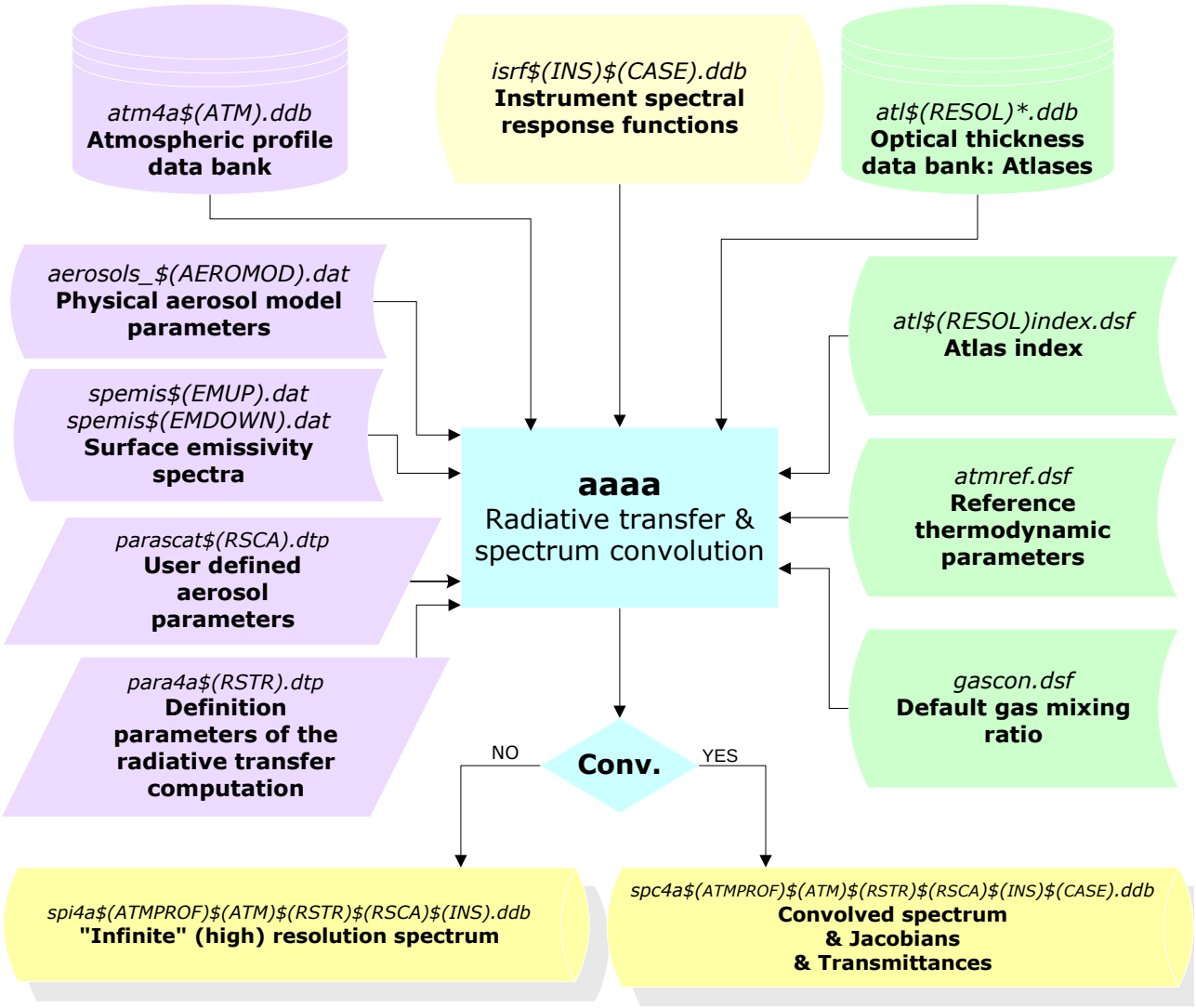


Figure 1: Flow chart of the software 4A/OP (* for several files).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

3.2. Directory description

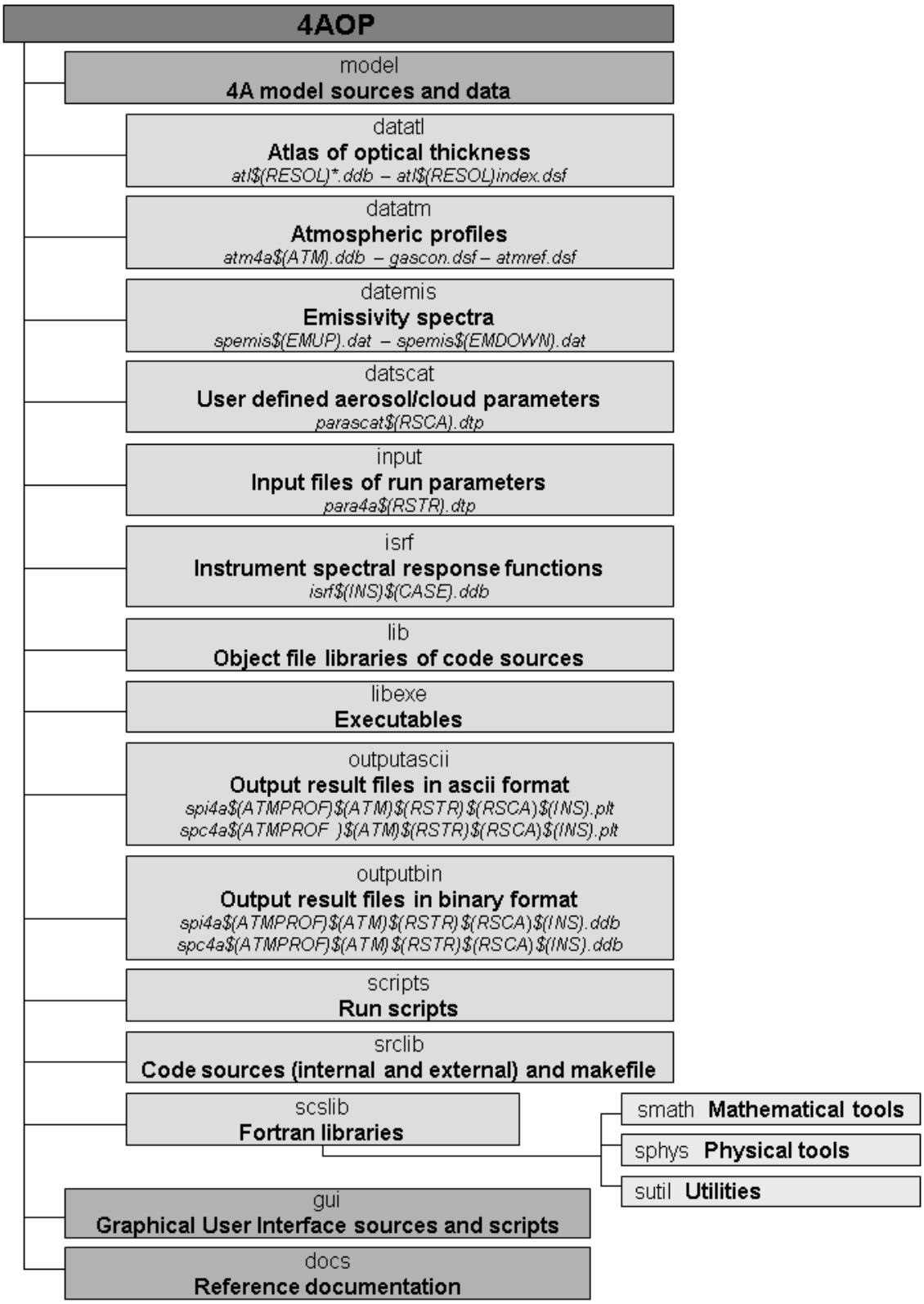


Diagram 1: Directory description of 4A/OP (* for several files).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

3.3. Data

3.3.1. Input data

Optical thicknesses in atlases.

Reference thermodynamical parameters.

Default gas mixing ratio.

Atmospheric profiles.

Simulation definition parameters.

Instrument spectral response functions.

Input data include algorithm fixed input data and user input data.

3.3.1.1. Algorithm fixed input data

Optical thickness atlases.

Reference thermodynamical parameters used for the computation of the atlases.

Default gas mixing ratio.

3.3.1.2. User-defined input data

Atmospheric profiles on user-defined pressure levels (see section 5.2.1).

Spectral emissivity (see section 5.2.2).

Instrument spectral response functions (see section 5.2.3) if it is necessary.

Aerosol models and cloud types (see section 5.2.4) if necessary.

Simulation definition parameters (see section 5.2.5).

3.3.2. Output data

Output data always include radiances and other quantities for an instrument simulation case.

3.3.2.1. "Pseudo-infinite" case

High-resolution radiance spectra.

3.3.2.2. "Instrument" case

Convolved radiance spectra.

Jacobians on user-defined pressure layers: partial derivatives of the radiance with respect to the temperature, the gas mixing ratio and the surface emissivity.

Transmittances on user-defined pressure layers.

3.4. Description of the software process management

The operator management is performed by the way of the Unix software Make.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

3.4.1. Use Make

Management controls are grouped in the files makefile.

3.4.1.1. Make description

The processing of the operator sequence is managed by the software Make. This software allows the administration of executables by creating a temporal dependency between executables. The compilation is performed if the date of the last modification of the source file is posterior to the date of the executable creation.

Because of its convenient faculties, the software Make is also used for the execution. It makes the execution of any operator possible if its output interface file does not exist, or has been created at a date anterior to the creation date of its input interface file. This way of using the software Make requires a good definition of the interface file names.

3.4.1.2. The file makefile

The file makefile organises the hierarchic execution of dependent operators from the building of instrument functions and atmospheric profiles, if that is required, to the radiative transfer computation by using the 4A method.

The set of dependencies linking the interface files between the main operators is grouped in only one main file makefile. When the creation date of the output interface file of a main operator is anterior to the one of the input interface file, this main operator runs after the required associated operators have been executed.

Variables describing the main operator state are initialised in the main file makefile and in the run shell-script.

Primary directories	File Names	Contents
srclib	makefile	Execution makefile of the main operator 4A: manages file's links and executions. It also manages compilation by using the instructions include Makefile.4a and include makefile_ifct_atm
	Makefile.4a / Makefile.4a.inc	Compilation makefile: manages the whole compilation and creates libraries of object files
	makefile_ifct_atm	Compilation makefile for the formatting programs of the atmospheric profiles and instrument functions (atmformat and ifct) and for the reading and ASCII writing programs of 4A output (lirespc4a and lirespi4a)

Table 2: description of the directory of files makefile

The user defines the variables related to the experiment in a run shell-script that are passed on to the execution makefile and also by setting up the required parameter files corresponding to the user input variables. A shell-script runs controls of the main file makefile. A target is specified in the script to select the output level of the code. Once the run is finished, the user can examine the result files.

Each operator is linked with files of parameters and data (measured, estimated or specified). The following main source dependency tree (Diagram 2) describes the dependency between main processes.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
lirespc4a.......................... § Part II.1.1.5
    |
    |
    |__ aaaa......................... § Part II.1.1.1
        |
        |
        |__ atmformat............ § Part II.1.1.2
        |__ ifct................. § Part II.1.1.3
lirespi4a.......................... § Part II.1.1.4
    |
    |
    |__ aaaa......................... § Part II.1.1.1
        |
        |
        |__ atmformat............ § Part II.1.1.2
        |__ ifct................. § Part II.1.1.3
```

Diagram 2: Main source dependency tree of the software 4A/OP

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

4. Installation of 4A/OP

The installation phase includes the installation of the algorithm folder tree with available files and the compilation of the full source code.

4.1. What is required ?

The 4A/OP distribution package is supplied in a single tar file `4AOP-version.tar.bz2`, where `version` corresponds to the current 4A/OP Software version number (presently `4AOP-2011-1-0`). The file is compressed using `bzip2`, available at no cost from the Free Software Foundation.

The following files are sufficient for the installation process:

- `README` (file containing important information);
- `4AOP-version.tar.bz2` (4A/OP as compressed (zip) tar file).

Installation of 4A/OP on a UNIX system is performed using UNIX b`unzip2` and `tar` commands and the GNU Make program.

`4AOP-version.tar.bz2` is a compressed tar file containing the entire 4A/OP program (possible exception: large data base files which might be stored separately). This comprises several subdirectories with FORTRAN 90 source files (`*.f90`), a GNU-`Makefile`, UNIX shell scripts, user input files (`*.dtp`) and several other files including all necessary input and test data for 4A/OP.

4.1.1. Compilers

The code has been developed in Fortran 77 on SUN workstations and has been ported in Fortran 90 on SUN Ultra Sparc workstations under Solaris using the Forte 6.2 Fortran 90 compiler. 4A/OP has also been tested on PC Linux using the pgf90 compiler (Portland Group Fortran 90 compiler), g95/gfortran (GNU compiler), ifort (Intel compiler) and f90 (SUN compiler). The 4A/OP source code is FORTRAN 90 compatible and contained within several files `*.f90`.

Note: The different Fortran 90 compilers are proposed in the following order: f90 (SUN) (Linux or SunOS), ifort (INTEL) (Linux only), pgf90 (PGI) (Linux only), gfortran (GNU) (Linux only) and g95 (GNU) (Linux or SunOS). Fortran compiler is limited to 32-bit mode if it is not default mode (except in ifort case, UNKNOWN 32-bit limitation option). The user can define his/her choice by >> export PATH_F90=... (compiler with full path).

Moreover, a new routine written in C has been introduced and needs that a C compiler is installed on your system. 4A/OP has been tested on PC Linux using the gcc compiler (GNU compiler) and cc (SUN compiler).

Note: The different C compilers are proposed in the following order: cc (SUN) (Linux or SunOS) and gcc (GNU) (Linux or SunOS). C compiler is limited to 32-bit mode if it is not default mode. The user can define his/her choice by >> export PATH_CC=... (compiler with full path).

Important note: 4A/OP has been developed and tested on a big-endian architecture (Sun). In particular the optical thickness atlases are Sun binary files. On little-endian architecture (i.e. x86), be sure your compiler has the option that swaps bytes at file reading.

4.1.2. Tcl/Tk

The 4A/OP GUI needs that Tcl/Tk (at least version 8.4) is installed on your system. See www.tcl.tk for more information on Tcl/Tk.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

4.1.3. Gnuplot

The visualization part of the 4A/OP GUI needs Gnuplot software to plot graphs. The GUI has been tested with the following version of Gnuplot: Unix version 4.0 patchlevel 0. See www.gnuplot.info for more information on Gnuplot.

4.1.4. Resources

The 4A executable occupies about one megabyte on disk only. It uses about 200 megabytes of memory. The associated files (sources and user input files) occupy about 20 megabytes of disk space. There are also the atlas files that can require up to 6.5 gigabytes of disk space. This volume depends on the user simulation needs.

The size of the output files is a function of the input simulation parameters, which you control via the file described in section Part I 5.2.5.2 (Jacobian calculations can use lot of disk space).

4.2. Installation procedure

Invoking the UNIX commands bunzip2 and tar creates the 4A/OP main directory `4AOP-version/` just below the current directory including all necessary subdirectories. Simply type

```
bunzip2 4AOP-version.tar.bz2
tar xvf 4AOP-version.tar
```

in order to uncompress and "untar" `4AOP-version.tar.bz2`.

Directory `4AOP-version/` should now have been created and should contain the following files and subdirectories:
`Makefile`, `README`, `README.user.in`, `VERSION`, `configure`, `configure.in` and the directories `model/`, `gui/` and `doc/`.

There are two levels of installation: system and user. The second one is useful when several users need to use the software.

4.2.1. System installation

4.2.1.1. Summary

Here are the instructions to build the main installation of 4AOP, from the main directory `4AOP-version/`:

```
./configure
make
```

The user has to install atlas files if they are not included in this archive (i.e. the directory `model/datatl` is empty). If the user atlas directory is `/aaaa/atlas`, the user can do either

```
cp /aaaa/atlas/* model/datatl
```

or

```
rmdir model/datatl
ln -s /aaaa/atlas model/datatl
```

4.2.1.2. Details

The configure step creates files (like makefiles) that need system configuration information. If this step leads to incorrect settings, the user can modify these files by hand. In particular, the user needs to edit `Makefile.4a` for compiler settings.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

The make step compiles. The compilation generates libraries, object files and executables.

4.2.1.3. Contents of the system installation

`4AOP-version.tar.bz2` contains the entire 4A/OP program (possible exception: large data base files which might be stored separately). There are several directories that should be included in the installation (i.e. the directory where `./configure` and `make` have been done; see above):

`model/` The model itself. This comprises several subdirectories with Fortran 90 source files, makefiles, UNIX shell scripts, user input files and several other files including all necessary input and test data for 4A/OP.

`gui/` The graphical user interface for the model. This comprises Tcl/Tk sources and library, gnuplot scripts and on-line help for the GUI.

`doc/` The documentation for 4A/OP.

The directory `model` should contain the following subdirectories:

datatl/	datscat/	isrf/	outputascii/	scslib/
datatm/	guiparam/	lib/	outputbin/	srclib/
datemis/	input/	libexe/	scripts/	

The subdirectory `model/libexe/` should contain the executable files `aaaa, atmformat, lirespc4a, lirespi4a, statanalys`.

Six user input files are necessary to completely specify all 4A/OP input parameters (section 5.2). These files are read by the 4A/OP main program `aaaa` through several subroutines which set up all input parameters required for the simulation definition.

- One user interface file has a predefined filename: `para4a.dtp`. It is located in the subdirectory `input`. This ASCII file has to be edited manually (if the user does not use the GUI) in order to set several switches (e.g., trace gas selection, convolution mode), and to set various computational parameters, to specify the wave number range of interest, etc. (see section 5.2.5.2);
- Another user interface file is needed if aerosol contribution is required; its predefined filename is `parascat.dtp`. It is located in the subdirectory `input`. This ASCII file has to be edited manually (if the user does not use the GUI) in order to set several parameters, to specify the aerosol model, the number of aerosol layers, etc. (see section 5.2.5.3);
- Four other user input files are necessary:
  - One deals with the atmospheric state (profiles) to be simulated (see section 5.2.1), located in the subdirectory `datatm`,
  - One with the spectral emissivity (see section 5.2.2), located in the subdirectory `datemis`,
  - One with the aerosol physical parameters (see section 5.2.4), located in the subdirectory `datscat`,
  - And one with the instrument function (see section 5.2.3) for an instrument simulation case, located in the subdirectory `isrf`.

4.2.2. User installation

If a user wants to install a local 4A/OP working directory, go to the 4A/OP installation directory (i.e. the directory where `./configure` and `make` have been done; see above) and type:

```
make INS_DIR=/home/user/4AOP installuser
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

where `/home/user/4AOP` is the 4A/OP install directory for the user. See the `README` file in the user directory (`/home/user/4AOP` in our example) for instructions to use 4A/OP as a user.

4.3. Was the installation successful ?

In order to check if 4A/OP has been correctly installed, some reference input files have been defined and some reference output files created (on SUN !). All relevant test input files are located in directory `model/input/` and test output files are located in directory `model/outputascii/`.

A UNIX shell script in the directory `model/srclib/`, `CasTest` (see section Part III.3 for details), directs 4A/OP to produce output corresponding to some sample problems. The output is written to files `spi4a0001testtest1infb1.plt`, `spc4a0001testtest1testb1.plt`, etc, in the directory `model/outputascii/`. To ensure the code is operating properly, you should compare your results with the contents of the corresponding files provided in the release package, named `spi4a0001testtest1infb1.plt_ref`, `spc4a0001testtest1testb1.plt_ref`, etc, which were generated by 4A/OP on our Solaris Sun workstation. The current directory should be `~/srclib/`. Please, perform the following steps:

1. Execute 4A/OP for several tests:

   ```
   cd model/srclib
   ./CasTest
   ```

   Please note that this run test as well as the test sequence described below will use the following user input files which already exist in `model/input/` as well as in `model/datatm/`, `model/datemis/`, `model/datscat/` and `model/isrf/`:

   ```
   input/para4atest*.dtp
   input/parascat*.dtp
   datatm/atm4atest*.dsf
   datemis/spemis*.dat
   datscat/aerosols_*.dat
   isrf/isrf*.dsf
   ```

2. Now the following files should have been created:

   ```
   datatm/atm4atest*.ddb
   isrf/isrf*.ddb
   outputbin/spc4a*test*.ddb
   outputbin/spi4a*test*.ddb
   outputascii/spc4a*test*.plt
   outputascii/isrf*.plt
   srclib/CasTest.log
   ```

3. The contents of these files should be essentially identical with the corresponding `*_ref` files (note that the numerical results may differ somewhat depending on the computer system and then the compiler used).

   The script performs fourteen tests for testing different simulation conditions with 4A. These tests are based on the `~/input/para4atest*i.dtp` (*i = 1,14) input files. After each execution of 4A/OP, the

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

corresponding `sp*i.plt_ref` reference files in `~/outputascii/` and the newly generated `sp*i.plt` files are compared using the UNIX `diff` command. The script performs also three more tests for testing the atmospheric profile and instrument function building.

For all tests, the basic scenario is the same. They only differ in wave number settings, simulated atmosphere state, convolution mode selection, simulated instrument, Jacobian computation (only in Test5) and scattering effects. Test8 is the most time consuming.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

5. How to use 4A/OP?

4A/OP processing is controlled by a set of global variables that identifies required input data files and files of run parameters. The user defines these variables through a script file and they are passed on to the execution makefile.

The execution principle is presented first. User input files, global variables and parameters are described in sub-sections of Part I 5.2. The following section presents the output results.

In the variable tables, the "Type" indication follows the notation described below:

int	Integer (4bytes)
r	Single precision real (4bytes)
d	Double precision real (8bytes)
char	Character
string	Character string

In the variable tables, the "Source/Destinat°" indication follows the notation described below:

Ini	Initialisation file (read via makefile)

Convention for additional notations:

NA	Not Applicable

5.1. General

5.1.1. Execution

After 4A/OP has been successfully installed, the program `aaaa` can be easily executed either by using the Graphical User Interface or by typing the file name of a run shell script. See the file `README` in the user installation directory.

The simplest way to use 4A/OP is to launch its Graphical User Interface:

```
./4Agui
```

This program is in the `gui/bin` directory of the system installation. Make sure this directory is in your PATH or launch `4Agui` from this directory. See sections Part I.6 and Part III.4.

If the user wants to use the software 4A/OP "by hand", the user can use the script `run4a_example` as an example:

```
./run4a_example
```

in directory `model/scripts/` or by invoking `run4a_example` by using the complete path to `scripts/` whilst being in a different directory.

Before starting the program, the user has to manually adjust this script file for other simulation conditions and to set up several user interface files in order to specify the simulation conditions. These user interface files are described in the next section.

In the running script, a target is specified in order to point at the process that the user wants to execute (`make target`). Presently, in order to compute the radiative transfer and produce ASCII result files, the target name is:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- in the case of "infinite" resolution spectra: `runlirespi4a`
- in the case of spectra convolved with an instrument function: `runlirespc4a`

If the user wants BINARY output only, the corresponding instruction is: `make runspi4a`

`make runspc4a`

The entire instruction used in the script is (for a convolved spectrum):

```
make -f ../srclib/makefile runlirespc4a INS=$INS CASE=$CASE ATMPROF=$ATMPROF
ATM=$ATM RSTR=$RSTR RSCA=$RSCA NUMIN=$NUMIN NUMAX=$NUMAX UNIT=$UNIT
```

where the list of variables are global variables described in section Part I 5.2.5.1.

The instruction "make help" (from the directory `model/srclib/`) produces a list of executions available when using the program make.

5.1.2. CPU runtime

The computation time for a spectrum alone is reasonable but depends on the machine used. For Jacobians, it can take a long time to get results. The most time consuming step of the simulation is the convolution. When Jacobian calculation is selected a convolution is done for each molecule and each level, then the computation is significantly longer. The computation time is a function of the number of levels, the number of implied molecular species, the spectral resolution and the instrumental function for the convolution case. Here is some computing time examples for two different machines in the case of a IASI spectrum:

Machine	IASI spectrum alone	IASI spectrum + 4 Jacobians
Linux Xeon Bipro 3.4 GHz (32bits - Fedora)	About 35 seconds	About 2.5 minutes
Linux Xeon 7 proc. 3 GHz (64bits -Debian)	About 21 seconds	About 2 minutes
Unix Sun V880 900 MHz	About 110 seconds	About 30 minutes

Table 3: CPU runtime examples

5.2. User input

Four user input files are necessary to completely specify all 4A/OP input data. The user has the choice to use the default installed files or to use her/his own files provided that they are compatible with the required input format.
For the two binary files (atmosphere state and instrument functions), the default procedure is to install ASCII files that are automatically formatted into binary files via interface subroutines. The user is free to modify these subroutines in order to code her/his own ASCII file format reading.

This section describes the format of the input files that are directly read by the main program 4A.

5.2.1. Atmospheric profiles

The file describing the atmosphere profiles used for a 4A run are placed in the directory **`datatm`**.

5.2.1.1. Content

The file of atmospheric profiles contains necessarily the user-defined temperature profile and the corresponding pressure levels. The surface pressure and the surface temperature are also specified. The default installed file contains humidity and ozone profiles too.

The user is free to specify any gas mixing ratio profile provided that this gas is defined in the GEISA database (see appendix Part III section 2). If the user defines any gas mixing ratio profile and if the user

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

wants to simulate the radiative transfer by including the effect of some gases, the gas mixing ratios are defined through the default gas concentration file `gascon.dsf` (placed in the same directory `datatm`).

For a satellite measurement simulation from a radiosonde atmospheric profile, we recommend the user to extend the profile until 0.1hPa at least in order to avoid wrong extrapolations in the radiative transfer computation. This extend can be made by using profile shapes from climatology databases for example.

The database is stored in a file named: **`atm4a$ATM.ddb`**.

`$ATM` = extension index of the database type given by the user (default possible values are test, test2, satigr_v4.0_moyclas, etc: see section 5.2.5.1).

5.2.1.2. Format

The sub-routine `atmsph` allows reading this kind of files. It is placed in the directory **`srclib`** (Part II.1.2.1.8). The program `atmformat` (section Part II.1.2.6) together with the subroutine `lecatmdsf` (section Part II.1.2.6.3) support the user for the creation of these files in the appropriate format from a user-defined file `atm4A$ATM.dsf`.

The atmosphere binary files are formatted according to the following characteristics:

- Suffix: `.ddb;`
- Access: direct binary;
- Length of a record block (buffer): 1024 words;
- Length of a word: 32 bits for real variables and for integer variables;
- Header records describe the file structure.

These files are composed of several first header records of maximum length 1024*4 bytes and a variable number of record blocks dependent on the number of pressure levels. Numbers in parentheses (1), (2), (3) and (4) indicate the real values of the corresponding variables in the used default files, respectively natm, 1+i, 1+natm+i and 1 + 2*natm +i. Variable names in parentheses (nlevel), (nbcor), etc are the names used in the sub-routine `atmsph` (see their definition in section Part II 1.2.1.8) and in the program `atmformat`.

Number of the record	Parameter name	Description	Units	Dimension	Type	Size (bytes)
1	natm (1)	Number of atmospheric profiles	-	1	Int	4
1 + i* (2)	iloc	Index number of the first record block	-	natm (3072 max)	Int	4
1 + natm + i* (3)	ilas	Number of significant words in the last record block	-	natm (3072 max)	Int	4
1 + 2*natm + i* (4)	nben	Number of record blocks	-	natm (3072 max)	Int	4
7+j**	buff	Profile characteristics	-	nmbuf	Real	4*dim
	buff(1)	Number of pressure levels of the profiles (nlevel)	-	1		4
	buff(2)	Number of various gas (nbcor)	-	1		4
	buff(2+nc, nc=1,nbcor)	Gas index (idcor), according to the GEISA database convention	-	2		4*2
	buff(nbcor+3)	Index of the profile (nopro)		1		4

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Number of the record	Parameter name	Description	Units	Dimension	Type	Size (bytes)
	buff(nbcor+4)	Surface temperature (tdown)	K	1		4
	buff(nbcor+5)	Surface pressure (pzdown)	hPa	1		4
	buff(nbcor+6)	Secant value of viewing angle (sectta1)	-	1		4
	buff(nbcor+7+ (nl-1)*(2+nbcor), nl=1,nlevel)	Pressure levels (pl)	hPa	nlevel		4*dim
	buff(nbcor+8+ (nl-1)*(2+nbcor), nl=1,nlevel)	Temperature profile (tl)	K	nlevel		4*dim
	buff(nbcor+8+nc +(nl-1) *(2+nbcor), nl=1,nlevel, nc=1,nbcor)	Gas concentration profile (rol)	g/g	nlevel*nbcor		4*dim

* $1 \leq i \leq (natm)/1024 + 1$; ** $1 \leq j \leq (nmbuf)/1024 + 1$, $nmbuf = nlevel*(nbcor+2)+nbcor+5$

Table 4: format description of atmospheric database files

5.2.2. Spectral emissivity

The files describing the infrared surface emissivity spectrum for different types of surface used for a 4A run are placed in the directory **`datemis`**.

5.2.2.1. Content

Files of emissivities contain spectral emissivities and the corresponding wave numbers. Each emissivity spectrum is stored in a file named: **`spemis$EMDOWN.dat`** (and **`spemis$EMUP.dat`** in the case of two paths).

`$EMDOWN` = emissivity index for the lower level

`$EMUP` = emissivity index for the top level

The emissivity indexes are equivalent to the emissivity codes described in appendices, section 1.2. You can also edit the files `readme.txt` and `index.txt` available in the directory **`datemis`** for the index meaning.

For example, if the user wants to create her/his own constant emissivity, she/he can copy one of the constant emissivity file in the emissivity database directory and change its second column (see below) to the constant emissivity value. Be sure the name of the new file is spemisXXX.dat where XXX is the new emissivity (i.e. 057 for a value of 0.57).

5.2.2.2. Format

The sub-routine `lireemis` allows reading this kind of files. It is placed in the directory **`srclib`**.

These files are formatted according to the following characteristics:

- Suffix: `.dat;`
- Access: sequential formatted (ASCII).

These data files are composed of one header line and two columns of a variable number of lines.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Number of the line	Parameter name	Description	Units	Dimension	Type	Format
1	EmisDim	Number of emissivity channels	-	1	Int	i4
2 to emisDim+1	emis (fisrt column)	Wave numbers	cm^{-1}	emisDim	Real (double)	Free
	emis (second column)	Mean spectral emissivity	-	emisDim	Real (double)	Free
	third to sixth columns	Statistics on emissivity (not read)	-	emisDim	Real (double)	Free

Table 5: format description of emissivity spectra

5.2.3. Instrument Spectral Response Function (ISRF)

The file describing the instrument functions used for a 4A run are placed in the directory **`isrf`**.

5.2.3.1. Content

The file of instrument functions contains the function characteristics in the file header and the functions as a function of the wave numbers in the file body. The instrument functions are stored in a file named: **`isrf$INS$CASE.ddb`**.

`$INS` = Indicator (name) of the simulated instrument (ISRF) (default possible values are inf, test, test2, iasi1c, etc. ; « inf » for "pseudo-infinite" spectrum (an ISRF file is not required), « iasi1c » for a level 1c IASI spectrum, or « test » and « test2 » for test functions; see test cases in Part III 3.1.5).

`$CASE` = Indicator of the instrument function version (case) for a given instrument defined by the variable `INS` (see section 5.2.5.1).

Important note: In the file, at least two instrument functions have to be defined at two different wave numbers included in the simulated spectral domain (even if they are the same).

Three types of instruments are possible: radiometer, interferometer or spectrometer. The second category of instrument is divided into two types: a constant sampling step and a non constant sampling step in wavenumber. An example of a radiometer ISRF is contained in the file **`isrftest.dsf`** (in the directory **`model/isrf`**), an example of an interferometer/spectrometer ISRF with a constant sampling step is given by the file **`isrftest2.dsf`** and an example of an interferometer/spectrometer ISRF with a non constant sampling step is given by the file **`isrfairs0802.dsf`**

5.2.3.2. Format

The sub-routine `lireisrf` allows reading this kind of files. It is placed in the directory **`srclib`**. The program `ifct` (Part II.1.1.3 and Part II.1.2.7) supports the user for the creation of these files in the appropriate format from an user-defined file `isrf$INS$CASE.dsf` (see 5.2.3.3 below).

These files are formatted according to the following characteristics:

- Suffix: `.ddb`;
- Access: direct binary;
- Length of a record block (buffer): 1024 words;
- Length of a word: 32 bits for real variables and for integer variables.

These files are composed of four first header records of maximum length 1024*4 bytes and a variable number of record blocks dependent on the number of instrument functions and their size.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Number of the record	Parameter name	Description	Units	Dimension	Type	Size (bytes)
1	fisrfu	File name of the instrument functions	-	1	String	2
1	titre	Description of the instrument functions	-	1	String	2
1	nbk	Number of spectral bands	-	1	Int	4
1	itypeconv	Indentifier of the convolution type	-	1	Int	4
2	nufdeb	First central wave number of the ISRFs	cm^{-1}	nbk (3 max)	Real	4*dim
2	nuffin	Last central wave number of the ISRFs	cm^{-1}	nbk (3 max)	Real	4*dim
2	dnds	Spectral step of central wave numbers	cm^{-1}	nbk (3 max)	Real	4*dim
2	nknu	Number of ISRFs	-	nbk (3 max)	Int	4*dim
2	pdnuco	Sampling step of the ISRFs	cm^{-1}	nknu*nbk (1000*3 max)	Real	4*dim
2	kfmo	Number of description points of the ISRF	-	nknu*nbk (1000*3 max)	Int	4*dim
3	nufo	Central wave numbers of each ISRF	cm^{-1}	nknu*nbk (1000*3 max)	Real	4*dim
3	wnuo	Width of each ISRF	cm^{-1}	nknu*nbk (1000*3 max)	Real	4*dim
4	dnuo	Spectral shift of each ISRF	cm^{-1}	nknu*nbk (1000*3 max)	Real	4*dim
4	irecf	Record number of the beginning of each ISRF	-	nknu*nbk (1000*3 max)	Int	4*dim
4+j*	crp	Instrument functions	-	30001 max	Real	4*dim

* *$1 \leq j \leq$ kfmo/1024 + 1, ((kfmo(k,nb), k=1,nknu(nb)),nb=1,nbk)*

Table 6: format description of instrument function files

5.2.3.3. Creation example

The section Part II.1.2.7 aims at supporting users to set up the instrument function files. However, some points could remain unclear for new users who are not familiar with Fortran code. That is why the present section proposes to help users to install her/his own ISRF files.

According to the flow chart (Diagram 12) in section Part II.1.2.7, the first step is to prepare the file `isrf$(INS)$(CASE).dsf`:

1. The file name has to correspond to the instrument name INS (and optionally to the version of the instrument specified by the variable CASE). For example, if INS=instru and CASE not used (empty character string), the name of the file is: `isrfinstru.dsf`. It is placed in the directory `/isrf`;
2. Then it is recommended to include a header in the file as in the ASCII file `isrftest.dsf` for example:
   a. Fisrt line: number of total functions,
   b. Line above each function: index number of the function, corresponding central wave number, number of description points of the function (number of samplings),
   c. The entire definition spectral domain of the functions has to be included in the limits of the radiative transfer simulation. The entire spectral domain of the functions are defined by the first and last central wave numbers and the function width.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

In a second step, the user has to prepare the corresponding reading subroutine *lecifctdsf* located in the file named `ifct$(INS).f90`. The easiest way is to take as an example an existing subprogram, for example `ifcttest.f90`:

1. In the directory `/srclib`, the user has to copy `ifcttest.f90` in `ifctinstru.f90`;
2. Then, it is recommended to edit the new file in order to correct some parameters if it is necessary in particular:
   a. the ISRF sampling step: change the variable `pdnuco` (0.1) into the user-defined sampling step value.
   b. the convolution type: change the variable **`itypeconv`** into the user-defined convolution type value. The value of **`itypeconv`** is choosen among 3 possibilities: CONV_RADIOMETER, CONV_INTERF_CS and CONV_INTERF_NCS. These constants are defined in the module instruments:
      i. CONV_RADIOMETER: used in case of a radiometer with n channels (Météosat, HIRS…);
      ii. CONV_INTERF_CS: used in case of an interferometer/spectrometer with a constant sampling step in wave number (IASI: step = 0.25cm^{-1},…);
      iii. CONV_INTERF_NCS: used in case of an interferometer/spectrometer with a non constant sampling step in wave number (AIRS,…).

Now, all is ready to compile and create automatically the binary file at the execution of the program 4A by specifying the correct instrument function name ("instru" in our example). Indeed the ISRF formatting program *ifct* (conversion from ASCII to 4A binary format) is compiled with the correct reading subroutine (`ifct$(INS).f90`) corresponding to the specified instrument INS at the moment of the execution of the program 4A The instrument function binary file is created (`isrfinstru.ddb`) and passed to 4A as an input.

5.2.4. Aerosol/ice cloud physical parameters

The files describing aerosol/ice cloud physical parameters are placed in the directory **`datscat`**.

5.2.4.1. Content

Aerosol physical parameters are provided in files, separately for a given aerosol model, placed in the directory `datscat`. The nomenclature is `aerosols_$(AEROMOD).dat`. Files for 12 aerosol models are included in the software package:

- inso00 Insoluble;
- micm00 Mineral (coarse mode);
- mitr00 Mineral (transported);
- soot00 Soot;
- ssam70 Sea salt (accumulation), 70% relative humidity;
- ssam95 Sea salt (accumulation), 95% relative humidity;
- sscm70 Sea salt (coalescence), 70% relative humidity;
- sscm95 Sea salt (coalescence), 95% relative humidity;
- suso70 Sulfate, 70% relative humidity;
- suso95 Sulfate, 95% relative humidity;
- waso70 Water soluble, 70% relative humidity;

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- waso95 Water soluble, 95% relative humidity.

The user may define new aerosol models but must provide aerosol parameter files in the same format.

Cirrus extinction is modelled in the same way as aerosol extinction. For the user's convenience, cloud physical parameters are provided for 13 different cirrus cloud models/size distributions from two different sources. These files have the same nomenclature as corresponding aerosol physical parameter files `aerosols_$(AEROMOD).dat`.

The spectral coverage of the three ice cloud models from OPAC (Part IV.[Ref. 25]) is limited to wave numbers above 1000 cm^{-1} with very poor sampling in the thermal Infrared.

- cir100 Liou size distribution 30-1300 µm at -25°C
- cir200 Liou size distribution 30-1300 µm at -50°C
- cir300 as cir200, with additional small-size particles down to 2 µm

The spectral coverage of the ten selected models from Part IV.[Ref. 34] based on the size distribution model Part IV.[Ref. 33] is limited to the range between 780 and 1500 cm^{-1} with good sampling in between.

The selected models correspond to five effective diameters (10, 30, 50, 70, 90 µm) and two shapes (aggregate and columns).

- agg010 Aggregate, effective diameter 10 µm
- …
- agg090 Aggregate, effective diameter 90 µm
- col010 Columns, effective diameter 10 µm
- …
- col090 Columns, effective diameter 90 µm

5.2.4.2. Format

The sub-routine `readaerosol` allows reading this kind of files. It is placed in the directory **`srclib`**.

These files are formatted according to the following characteristics:

- Suffix: `.dat;`
- Access: sequential formatted (ASCII).

These data files are composed of

✦ 24 header lines: at the 16th line of the header, the value corresponds to the reference aerosol/cloud optical depth at 1000 cm^{-1}.

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
aotref	Reference aerosols optical depth at 10 µm	r	µm	-		

✦ and nine columns of a variable number of lines:

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
wl	Wavelengths for which aerosol model properties are provided	r[ksmax]	µm	-		in increasing order
ecoef	Extinction coefficient at wavelengths wl	r[ksmax]	km^{-1}	-		

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
scoef	Scattering coefficient at wavelengths wl	r[ksmax]	km^{-1}	o	*NOT USED*	
acoef	absorption coefficient	r[ksmax]	km^{-1}		*NOT USED*	
ssa	Single scattering albedo at wavelengths wl	r[ksmax]	-	-		
asp	Asymmetry parameter at wavelengths wl	r[ksmax]	-	-		
eno	normalized extinction coefficient	r[ksmax]	-		*NOT USED*	
rir	refractive index - real part	r[ksmax]	-		*NOT USED*	
rii	refractive index - imaginary part	r[ksmax]	-		*NOT USED*	

Table 7: format description of aerosol/ice cloud physical parameters file

5.2.5. Simulation definition parameters

5.2.5.1. Global variables

Global variables are initialised through the run script file and are passed to the file makefile. They are listed below.

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
Radiative transfer computation						
JOB	Indicator of the program to use	string	-	`makefile`	JOB is fixed to 4a	4a
ATM	Indicator of the atmospheric database stored in the file `atm4a$(ATM).ddb`	char	-	`makefile` / directory `datatm`	(Presently, possible values are test, test2, satigr_v4.0_moyclas: TIGR atmosphere type)	test
ATMPROF	Index of the atmospheric profile read in the atmospheric database stored in the file named `datatm/atm4a$(ATM).ddb`. The range of profile indexes and their meaning depend on the atmospheric database.	int	-	`makefile`	Presently, possible values are 1 to 5. In the case of the total TIGR databank, not available with the default 4A/OP installation, 1≤ATMPROF≤ 1761 1 to 322: tropical class 323 to 710: warm temperate 711 to 1064: cold temperate 1065 to 1761: arctic class	0001

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
RSTR	Indicator of the parameter file `para4a$(RSTR).dtp`	char	-	`makefile` / directory `input`	(Presently, possible values are test1 to test10)	NA
RSCA	Indicator of the aerosol parameter file `parascat$(RSCA).dtp`	char	-	makefile / directory input	if empty string: no aerosol is considered	NA
RESOL	Indicator of the resolution of atlases described in the file `atl$(RESOL)index.dsf`	char	-	`makefile` / directory `datatl`	(Presently, possible values are hr09, that corresponds to the high resolution atlas database using spectroscopic parameters from the GEISA 2009 edition)	hr09
INS	Indicator (name) of the simulated instrument (instrument function ISRF) stored in the file named `isrf$(INS)$(CASE).ddb`	string	-	`makefile` / directory `isrf`	Only used if the convolution is performed (presently, possible values are inf, test, test2, iasi1c)	inf
CASE	Indicator of the instrument function version (case) for a given instrument defined by the INS	char	-	`makefile` / directory `isrf`	(Presently, possible values are " ")	" "
Reading the output of the radiative transfer and convert it into ASCII format						
NUMIN	Beginning of the extraction for the conversion (in wave number)	r	cm^{-1}	`makefile`	NUMIN must be ≥ *forig* (see Table 9)	635
NUMAX	End of the extraction for the conversion (in wave number)	r	cm^{-1}	`makefile`	NUMAX must be ≤ *fstop* (see Table 9)	2880
UNIT	Jacobian unit index for Jacobian ASCII outputs	i	-	`makefile`	• UNIT = 0: NedT at TREF - Kelvin per Kelvin.layer - Kelvin per layer for 1 g/g variation of the mixing ratio ro - Kelvin for 1% variation of surface emissivity • UNIT = 1: NedT at TREF - Kelvin per Kelvin.Km - Kelvin per Km for 10% variation of ro - Kelvin for 1% variation of surface emissivity • UNIT = 2: Decibel - Decibel per Kelvin.Km - Decibel per Km for 10% variation of ro - Decibel for 1% variation of surface emissivity • UNIT = 3: NedT at brightness temperature Tb - Kelvin per Kelvin.Km - Kelvin per Km for 10% variation of ro - Kelvin for 1% variation	4

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
					of surface emissivity • UNIT = 4: NedT at Tb - Kelvin per Kelvin.layer - Kelvin per layer for 1 g/g variation of ro - Kelvin for 1% variation of surface emissivity • UNIT = 5: Initial unit - W/(m^2 str cm^{-1}) per Kelvin.layer - W/(m^2 str cm^{-1}) per layer for 1 g/g variation of ro - W/(m^2 str cm^{-1}) for 100% variation of the surface emissivity	
TREF	Reference temperature for conversion into NedT	r	K	`makefile`		280
TEBR	Radiometric noise estimation for conversion into Decibel	r	K	`makefile`		0.28
WCLASS	Width of the class in wave number for statistical characteristics of the difference between two spectra	d	cm^{-1}	`makefile`		50
WNMIN	Limits of the computation of the difference between two spectra in wave numbers	d	cm^{-1}	`makefile`		645
WNMAX		d	cm^{-1}	`makefile`		2760

Table 8: global variables initialised through the file `makefile`

5.2.5.2. Run characterization parameters

The file containing the run parameters is named `para4a$RSTR.dtp`, where `$RSTR` is the value of the variable defining the experiment parameters, and it is stored in the directory `input`. An example of such a file is displayed below. Parameters introduced after limb geometry implementation are highlighted in bold letters. Then a table (Table 9) describes every parameter.

```
'AAAA'
' Geometry Identifier (EVIEW/LIMB)                    GEOMID'      'LIMB'
' Viewing configuration (UP, DOWN, DOWNUP, UPDOWN)    TRAJET'      'DOWNUP'
' Upper limit of simulation (hPa/km)                  PZUP'        100.00
' Lower limit of simulation (hPa/km)                  PZDOWN'      0.00
' Observation level (hPa if EVIEW/km if LIMB)         PZOBS'       100.00
' Index for the Emissivity of the upper level         EMUP'        100
' Temperature of the upper level                      TUP'         4.00
' Index for the Emissivity of the lower level         EMDOWN'      100
' Temperature of the lower level                      TDOWN'      4.00
' Geometric tangent height (km) (LIMB only)           ZGTAN'      30.00
' Angle type definition  (ANGLE,SECANT,UNDEF)         TYP_ANG'    'ANGLE'
' Viewing angle (in degrees or secant value)          SECTTA1'    0.
' Reflexion/emission angle (same unit as above)       SECTTA2'    55.
' Solar zenith angle (same unit as above)             SECTTASUN'  90.
' Earth-Sun distance in astronomical units            ESDIST'     1.
' Atmospheric Refraction                              REFRA'      'YES'
' Altitude of lowest atm. pressure level (km)         ZATMBOT'    0.
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
' Continua (H2O, N2, O2) contribution                    CONTIN'    'YES'
' Spectral resolution                                    RESPEC'    0.01
' First wave number of the simulation                    FORIG'      2045.
' Last wave number of the simulation                     FSTOP'      2050.
' Width of each Atlas spectral band                      DELDS'      15.
' Optical thickness interpolation in temperature         INTOPT'     'YES'
' Indicator of convolution with the ISRF                 CONV'       'NO'
' Indicator of hole algorithm for the convolution        HOLE' 'NO'
' Spectral shift introduction due to the ISRF            SHIFT'      'NO'
' Value of the global spectral shift                     RNU'       0.
' Convolution step                                       PASCON'    0.25
' Transmittance calculation flag                         FPOID'     'NO'
' Jacobian calculation flag                              JACOB'     'NO'
' Selection indexes for the 40 gases of the GEISA database (1 or 0)IDSEL'
'   h2o   co2    o3   n2o    co   ch4    o2    no   so2   no2'
    1    , 1   , 1   , 1   , 1   , 1   , 1   , 0   , 0   , 0   ,
'   nh3   ph3  hno3    oh    hf   hcl   hbr    hi   clo   ocs'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  h2co  c2h6  ch3d  c2h2  c2h4  geh4   hcn  c3h8  c2n2  c4h2'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  hc3n  hocl    n2 ch3cl  h2o2   h2s hcooh  cof2   sf6  c3h6'
    0    , 0   , 1   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  ho2  clono2  xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  xxx    xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  cfc11 cfc12 ccl4                                            '
    1    , 1   , 1   ,
' Weighting coefficients of the mixing ratio          ROCOEF'
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
' Selection indexes for the gas Jacobian calculation (1 or 0) IDJAC'
'   h2o   co2    o3   n2o    co   ch4    o2    no   so2   no2'
    1    , 0   , 1   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'   nh3   ph3  hno3    oh    hf   hcl   hbr    hi   clo   ocs'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  h2co  c2h6  ch3d  c2h2  c2h4  geh4   hcn  c3h8  c2n2  c4h2'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  hc3n  hocl    n2 ch3cl  h2o2   h2s hcooh  cof2   sf6  c3h6'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  ho2  clono2  xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  xxx    xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  cfc11 cfc12 ccl4                                            '
    0    , 0   , 0   ,
```

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
AAAA	Main run parameter that identifies the program to use	string (len=60 max)	-	Ini: parameter file para4a.dtp	Only for information: not used	AAAA
GEOMID	Geometry type identifier	string (len=6)	-	Ini: para4a.dtp	LIMB or EVIEW	EVIEW
TRAJET	Viewing configuration. The various configurations are defined in Table 10 and illustrated by Figure 2 to Figure 5	string (len=6 max)	-	Ini: parameter file para4a.dtp	UP, DOWNUP, DOWN, UPDOWN UP: nadir viewing DOWN: zenith viewing	UP
PZUP	TOA or cloud bottom pressure if GEOMID = 'EVIEW' TOA or cloud bottom height if GEOMID = 'LIMB'	r	hPa km	Ini: para4a.dtp		0.05
PZDOWN	Surface or cloud top pressure if GEOMID = 'EVIEW' Surface or cloud top height if GEOMID = 'LIMB'	r	hPa km	Ini: para4a.dtp	if "Earth view" and if *pzdown* undefined (negative): set *pzdown* to the surface pressure level read in the atmospheric database	1013.25
PZOBS	Pressure at observation level Height at observation level	r	hPa km	Ini: para4a.dtp	if GEOMID = 'EVIEW': pzup≤*pzobs*≤pzdown if GEOMID = 'LIMB': pzdown≤*pzobs*≤pzup	0.05
EMUP	Index for the emissivity of the top level read in the file datemis/ spemis$EMUP.dat	r	-	Ini: parameter file para4a.dtp	Used if *TRAJET*≠UP	100
TUP	Temperature of the top level	r	K	Ini: parameter file para4a.dtp	4 K in general (cold space temperature) If equal to 0, set to temperature of top level in atmospheric database.	4.
EMDOWN	Index for the emissivity of the lower level read in the file datemis/ spemis$EMDOWN.dat	r	-	Ini: parameter file para4a.dtp	Used if *TRAJET* ≠ DOWN	100

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
TDOWN	Surface temperature (temperature of the lower level)	r	K	Ini: parameter file para4a.dtp	If *TDOWN < 0* set to temperature of level2 If *TDOWN*=0: - for level2 < nlevtam, *TDOWN* = temperature of level2 - for level2 = nlevtam, *TDOWN* = surface temperature defined in the atmospheric database Else *TDOWN*=user input value	0.
ZGTAN	Geometric tangent height	r	km	Ini: para4a.dtp	Used if GEOMID = 'LIMB'	0.
TYP_ANG	Angle type definition for sectta1, sectta2 and secttasun (see below). This variable are also detailed in Table 10 and illustrated in Figure 2 to Figure 5	string (len=6 max)	-	Ini: parameter file para4a.dtp	ANGLE, SECANT or UNDEF ANGLE: secttax=1/cos(secttax), x = 1, 2 or sun SECANT: secttax=secttax UNDEF: sectta1=0, sectta2=1.7434468, secttasun=0	ANGLE
SECTTA1	Primary path angle (viewing angle) scaled to ground (altitude z=0).	r	degree	Ini: parameter file para4a.dtp	If *TYP_ANG*=SECANT, *SECTTA1* must be ≥1 If *TYP_ANG*=UNDEF, *SECTTA1* is set to 0 and then it is set to the value read in the atmosphere file datatm/ atm4a$(ATM).ddb	0. new: *sectta1* <1 or *sectta1* ≥90° admitted
	Or secant value	r	-			
SECTTA2	Secondary path angle (reflexion incidence angle) at z=0.	r	degree	Ini: parameter file para4a.dtp	Used if *TRAJET* = DOWNUP or UPDOWN If *TYP_ANG*=SECANT, *SECTTA2* must be ≥1 If *TYP_ANG*=UNDEF, *SECTTA2* is set to 1.7434468 (cos(55°))	55.
	Or secant value	r	-			

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
SECTTASUN	Solar zenith angle	r	degree	Ini: parameter file para4a.dtp	Used if *TRAJET*= DOWN or DOWNUP If *TYP_ANG*=SECANT, *SECTTASUN* must be ≥1 If *TYP_ANG*=UNDEF, *SECTTASUN* is set to 0. To suppress the solar contribution, specify SECTTASUN outside [0,90[deg.	90.
	Or secant value	r	-			
ESDIST	Earth-Sun distance in astronomical units	r	astro. units	Ini: parameter file para4a.dtp	Natural values are [0.98, 1.02]	1
REFRA	Refraction computation identifier	string (len=6)	-	Ini: para4a.dtp	YES or NO	NO
ZATMBOT	Height of lower atmospheric level	r	km	Ini: para4a.dtp	if < 0: computed in atmsph through barometric height formula	0.
CONTIN	Continua (H_2O, N_2, O_2) contribution	string (len=6 max)	-	Ini: parameter file para4a.dtp	YES or NO If yes, the continua of the selected molecules (*IDSEL*=1) contribute to the radiative transfer computation	YES
RESPEC	Spectral resolution: contraction of the spectra before convolution or to define the spectral discretization with an "infinite" resolution simulation	r	cm^{-1}	Ini: parameter file para4a.dtp	<u>If *RESPEC*<5.10^{-4}</u>: – For *CONV*=YES, the resolution for the computation = 5.10^{-4} – For *CONV*=NO, the resolution for the computation = Atlas resolution <u>Else</u> *RESPEC* = user input value	0.00
FORIG	First wave number of the simulation	r	cm^{-1}	Ini: parameter file para4a.dtp	*FORIG* must be ≥ forig_atlas*	600.
FSTOP	Last wave number of the simulation	r	cm^{-1}	Ini: parameter file para4a.dtp	*FSTOP* must be ≤ fstop_atlas*	3000.
DELDS	Width of each Atlas spectral band	r	cm^{-1}	Ini: parameter file para4a.dtp	generally 15 cm^{-1}	15.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
INTOPT	Indicator for interpolation in temperature of the optical thicknesses	string (len=6 max)	-	Ini: parameter file para4a.dtp	YES or NO Note: If no, the computation time is decreased but the results are deteriorated	YES
CONV	Indicator of convolution with the ISRF	string (len=6 max)	-	Ini: parameter file para4a.dtp	YES or NO ("infinite" resolution)	NO
HOLE	Indicator of hole algorithm use for the convolution (see Part III1.4)	string (len=6 max)	-	Ini: parameter file para4a.dtp	YES or NO	NO
SHIFT	Indicator of a spectral shift introduction due to the ISRF	string (len=6 max)	-	Ini: parameter file para4a.dtp	YES or NO Only used if *CONV*=YES If yes, a spectral shift is applied due to ISRF not centred	NO
RNU	Value of the global spectral shift	r	cm^{-1}	Ini: parameter file para4a.dtp	Only used if *CONV*=YES Application of a possible global spectral shift of the spectra if *RNU* ≠ 0	0.
PASCON	Convolution step: step of the ISRF application	r	cm^{-1}	Ini: parameter file para4a.dtp	Only used if *CONV*=YES Must be consistent with the instrument function file header; *PASCON* only used if itypeconv=2 (see §5.2.3, Part II 1.2.7) it must be >*RESPEC*	0.
FPOID	Transmittance calculation flag	string (len=6 max)	-	Ini: parameter file para4a.dtp	YES or NO Jacobians and transmittances are not permitted if *CONV*=NO	NO

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	Source / Destinat°	References / Remarks	Default value
JACOB	Jacobian calculation flag	string (len=6 max)	-	Ini: parameter file para4a.dtp	YES or NO If *JACOB*=YES: – INTOPT must be YES – The Jacobian with respect to the temperature is always calculated. – The Jacobian with respect to the mixing ratio is calculated if the corresponding indicator is equal to 1 (see IDSEL and idjac below) – The Jacobian with respect to the surface emissivity is calculated if the corresponding indicator is equal to …	NO
IDSEL	Table to select molecules among the 42 gases available in the GEISA database and 3 CFCS coded between 61 to 63	int[63]	-	Ini: parameter file para4a.dtp	1 or 0 1=yes 0=no	0, …
ROCOEF	Table of weighting coefficients for the mixing ratio of the gases and CFCS	r[63]	-	Ini: parameter file para4a.dtp	0 ≤ *ROCOEF*	1.0, …
IDJAC	Table to select the calculation of the Jacobian for each gas in the 42 available gases and 3 CFCS	int[63]	-	Ini: parameter file para4a.dtp	1 or 0 (yes or no) Warning: The Jacobian calculation of a molecule is effective only if the corresponding molecule is selected too (idsel=1)	0, …

* forig_atlas and fstop_atlas are the limits of the atlas in wave numbers and are defined in the file `atlas$(RESOL)index.dsf`

Table 9: Run characterization parameters read in the file para4a.dtp

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

5.2.5.2.1 Viewing and geometric configuration definition

This section defines the terminology used for the viewing and geometric configuration of the radiative transfer computation in Table 10. The various viewing configurations are illustrated by Figure 2 to Figure 5. Viewing configurations in limb geometry are illustrated by Figure 6 to Figure 10.

<table>
<tr><th>TRAJET</th><th>SECTTA1</th><th>SECTTA2</th><th>SECTTAS</th></tr>
<tr><td>Earth View-UP</td><td>Viewing (sat.) zenith angle (at surface z=0)</td><td rowspan="2">-</td><td>-</td></tr>
<tr><td>Earth View-DOWN</td><td>Viewing (obs.) zenith angle (at surface z=0)</td><td rowspan="2">Solar zenith angle (at surface z=0)</td></tr>
<tr><td>Earth View-DOWNUP</td><td>Viewing (sat.) zenith angle (at surface z=0)</td><td rowspan="2">Reflexion incidence angle (scaled to surface z=0)</td></tr>
<tr><td>Earth View-UPDOWN</td><td>Viewing (obs.) zenith angle (at surface z=0)</td><td>-</td></tr>
<tr><td>Limb-UP</td><td rowspan="5">< 1 or ≥ 90°

zgtan must be specified</td><td>-</td><td>-</td></tr>
<tr><td>Limb-DOWN</td><td>-</td><td>-</td></tr>
<tr><td>Limb-DOWNUP</td><td>55° incidence zenith angle at cloud top (cloud above tangent point)</td><td>solar zenith angle at cloud top (cloud above tangent point)</td></tr>
<tr><td>Limb-DOWNUP</td><td>set to sectta1 (limb view through tangent point)</td><td>-</td></tr>
<tr><td>Limb-UPDOWN</td><td>55° incidence nadir angle at cloud bottom (cloud above tangent point)</td><td>-</td></tr>
</table>

Table 10: Terminology for the viewing and geometric configuration of 4A/OP.

Important note 1: User-defined angles need to be scaled to surface at z=0 in order to take into account spherical atmospheres (see section Part III 1.3). The zenith angle at ground (z=0) is given by:

$$\theta_0 = \arcsin [(R+z)/R \sin\theta(z)]$$

where R is the Earth radius and $\theta(z)$ the zenith angle as function of altitude z.

Important note 2: If the user has specified in configuration DOWN a solar zenith angle different from the viewing zenith angle, no sunlight is observed if the sun is completely outside the instrument's field of view. The simulation result will be identical to that obtained in nighttime conditions.
If the user has specified in configuration DOWN a solar zenith angle equal to the viewing zenith angle, the simulation considers that the instrument's field of view is fully occupied by the solar disk (SECTTASUN is set to 0, TUP to 6000K and EMUP to 100). If the user wishes to simulate a situation where the solar disk partly covers the instrument's field of view, he has to adjust the following specifications:

- Specify SECTTASUN as for a nightime situation;
- Specify the upper temperature TUP to the solar surface temperature (6000K);
- Specify the upper emissivity EMUP according to the fraction of the solar disk within the instrument's field of view. Do not forget to generate the file spemis$(EMUP).dat if it does not exist already.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

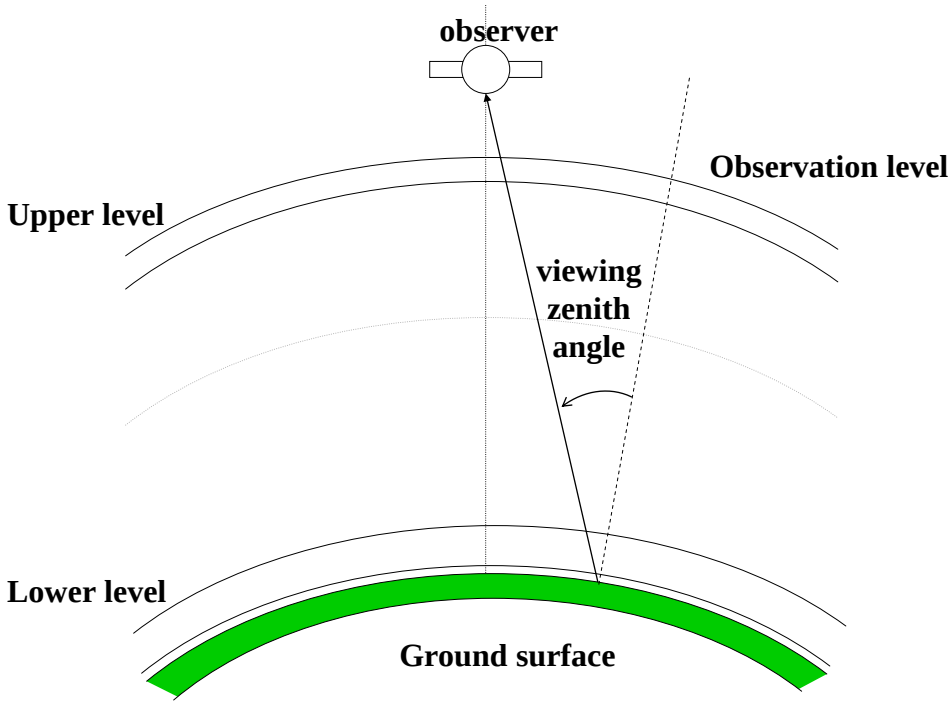


Figure 2: Nadir viewing (*TRAJET* = UP)

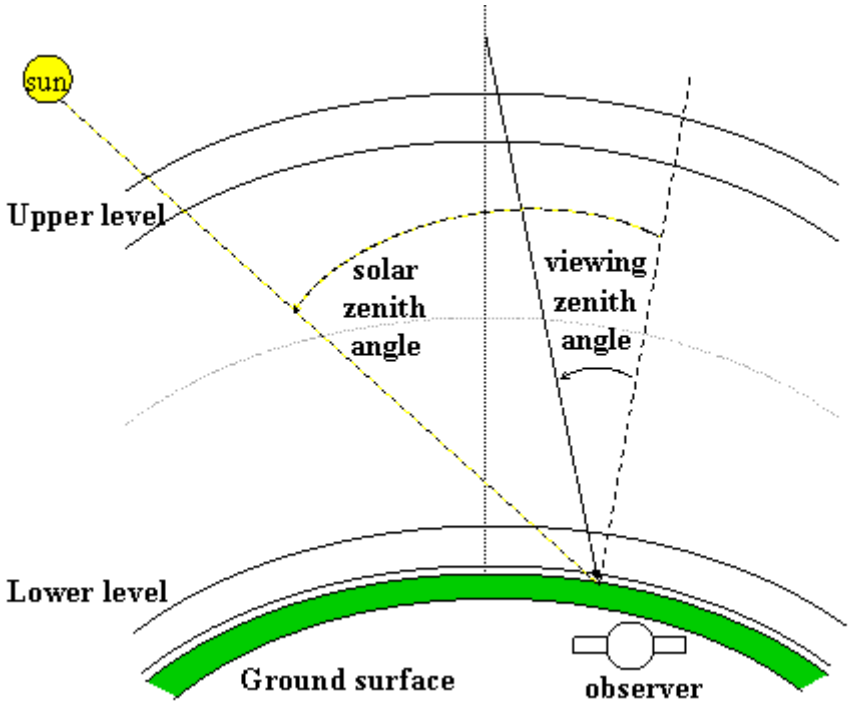


Figure 3: Zenith viewing (*TRAJET* = DOWN)

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

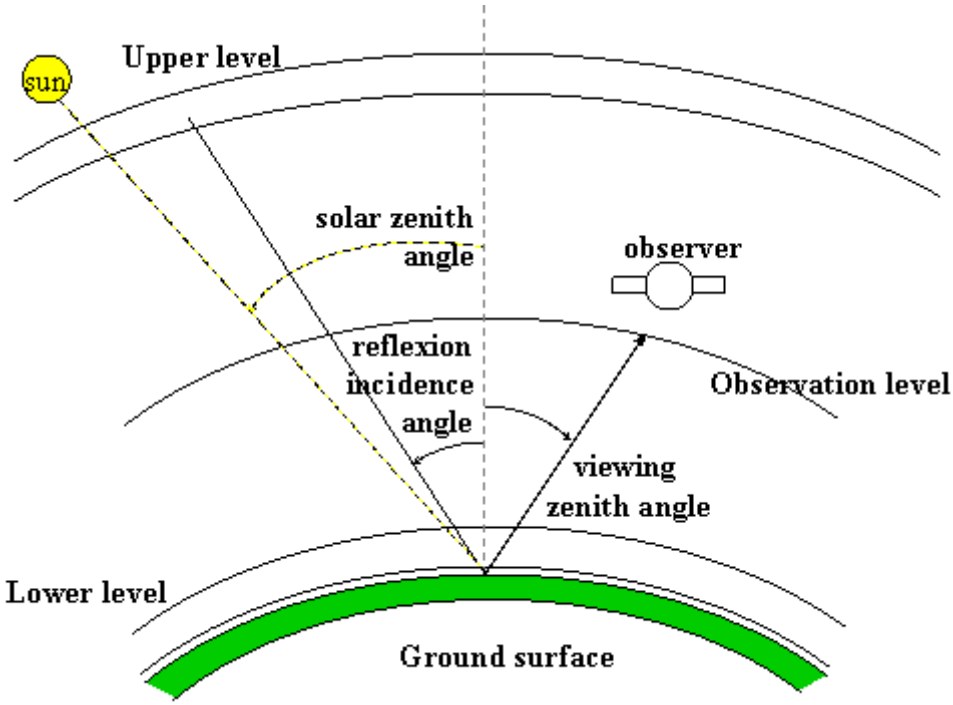


Figure 4: Viewing with ground surface reflexion (*TRAJET* = DOWNUP)

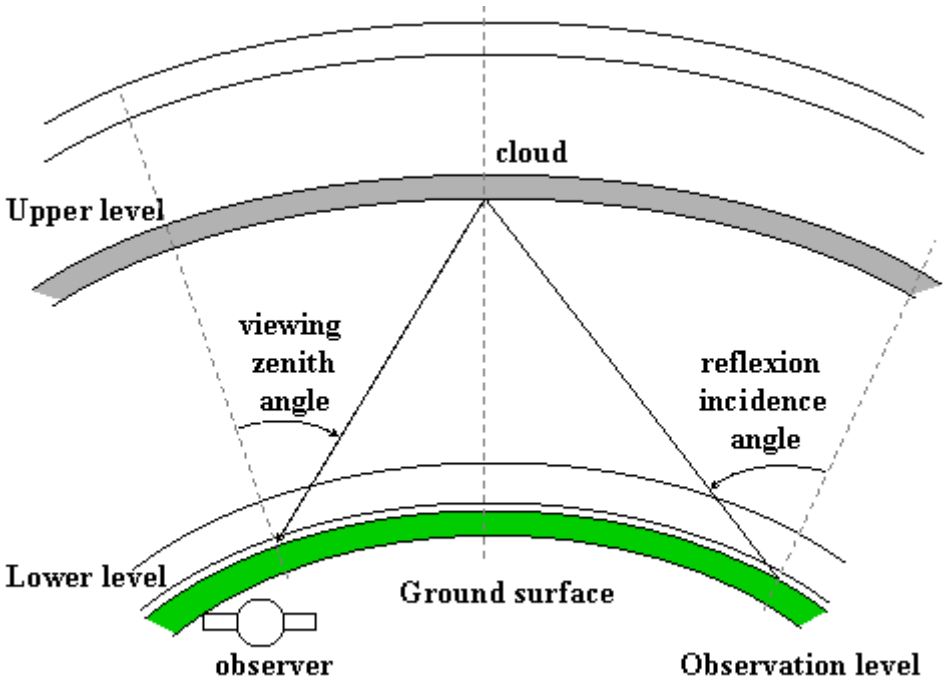


Figure 5: Viewing with cloud reflexion (*TRAJET* = UPDOWN)

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

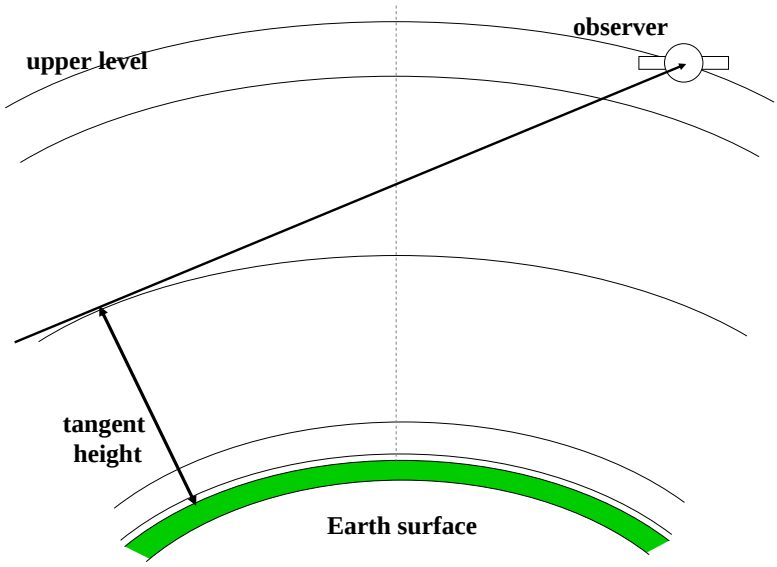


Figure 6: Limb view through the tangent point (*TRAJET* = DOWNUP). The user specifies background temperature and emissivity (cloud or sky), tangent height and the lower level below tangent height. The observer must not be specified above the background level.

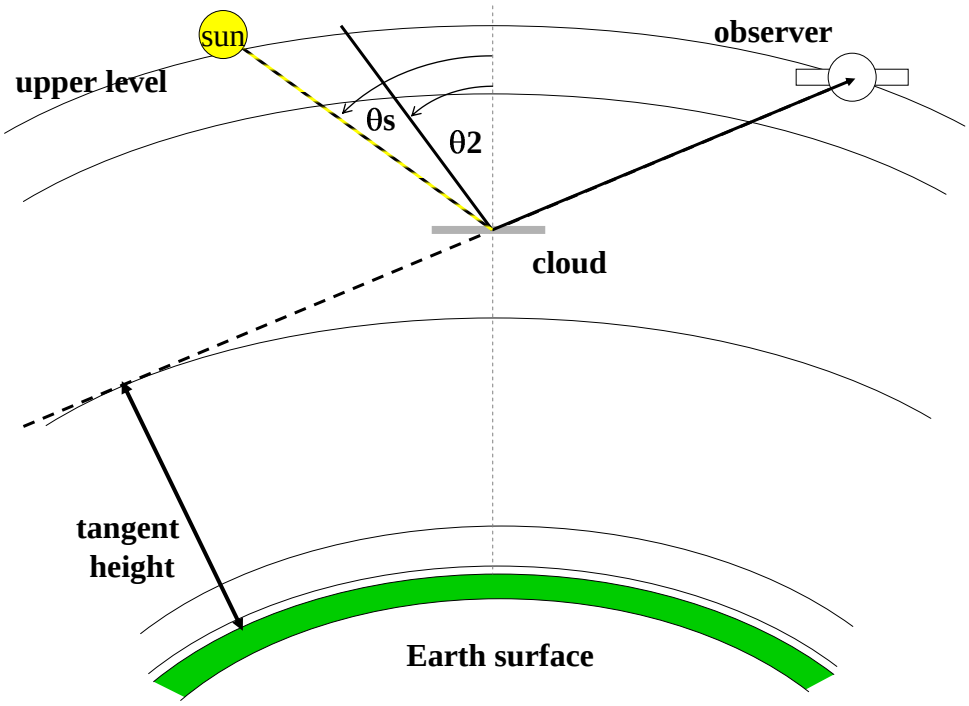


Figure 7: Limb view of a cloud top, including reflection of downward terrestrial and solar radiation (*TRAJET* = DOWNUP). The user specifies background temperature and emissivity (sky or cloud), cloud top temperature and emissivity, tangent height, the lower level above tangent height, and the solar zenith angle θs (if background is sky). θ2 is fixed to 55°. The observer must not be specified above the background level.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

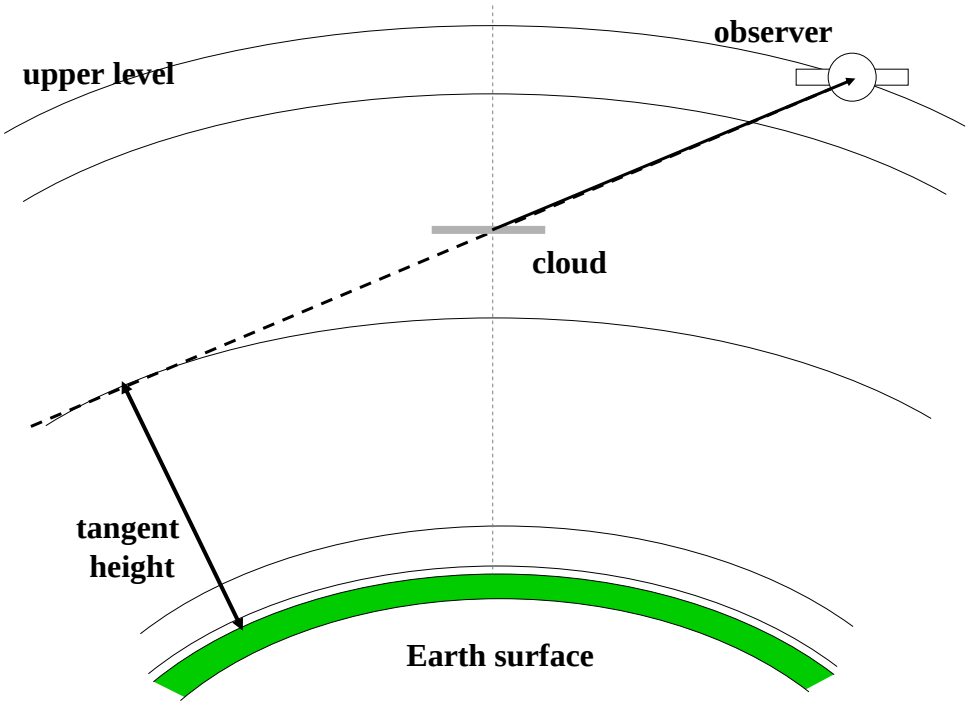


Figure 8: Limb view of a cloud top (*TRAJET* = UP). The user specifies cloud top temperature and emissivity, tangent height, and the lower level above tangent height.

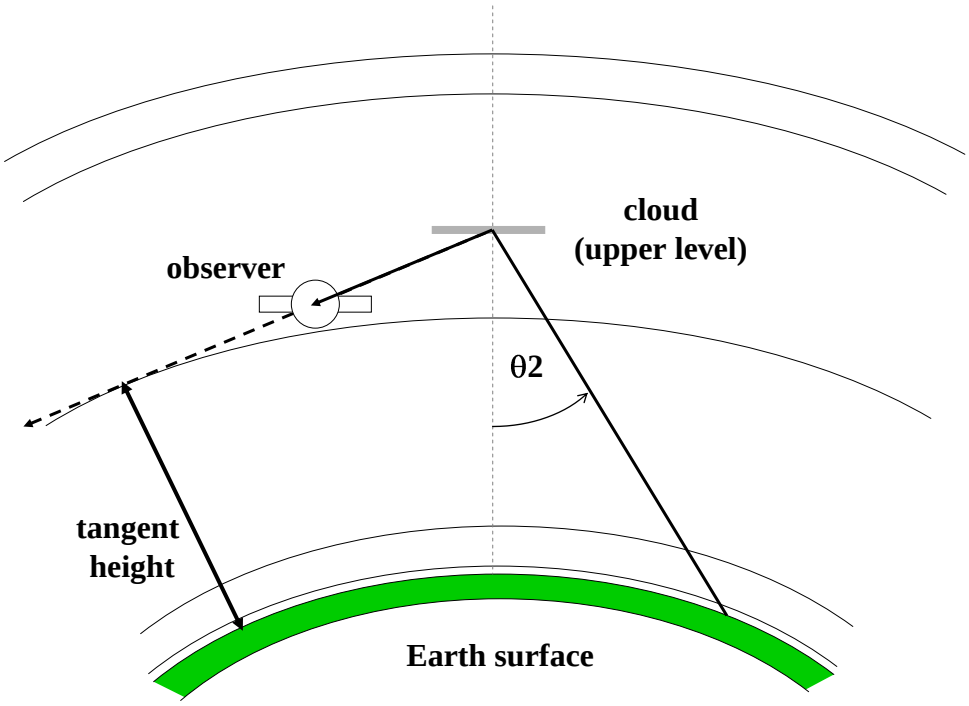


Figure 9: Backward limb view of a cloud bottom, including reflection of upward terrestrial radiation (*TRAJET* = UPDOWN). The user specifies background temperature and emissivity (Earth surface or lower cloud), cloud bottom temperature and emissivity, tangent height, and the upper level above tangent height. θ2 is fixed to 55°. The observer must be specified between upper level and tangent height.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

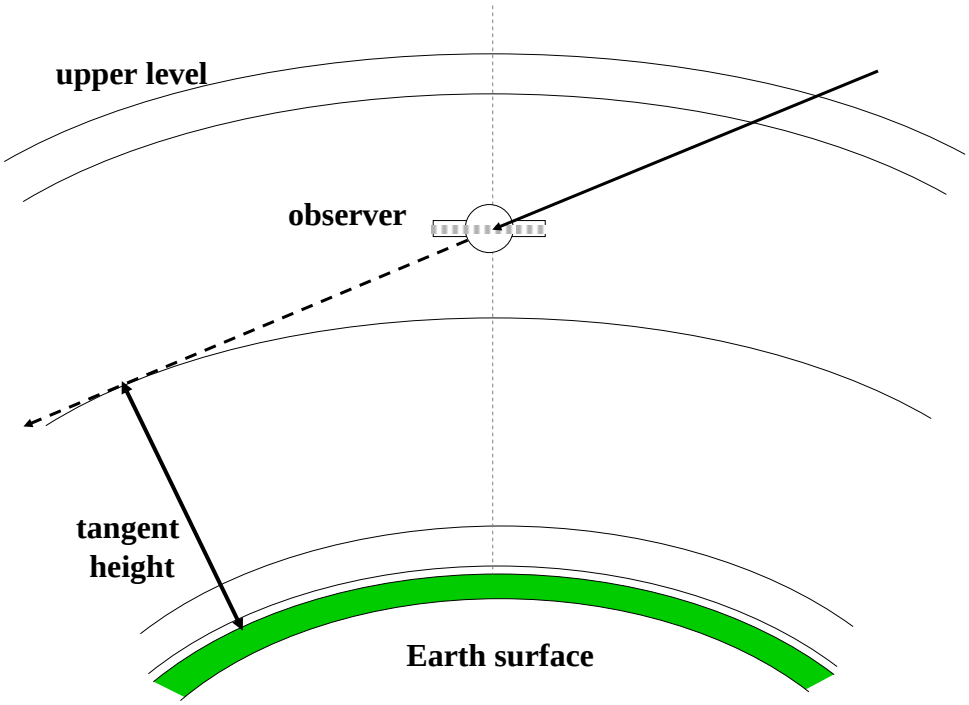


Figure 10: Backward limb view of sky or a cloud bottom (*TRAJET* = DOWN). The user specifies background temperature and emissivity, tangent height, and the observation level above tangent height.

5.2.5.3. Aerosol/ice cloud run characterization parameters

The variable `$RSCA` initialized through the makefile identifies the parameter file `parascat$RSCA.dtp` that contains the aerosol/ice cloud microphysical and optical user parameterisations.

Specifying `$RSCA` to the empty string, 4A/OP is executed in the same way as the reference code by reading the file `parascat.dtp`, delivered with the software package.

The file containing the user defined aerosol/ice cloud parameters (`parascat$RSCA.dtp)` is stored in the directory `input`. An example is displayed below. The file is read by the routine `parascat.f90`. The number of read lines depends on the value of FLAG_SCAT. If equal to 'NO', only the first line is read, if equal to 'SOS', the first eight lines are read, otherwise (FLAG_SCAT equal to 'DIS'), all 16 lines must be specified.

```
' Disort(DIS), SOS(SOS) or no scattering (NO)   FLAG_SCAT'     'DIS'
' Number of aerosol layers (<=Cst_nlaeromax)       NLAERO'     2
' Aerosol model                                   AEROMOD'     'mitr00' 'soot00'
' reference optical thickness at 10 micron         AOTREF'     1.5 0.5
' Vertical level where aerosol is present         LAYAERO'     33 35 38 38
' Single scattering albedo (const)                OMEAERO'     0. 0.
' Asymmetry parameter (const)                       GAERO'     0. 0.
' Number of phase function Legendre coeffs.          NMOM'     10
' Number of DISORT streams (even and >2)          NSTREAM'     10
' DISORT conv. crit. for beam source ]0.d0,0.1d0]   ACCUR'     0.009
' DISORT print flag, input parameters             PRNT(1)'     .FALSE.
' DISORT print flag, fluxes                       PRNT(2)'     .FALSE.
' DISORT print flag, radiances, user polar ang.   PRNT(3)'     .FALSE.
' DISORT print flag, transmittivity/albedo        PRNT(4)'     .FALSE.
' DISORT print flag, phase function moments       PRNT(5)'     .FALSE.
' DISORT header string (<128 characters)        DISHEADER'     '4AOP_DISORT
test'
```


© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks	Default value
flag_scat	Aerosol presence flag	string (len=3)	-	o	calcrad	set to 'NO' in parascat.dtp	NO
nlaero	Number of aerosol layers	int	dl.	o	pre_disort pre_sos	If flag_scat≠ 'NO'	1
aeromod	Identifier of the aerosol model, from top to bottom layer	string(len=6) [Cst_nlaeromax]	-	o	readaerosols	If flag_scat≠ 'NO'	UNDEF
aotref	Aerosol optical thickness at reference wave number 1000cm^{-1}, from top to bottom layer	r [Cst_nlaeromax]	dl.	o	pre_disort pre_sos	If flag_scat≠ 'NO': Initialisation to a spectrally constant value if positive	0
layaero	Vertical atmospheric layers where aerosol is present (upper and lower layer, then from top to bottom)	int[2, Cst_nlaeromax]	-	o	pre_disort pre_sos	If flag_scat≠ 'NO'	1 1
omeaero	Single scattering albedo, from top to bottom layer	r [Cst_nlaeromax]	dl.	o	pre_disort pre_sos	If flag_scat≠ 'NO': Initialisation to a spectrally constant value if positive	0.5
gaero	Asymetry parameter, from top to bottom layer	r [Cst_nlaeromax]	dl.	o	pre_disort pre_sos	If flag_scat≠ 'NO': Initialisation to a spectrally constant value if positive	0.5
nmom	Number of phase function Legendre polynoms	int	dl.	o	pre_disort pre_sos	If flag_scat≠'NO'	10
nstream	Number of streams	int	dl.	o	pre_disort	If flag_scat= 'DIS'	10
accur	DISORT convergence criterion if beam source is present	d	relative radiance unit	o	pre_disort	If flag_scat= 'DIS'	0.009
prnt	Header for prints, embedded in a DISORT banner	logical (kind=4) [5]	-	o	pre_disort	If flag_scat= 'DIS'	FALSE
disheader	Array of DISORT logical print flags	string (len=127)	-	o	pre_disort	If flag_scat= 'DIS'	4AOP_DISORT test

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Warning:
This warning concerns the **use of the RESPEC parameter** (in the `para4a$RSTR.dtp` file) in the case of a run requiring the use of **4AOP with DISORT** (in order to include scattering for aerosol contribution for example). It is recalled that, in script mode, this can be done by setting up the parameter `RSCA` as follows: `RSCA` ≠ ' ' (empty string).

Eventually, the call of DISORT may considerably increase (up to 15 times depending on spectral regions) the computing time with respect to 4AOP in stand alone.

A way to attenuate this slowing down has been to perform a contraction of the optical thicknesses in each layer before calling DISORT. This contraction is left to the choice of the user and is defined by the value of the `RESPEC` parameter. Since this value may impact the quality of the results, the value for this parameter has to be carefully chosen.

For indication only:

- A contraction at 0.02 cm^{-1} (`RESPEC`=0.02) is an acceptable compromise between computing time (about 3 times slower "only" than 4AOP) and accuracy for "window" channels but can lead to errors greater than 5K in channels sounding the upper part of the atmosphere.
- A contraction of 0.002 cm^{-1} (`RESPEC`=0.002) increases the computing time by about 15 times, but provides a good accuracy for all the wavelengths range (error smaller than 0.2K, except in region at 15 µm, where it can reach 0.5K).

As a consequence, and in order to correctly and coherently assess the impact of aerosols with respect to any reference "clear sky" situation, it is recommended to compute this "clear sky" radiance using 4AOP with DISORT. (ie `RSCA` ≠ ' ')

NB: A way to accelerate the computing time in case of the use of 4AOP with DISORT is in progress.

5.3. Output

5.3.1. Output file description

5.3.1.1. Content

Radiative transfer computation considers each middle of the layers *pc(nl)* delimited by the atmospheric vertical levels *p(nl)* and *pl(nl+1)* specified by the user. Output Jacobians are given for these specific layers.

4A/OP produces only one binary file containing all results. An interface program (sections Part II 1.1.4 and 1.1.5) transforms the binary file into ASCII files. Two simulation configurations are possible: "Pseudo-infinite" case and "Instrument" case. The output file names produced are listed in the table (Table 11) below according to the simulation case.

Output data in ASCII files always include radiances and brightness temperature and can also include other quantities (Jacobians and/or transmittances) for an instrument simulation case.

The output unit for Jacobians depends on the user choice (see global variable UNIT in Table 8 of section Part I 5.2.5.1 and section 5.3.3.1 below). The Jacobian unit specified in tables of the following section is the original unit (UNIT=5).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Simulation case	Content	Binary file names In directory outputbin	ASCII file names In directory outputascii
"Pseudo-infinite"	High-resolution radiances	spi4a$(ATMPROF)$(ATM)$(RSTR)$(RSCA)$(INS).ddb	spi4a$(ATMPROF)$(ATM)$(RSTR)$(RSCA)$(INS)b1.plt
"Instrument"	- Convolved radiances	spc4a$(ATMPROF)$(ATM)$(RSTR)$(RSCA)(INS)(CASE).ddb	spc4a$(ATMPROF)$(ATM)$(RSTR)$(RSCA)(INS)(CASE)b1.plt
	Possible additional quantities: - Partial derivatives of the radiance (Jacobians) with respect to the temperature		dtj4a$(ATMPROF)$(ATM)$(RSTR)$(RSCA)(INS)(CASE)b1.plt
	- Jacobians with respect to the molecule mixing ratio - Jacobians with respect to the surface emissivity		dcj4a$(ATMPROF)$(ATM)$(RSTR)$(RSCA)(INS)(CASE)b1.plt
	- Transmittances		dto4a$(ATMPROF)$(ATM)$(RSTR)$(RSCA)(INS)(CASE)b1.plt

Table 11: 4A/OP output file description

5.3.1.2. Format

Only ASCII file format is described (see also section Part II 1.2.8 and 1.2.9 for the description of the source code). The output ASCII files are formatted according to the following characteristics:

- Suffix: `.plt;`
- Access: sequential formatted (ASCII).
- The constant values on the tables below are defined in a constant module `model`. The constants used are:

- Cst_maxcnv = 15000: Maximum number of convolution wave numbers;
- Cst_nlevelmax = 101: Maximum number of pressure levels;
- Cst_ncjacmax = 5: Maximum number of molecules allowed in a spectral domain to calculate their Jacobians.

5.3.1.2.1 High-resolution radiance spectra

These data files are composed of three columns of a variable number of lines. The file name is `spi4a$(ATMPROF)$(ATM)$(RSTR)$(INS)b1.plt`.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Number of the line	Parameter name	Description	Units	Dimension (maximum)	Type	Format
1 to (numax-numin)/pdnu+1	ond (fisrt column)	Wave numbers	cm^{-1}	2**19+1	r	e16.8
	spi (second column)	Radiances	W/ (m^2.strd.cm^{-1})	2**19+1	r	e16.8
	temp (third column)	Brightness temperature	K	2**19+1	r	e16.8

Table 12: format description of 4A/OP output ASCII file for the high-resolution radiance spectra

5.3.1.2.2 Convolved radiance spectra

These data files are composed of three columns of a variable number of lines. The file name is `spc4a$(ATMPROF)$(ATM)$(RSTR)$(INS)$(CASE)b1.plt`.

Number of the line	Parameter name	Description	Units	Dimension (maximum)	Type	Format
1 to (numax-numin)/pascon+1	ond (fisrt column)	Wave numbers	cm^{-1}	Allocated in readconvdata (module spectrumio) Size: number of convolution wave numbers	r	e16.8
	spf (second column)	Radiances	W/ (m^2.strd.cm^{-1})	Allocated in readconvdata (module spectrumio) Size: number of convolution wave numbers	r	e16.8
	temp (third column)	Brightness temperature	K	Allocated in lirespc4a Size: number of convolution wave numbers	r	e16.8

Table 13: format description of 4A/OP output ASCII file for the convolved radiance spectra

5.3.1.2.3 Convolved Jacobians with respect to temperature

These data files are composed of six columns of a variable number of lines. The file name is `dtj4a$(ATMPROF)$(ATM)$(RSTR)$(INS)$(CASE)b1.plt`.

Number of the line	Parameter name	Description	Units	Dimension (maximum)	Type	Format
(i-1)*(level2-level1+1)+i to i*(level2-level1+2)-1,	pc(nl), nl = 1 to level2-level1+1 (first column)	Pressure of the layer between pl(nl) and pl(nl+1)	hPa	Cst_nlevelmax+1	r	e16.8

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

i = 1 to (numax-numin)/pascon +1	pl(nl) (second column)	Pressure of level nl	hPa	Cst_nlevelmax	r	e16.8
	pl(nl+1) (third column)	Pressure of level nl+1	hPa	Cst_nlevelmax	r	e16.8
	ond (fourth column)	Wave numbers	cm^{-1}	Allocated in readconvdata (module spectrumio) Size: number of convolution wave numbers	r	e16.8
	ctjac (fifth column)	Jacobians in temperature	W/ (m^2.strd. cm^{-1})/K	Allocated in readconvdata (module spectrumio) Size: number of convolution wave numbers * (Cst_nlevelmax+1)	r	e16.8
	nl (sixth column)	Level pressure number	-	1	i	i3
i*(level2-level1+2)	Line-jump					

Table 14: format description of 4A/OP output ASCII file for the convolved jacobians in temperature

5.3.1.2.4 Convolved Jacobians with respect to gas mixing ratio

These data files are composed of seven columns of a variable number of lines. The file name is `dcj4a$(ATMPROF)$(ATM)$(RSTR)$(INS)$(CASE)b1.plt`.

Number of the line	Parameter name	Description	Units	Dimension (maximum)	Type	Format
nc*[(i-1)*(level2-level1+1)+i to i*(level2-level1+2)-1], i = 1 to (numax-numin)/pascon +1, nc=1, jselds	pc(nl), nl = 1 to level2-level1+1 (first column)	Pressure of the layer between pl(nl) and pl(nl+1)	hPa	Cst_nlevelmax+1	r	e16.8
	pl(nl) (second column)	Pressure of level nl	hPa	Cst_nlevelmax	r	e16.8
	pl(nl+1) (third column)	Pressure of level nl+1	hPa	Cst_nlevelmax	r	e16.8
	ond (fourth column)	Wave numbers	cm^{-1}	Allocated in readconvdata (module spectrumio) Size: number of convolution wave numbers	r	e16.8

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

	ccjac (fifth column)	Jacobians in gas mixing ratio	W/ (m^2.strd. cm^{-1})/(g/g)	Allocated in readconvdata (module spectrumio) Size: number of convolution wave numbers * (Cst_nlevelmax+1)*Cst_ncjacmax	r	e16.8
	nc (sixth column)	Gas index number	-	1	i	I3
	nl (seventh column)	Level pressure number	-	1	i	i3
i*(level2-level1+2)	Line-jump					
For nc+1 ≠ nc	Line-jump					

Table 15: format description of 4A/OP output ASCII file for the convolved jacobians in gas mixing ratio

5.3.1.2.5 Convolved Jacobians with respect to the surface emissivity

They are contained in the file for the Jacobians with respect to gas mixing ratio `dcj4a$(ATMPROF)$(ATM)$(RSTR)$(INS)$(CASE)b1.plt`. At the surface level (nl=level2), the Jacobians in gas mixing ratio ccjac do not exist. At the surface level, ccjac is set to the Jacobian with respect to the surface emissivity.

5.3.1.2.6 Convolved transmittances

These data files are composed of six columns of a variable number of lines. The file name is `dto4a$(ATMPROF)$(ATM)$(RSTR)$(INS)$(CASE)b1.plt`.

Number of the line	Parameter name	Description	Units	Dimension (maximum)	Type	Format
(i-1)*(level2-level1+1)+i to i*(level2-level1+2)-1, i = 1 to (numax-numin)/pascon+1	pc(nl), nl = 0 to level2-level1 (first column)	Pressure of the layer between pl(nl) and pl(nl+1)	hPa	Cst_nlevelmax+1	r	e16.8
	pl(nl) (second column)	Pressure of level nl	hPa	Cst_nlevelmax	r	e16.8
	pl(nl+1) (third column)	Pressure of level nl+1	hPa	Cst_nlevelmax	r	e16.8
	ond (fourth column)	Wave numbers	cm^{-1}	Allocated in readconvdata (module spectrumio) Size: number of convolution wave numbers	r	e16.8
	cdto (fifth column)	Transmittances	-	Allocated in readconvdata (module spectrumio) Size: number of convolution wave numbers * (Cst_nlevelmax+1)	r	e16.8

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Number of the line	Parameter name	Description	Units	Dimension (maximum)	Type	Format
	nl (sixth column)	Level pressure number	-	1	i	i3
i*(level2-level1+2)	Line-jump					

Table 16: format description of 4A/OP output ASCII file for the convolved transmittances

5.3.2. Examples

The examples provide illustrations of both the input and output files. See the content of each input file for a description of simulation definition parameters (section Part I 5.2.4).

5.3.2.1. Example 1

As a first example, here are the script which performs the required calculations and the corresponding input parameter file (`para4a$RSTR.dtp`) that cause 4A/OP to compute the very high-resolution radiance from 719 to 721 cm^{-1} ("infinite" spectrum):

```
make -f makefile runlirespi4a ATMPROF=0001 ATM=test RSTR=test1 INS=inf
NUMIN=719 NUMAX=721
```

Corresponding input parameter file: `para4atest1.dtp`

```
'AAAA'
' Geometry Identifier (EVIEW/LIMB)                  GEOMID'     'EVIEW'
' Viewing configuration (UP, DOWN, DOWNUP, UPDOWN) TRAJET'       'UP'
' Upper limit of simulation (hPa/km)                 PZUP'        0.05
' Lower limit of simulation (hPa/km)                PZDOWN'     1013.25
' Observation level (hPa if EVIEW/km if LIMB)       PZOBS'        0.05
' Index for the Emissivity of the upper level        EMUP'       100
' Temperature of the upper level                      TUP'        4.
' Index for the Emissivity of the lower level       EMDOWN'       100
' Temperature of the lower level                    TDOWN'        0.0
' Geometric tangent height (km) (LIMB only)         ZGTAN'        0.
' Angle type definition  (ANGLE,SECANT,UNDEF)      TYP_ANG'     'ANGLE'
' Viewing angle (in degrees or secant value)       SECTTA1'        0.
' Reflexion/emission angle (same unit as above)    SECTTA2'       55.
' Solar zenith angle (same unit as above)        SECTTASUN'       90.
' Earth-Sun distance in astronomical units          ESDIST'        1.
' Atmospheric Refraction                             REFRA'      'NO'
' Altitude of lowest atm. pressure level (km)      ZATMBOT'        0.
' Continua (H2O, N2, O2) contribution               CONTIN'     'YES'
' Spectral resolution                               RESPEC'        0.00
' First wave number of the simulation               FORIG'      719.
' Last wave number of the simulation                FSTOP'      721.
' Width of each Atlas spectral band                 DELDS'       15.
' Optical thickness interpolation in temperature   INTOPT'     'YES'
' Indicator of convolution with the ISRF              CONV'      'NO'
' Indicator of hole algorithm for the convolution    HOLE'      'NO'
' Spectral shift introduction due to the ISRF       SHIFT'      'NO'
' Value of the global spectral shift                  RNU'        0.
' Convolution step                                 PASCON'        0.25
' Transmittance calculation flag                    FPOID'      'NO'
' Jacobian calculation flag                         JACOB'      'NO'
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
' Selection indexes for the 40 gases of the GEISA database (1 or 0)   IDSEL'
'   h2o   co2    o3   n2o    co   ch4    o2    no   so2   no2'
    1    , 1   , 1   , 1   , 1   , 1   , 0   , 0   , 0   , 0   ,
'   nh3   ph3  hno3    oh    hf   hcl   hbr    hi   clo   ocs'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  h2co  c2h6  ch3d  c2h2  c2h4  geh4   hcn  c3h8  c2n2  c4h2'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  hc3n  hocl    n2 ch3cl  h2o2   h2s hcooh  cof2   sf6  c3h6'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  ho2  clono2  xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  xxx    xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  cfc11 cfc12 ccl4                                            '
    1    , 1   , 1   ,
' Weighting coefficients of the mixing ratio          ROCOEF'
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
' Selection indexes for the gas Jacobian calculation (1 or 0) IDJAC'
'   h2o   co2    o3   n2o    co   ch4    o2    no   so2   no2'
    1    , 0   , 1   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'   nh3   ph3  hno3    oh    hf   hcl   hbr    hi   clo   ocs'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  h2co  c2h6  ch3d  c2h2  c2h4  geh4   hcn  c3h8  c2n2  c4h2'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  hc3n  hocl    n2 ch3cl  h2o2   h2s hcooh  cof2   sf6  c3h6'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  ho2  clono2  xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  xxx    xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  cfc11 cfc12 ccl4                                            '
    0    , 0   , 0   ,
```

The spi4a0001testtest1infb1.plt file contains the 4A/OP output corresponding to this input. Columns 1 through 3 are: the wave numbers (cm^{-1}), the radiance at the top of the atmosphere ($W/(m^2\ str\ cm^{-1})$) and the corresponding brightness temperature (K). The results for this first example are shown in Figure 11. A good way to verify that 4A/OP is operating correctly on your system is to use your favourite graphics software to read spi4a0001testtest1infb1.plt and compare the results visually.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

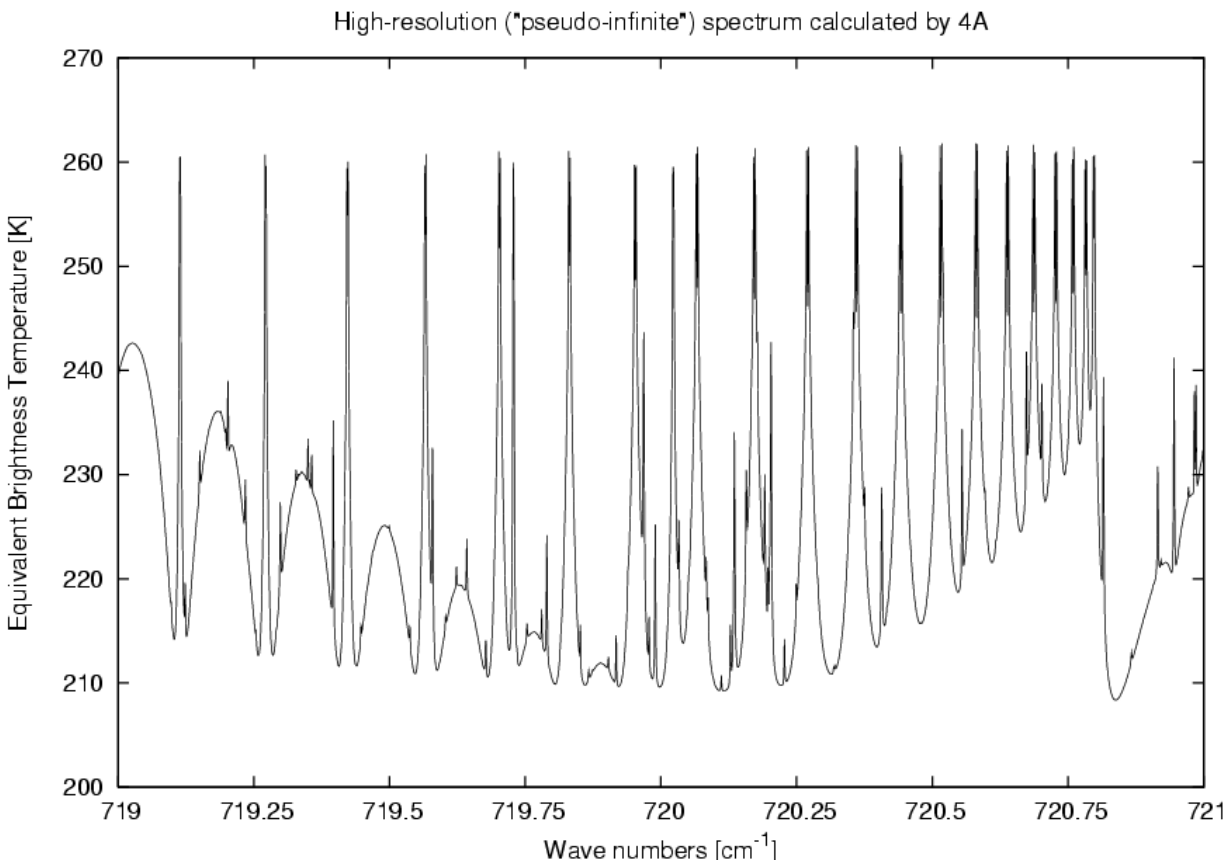


Figure 11: High-resolution radiance computed with 4A from 719 to 721 cm^{-1} in equivalent brightness temperature (test1).

5.3.2.2. Example 2

The second example is the computation of the convolved radiance that would be measured by the instrument IASI, from 645 to 2760 cm^{-1}.

```
make -f makefile runlirespc4a NUMIN=645 NUMAX=2760 RSTR=test8 INS=iasi1c
ATM=test ATMPROF=0001
```

Corresponding input parameter file: `para4atest8.dtp`

```
'AAAA'
' Geometry Identifier (EVIEW/LIMB)                  GEOMID'     'EVIEW'
' Viewing configuration (UP, DOWN, DOWNUP, UPDOWN) TRAJET'       'UP'
' Upper limit of simulation (hPa/km)                 PZUP'        0.05
' Lower limit of simulation (hPa/km)                PZDOWN'     1013.25
' Observation level (hPa if EVIEW/km if LIMB)       PZOBS'        0.05
' Index for the Emissivity of the upper level        EMUP'       100
' Temperature of the upper level                      TUP'        4.
' Index for the Emissivity of the lower level       EMDOWN'       100
' Temperature of the lower level                     TDOWN'        0.0
' Geometric tangent height (km) (LIMB only)          ZGTAN'        0.
' Angle type definition  (ANGLE,SECANT,UNDEF)      TYP_ANG'     'ANGLE'
' Viewing angle (in degrees or secant value)       SECTTA1'        0.
' Reflexion/emission angle (same unit as above)    SECTTA2'       55.
' Solar zenith angle (same unit as above)        SECTTASUN'       90.
' Earth-Sun distance in astronomical units          ESDIST'        1.
```


© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
' Atmospheric Refraction                                REFRA'      'NO'
' Altitude of lowest atm. pressure level (km)      ZATMBOT'        0.
' Continua (H2O, N2, O2) contribution               CONTIN'     'YES'
' Spectral resolution                               RESPEC'        0.00
' First wave number of the simulation                FORIG'      645.
' Last wave number of the simulation                 FSTOP'     2760.
' Width of each Atlas spectral band                  DELDS'       15.
' Optical thickness interpolation in temperature    INTOPT'     'YES'
' Indicator of convolution with the ISRF              CONV'     'YES'
' Indicator of hole algorithm for the convolution     HOLE'      'NO'
' Spectral shift introduction due to the ISRF        SHIFT'      'NO'
' Value of the global spectral shift                   RNU'        0.
' Convolution step                                  PASCON'        0.25
' Transmittance calculation flag                     FPOID'      'NO'
' Jacobian calculation flag                          JACOB'      'NO'
' Selection indexes for the 40 gases of the GEISA database (1 or 0)   IDSEL'
'   h2o   co2    o3   n2o    co   ch4    o2    no   so2   no2'
    1    , 1   , 1   , 1   , 1   , 1   , 1   , 0   , 0   , 0   ,
'   nh3   ph3  hno3    oh    hf   hcl   hbr    hi   clo   ocs'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  h2co  c2h6  ch3d  c2h2  c2h4  geh4   hcn  c3h8  c2n2  c4h2'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  hc3n  hocl    n2 ch3cl  h2o2   h2s hcooh  cof2   sf6  c3h6'
    0    , 0   , 1   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  ho2  clono2  xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  xxx    xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  cfc11 cfc12 ccl4                                            '
    1    , 1   , 1   ,
' Weighting coefficients of the mixing ratio          ROCOEF'
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
    1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 , 1.0 ,
' Selection indexes for the gas Jacobian calculation (1 or 0) IDJAC'
'   h2o   co2    o3   n2o    co   ch4    o2    no   so2   no2'
    1    , 0   , 1   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'   nh3   ph3  hno3    oh    hf   hcl   hbr    hi   clo   ocs'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  h2co  c2h6  ch3d  c2h2  c2h4  geh4   hcn  c3h8  c2n2  c4h2'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  hc3n  hocl    n2 ch3cl  h2o2   h2s hcooh  cof2   sf6  c3h6'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  ho2  clono2  xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  xxx    xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx   xxx'
    0    , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   , 0   ,
'  cfc11 cfc12 ccl4                                            '
    0    , 0   , 0   ,
```

You can compare your results written in the file `spc4a0001testtest8iasi1cb1.plt` to those shown in Figure 12.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

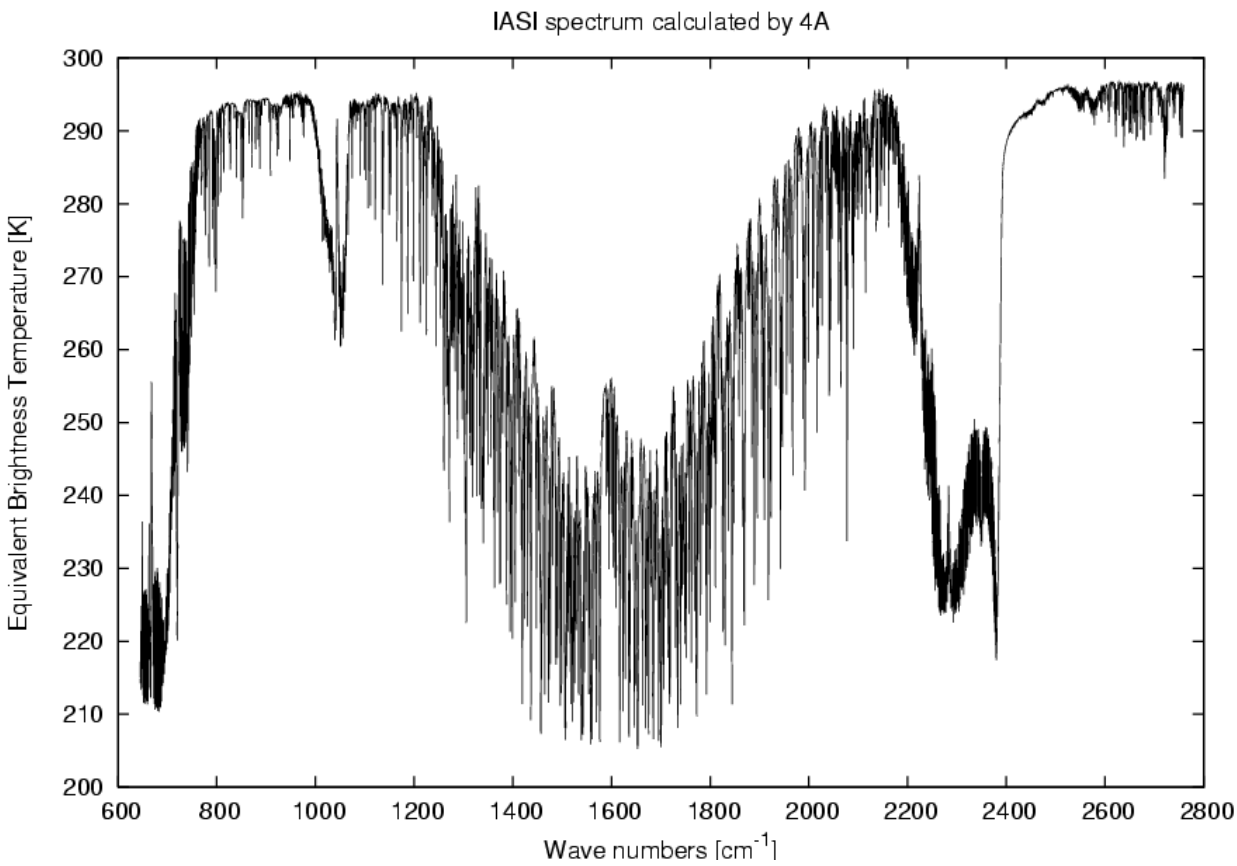


Figure 12: Convolved radiance (in in equivalent brightness temperature) that would be measured by the instrument IASI, from 645 to 2760 cm^{-1} (test8).

5.3.3. Result visualization

Currently ASCII result files are produced and their visualization can be performed with the free Unix tool Gnuplot, or with IDL, or with some other plotting program.

5.3.3.1. Jacobian unit recommendation

The output unit for Jacobians depends on the user choice (see global variable UNIT in Table 8 of section Part I 5.2.5.1 and the subroutine *converjacob* in section 1.2.9.4 of Part II). The output unit in binary file is the computation unit (W/(m^2.strd.cm^{-1})/K/layer or W/(m^2.strd.cm^{-1})/(g/g)/layer): UNIT=5); for ASCII extraction, it can be converted into another unit available in the subroutine *converjacob*. Among the six possible units, we recommend the user to visualize Jacobians with normalized values, that is for UNIT=1: NedT at a reference temperature (usually 280K) in order to be independent of the temperature profile and per kilometre in order to be independent of the radiative transfer discretization. The corresponding unit for each Jacobian type is the following:

- Temperature: Kelvin (Nedt at tref) per Kelvin.Km;
- Gas mixing ratio: Kelvin (Nedt at tref) per Km for 10% variation of the mixing ratio;
- Emissivity: Kelvin (Nedt at tref) for 1% variation.

However, the user is free to visualize Jacobians in any other unit by converting it with his own tool.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

5.3.4. Statistics tool

A statistics tool is available for the computation of the difference between two spectra and the corresponding statistics (see section Part II 1.2.10 for the source code description). It does work for convolved spectra only and both spectra must be sampled with the same sampling step.

As an example, here is a script which performs the required calculations that cause the statistics program to compute the difference between a spectrum to be compared stored in the file `spc4a$(SPC1).ddb` and a reference spectrum stored in the file `spc4a$(SPC2).ddb`:

```
make -f makefile runstatanalys TREF=280 WCLASS=50 WNMIN=645 WNMAX=2760
SPC1=0002test8iasi1c SPC2=0001test8iasi1c
```

See Table 8 for the description of the required input variables.

Two ASCII files are generated and stored in the directory `outputascii`:

- `stat_$(SPC1)_$(SPC2).plt`: difference in radiance and in NedT;
- `classstat_$(SPC1)_$(SPC2).plt`: statistical characteristics of the difference per wave number class with the width WCLASS.

The output format of these files is described below.

5.3.4.1. File of the spectra difference

This file is composed of five columns of a variable number of lines (nbwnScomspect: number of spectra samples). The file name is `stat_$(SPC1)_$(SPC2).plt`.

Number of the line	Parameter name	Description	Units	Dimension (maximum)	Type	Format
1 to nbwnScomspect	wnScomspect (fisrt column)	Wave numbers	cm^{-1}	Allocated in statanalys Size: number of convolution wave numbers	d	e21.12
	RdScomspect (second column)	Spectrum (radiance) to be compared	W/ (m^2.strd.cm^{-1})	Allocated in statanalys Size: number of convolution wave numbers	r	e21.12
	RdSrefspect (third column)	Reference spectrum	W/ (m^2.strd.cm^{-1})	Allocated in statanalys Size: number of convolution wave numbers	r	e21.12
	diff (fourth column)	Normalised difference between both spectra in NedT at TREF	K	Allocated in statanalys Size: number of convolution wave numbers	d	e21.12
	RdScomspect – RdSrefspect (fifth column)	Difference between both spectra	W/ (m^2.strd.cm^{-1})	Allocated in statanalys Size: number of convolution wave numbers	r	e21.12

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Table 17: format description of the first output ASCII file for the spectra difference

5.3.4.2. File of the statistical characteristics per class

This file is composed of six columns of a variable number of lines (nbClass: number of classes calculated according to WCLASS). The file name is `classstat_$(SPC1)_$(SPC2).plt`.

See appendix 4, p294 for description of percentiles.

Number of the line	Parameter name	Description	Units	Dimension (maximum)	Type	Format
1 to nbClass	wnSclass (fisrt column)	Wave numbers corresponding to the classes	cm^{-1}	Allocated in statcharclass Size: number of convolution wave numbers	d	e21.12
	rmsClass(1) (second column)	Root mean square of the difference in NedT per class	K	Allocated in statcharclass Size: number of convolution wave numbers	d	e21.12
	rmsClass(2) (third column)	Standard deviation of the difference in NedT per class	K	Allocated in statcharclass Size: number of convolution wave numbers	d	e21.12
	percentil(1) (fourth column)	Percentiles at 90 %	-	Allocated in statcharclass Size: number of convolution wave numbers	r	e21.12
	percentil(2) (fifth column)	Percentiles at 95 %	-	Allocated in statcharclass Size: number of convolution wave numbers	r	e21.12
	percentil(3) (sixth column)	Percentiles at 100 %	-	Allocated in statcharclass Size: number of convolution wave numbers	r	e21.12

Table 18: format description of the second output ASCII file for the statistical characteristics per class of the spectra difference

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

6. Graphical User Interface (GUI)

A Graphical User Interface (GUI) for 4A/OP is available. The GUI allows the user to create a basic 4A/OP input file by selecting values with buttons, pull-down menus, and text fields. It should be particularly useful to the new user.

The GUI aims at wrapping the existing software (and the process management) in order to avoid source modifications. The dependency between parameters are kept and are implemented as a black box for the user. The GUI guides the user's choices according to the parameter dependencies: some entries (labels) are disabled if not required and warning messages appear if the user enters no compatible values with his experiment definition.

The programming language used to implement the GUI is Tcl/Tk (Tool command language / Tool kit), version 8.4 at least, and it uses the library BWidget.

6.1. Produce or open a GUI parameter file

A parameter file with a format specific to the GUI is created in order to store all the parameters required to run the 4A model from the GUI. This file allows to save the experiment state chosen by the user. This parameter file can be saved every new experiment (menu file access: Save or Save as) into a file named `filename.4a`; the character string `filename` is suggested by the user if the file is saved with the menu file access "Save" or "Save as". Available GUI parameter files can be open (menu "File / Open") for loading parameters that have been already used and recorded. This kind of parameter file can be saved in the directory `model/guiparam/`.

6.2. Produce a script file

The GUI produces a script file to run 4A from the user's choices presented below. The parameter file `para4a$RSTR.dtp` (see section 5.2.5.2) attached to the run experiment is created and remains hidden for the user. The variable `$RSTR` is equal to the character string `filename`. For example, if one creates a set of parameters that is saved under the name `test.4a`, the GUI creates the required 4A/OP parameter file called `input/para4atest.dtp` and an other file called `scripts/test` in the model installation directory (set with menu File/Preferences). Then `scripts/test` is executed.

It is important to notice that the parameter DELDS has been fixed to 15 cm^{-1} and the parameter SHIFT has been set to "NO": the user can not modify them.

6.3. User's choices

The user's choices are presented through a notebook that displays several pages for each type of parameters. Examples of pages are described by the following screenshots.

- Menu description
  - Menu File: New, Open, Save, Save as, Preferences, Exit;
    - Preferences: This functionality indicates the locations of both the 4A model sources and Gnuplot scripts. This allows the user the possibility to specify another location of the model sources (for the recording of the parameter, script and result files) or/and of the Gnuplot scripts (in order to enable the user to use its own scripts):
  - Menu Run: Run 4A/OP, View output log;

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Run 4A/OP: It launches the 4A/OP model execution. This action requires that the current set of parameters has been saved on disk,
- View output log: This functionality allows the visualization of the standard output and standard error of the run in progress ; that can be saved,

- Menu Visualization: This menu allows visualization (and then printing) of graphs of 4A/OP model products (spectra, jacobians, transmittances, atmospheric profiles and difference between two spectra). Those graphs are governed by Gnuplot scripts;
- Menu Help: Help on 4A interactive, Credits:
  - Help on 4A interactive: It is the on-line help of the 4A GUI.

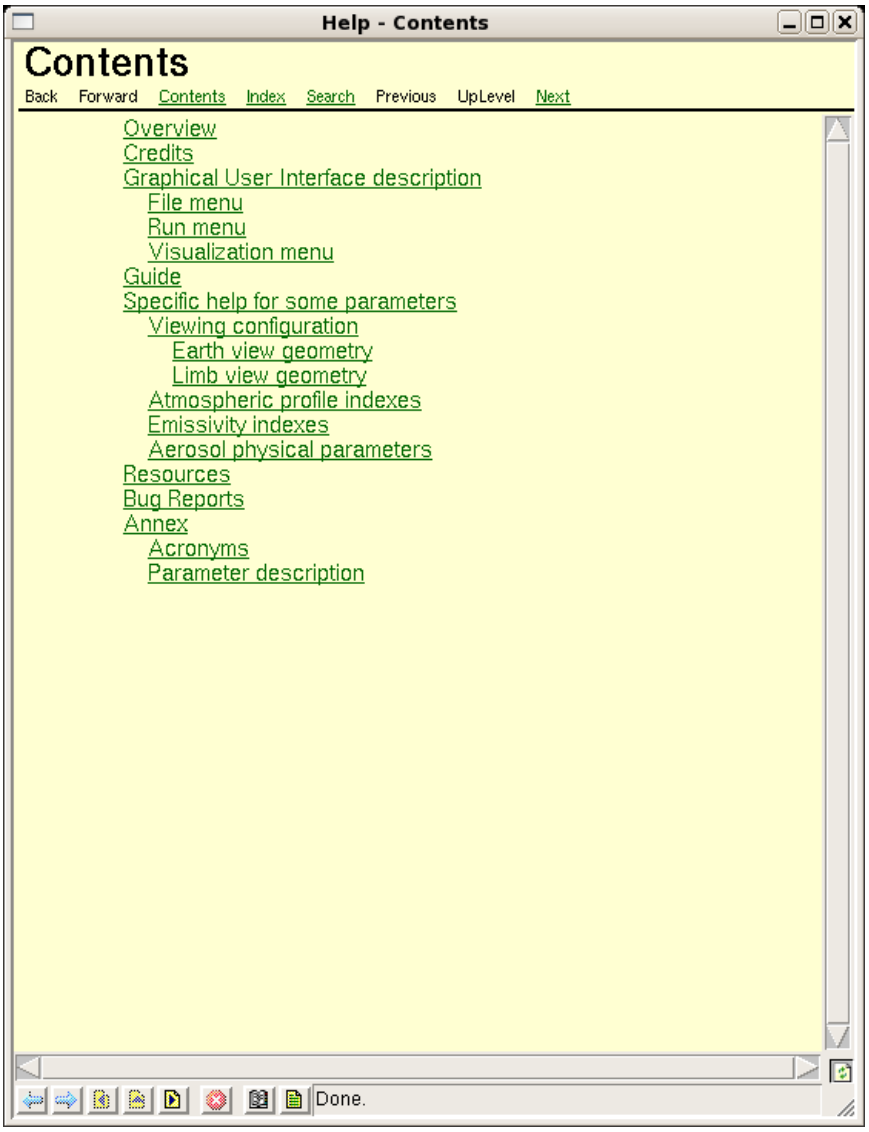


Figure 13: Screenshot of on-line help page that appears by choosing the menu "Help/Help on 4A/OP interactive".

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Page "General"

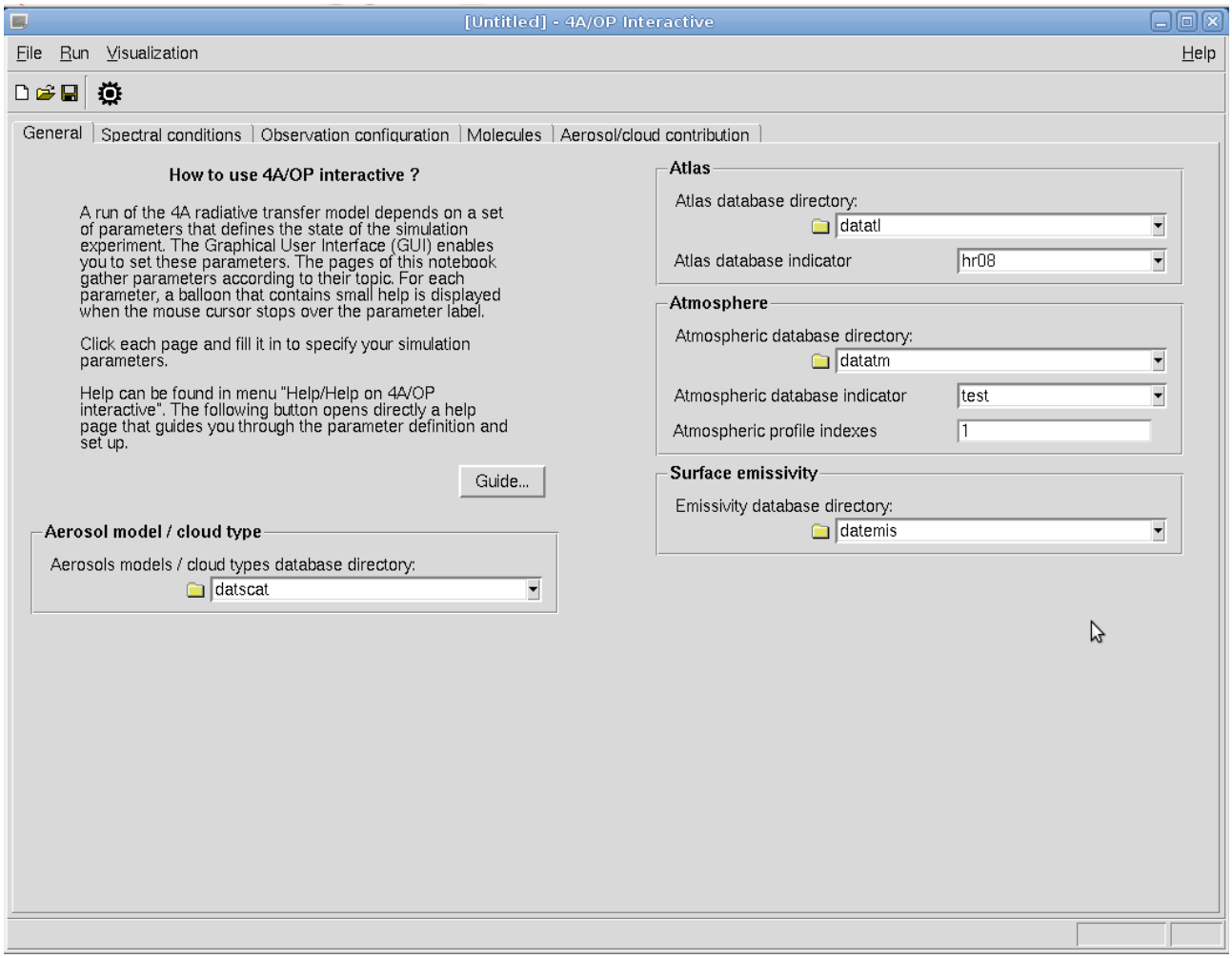


Figure 14: Screenshot of the GUI first page that shows the general parameters required for a 4A run.

- Additional help on parameters:

  Additional help is available for every parameter and is displayed when the mouse pointer stops over the parameter label (a frame appears);

- Section "Atmosphere":

  The user is able to enter several atmospheric profile indexes if several profiles are available in the atmospheric database file chosen by the user. Then the model runs with a loop on several profiles;

- Guide:

  By clicking the button "Guide" available in the page "General", a help/guide page appears. It explains step by step the use of the GUI from the parameter definition to the 4A/OP model output visualization.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

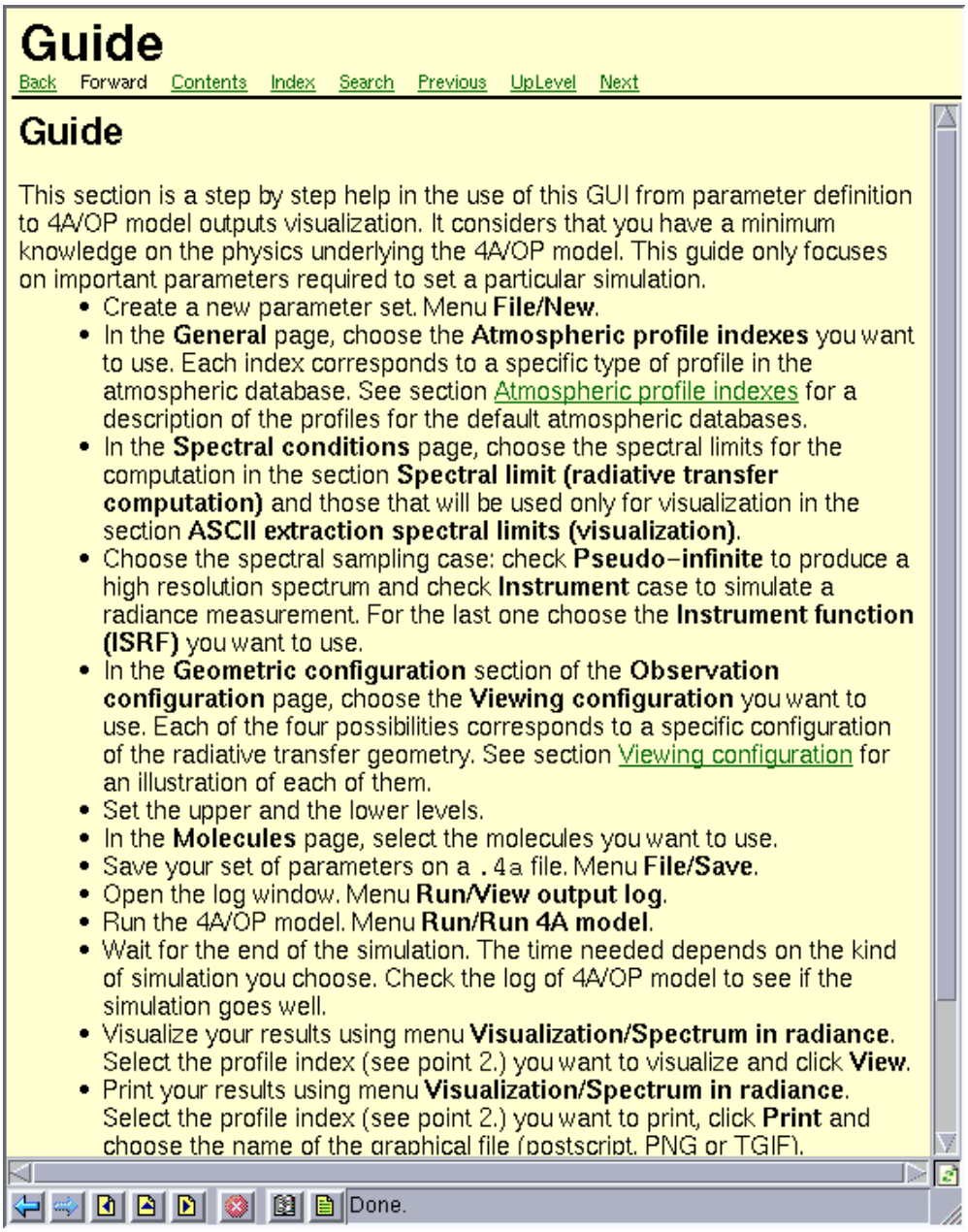


Figure 15: Screenshot of the GUI guide page that appears by clicking the button "Guide".

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Page "Spectral conditions"

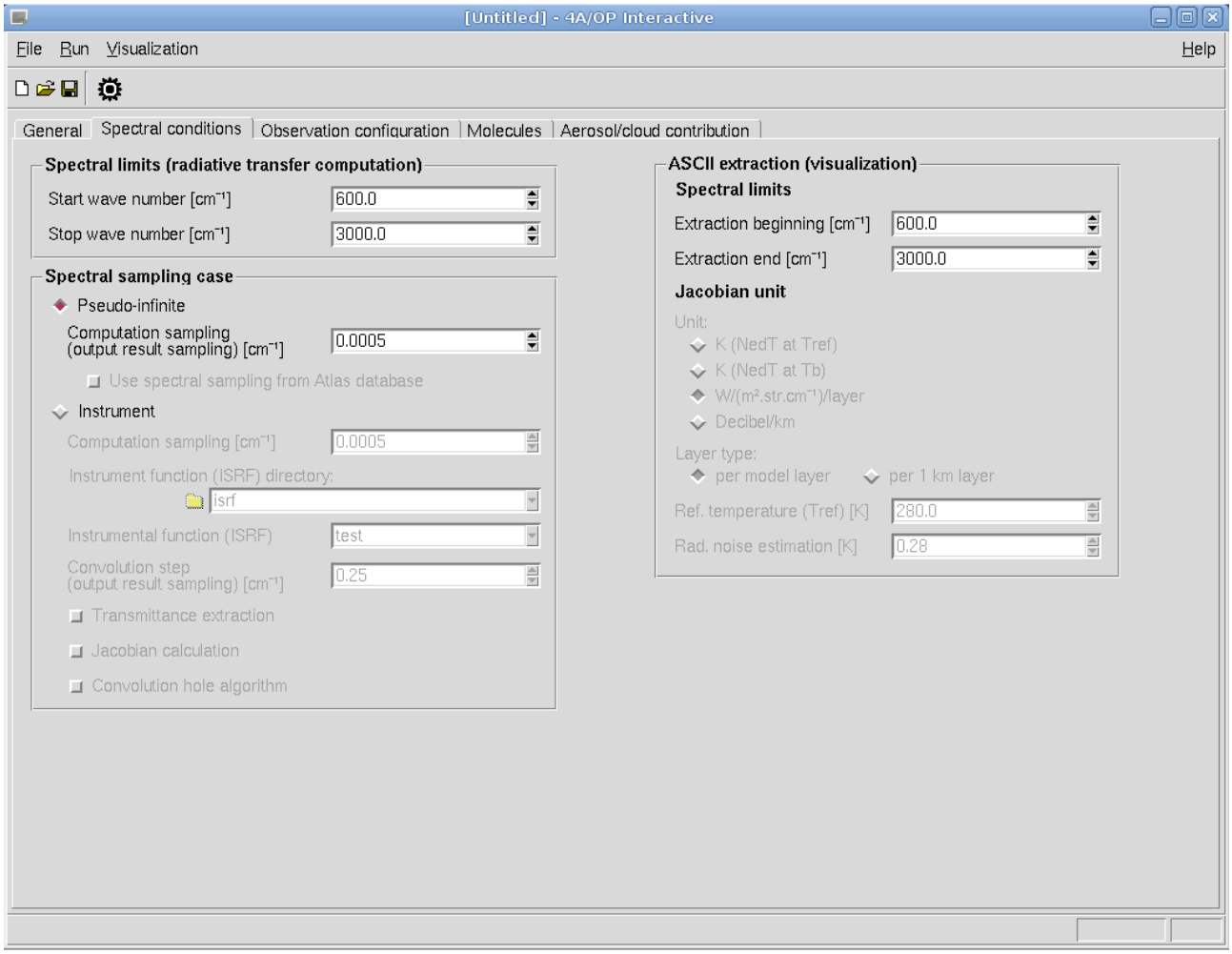


Figure 16: Screenshot of the page concerning the spectral condition parameters for the simulation.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Page "Observation configuration"

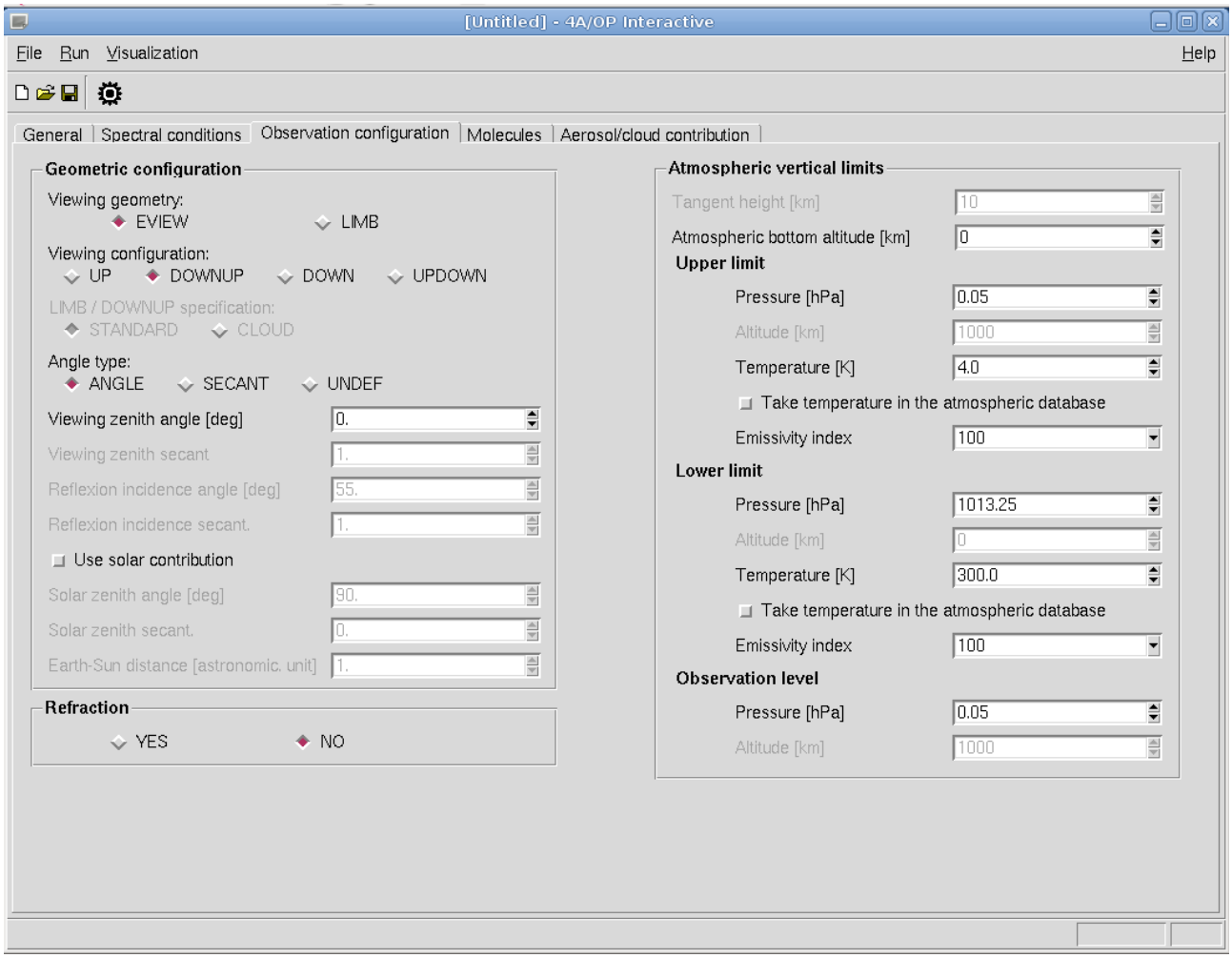


Figure 17: Screenshot of the page concerning the observation configuration parameters.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Page "Molecules "

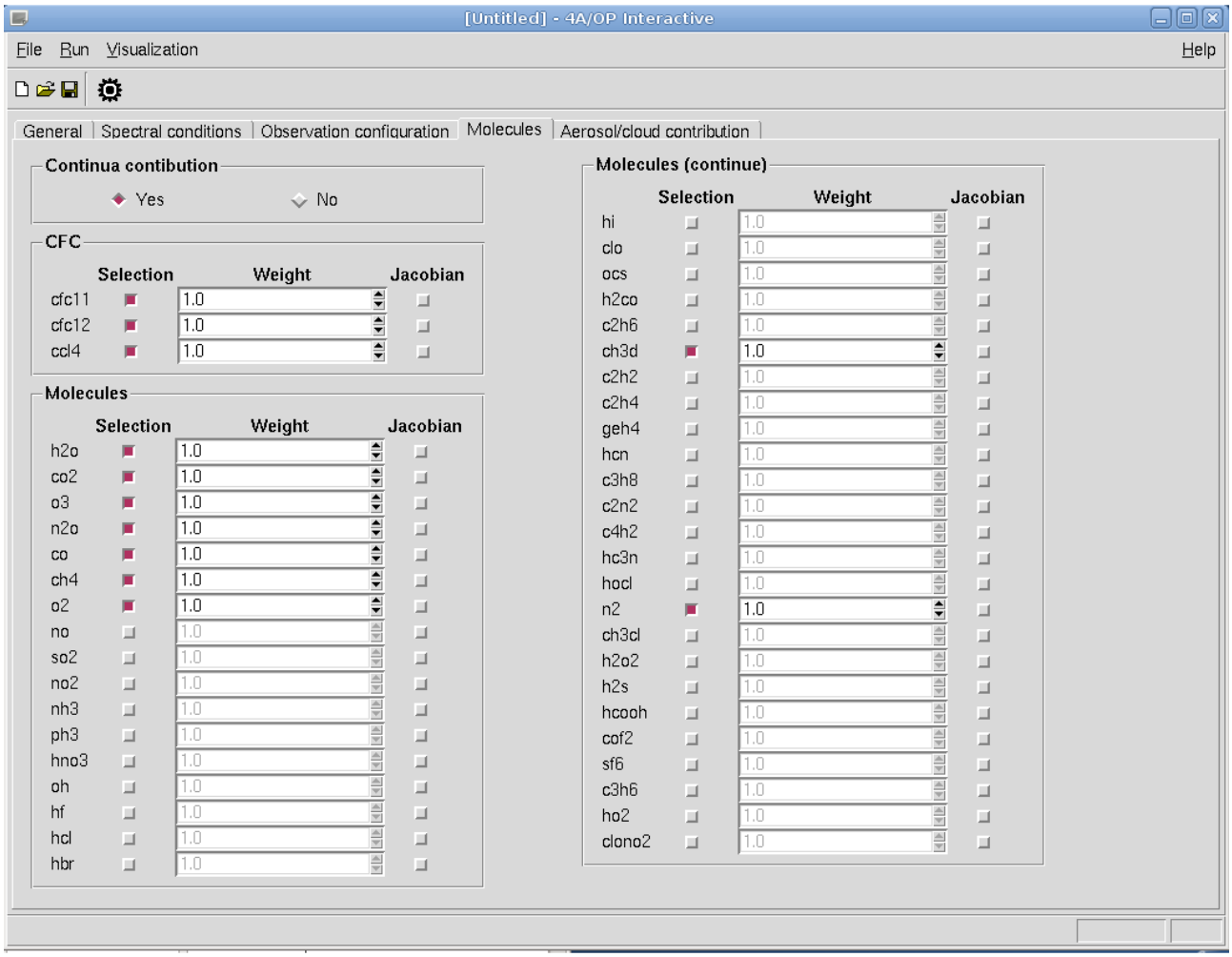


Figure 18: Screenshot of the page concerning the molecule parameters.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Page "Aerosol/cloud contribution"

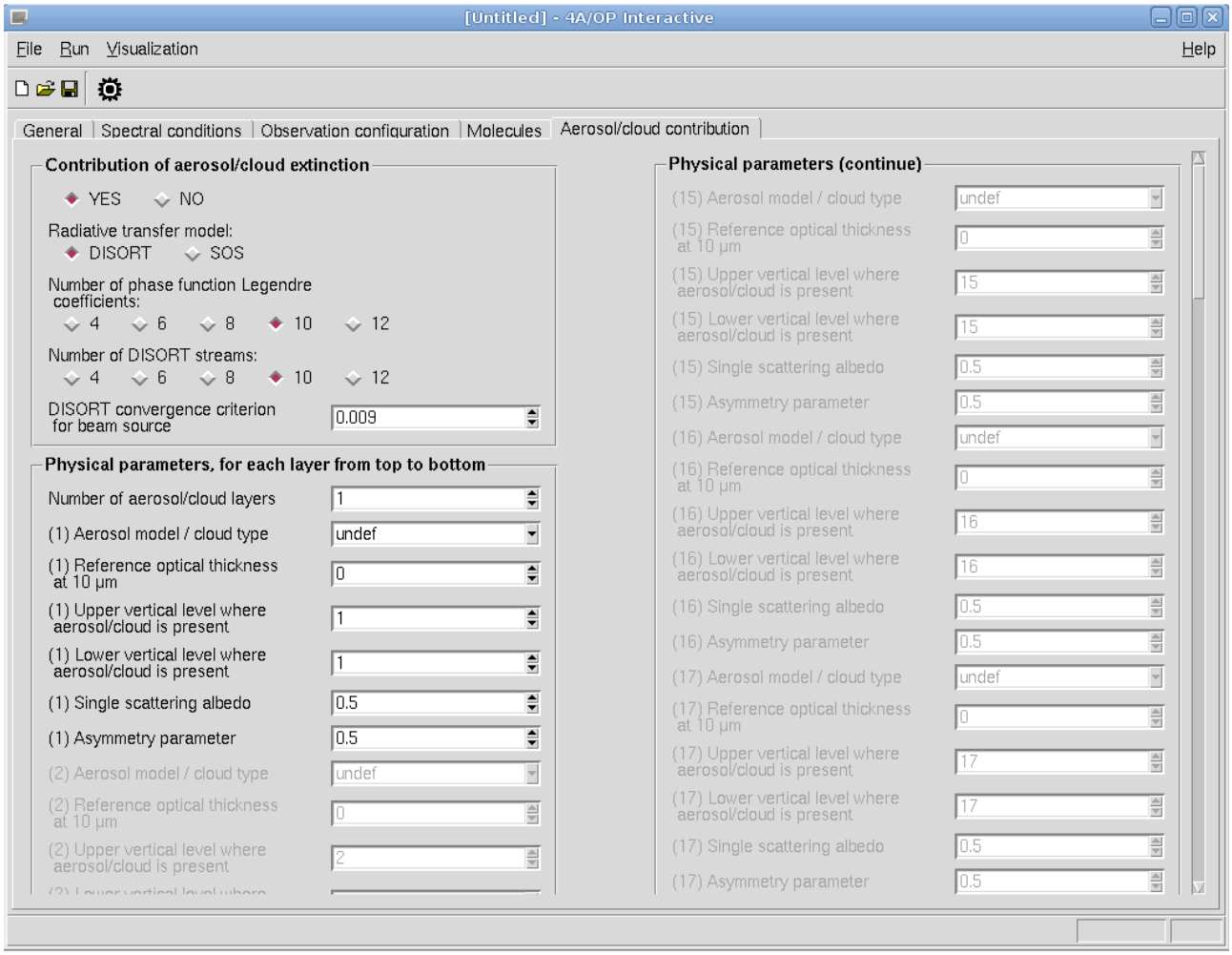


Figure 19: Screenshot of the page concerning the aerosol/cloud contribution parameters.

6.4. Save

The GUI enables the user to save the experiment parameters in order to use it again with the menu "File / Save as" (in the file `$filename.4a`: see section 6.1).

6.5. Run

The entry "Run 4A/OP" in the menu "Run" creates a run script and the parameter file `para4a$filename.dtp`. Then, the model runs.

The user has the possibility to monitor the run development by watching the log window that can be open and closed at any time with the menu "Run / View output log". The model runs in background then the GUI remains active and the user can move all over the GUI. The user is able to stop the run at any time thanks to an aside "Stop" button.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Figure 20: Screenshot of the log window showing a run development example. This window appears by choosing the menu entry "Run / View output log".

6.6. Visualization

The visualization displays Gnuplot graphs of results that are the spectra, Jacobians (if they are calculated) and transmittances (if they are extracted). The "Print" entry (figure below) appears directly from the graph and allows to save it in a file (postscript, png or gif file).

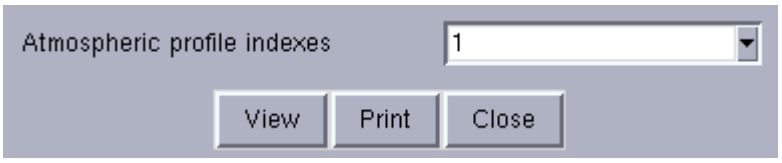


Figure 21: Screenshot of the window that appears by choosing the menu "Visualization / Spectrum in ...".

The visualization can also display Gnuplot graphs of used atmospheric profiles and the difference between two spectra.

6.6.1. Zoom

The user can select a new visualization range by zooming on the graph with the right mouse button. To go back to the previous range, the user can simply type "p" on the Gnuplot window.

6.6.2. User specific plots

You can personalize the graphs produced by the GUI. To do that:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1. Copy the default Gnuplot scripts which are installed in the directory `gnuplot` of the GUI installation. Put them on your own directory. Be sure that these scripts keep their original names;
2. Edit these scripts to fit your needs using standard Gnuplot commands;
3. Modify the field "Gnuplot scripts directory" in the "Preferences" window (menu "File/Preferences") to point at the directory of your Gnuplot scripts.

Original Gnuplot scripts contain some special codes used by the GUI which are names of variables that are substituted when the graph is actually plotted. They begin with the sign `$` (dollar). For example, `$dataFile` is replaced by the name of the 4A/OP output file to plot.

6.6.3. Examples

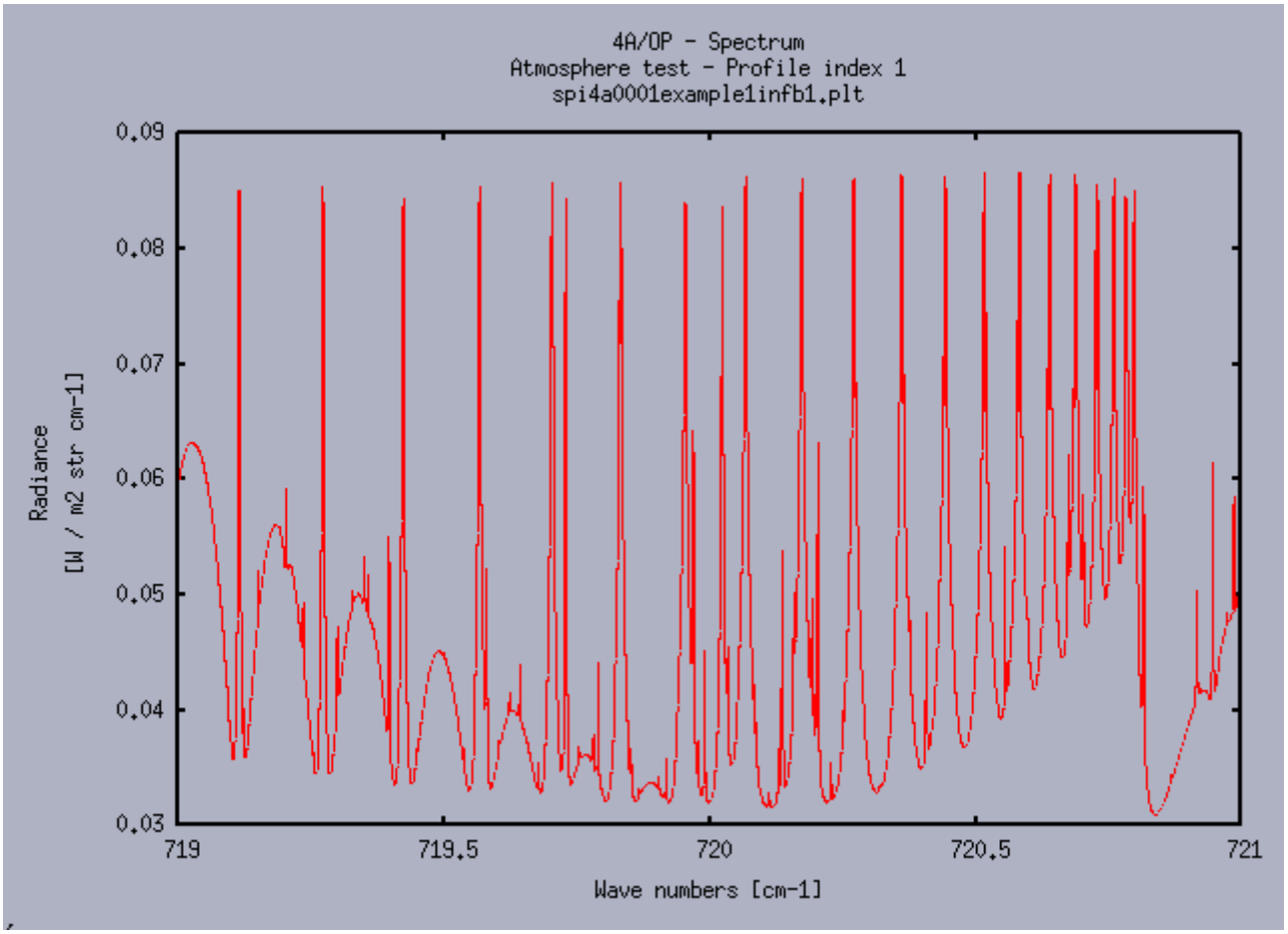


Figure 22: Screenshot of the Gnuplot window showing an example of a spectrum resulting from the current simulation. This window appears by choosing the menu entry "Visualization / Spectrum in radiance". The name of the ASCII file (*.plt) storing the spectrum is below the title "4A/OP – Spectrum".

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

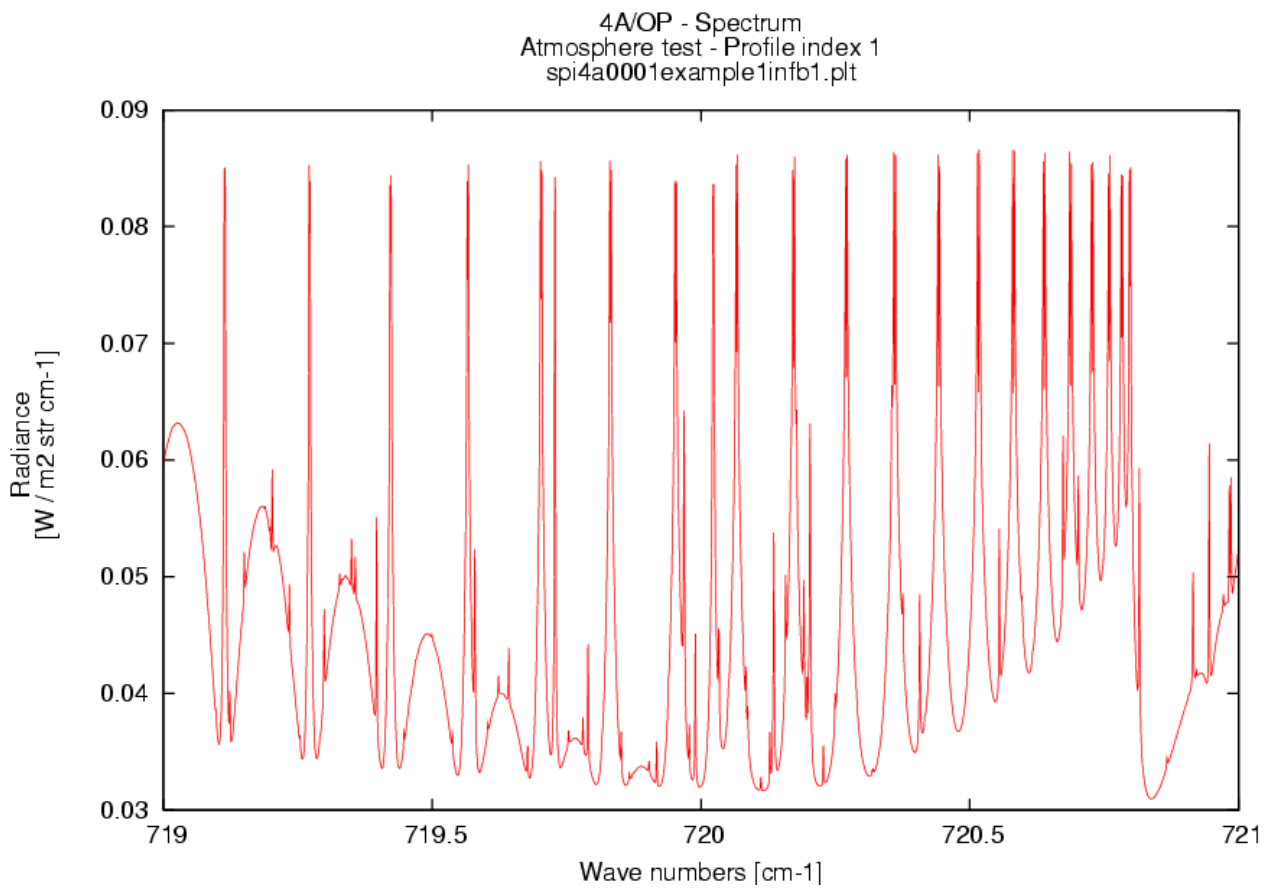


Figure 23: Display of the postscript file illustrating the spectrum of Figure 22. The postscript file is created by clicking the button "Print" in the window associated to the graph (see Figure 21).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

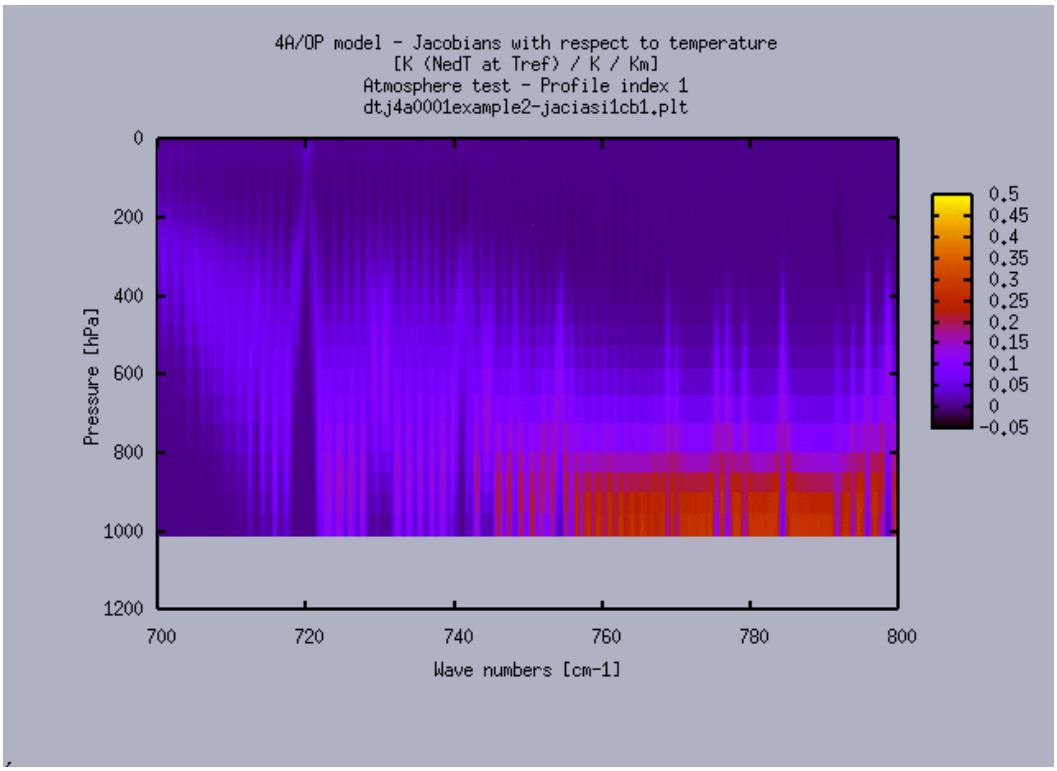


Figure 24: Screenshot of the Gnuplot window showing an example of Jacobians with respect to the temperature. This Gnuplot window appears by choosing the menu entry "Visualization / Jacobians (temperature)".

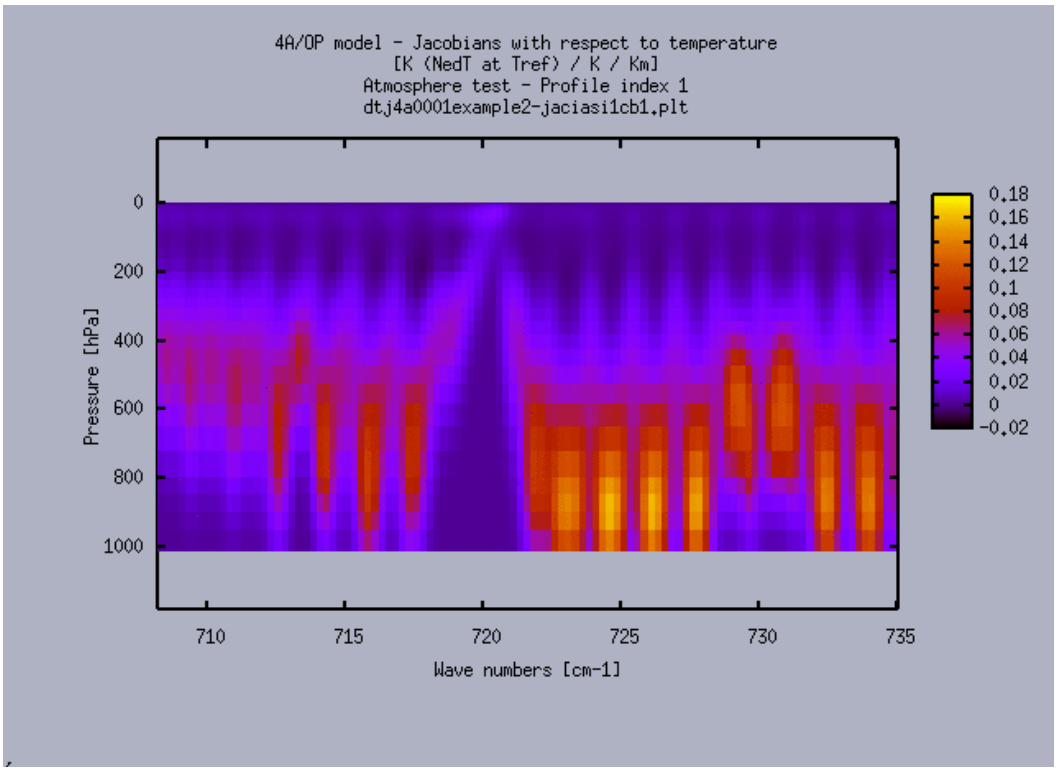


Figure 25: Same as Figure 24 for a selected range obtained by zooming on the graph with the right mouse button.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

6.7. Future improvements

6.7.1. Run

An extra useful functionality is the possibility to run the GUI from the shell with a parameter file, by using the following instruction for example:

```
% 4Agui filename.4a &
```

A practical option could be added in order to enable the user to run 4A with a loop on one or several parameters other than the atmospheric profiles.

6.7.2. Visualization

More functionalities can be added in the menu visualization access:

- View instrumental functions: in order to visualize the ISRF used in the simulation if it is the case of a convolved spectrum;
- View Jacobian for a 1D section of the map at a constant vertical level;
- View statistics;
- A succinct on-line help for Gnuplot.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

7. Revision History

This is update notice for users of 4A/OP. It shows major revisions. Minor revisions are not listed. The following are descriptions of significant changes contained in 4AOP2012v1.0.

2010-2012

- Added user option to simulate scattering effects of clouds (cirrus, …)
- New atlases of pre-calculated optical thicknesses that include:
    - an improvement for effects of CO_2 line-mixing,
    - the new GEISA 2009 spectroscopy,
    - the pressure shift for H_2O, CO_2 and N_2O,
    - updated reference gas mixing ratio profiles,
    - improved TIPS' formulation;
- Minor bug fixes in a few routines of the model;
- Updated and improved Graphical User Interface.

2007-2009

- Implementation of the limb viewing geometry:
    - Refined geometry module,
    - Option for limb view simulation,
    - Height dependence of gravity acceleration;
- Implementation of the user option to simulate scattering effects (coupling with DISORT and SOS codes). Option is active only in Earth view geometry and for aerosols;
- New atlases of pre-calculated optical thicknesses:
    - that make the code more precise,
    - that include:
        - an updated line-mixing for CO_2,
        - the new GEISA 2008 spectroscopy,
        - the pressure shift for H_2O,
        - updated gas mixing ratio profiles,
        - improved CFC's formulation;
- Minor bug fixes in a few routines of the model;
- Updated and improved Graphical User Interface.

2005-2006

- Optical thicknesses of CFC molecules are now stored in Atlases (then the corresponding explicit computation has been removed from the 4A/OP code). This modification allows users to use their own CFC mixing ratio profiles and to calculate the corresponding Jacobians;
- Implementation of user option for simulation of the solar contribution to spectral radiance and Jacobians;

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Replacement of optical thickness atlases computed with GEISA 97 by atlases computed with up-to-date spectroscopy from the GEISA 2003 Edition;
- Improved capabilities of the updated Graphical User Interface;
- Miscellaneous minor corrections.

2003-2004

- Minor bug fixes in a few routines of the model;
- Updated and improved Graphical User Interface.

December 2002

- Added various possible units for Jacobians in output ASCII files (see Table 8, in section Part I 5.2.5.1).

November 2002

- Added hole convolution option for computer time saving in some particular cases. This option speeds up convolution computing when Jacobian calculation is selected. It slightly slows it when convolution is done with a spectrum only. Note that results computed with this option are physically the same as those computed without it, even if there is a numerical difference between them;
- Added statistics tool for the computation of the difference between two spectra and the corresponding statistics (see section Part I 1.2.10).

October 2002

- Added user-defined spectral emissivity functions;
- The plan-parallel atmosphere assumption has been replaced by a spherical atmosphere approximation.

September 2002

- The original code 4A2000 written in Fortran 77 developed at LMD has been ported in Fortran 90;
- The wave numbers initially in single precision are now in double precision.

A detailed description of the revision is available in the appendices Part III 1.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Part II. Reference Manual

1. Algorithm description

This part presents the structural design of the software sources by displaying first the program dependencies and then the detailed description of each module.

1.1. Description of the programs and dependencies

The previous main source dependency tree (Diagram 2) describes the dependency between main processes of the software 4a. The following dependency trees list all the routines used by the software. Each main process is described by a section presenting the corresponding dependency tree. Each program is referenced by its description section in brackets. A routine calls directly all routines to which it is connected by a solid line in the column immediately to its right; it calls indirectly the connected routines in all columns to its right. The smaller, slanted font is used for the second and subsequent occurrences of a routine in a single dependency tree. The character string "s" means "subprogram", the character string "f" means "function" and the character string "ms" means "module subprogram". For subprograms (or functions), the name of the corresponding directory (if different from the main program one) is specified in parentheses. For module subprograms, the name of the corresponding module is specified in parentheses. The number of calls for a subprogram is indicated in square brackets.

Convention for the dependency tree illustrated below:

`s`: subprogram

`f`: function

`ms`: module subprogram

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.1.1. Routines used for the radiative transfer computation

```
aaaa
├── para4a: s
├── parascat: s
├── atmref: s
├── gascon: s
├── lireemis: s
├── readaerosols: s
├── convini: s
│   │
│   └── dichotomd: s [2] (scslib/sutil)
├── atmsph: s
│   │
│   └── inttab: s (scslib/smath)
├── calcsphtraj: s
│   │
│   ├── inttab: s [2]
│   └── dichotomd: s [6] (scslib/sutil)
├── init_contin: s
├── peicom: s [3]
├── liread: s
│   │
│   ├── readatlas_header: ms (modatlas)
│   └── readatlas: ms (modatlas)
├── calcsig: s
│   │
│   ├── dichotom: s
│   ├── liread: s [2]
│   │   │
│   │   ├── readatlas_header: ms
│   │   └── readatlas: ms
│   ├── inttab2: s [5]
│   └── continua: s
│       │
│       ├── contN2: f [2]
│       └── contO2: f [2]
├── calcrad: s
│   │
│   ├── plkdir: s [5] (scslib/sphys)
│   ├── inttab2: s [11]
│   └── plkder: s [2] (scslib/sphys)
├── contract/contract_sig: s
│   │
│   ├── inttab2: s [4]
│   ├── echantini: s
│   └── echantillon: s [4]
├── inttab2: s [5]
├── plkdir: s
├── pre_disort: s
│   │
│   └── disort: s (EXTERNAL)
├── pre_sos: s
│   │
│   ├── plkdir: s [2]
│   └── diffnd: s (EXTERNAL)
├── cnvspe/cnvspehole: ms (convolution)
│   │
│   ├── dichotomd: s [3]
│   └── calcisrf: s
│       │
│       ├── dichotomd: s
│       └── lisreisrf: s [2]
│           │
│           └── intspline: s (scslib/math)
└── Err_notifyError, Err_outputErrorMessage: ms (error)
```

Diagram 3: Dependency tree for the radiative transfer computation (see section 1.2.1)

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.1.2. Routines used to format the atmospheric profiles

```
atmformat
│
└─ lecatmdsf: s
```

Diagram 4: Dependency tree for the atmospheric profile formatting (see section 1.2.6)

1.1.3. Routines used to format the instrument functions

```
ifct
│
└─ lecifctdsf: s (ifct$(INS))
     │
     └─ inttab: s (scslib/smath)
```

Diagram 5: Dependency tree for the instrument function formatting (see section 1.2.7)

1.1.4. Routines used for the formatted writing of the 4A outputs concerning high-resolution spectra

```
lirespi4a
│
└─ tbrillans: f (scslib/sphys)
```

Diagram 6: Dependency tree for the formatted writing of the 4A output high-resolution spectra (see §1.2.8)

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.1.5. Routines used for the formatted writing of the 4A outputs concerning convolved spectra and Jacobians

```
lirespc4a
│
├── readconvdata: ms (spectrumio)
│      │
│      └── Err_notifyError: ms (error)
├── tbrillans: f (scslib/sphys)
├── converjacob: s
│    │
│    ├── plkder: s [2] (scslib/sphys)
│    └── Err_notifyError: ms (error)
└── Err_outputErrorMessage: ms (error)
```

Diagram 7: Dependency tree for the formatted writing of the 4A output convolved spectra and Jacobians (§1.2.9)

1.1.6. Routines used for the computation of the difference between two spectra and the corresponding statistics

```
statanalys
│
├── readconvsp: ms [2] (spectrumio)
│      │
│      └── readconvdata: ms (spectrumio)
│             │
│             └── Err_notifyError: ms (error)
├── plkderdble: s (scslib/sphys)
├── dichotomd: s [2] (scslib/sutil)
├── statchar : ms (spstats)
└── statcharClass : ms (spstats)
       │
       ├── dichotomd: s [2] (scslib/sutil)
       └── inttab2: s (scslib/smath)
```

Diagram 8: Dependency tree for the statistics tool (see section 1.2.10)

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2. Component design specifications

The code has been developed in Fortran77 under the UNIX system. Now it is ported in Fortran90.
In the sections below, the functionality of the radiative transfer computation is specified in detail. Each module presented in Diagram 3 is described. For each algorithmic module, the specification includes:

- Type: logical and physical characteristics;
- Function;
- Subordinates: used subsidiary functions;
- Dependencies: defined by listing constraints placed upon the component use by other components or/and the name of the upon components;
- Interfaces: table of variables (arguments in the case of sub-programs);
- Processing;
- Variable classification. The variables are classified as:
    - Double (same variable but with a different symbol),
    - Parameter,
    - Constant.

In the variable tables, the "Type" indication follows the notation described below:

int*2	Integer (2bytes)
int	Integer (4bytes)
r	Single precision real (4bytes)
d	Double precision real (8bytes)
l	Logical (1bytes)
char string	Character (1bytes) Character string

In the variable tables, the "I/O" (Input/Output) indication follows the notation described below:

i	Input to the module
i/o	Input and output to the module
o	Output to the module

In the variable tables, the "Source/Destinat°" indication follows the notation described below:

Ini	Initialisation file (read via makefile)

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1. Radiative transfer computation with 4AOP

1.2.1.1. Flow chart

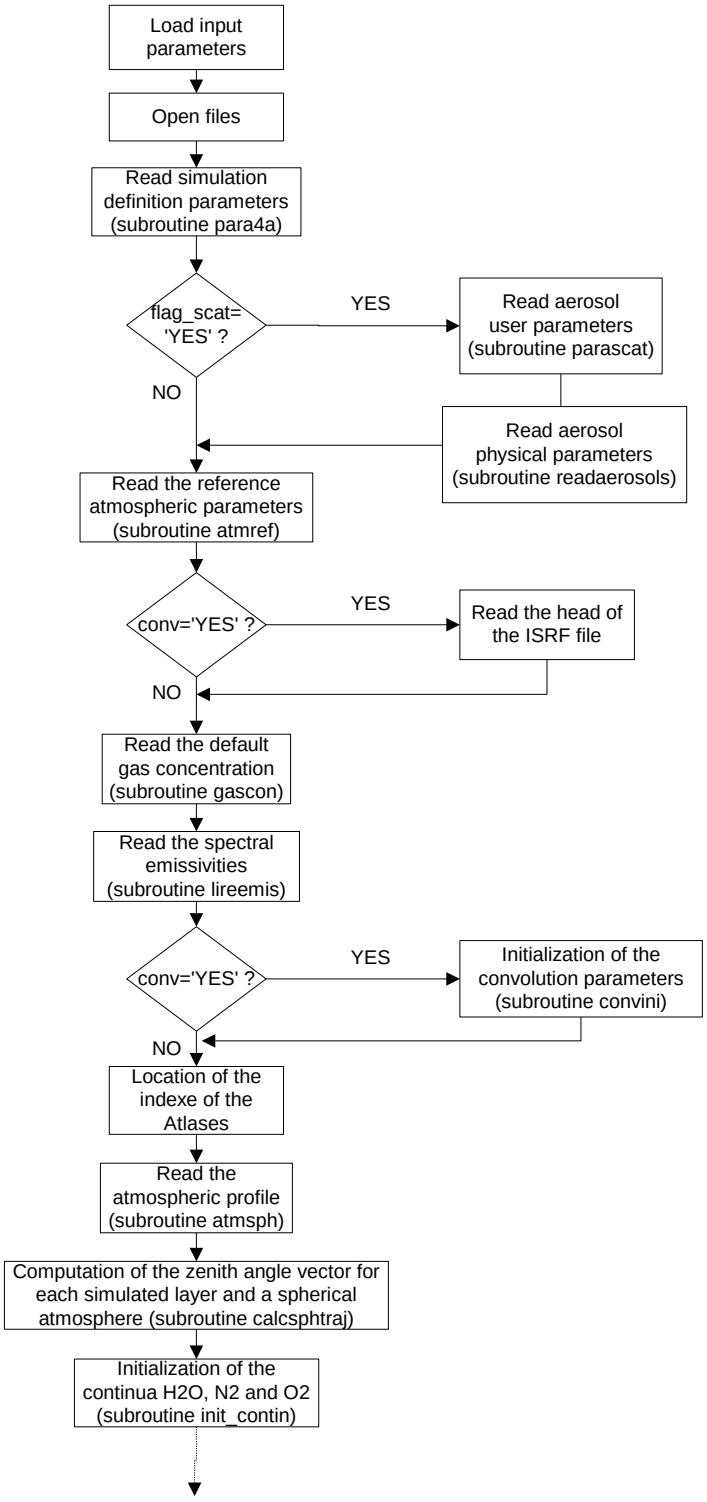


Diagram 9: First part of the flow chart of the main program for the radiative transfer computation aaaa

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

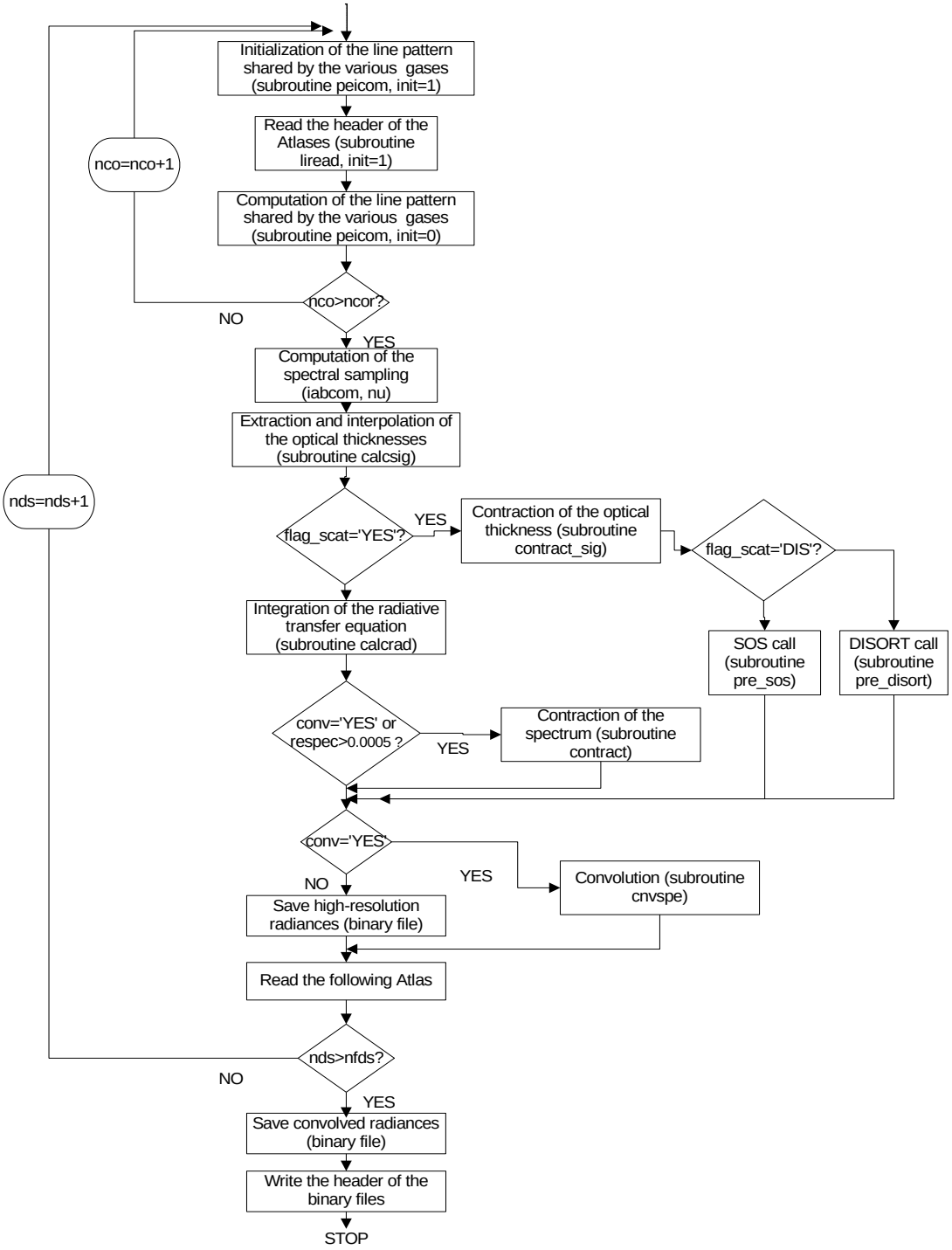


Diagram 10: Last part of the flow chart of the main program for the radiative transfer computation `aaaa`

1.2.1.2. aaaa

1.2.1.2.1 Type

It is the main program of the software 4A/OP. It is stored in the directory `srclib`.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.2.2 Function

The main program *aaaa* is dedicated to the computation of the radiative transfer through an interpolation of the absorption optical thicknesses and the integration of the radiative transfer equation in a spherical atmosphere including the solar contribution.

1.2.1.2.3 Subordinates

See Diagram 3. It uses the modules *error*, *model*, *constants*, *convolution*, *modatlas*, *precision* and *instruments*.

1.2.1.2.4 Dependencies

It depends on the existence of the required input files (section 3.3.1).

1.2.1.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Inputs						
instru	Simulated instrument	string	-	i	Ini: makefile	
nopro	Atmospheric profile number	int	-	i	Ini: makefile	
fatm	File name of the reference atmosphere	string	-	i	Ini: makefile	datatm/atmref.dsf
fgas	File name of the minor gases	string	-	i	Ini: makefile	datatm/gascon.dsf
fatmo	File name of the atmospheric profile databank	string	-	i	Ini: makefile	datatm/atm4a$(ATM).ddb
fpara	File name of the simulation parameters	string	-	i	Ini: makefile	input/para4a$(RSTR).dtp
fpars	File name of user defined aerosol parameters	string	-	i	Ini: makefile	input/parascat$(RSCA).dtp
name_h2o	Filename for H_2O continuum	string	-	i	Ini: makefile	input/foreignself_h2o.coef
name_n2	Filename for N_2 continuum	string	-	i	Ini: makefile	input/foreign_n2.coef
name_o2	Filename for O_2 continuum	string	-	i	Ini: makefile	input/foreign_o2.coef
findex	File name of the Atlas indexes	string	-	i	Ini: makefile	datatl/atl$(RESOL)index.dsf
fisrf	File name of the instrument function	string	-	i	Ini: makefile	isrf/isrf(INS)(CASE).ddb
fresui	File name of the resulting high resolution spectra ("pseudo-infinite")	string	-	i	Ini: makefile	outputbin/spi4a$(ATMPROF) (ATM)(RSTR)$(RSCA)$(INS).ddb

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
fresuc	File name of the resulting convolved spectra	string	-	i	Ini: makefile	outputbin/spc4a$(ATMPROF) $(ATM)(RSTR)$(RSCA)$(INS).ddb
racine	Root name of the optical thickness files	string	-	i	Ini: makefile	
racine3	Root name of the emissivity files	string	-	i	Ini: makefile	
racine4	Extension name of the emissivity files	int	-	i	Ini: makefile	
racine5	Root name of aerosol model parameter files	string	-	i	Ini: makefile	
Outputs						
crad	Convolved radiances	r[allocatable]	W/ (m^2.str. cm^{-1})		fresuc	spc4a$(ATMPROF)$(ATM) $(RSTR)$(RSCA)$(INS).ddb Size: number of convolved wave numbers
rad	High resolution radiances	r[maxpt]	W/ (m^2.str. cm^{-1})		fresui	spi4a$(ATMPROF)$(ATM) $(RSTR)$(RSCA)$(INS).ddb

1.2.1.2.6 Processing

The main functions of *aaaa* are:

- Load parameters from standard input;
- Load input files;
- Open files;
- Read the simulation definition parameters (subroutine *para4a*);
- Loading of the additional global variables *fpars* and *racine5* from standard input;
- Loading of the user defined aerosol parameters (subroutine *parascat*);
- Additional consistency check on user parameterizations (overlap of aerosol layers forbidden, Jacobian computation forbidden in presence of aerosol);
- Read the header of ISRF file;
- Read the reference atmospheric parameters (subroutine *atmref*);
- Read the default concentration values of gases required for 4A (subroutine *gascon*);
- Read the spectral emissivities (subroutine *lireemis*);
- The determination of the indexes of the wave number limits, by using the dichotomy technique (subroutine *dichotomd*) in the case itypeconv=3
- The determination of the total number of convolution points: iadcnv
- Initialization of the parameters required for the convolution (subroutine *convini*);
- Location of the index of the Atlases;

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Read the atmospheric profile databank that identifies the thermodynamical conditions of the simulation, independently of viewing geometry and computation of altitude, gravity acceleration and (optional) refraction (subroutine *atmsph*);
  - after call *atmsph*: in Earth view geometry and if specified by a negative value, the surface level *pzdown* is set to the value *pdown* loaded from the atmospheric profile databank;
- Geometry module: Positioning of user specified optical path (in Earth view or in limb geometry) within the vertical levels of the atmospheric profile; computation of the local zenith angle for each simulated layer and a spherical atmosphere (subroutine *calcsphtraj);*
  - after call *calcsphtraj*:
    1. initialisation of layer depth and mixing ratios in pseudo-layers: zero,
    2. initialisation of pseudo-layer temperatures: user specified values tdown, tup,
    3. in limb view and if the lower pseudo-layer is at tangent height: set surface emissivity to zero (neutralise the lower pseudo-layer);
- Initialization of the continua H_2O, N_2 and O_2 (subroutine *init_contin*);
- Initialization of arrays;
- Initialization of the line pattern shared by the various gases (subroutine *peicom*, init=1);
- Read the header of the optical thickness Atlases (subroutine *liread*, init=1);
- Location of the calculated points shared by all the layers and all the molecules (subroutine *peicom*, init=0);
- Initialization of the spectral sampling (nu: wave numbers common to the extraction of the optical thicknesses for the various involved molecules, iabcom: indexes of the spectrum non regular in wave numbers);
- Computation of the optical thicknesses corrected by height dependent gravity acceleration for each molecule and each layer (subroutine *calcsig*) with the incorporation of continua contribution (subroutine *continua*);
- If aerosol flag indicates DISORT and daytime simulation: Initialisation of the solar zenith angle for DISORT computation;
- If aerosol flag indicates SOS: Initialisation of the optical thickness tabulation;
- If aerosol flag indicates DISORT or SOS and if aerosol model flag is different from '*undef*': Loading of aerosol model parameters (subroutine *readaerosols*);
- Subroutines *calcrad* and *contract* are called only if aerosol flag indicates no aerosol. Otherwise:
  - Spectral resampling (contraction) of the gaseous optical thickness (subroutine *contract_sig*);
  - Within a loop of spectral channels: Spectral interpolation of aerosol extinction parameters (aerosol optical thickness, single scattering albedo, asymmetry parameter) and of top and bottom spectral emissivity. If aerosol flag indicates DISORT and daytime simulation, computation of the spectral solar irradiance;
  - If aerosol indicates DISORT: computation of DISORT input parameters and call of DISORT (subroutine *pre_disort*);
  - If aerosol indicates SOS: computation of SOS input parameters and call of SOS (subroutine *pre_sos*);
- Integration of the radiative transfer equation in the atmosphere (subroutine *calcrad*);
- Contraction of the spectrum on a given sampling step (subroutine *contract*);
- Convolution of the spectrum (subroutine *cnvspe* or *cnvhole*);
- Save results in output binary files.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.2.7 Variable classification

Symbol	Description	Type	Unit	Classification
maxb	Maximum number of computed Plank functions in a spectral domain	int	-	Parameter (5)
kpmax	Maximum dimension of the initial instrumental functions	int	-	Parameter = Cst_kpmax
Cst_kpmax	See module *model*	int	-	constant
Cst_maxbuf	See module *model*	int	-	constant
Cst_maxpt	See module *model*	int	-	constant
Cst_maxpt2	See module *model*	int	-	constant
Cst_maxpth2o	See module *model*	int	-	constant
Cst_nlevelmax	See module *model*	int	-	constant
Cst_ncjacmax	See module *model*	int	-	constant
Cst_npttabexp	See module *model*	int	-	constant
Cst_ldmax	See module *model*	int	-	constant
Cst_cnvmax	See module *model*	int	-	constant
Cst_nlevref	See module *model*	int	-	constant
Cst_nprof	See module *model*	int	-	constant
Cst_ncorps	See module *model*	int	-	constant
Cst_ncmax	See module *model*	int	-	constant
Cst_nValO2	See module *model*	int	-	constant
Cst_nValN2	See module *model*	int	-	constant
Cst_StpSigO2	See module *model*	int	-	constant
Cst_StpSigN2	See module *model*	int	-	constant
Cst_knumax	See module *model*	int	-	constant
Cst_nbmax	See module *model*	int	-	constant
Cst_ndsmax	See module *model*	int	-	constant
Cst_kemax	See module *model*	int	-	constant

1.2.1.2.7.1 Variables and constants introduced with implementation of aerosol extinction

Symbol	Description	Type	Unit	Comment
fpars	File name of user defined aerosol parameters	string (len=400)	-	loaded from standard input
racine5	Root name of aerosol model parameter files	string (len=400)	-	loaded from standard input

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Description	Type	Unit	Comment
jpars	Unit of *fpars*	i	-	parameter
jscat	Unit of aerosol model file	i	-	parameter
nla	Loop variable for aerosol layers	i	-	
flag_scat	Aerosol presence flag	string (len=3)	-	'NO','DIS' or 'SOS' loaded by *parascat*
disheader	Header for prints, embedded in a DISORT banner	string (len=127)	-	If flag_scat='DIS': loaded by *parascat*
nlaero	Number of aerosol layers	i	dl.	If flag_scat≠'NO': loaded by *parascat*
nmom	Number of phase function Legendre polynoms	i	dl.	If flag_scat≠'NO': loaded by *parascat*
nstream	Number of streams	i	dl.	If flag_scat='DIS': loaded by *parascat*
fbeam_s, fbeam	Spectral solar radiance at TOA	r, d	$Wm^{-2}sr^{-1}cm$	If flag_scat='DIS': initialised in *aaaa*
umu0	Cosine of the solar zenith angle at surface	d	dl.	If flag_scat='DIS': initialised in *aaaa*
accur	DISORT convergence criterion if beam source is present	d	relative radiance unit	If flag_scat='DIS': loaded by *parascat*
prnt(5)	Array of DISORT logical print flags	logical (kind=4)	-	If flag_scat='DIS': loaded by *parascat*
eup_s, edown_s	Top/bottom surface emissivity	r	dl.	If flag_scat≠'NO': interpolated in *aaaa*
rad_diff	Spectral radiance	r	$Wm^{-2}sr^{-1}cm$	Output from SOS or DISORT
aeromod (Cst_nlaeromax)	Identifier of the aerosol model (from top to bottom layer)	string (len=6)	-	If flag_scat≠'NO': loaded by *parascat*
layaero (2, Cst_nlaeromax)	Vertical atmospheric layers where aerosol is present (upper and lower layer, then from top to bottom)	i	-	If flag_scat≠'NO': loaded by *parascat*
aotref(Cst_nlaeromax)	Aerosol optical thickness at reference wave number $1000cm^{-1}$, from top to bottom layer	r	dl.	If flag_scat≠'NO': initialised in *parascat*, then updated for the processed wave number
otaero(Cst_nlaeromax)	Aerosol optical thickness, from to to bottom layer	r	dl.	Initialised by aotref, then updated for the processed wave number
omeaero (Cst_nlaeromax)	Single scattering albedo, from top to bottom layer	r	dl.	If flag_scat≠'NO':

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Description	Type	Unit	Comment
				initialised in *parascat*, then updated for the processed wave number
gaero(Cst_nlaeromax)	Asymmetry parameter, from top to bottom layer	r	dl.	If flag_scat≠'NO': initialised in *parascat*, then updated for the processed wave number
nwn(Cst_nlaeromax)	Number of wave numbers for which aerosol model properties are provided, from top to bottom layer	i	dl.	If flag_scat≠'NO' and aeromod(nla) ≠'undef': loaded in *readaerosols*
wnscat(Cst_ksmax, Cst_nlaeromax)	Wave numbers for which aerosol model properties are provided, from top to bottom layer	r	cm^{-1}	If flag_scat≠'NO' and aeromod(nla) ≠'undef': loaded in *readaerosols*
ot_aero(Cst_ksmax, Cst_nlaeromax)	Aerosol optical thickness at wave numbers for which aerosol model properties are provided, from top to bottom layer	r	dl.	If flag_scat≠'NO' and aeromod(nla) ≠'undef': loaded in *readaerosols*
ome_aero(Cst_ksmax,Cst_nlaeromax)	Single scattering albedo at wave numbers for which aerosol model properties are provided, from top to bottom layer	r	dl.	If flag_scat≠'NO' and aeromod(nla) ≠'undef': loaded in *readaerosols*
g_aero(Cst_ksmax, Cst_nlaeromax)	Asymmetry parameter at wave numbers for which aerosol model properties are provided, from top to bottom layer	r	dl.	If flag_scat≠'NO' and aeromod(nla) ≠'undef': loaded in *readaerosols*
pastabexp_d	Exponential function tabulation step for use in SOS	d	cm^{-1}	If flag_scat='SOS': initialised in *aaaa*
tabexp_d (Cst_npttabexp_d)	Exponential function table for use in SOS	d	dl.	If flag_scat='SOS': initialised in *aaaa*
Cst_npttabexp_d	Number of exponential tabulation points for use in SOS	i	dl.	defined in *model*
Cst_ksmax	Maximum number wave numbers for which scattering properties are provided	i	dl.	defined in *model*
Cst_nlaeromax	Maximum number of aerosol layers	i	dl.	defined in *model*
PI	Approximation for reflection of diffuse radiation	d	-	defined in *constants*
T_SUN	Solar surface temperature	r	K	defined in *constants*
SUN_EMM_APP	Apparent solar emissivity at TOA at mean Earth-Sun distance	r	dl.	defined in *constants*

Remark:

File name units are declared as parameters: jatm=1, jatmo=2, jgas=3, jindex =4, jlist=6, jemis=7, jap=8, jresui=9, jresuc=10, jpara=11, jpars=12, jscat=13.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.2.7.2 Variables and constants introduced with implementation of limb geometry

Symbol	Description	Type	Unit	Comment
zl(Cst_nlevelmax)	Altitude of each pressure level	real	km	
dzc(0:Cst_nlevelmax)	Geometric layer depth	real	km	
refil(Cst_nlevelmax)	Refractive index at each pressure level	real	-	
gc(0:Cst_nlevelmax)	Gravity acceleration at layer centre	real	m s^{-2}	
geomid	Geometry identifier	character	-	'EVIEW' or 'LIMB'
refra	Identifier for refraction computation	character	-	'YES' or 'NO'
pzdown	maximum pressure/ minimum altitude along optical path	real	hPa km	EVIEW LIMB
pzup	minimum pressure/ maximum altitude along optical path	real	hPa km	EVIEW LIMB
pzobs	pressure/altitude at observation level	real	hPa km	EVIEW LIMB
zgtan	Geometric tangent height of limb optical path	real	km	
zatmbot	Height of lowest, defined atmospheric level	real	km	
RI1_0	Factor for refractive index computation (dry air)	real	K hPa^{-1}	77.48
RI2_0	Factor for refractive index computation (water vapour)	real	K hPa^{-1}	-12.79
SECTTA2_0	Approximation for reflection of diffuse radiation	real	-	1.7434468
MINLAY_0	Minimum geometric layer depth	real	km	0.02

1.2.1.3. para4a

1.2.1.3.1 Type

para4a is a subroutine used by the main program `aaaa.f90`.
para4a is referenced in the library `aaaa.a`.

1.2.1.3.2 Function

The subroutine *para4a* has been made to read the simulation definition parameters listed in the file para4a$(RSTR).dtp (directory `input`, see section Part I. 5.2.5.2).

1.2.1.3.3 Subordinates

It uses the modules *error* and *constants*.

1.2.1.3.4 Dependencies

It depends on the existence of the parameter file para4a$(RSTR).dtp.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.3.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
ncorps	Maximum number of atmospheric molecules	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncorps
jpara	Logical unit defining the input file to read	int	-	i	Main program aaaa	Defined in aaaa: 3
Jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Output						
job	Main run parameter that identifies the program to use	string (len=60 max)	-	o	Ini: parameter file para4a.dtp	Only for information: not used
geomid	Geometry type identifier	string (len=6)	-	o	Ini: parameter file para4a.dtp	'LIMB' or 'EVIEW' default: 'EVIEW'
trajet	Viewing configuration	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	UP, DOWNUP, DOWN, UPDOWN UP: nadir viewing DOWN: zenith viewing
pzup	TOA or cloud bottom pressure TOA or cloud bottom height	r	hPa km	o	Ini: parameter file para4a.dtp	if geomid = 'EVIEW' if geomid = 'LIMB'
pzdown	Surface or cloud top pressure Surface or cloud top height	r	hPa km	o	Ini: parameter file para4a.dtp	if geomid = 'EVIEW' if geomid = 'LIMB'
pzobs	Pressure at observation level Height at observation level	r	hPa km	o	Ini: parameter file para4a.dtp	if geomid = 'EVIEW' pzup ≤ *pzobs* ≤ pzdown if geomid = 'LIMB' pzdown ≤ *pzobs* ≤ pzup
iemup	Index for the emissivity of the top level	i	-	o	Ini: parameter file para4a.dtp	Used if *trajet*≠UP
tup	Temperature of the top level	r	K	o	Ini: parameter file para4a.dtp	4 K in general If = 0, the temperature of the atmospheric database is kept (*tup* = level1 temperature)
iemdown	Index for the emissivity of the lower level	i	-	o	Ini: parameter file para4a.dtp	Used if *trajet*≠ DOWN

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
tdown	Temperature of the lower level	r	K	o	Ini: parameter file para4a.dtp	If *tdown* < 0 *tdown* set to temperature of level2 If *tdown*=0: - for level2<nlevtam *tdown*=temperature of level2 - for level2=nlevtam *tdown*=surface temperature defined in the atmospheric database Else *tdown*=user input value
zgtan	Geometric tangent height	r	km	o	Ini: parameter file para4a.dtp	Used if *geomid* = 'LIMB'
sectta1	Secant value of the incident angle (viewing angle)	r	- or °	o	Ini: parameter file para4a.dtp	If *typ_ang* = SECANT, *sectta1* must be ≥1 new: *sectta1* < 1. or *sectta1* ≥90° admitted
sectta2	Secant value of the reflexion angle	r	-	o	Ini: parameter file para4a.dtp	Idem *sectta1.* Used if *trajet* = DOWNUP or UPDOWN
secttas	Secant value of the solar indicence angle	r	-	o	Ini: parameter file para4a.dtp	Idem *sectta1.* Used if *trajet* = DOWN or DOWNUP. If secttas ≥1, the solar contribution is included in the radiative transfer simulation.
esdist	Earth-Sun distance in astronomical units	r	Astronomical unit	o	Ini: parameter file para4a.dtp	Natural values are [0.98, 1.02]
refra	Refraction computation identifier	string (len=6)	-	o	Ini: para4a.dtp	'YES' or 'NO'
zatmbot	Height of lower atmospheric level	r	km	o	Ini: para4a.dtp	if specified negative: computed in atmsph through barometric height formula

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
contin	Continua (H_2O, N_2, O_2) contribution	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	YES or NO If yes, the continua of the selected molecules (*idsel*=1) contribute to the radiative transfer computation
respec	Spectral resolution: contraction of the spectra before convolution or to define the spectral discretization with an "infinite" resolution simulation	d	cm^{-1}	o	Ini: parameter file para4a.dtp	If *respec*<5.10^{-4}: – For *conv*=YES, the resolution for the computation = 5.10^{-4} – For *conv*=NO, the resolution for the computation = Atlas resolution Else *respec* = user input value of *respec*
forig	First wave number of the simulation	r	cm^{-1}	o	Ini: parameter file para4a.dtp	*forig* must be ≥ forig_atlas*
fstop	Last wave number of the simulation	r	cm^{-1}	o	Ini: parameter file para4a.dtp	*fstop* must be ≤ fstop_atlas*
delds	Width of each Atlas spectral band	d	cm^{-1}	o	Ini: parameter file para4a.dtp	generally 15 cm^{-1}
intopt	Indicator for interpolation in temperature of the optical thicknesses in the atlases	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	YES or NO Note: If no, the interpolation in temperature is skipped: The closest atlas temperature profile is kept. The computation time is decreased but the results are deteriorated The option "NO" is not compatible with Jacobian computation.
conv	Indicator of convolution with the ISRF	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	YES or NO ("infinite" resolution)
hole	Indicator of the hole algorithm use for the convolution	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	YES or NO
shift	Indicator of a spectral shift introduction due to the ISRF	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	YES or NO Only used if *conv*=YES If yes, a spectral shift is applied due to the no centred ISRF
rnu	Value of the global	d	cm^{-1}	o	Ini: parameter	Only used if *conv*=YES

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	spectral shift				file para4a.dtp	Application of a possible global spectral shift of the spectra if *rnu* ≠ 0
pascon	Convolution step: step of the ISRF application	d	cm^{-1}	o	Ini: parameter file para4a.dtp	Only used if *conv*=YES Must be consistent with the instrument function file header; *pascon* only used if itypeconv=2 (Part I. 5.2.3 and Part II.1.2.7); must be greater than *respec*
fpoid	Transmittance calculation flag	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	YES or NO Computation of Jacobians and transmittances is not permitted if *conv*=NO
jacob	Jacobian calculation flag	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	YES or NO If *jacob*=YES: • *intopt* must be YES • The Jacobian with respect to the temperature is always calculated. • The Jacobian with respect to the mixing ratio is calculated if the corresponding indicator is equal to 1 (see *idsel* and *idjac* below)
idsel	Array of indexes to select molecules for the 42 gases available in the GEISA database and 3 CFCS coded between 61 to 63	int[63]	-	o	Ini: parameter file para4a.dtp	1 or 0 1=yes 0=no
rocoef	Array of weighting coefficients of the mixing ratio for the 42 available gases and 3 CFCS	r[63]	-	o	Ini: parameter file para4a.dtp	0 ≤ *rocoef*
idjac	Array of indexes to select the calculation of the Jacobian for each gas in the 42 available gases and 3 CFC	int[63]	-	o	Ini: parameter file para4a.dtp	1 or 0 1=yes 0=no
ErrCode	Error code	int	-	o		Err_ok or Err_error

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
						Err_ok if reading is successful Err_error otherwise

1.2.1.3.6 Processing

para4a reads all the parameters already described in the first part of the document plus the variable typ_ang that represents the angle type definition for sectta1, sectta2 and secttas:

- if typ_ang equal to the character string UNDEF then sectta1 is equal to 0 (then sectta1 is set to the value read in the atmosphere file atm4a$(ATM).ddb: see *atmsph*, section 1.2.1.8);
- if typ_ang equal to the character string UNDEF then sectta2 is set to 1;
- if typ_ang equal to the character string UNDEF then secttas is set to 0;
- if typ_ang equal to the character string ANGLE then sectta1 and sectta2 are angles given in degree and sectta1 (sectta2) is converted into secant as follows: sectta1 = 1/cos(sectta1);
- if typ_ang equal to the character string ANGLE and 0≤secttas<90 then secttas is an angle given in degree and secttas is converted into secant;
- If typ_ang equal to the character string ANGLE and secttas≥90 then secttas is set to 0;
- if typ_ang equal to the character string SECANT then sectta1, sectta2 and secttas remain unchanged.

Constraints for user defined upper, lower and observation levels

T_{up}, ε_{up} represent temperature and emissivity of the upper limit, i.e., either sky (TOA) or a cloud bottom. T_{down}, ε_{down} represent temperature and emissivity of the lower limit, i.e., either Earth surface or a cloud top. An 'X' indicates the need for specification, 'C' means that this specification makes sense only for a cloud, '0' indicates that the surface layer is neutralised by setting its emissivity to zero.

Negative specifications of pzdown, pzup are admitted and replaced by lower and upper level of the atmopheric profile loaded in atmsph. A negative specification of pzobs causes the observation level to be set to the upper level. Exceptions for upper and lower level specifications outside the range of the defined atmosphere are discussed for the module *calcsphtraj*.

Earth view configurations (geomid = 'EVIEW')

trajet	p_{up}=pzup [hPa]	p_{down}=pzdown [hPa]	p_{obs}=pzobs [hPa]	T_{up}, ε_{up}	T_{down}, ε_{down}
UP		$p(T_{down}, \varepsilon_{down})$	$p_{down} > p_{obs} \geq p_{up}$		X
DOWNUP	$P(T_{up}, \varepsilon_{up})$	$p(T_{down}, \varepsilon_{down})$	$p_{down} > p_{obs} \geq p_{up}$	X	X
DOWN	$P(T_{up}, \varepsilon_{up})$		$p_{down} \geq p_{obs} > p_{up}$	X	
UPDOWN	$P(T_{up}, \varepsilon_{up})$	$p(T_{down}, \varepsilon_{down})$	$p_{down} \geq p_{obs} > p_{up}$	C	X

Limb view configurations (geomid = 'LIMB')

trajet	z_{up}=pzup [km]	z_{down}=pzdown [km]	z_{obs}=pzobs [km]	T_{up}, ε_{up}	T_{down}, ε_{down}

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

UP		$z(T_{down}, \varepsilon_{down})$ > zgtan	$z_{down} < z_{obs} \leq z_{up}$		C
DOWNUP-1	$z(T_{up}, \varepsilon_{up})$	$z(T_{down}, \varepsilon_{down})$ > zgtan	$z_{down} < z_{obs} \leq z_{up}$	X	C
DOWNUP-2	$z(T_{up}, \varepsilon_{up})$	≤ zgtan	zgtan $< z_{obs} \leq z_{up}$	X	0
DOWN	$z(T_{up}, \varepsilon_{up})$		$z_{down} \leq z_{obs} < z_{up}$ zgtan $\leq z_{obs}$	X	
UPDOWN	$Z(T_{up}, \varepsilon_{up})$ > zgtan	$z(T_{down}, \varepsilon_{down})$	$z_{down} \leq z_{obs} < z_{up}$ zgtan $\leq z_{obs}$	C	X

1.2.1.3.7 Variable classification

Symbol	Description	Type	Unit	Classification
tdown	Temperature of the lower level	r	K	double: tdowno in *aaaa*

1.2.1.4. atmref

1.2.1.4.1 Type

atmref is a subroutine used by the main program `aaaa.f90`.

atmref is referenced in the library `aaaa.a`.

1.2.1.4.2 Function

The subroutine *atmref* has been made to read the reference atmospheric parameters stored in the file atmref.dsf (directory datatm). These parameters are used for the computation of the optical thicknesses in the Atlases.

1.2.1.4.3 Subordinates

It uses the module *error* and *precision*.

1.2.1.4.4 Dependencies

It depends on the existence of the parameter file atmref.dsf.

1.2.1.4.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
ncorps	Maximum number of atmospheric molecules	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncorps
nlevel	Number of reference pressure levels	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nlevref
nprof	Number of temperature profiles	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nprof
jatm	Logical unit defining the input file to read	int	-	i	Main program aaaa	Defined in aaaa: 1

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Output						
plref	Reference pressure levels	r[nlevel]	hPa	o	Ini: file atmref.dsf	
pcref	Reference pressure values for each layer delimited by the pressure levels plref	r[nlevel]	hPa	o	Ini: file atmref.dsf	
tlref	Reference temperature profiles (at the pressure levels plref)	r[nlevel,nprof]	K	o	Ini: file atmref.dsf	
tcref	Reference temperature values for each layer	r[nlevel,nprof]	K	o	Ini: file atmref.dsf	
rolref	Reference molecule mixing ratio profiles (at the pressure levels)	r[nlevel,ncorps]	g/g	o	Ini: file atmref.dsf	
rocref	Reference molecule mixing ratio for each layer	r[nlevel,ncorps]	g/g	o	Ini: file atmref.dsf	
secref	Secant value of the reference angle of incidence	r	-	o	Ini: file atmref.dsf	
pzdown	Surface pressure	r	hPa	o	Ini: file atmref.dsf	
tdown	Surface temperature	r	K	o	Ini: file atmref.dsf	
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if reading is successful Err_error otherwise

1.2.1.4.6 Processing

The input file atmref.dsf contains the 12 temperature profiles, the 42 mixing ratio profiles of the GEISA databank molecules and the mixing ratio profiles of 3 CFCS used to generate the optical thickness Atlases.

atmref reads the reference atmospheric parameters described previously. The parameter values are read for each level (suffix "lref") and are arranged in layers in the arrays with the suffix "cref".

Concerning the pressure plref (pcref):

pcref(nl) = (plref(nl)+plref(nl+1))/2, with nl starting at 1 and stopping at nlevel-1.

pcref(nlevel)=plref(nlevel)

Concerning the temperature tlref (tcref):

tcref(nl,np) = (tlref(nl,np)+tlref(nl+1,np))/2, with np starting at 1 and stopping at nprof.

Concerning the mixing ratio rolref (rocref):

rocref(nl,nc) = secref*(rolref(nl,nc)+rolref(nl+1,nc))/2, with nc starting at 1 and stopping at ncorps.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.4.7 Variable classification

Symbol	Description	Type	Unit	Classification
pzdown	Surface pressure	r	hPa	double: psal in aaaa
tdown	Surface temperature	r	K	double: tsal in aaaa

1.2.1.5. gascon

1.2.1.5.1 Type

gascon is a subroutine used by the main program `aaaa.f90`.
gascon is referenced in the library `aaaa.a`.

1.2.1.5.2 Function

The subroutine *gascon* has been made to read the default concentration values of gases required for 4A stored in the file gascon.dsf (directory `datatm`).

1.2.1.5.3 Subordinates

It uses the module *error* and *precision*.

1.2.1.5.4 Dependencies

It depends on the existence of the parameter file gascon.dsf.

1.2.1.5.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
ncorps	Maximum number of atmospheric molecules	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncorps
nlevelmax	Maximum number of pressure levels	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ nlevelmax
jgas	Logical unit defining the input file to read	int	-	i	Main program aaaa	Defined in aaaa: 7
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Output						
nlevelgas	Number of pressure levels	int	-	o	Ini: file gascon.dsf	
plgas	Reference pressure levels	r[nlevelmax]	hPa	o	Ini: file gascon.dsf	
rolgas	Gas mixing ratio profiles (for each molecule)	r[nlevelmax,ncorps]	g/g	o	Ini: file gascon.dsf	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if reading is successful Err_error otherwise

1.2.1.5.6 Processing

gascon reads the default values of the mixing ratio for each required gas molecule. The values are read for each pressure level.

1.2.1.5.7 Variable classification

Symbol	Description	Type	Unit	Classification
nlevelgas	Number of pressure levels	int	-	double: nlevgas in aaaa

1.2.1.6. lireemis

1.2.1.6.1 Type

lireemis is a subroutine used by the main program `aaaa.f90`.
lireemis is referenced in the library `aaaa.a`.

1.2.1.6.2 Function

The subroutine *lireemis* has been made to read the surface spectral emissivities.

1.2.1.6.3 Subordinates

It uses the module *error* and *precision*.

1.2.1.6.4 Dependencies

It depends on the existence of the spectral emissivity files spemis$(EMDOWN).dat and spemis$(EMUP).dat (directory `datemis`). Before this component is called, the parameters listed in the file para4a$(RSTR).dtp must have been already read via the subroutine *para4a.*

1.2.1.6.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
kemax	Maximum number of read emissivities per file	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_kemax
racine3	Root name of the emissivity files	string (len=400)	-	i	makefile	
racine4	Extension name of the emissivity files	string (len=400)	-	i	makefile	
iemup	Emissivity index for upper limit	int	-	i	Ini: parameter file para4a.dtp	
iemdown	Emissivity index for lower limit	int	-	i	Ini: parameter file para4a.dtp	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

jemis	Logical unit defining the input file to read	int	-	i	Main program aaaa	Defined in aaaa: 17
Output						
nonde	Number of available spectral emissivity values; upper and lower limits	int[2]	-	o	Ini: file spemis.dat	
tabe1	Wave numbers for which emissivity is read; upper and lower limits	r[kemax,2]	cm^{-1}	o	Ini: file spemis.dat	
tabe2	Spectral emissivity at available wave numbers; upper and lower limits	r[kemax,2]	-	o	Ini: file spemis.dat	
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if reading is successful Err_error otherwise

1.2.1.6.6 Processing

lireemis reads the user-defined surface and top-of-atmosphere spectral emissivities for each limit (upper and lower). The user is free to create new files with its own emissivity values (consult appendices section Part III 1.2).

1.2.1.7. convini

1.2.1.7.1 Type

convini is a subroutine used by the main program `aaaa.f90`.
convini is referenced in the library `aaaa.a`.

1.2.1.7.2 Function

The subroutine *convini* is used only if the convolution is required. It allows the initialization of the parameters required for the convolution. It computes the limits in wave numbers used for the convolution as a function of the instrument type.

1.2.1.7.3 Subordinates

The subroutine *convini* calls the subroutine *dichotomd*. It uses the modules *error* and *instruments*.

1.2.1.7.4 Dependencies

It depends on the existence of the instrument function file isrf.ddb (directory `isrf`). Before this component is called, the function characteristics written in the header of the file isrf.ddb must have been already read in the main program *aaaa.*

1.2.1.7.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
nudeb	Starting convolution wave number	d	cm^{-1}	i	Main program aaaa	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
nufin	Stopping convolution wave number	d	cm^{-1}	i	Main program aaaa	
pascon	Convolution step: step of the ISRF application	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
irdeb	Index of the wave number limits	int	-	i	Main program aaaa	
nknu	Number of ISRF	int	-	i	Ini: ISRF file isrf.ddb	
pdnuco	Initial sampling step of the instrument functions	d	cm^{-1}	i	Ini: ISRF file isrf.ddb	
wnuo	Definition domain (width) of each ISRF	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	
nufo	Central wave numbers of each ISRF	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	
kfmo	Number of description points of a ISRF	int[nknu]	-	i	Ini: ISRF file isrf.ddb	
itypeconv	Identifier of the convolution	int	-	i	Ini: ISRF file isrf.ddb	1: Radiometer with n channels (HIRS) 2: Interferometer / spectrometer with a constant sampling step in wave number (IASI: step = 0.25 cm^{-1}) 3: Interferometer / spectrometer with a non constant sampling step and fixed wave numbers (AIRS)
ndsmax	Maximum number of spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ndsmax
delds	Width of each Atlas spectral band	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Output						
pdnuc	Sampling step of the instrument functions	d[nknu]	cm^{-1}	o		
cnu	Central wave numbers of each ISRF in the convolution domain (iadcnv wave numbers)	d[iadcnv]	cm^{-1}	o		
wnu	Width (definition domain) of each ISRF in the convolution domain	d[iadcnv]	cm^{-1}	o		

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
cwnu1	Last wave number limit of each ISRF involved in the convolution	d[iadcnv]	cm^{-1}	o		
cwnu2	First wave number limit of each ISRF involved in the convolution	d[iadcnv]	cm^{-1}	o		
idatl	Atlas flags used for the computation of the convolved spectrum	l[ndsmax]	-	o		false or true *iadtl* allows to avoid the computation of the infinite spectrum for all the wave numbers located between two ISRF (mainly used in the radiometer case)
nuinf	First wave number of the convolution	d	cm^{-1}	o		
nusup	Last wave number of the convolution	d	cm^{-1}	o		
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if reading is successful Err_error otherwise
Input/Output						
iadcnv	Number of convolution points	int	-	i/o		

1.2.1.7.6 Processing

convini determines:

- ~~the indexes of the wave number limits, by using the dichotomy technique (subroutine *dichotomd*) in the case itypeconv=3~~
- ~~the total number of convolution points: iadcnv~~
- the corresponding wave numbers: cnu
- the sampling step of the instrument functions: pdnuc
- the wave number limits:
  - First and last wave number limits of each ISRF involved in the convolution: cwnu2 and cwnu1
  - First and last wave numbers of the convolution: nuinf and nusup
- the Atlas flags: idatl. If the Atlas flag is true, the corresponding atlas is read, else if it is false, the atlas is not read. This parameter is mainly used for the radiometer case (broad spectral bands). It allows to avoid the computation of the infinite spectrum for all the wave numbers located between two filters (or ISRF), that is when two consecutive ISRF have no common spectral domain.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.8. atmsph

1.2.1.8.1 Type

atmsph is a subroutine used by the main program `aaaa.f90`.

atmsph is referenced in the library `aaaa.a`.

1.2.1.8.2 Function

The subroutine *atmsph* has been made to read the atmospheric profile databank stored in the file atm4a$(ATM).ddb (directory `datatm`), that identifies the thermodynamical conditions of the simulation.

1.2.1.8.3 Subordinates

The subroutine *atmsph* calls the subroutine *inttab*. It uses the module *error*, *precision*, *constants* and *model*.

1.2.1.8.4 Dependencies

It depends on the existence of the atmospheric profile databank file atm4a$(ATM).ddb. Before this component is called, the parameters listed in the files para4a$(RSTR).dtp and gascon.dsf must have been already read via the subroutines *para4a* and *gascon*.

1.2.1.8.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
rocoef	Array of weighting coefficients of the mixing ratio for the 42 gases available in the GEISA database and 3 CFCS	r[63]	-	i	Ini: parameter file para4a.dtp	
idsel	Array of indexes to select molecules for the available 42 gases and 3 CFCS (1 or 0)	int[63]	-	i	Ini: parameter file para4a.dtp	
refra	Refraction computation identifier	string (len=6)	-	i	Ini: parameter file para4a.dtp	'YES' or 'NO'
nlevgas	Number of vertical levels of the default gas concentration (mixing ratio) values	int	-	i	Ini: file gascon.dsf	
nlevmax	Maximum number of vertical levels	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ nlevelmax
plgas	Default pressure levels	r[nlevmax]	hPa	i	Ini: file gascon.dsf	
rolgas	Default gas mixing ratio per level and per molecule	r[nlevmax, ncorps]	g/g	i	Ini: file gascon.dsf	
ncorps	Maximum number of atmospheric molecules	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
						Cst_ncorps
maxbuf	Maximum size of a record block (buffer)	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxbuf
iat	Index number of the selected atmospheric profile	int	-	i	Ini: makefile	
jatmo	Logical unit defining the input file to read	int	-	i	Main program aaaa	Defined in aaaa: 2
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Input/Output						
zatmbot	Height of lower atmospheric level	r	km	i/o	Ini: parameter file para4a.dtp	
sectta1	Secant value of the viewing angle	r	-	i/o	Ini: parameter file para4a.dtp	neutralised in *atmsph*
tdown	Temperature of the lower level	r	K	i/o	Ini: parameter file para4a.dtp	
Output						
pl	Pressure levels of the atmospheric profiles	r[nlevmax]	hPa	o	Ini: file atm4a$(ATM).ddb	
tl	Temperature profile at the pressure levels pl	r[nlevmax]	K	o	Ini: file atm4a$(ATM).ddb	
rol	Molecule mixing ratio profiles at the pressure levels pl	r[nlevmax, ncorps]	g/g	o	Ini: file atm4a$(ATM).ddb	
pc	Pressure values at the middles of the layers delimited by the levels pl	r[0:nlevmax]	hPa	o		
tc	Temperature values at the layer middles	r[0:nlevmax]	K	o		
roc	Molecule mixing ratio values at the layer middles	r[nlevmax, ncorps]	g/g	o		
zl	Geometric height of pressure levels pl	r[nlevmax]	km	o		
refil	Refractive index at pressure levels pl	r[nlevmax]	-	o		
dzc	Geometric layer height	r[0:nlevmax]	km	o		

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
gc	Gravity acceleration at geometric layer centre	r[0:nlevmax]	m s^{-2}	o		
pdown	Pressure value at the lower level of the atmosphere, only used if PZDOWN not properly specified	r	hPa	o	Ini: file atm4a$(ATM).ddb	
ncsel	Total number of selected molecules	int	-	o	Ini: file atm4a$(ATM).ddb	
ncosel	Indentication numbers of the selected molecules according to the GEISA notation	int[ncorps]	-	o		
nlevatm	Number of discretization pressure levels of the atmospheric databank	int	-	o	Ini: file atm4a$(ATM).ddb	≤ nlevmax
buff	Buffer: work array	r[nlevmax*(ncorps+2) + ncorps+5]	-	o	Main program aaaa	
idcor	Index (flag) array of the molecules of which the concentration profile is defined in the input atmospheric databank file (0 or 1)	int[ncorps]	-	o		
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if reading is successful, Err_error otherwise

1.2.1.8.6 Processing

The input file atm4a$(ATM).ddb contains the user input profiles of the temperature, the water vapour mixing ratio and other gases (with their corresponding GEISA identification index) as a function of pressure levels.

With the implementation of limb geometry, any computation related to the observation geometry has been suppressed. Vertical levels and layers are defined as function of the number of vertical levels in atm4a$(ATM).ddb. Limits of the radiative transfer integration, read in the file para4a$(RSTR).dtp, are not located anymore.

The routine computes all necessary information for the definition of the optical path in the geometry module *calcsphtraj*, including

- the computation of the vertical height at each pressure level zl(nl);
- the computation of the geometric depth of vertical layers dzc(nc);
- the computation of the refractive index at each vertical level refil(nl);
- the computation of the gravity acceleration at each later centre gc(nc).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Values at the lower atmospheric level *nlevatm* are initialised as follows:

- altitude zl(nlevatm) = zatmbot

The various steps of *atmsph* are:

1) read information linked to the address of each thermodynamical profile (iloc, ilas, nben);
2) read and decode a buffer in order to determine the atmospheric profiles of each thermodynamical parameter (pressure, temperature and gas mixing ratio);
3) addition of missing gases from the default values read in the file gascon.dsf by using a linear interpolation on the pressure levels pl (subroutine *inttab*);
4) Height computation for each pressure level;
5) refractive index computation for each pressure level
6) computation of the physical parameter values for each middle of the layers delimited by the pressure levels.

The various steps are detailed below:

1) iloc, ilas and nben are address indexes in order to locate a profile among the file record blocks;
2) atmsph reads the atmospheric parameters of the databank. The parameter values are read for each level (suffix "l"). If the user input value of sectta1 has been set to 0 (typ_ang=UNDEF), sectta1 is set to the value read in the header of the file atm4a$(ATM).ddb;
3) A test is made by checking the molecules selected in the file para4a$(RSTR).dtp compared to those defined in the file atm4a$(ATM).ddb:
4) If idsel(nc) is not equal to 0 and if idcor(nc) is not equal to 0 then a linear interpolation of rolgas is performed on the pressure levels pl in order to make up the array rol (with nc starting at 1 and stopping at ncorps): (rolgas(nc), plgas) => (rol(nc), pl)
   a. Conditions: pzup=pl(level1),
   b. if tup=0 then tup=tl(level1),
   c. if tdown=0 then tdown=tsal (surface temperature defined in the atmospheric databank);
5) The physical parameter values are arranged in layers in the arrays with the suffix "c":

   Concerning the pressure pc (pl):

   pc(nl) = (pl(nl)+pl(nl+1))/2, with nl starting at level1 and stopping at level2-1.

   pc(level2)=pzdown

   pc(level1-1)=pzup

   Concerning the temperature tc (tl):

   tc(nl) = (tl(nl)+tl(nl+1))/2

   tc(level2)=tdown

   tc(level1-1)=tup

   Concerning the mixing ratio roc (rol), a correction of the profile weighting coefficient (rocoef) is applied: rol(nl,nc) = rol(nl,nc)*rocoef(nc).

   If idsel(nc) is not equal to 0 then ncsel=ncsel+1 and ncosel(ncsel)=nc

   roc(nl,nc) = (rol(nl,nc)+rol(nl+1,nc))/2,

   with nc starting at 1 and stopping at ncorps.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.8.7 Variable classification

Symbol	Description	Type	Unit	Classification
buff	Buffer: work array	r[nlevmax*(ncorps+2) + ncorps+5]	-	double: tab1 in aaaa
iat	Index number of the selected atmospheric profile	int	-	double: nopro in aaaa

Local variables and constants introduced with the implementation of limb geometry :

Symbol	Descriptive Name	Type	Units	References/Remarks
Local variables				
pel	Partial pressure of water vapour	r	hPa	
Constants				
RI1_0	Factor for refractive index computation	r	K hPa^{-1}	
RI2_0	Factor for refractive index computation	r	K hPa^{-1}	

1.2.1.9. calcsphtraj

1.2.1.9.1 Type

calcsphtraj is a subroutine used by the main program `aaaa.f90`.

calcsphtraj is referenced in the library `aaaa.a`.

1.2.1.9.2 Function

The subroutine *calcsphtraj* constitutes the geometry module of 4A/OP. The first functionality is to reference the user defined optical path limits within the vertical level/layer coordinates defined in atm4a$(ATM).ddb. It determines the length of the optical path in each vertical layer through computation of local zenith angles of the optical path at each vertical level in a spherical atmosphere.

1.2.1.9.3 Subordinates

The subroutine *calcsphtraj* calls the subroutines *inttab* and *dichotom*. It uses the modules *error* and *constants*.

1.2.1.9.4 Dependencies

Before this component is called, the atmospheric parameters described in the file atm4a$(ATM).ddb must have been already read via the subroutine *atmsph*.

1.2.1.9.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
nlevmax	Maximum number of vertical levels	int	-	i	aaaa	Parameter defined in module model

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
						used in aaaa: Cst_nlevelmax
nlevatm	Number of vertical levels	int	-	i	atmsph	
trajet	Viewing configuration	string (len=6)	-	i	i: para4a.dtp	UP, DOWNUP, DOWN, UPDOWN
geomid	Geometry type identifier	string (len=6)	-	i	para4a	EVIEW, LIMB
refra	Refraction computation identifier	string (len=6)	-	i	i: para4a.dtp	YES, NO
sectta1	Secant value of the primary path angle (viewing angle)	r	-	i	i: para4a.dtp	
secttas	Secant value of the solar incidence path zenith angle	r	-	i	i: para4a.dtp	
zgtan	Geometric tangent height	r	km	i	i: para4a.dtp	
ncorps	Maximum number of atmospheric molecules	int	-	i	aaaa	Parameter defined in module model used in aaaa: Cst_ncorps
jlist	Writing logical unit (standard output)	int	-	i	aaaa	defined in aaaa: 6
Input/Output						
pzup	TOA or cloud bottom pressure TOA or cloud bottom height	r	hPa km	i/o	i: para4a.dtp	geomid = 'EVIEW' geomid = 'LIMB'
pzdown	surface or cloud top pressure surface or cloud top height	r	hPa km	i/o	i: para4a.dtp	geomid = 'EVIEW' geomid = 'LIMB'
pzobs	Pressure at observation level Height at observation level	r	hPa km	i/o	i: para4a.dtp	geomid = 'EVIEW' geomid = 'LIMB'
zl	level altitude	r[nlevmax]	km	i/o	i: atmsph	adjustment to

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
pl	level pressure	r[nlevmax]	hPa	i/o	i: atm4a$(ATM).ddb	pzup and pzdown
tl	Temperature profile at the pressure levels	r[nlevmax]	K	i/o	i: atm4a$(ATM).ddb	
rol	Molecule concentration profile at the pressure levels	r[nlevmax, ncorps]	g/g	i/o	i: atm4a$(ATM).ddb	
refil	level refractive index	r[nlevmax]	-	i/o	i: atmsph	
pc	layer pressure	r[0:nlevmax]	hPa	i/o	i: atmsph	
tc	layer temperature	r[0:nlevmax]	K	i/o	i: atmsph	
roc	Layer molecule concentration values	r[nlevmax, ncorps]	g/g	i/o	i: atmsph	
dzc	geometric layer depth	r[0:nlevmax]	km	i/o	i: atmsph	
gc	gravity acceleration at geometric layer centre	r[0:nlevmax]	m s^{-2}	i/o	i: atmsph	
sectta2	Secant value of the secondary path angle (reflexion/emission angle)	r	-	i/o	i: para4a.dtp	
Output						
sectta1lev	Secant value of the primary path angle per pressure layer	r[0:nlevmax]	-	o	calcrad	
sectta2lev	Secant value of the secondary path angle per pressure layer	r[0:nlevmax]	-	o	calcrad	
secttaslev	Secant value of the solar incidence path angle per pressure layer	r[0:nlevmax]	-	o	calcrad	
ztan	tangent height (refraction corrected)	r	km	o		
level1	Upper level of the optical path	int	-	o	calcsig, calcrad	
level2	Lower level of the optical path	int	-	o	calcsig, calcrad	
levobs	Observation level	int	-	o	calcrad	
ErrCode	Error code	int	-	o		Err_ok or Err_error

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Constants						
R_EARTH	Earth radius	r	km			6371.0
Cst_gpf	Universal gas constant	r	J/K/mol			8.316
Cst_pdsair	Mol mass of dry air	r	g/mol			28.964
SECTTA2_0	Secans of standard incidence angle 55°	r	-			1.7434468
MINLAY_0	Minimum geometric layer depth	r	km			0.02
Local variables						
tta1	local zenith angle per level of primary path	r[nlevatm]	rad			
tta2	local zenith angle per level of secondary path	r[nlevatm]	rad			
ttas	local zenith angle per level of solar incidence path	r[nlevatm]	rad			
pl_orig	pressure level to be adjusted to pzdown	r	hPa			
refitan	refractive index at tangent height	r	-			

1.2.1.9.6 Processing

In its current version the subroutine computes the altitude for each pressure level defined in atm4a$(ATM).ddb by the barometric height formula (independent of the atmospheric state and thus approximate). Subsequently, the local zenith angles of the incoming and outgoing trajectory for each layer (at central pressure pc) in a spherical atmosphere are computed.

The modified version of *calcsphtraj* becomes a pure geometry module that links the user specified parameters defining the observation geometry (*trajet, sectta1, sectta2, secttas, pzup, pzdown, pzobs, zgtan*) with the vertical coordinates (*pl, zl*) and their atmospheric state (in particular *tl, refil*), defined in the preceding routine *atmsph*.

The overall purpose of *calcsphtraj* is unchanged, it provides the local zenith angle of the optical path for each vertical layer. Its secant value determines the length of the optical path in each layer, thus the weighting factor for optical thickness with respect to a reference secant value. Modifications of this calculation are required due to the implementation of limb observation geometry and due to the implementation of atmospheric refraction. Moreover, at large zenith angles the effective layer zenith angle cannot be approximated by the zenith angle at the layer centre but must be explicitly calculated as function of the zenith angles at the upper and lower layer limits.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Functional modifications are justified by the strict separation of the one-dimensional (vertical) description of the atmospheric state from the computation of the viewing geometry (all information that is independent of the viewing geometry is generated in routine *atmsph*):

- The height calculation requires higher precision for applications in limb viewing geometry. It is made dependent on the atmospheric state (hydrostatical equilibrium). Consequently, its computation is carried out in *atmsph* (section 1.2.1.8). Altitude at atmospheric pressure levels (zl) together with geometric layer depths (dzc), gravity acceleration at layer centre (gc) and refractive index at atmospheric pressure levels (refil) become input variables of the geometry model;
- Pseudo-layers (surface, cloud, TOA) and the observation level are delimitors of the optical path. In the reference release of 4A/OP, the user is required to specify level numbers, linked to the atmospheric pressure level numerotation (variables *level1, level2, levobs*). User friendliness is limited, because a viewing geometry cannot be defined without knowledge of the atmospheric pressure levels, i.e., the same optical path may have to be defined differently for different atm4a$(ATM).ddb. Moreover, optical path delimitors are discrete since linked to the atmospheric pressure levels. In the modified release, the user specifies the optical path delimitors independently of atmospheric pressure levels and continuously. For convenience, the user specifies these delimitors in pressure coordinates for Earth viewing geometry, and in height coordinates in limb viewing geometry. The geometric module *calcsphtraj* computes the corresponding layer indices *level1, level2, levobs* and adjusts the thermodynamic state of the upper and lower layers if user specifications do not correspond to atmospheric pressure levels.

User specification of the optical path are summarised, separately for Earth view and limb geometry

parameter		Earth view	Limb view
pzdown	lower limit of optical path	pressure in hPa	altitude in km
pzup	upper limit of optical path	pressure in hPa	altitude in km
pzobs	observation level	pressure in hPa	altitude in km
trajet	optical path orientation	unchanged	unchanged
sectta1	(secant of) primary path angle	scaled to z=0	< 1, otherwise without function
sectta2	(secant of) secondary path angle	scaled to z=0	without function
secttas	(secant of) solar incidence angle	scaled to z=0	with function only if trajet=DOWNUP and pzdown > zgtan
zgtan	geometric tangent height	without function	defines height where local zenith angle 90°

The various steps of the modified geometry module *calsphtraj* are:

- if limb view and if refraction: correction of the geometric tangent height:
  - computation of refractive index at tangent height (2 iterations),
  - tangent height correction (2 iterations),
  - if corrected tangent height ztan negative: compute zenith angle at surface and stop program;
- if limb view, trajet =DOWNUP, pzdown below ztan: set pzdown=ztan;
- locate pzdown within atmospheric levels:
  - if below the lowest level nlevatm: depth of layer nlevatm is increased, the layer temperature and mixing ratios as well as the surface temperature are conserved:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

i. set level2 = nlevatm; the lowest level pl(nlevatm), zl(nlevatm) is lowered to pzdown (in the limits zl(nlevatm) ≥ 0); correct dzc(nlevatm-1), gc (nlevatm-1), pc(nlevatm-1), gc(nlevatm), pc(nlevatm), refil(nlevatm) accordingly,

- else if between levels nl-1 and nl, except if limb view and trajet=DOWNUP and zl(nl-1)-ztan < minimum layer depth (Cst_minlay): depth of layer nl-1 is reduced, layer nl becomes pseudo-layer of depth 0, the surface temperature is conserved:
  i. set level2 = nl; the lower level pl(nl), zl(nl) is raised according to pzdown; correct dzc(nl-1), gc(nl-1), pc(nl-1), gc(nl), pc(nl), refil(nl) accordingly,
  ii. if Earth view, the layer temperature and mixing ratios are corrected according to the new layer depth nl-1;
- else (the lower layer depth is too small that its effective zenith angle can be computed numerically): layer nl is discarded:
  i. set level2 = nl-1; the lowest level zl(nl-1) is lowered to pzdown=ztan; correct dzc(nl-2); set pc(nl-1)=pl(nl-1);

- locate pzup within atmospheric levels
  - if above the highest level nl=1: there is no atmosphere between the upper atmospheric level and the specified upper level, no correction of the upper layer/level is applied:
    i. set level1 = 1; set pc(0) = pl(1);
  - else if between levels nl+1 and nl: depth of layer nl is reduced, layer nl-1 becomes pseudo-layer of depth 0; the layer temperature and mixing ratios as well as the top temperature are conserved,
    i. set level1 = nl; the upper level pl(nl), zl(nl) is lowered to pzup; correct dzc(nl), gc(nl), pc(nl), gc (nl-1), pc(nl-1), refil(nl) accordingly;
- define observation level levobs: the observer is situated at one of redefined levels nl=level1, level2:
  - if pzobs equal to or above pzup: set levobs=level1,
  - else if pzobs equal to or below pzdown: set levobs=level2,
  - otherwise locate pzobs within atmospheric levels level2 ≥ nl+1 > nl ≥ level1:
    i. if trajet DOWN or UPDOWN: set levobs=nl+1 (choose the lower level),
    ii. else if trajet UP or DOWNUP: set levobs=nl (choose the upper level);
- computation of the effective local zenith angle per layer nc=level1,level2-1 and their secant values sectta1lev(:), sectta2lev(:), secttaslev(:):
  - initialise to 0,
  - if Earth view geometry: calculated as function of the user specified values sectta1, sectta2, secttas at altitude 0 and of the refractive index refil(level2):
    i. at each level nl=level1,level2:

$$tta(nl) = a\sin\left[\frac{refil(level2)}{refil(nl)} \cdot \frac{R_Earth}{R_Earth + z(nl)} \cdot \sin\left(a\cos\left(1/sectta\right)\right)\right] \qquad (1.),$$

ii. sectta1lev is calculated in all cases,

iii. sectta2lev is calculated if trajet DOWNUP or UPDOWN,

iv. secttaslev is calculated if trajet DOWN or DOWNUP and if secttas ≥ 1,

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- if limb view geometry: sectta1lev is calculated as function of the tangent height ztan and the refraction index at tangent height refitan, sectta2lev and secttaslev are calculated as function of a standard value at the height of the reflection layer,
  i. at each level nl=level1,level2,
  ii. sectta1lev is calculated in all cases:

$$tta1(nl) = a\sin\left[\frac{refi\tan}{refil(nl)} \cdot \frac{R_Earth + z\tan}{R_Earth + z(nl)}\right] \quad (2.),$$

  iii. sectta2lev is calculated if trajet DOWNUP and pzdown=ztan (unobstructed limb view):

$$tta2(nl) = tta1(nl) \quad (3.),$$

  iv. sectta2lev is also calculated if trajet DOWNUP and pzdown > ztan with 55° incidence angle at level2 (reflection from sky):

$$tta2(nl) = a\sin\left[\frac{refil(level2)}{refil(nl)} \cdot \frac{R_Earth + z(level2)}{R_Earth + z(nl)} \cdot \sin\left(a\cos\left(/SECTTA2_0\right)\right)\right] \quad (4.),$$

  v. sectta2lev is also calculated if trajet UPDOWN with 55° incidence angle at level1 (reflection from surface):

$$tta2(nl) = a\sin\left[\frac{refil(level1)}{refil(nl)} \cdot \frac{R_Earth + z(level1)}{R_Earth + z(nl)} \cdot \sin\left(a\cos\left(/SECTTA2_0\right)\right)\right] \quad (5.),$$

  vi. secttaslev is calculated if secttas ≥ 1, if trajet DOWNUP and pzdown > ztan with specified solar incidence angle at level2:

$$ttas(nl) = a\sin\left[\frac{refil(level2)}{refil(nl)} \cdot \frac{R_Earth + z(level2)}{R_Earth + z(nl)} \cdot \sin\left(a\cos\left(/secttas\right)\right)\right] \quad (6.);$$

- calculate secant of effective layer zenith angles secttalev(:) from level zenith angles tta(:) in each layer nc=level1,level2-1

$$secttalev(nc) = \cos\left[\left(tta(nl) - tta(nl+1)\right)/2\right] / \cos\left[\left(tta(nl) + tta(nl+1)\right)/2\right] \quad (7.).$$

1.2.1.10. init_contin

1.2.1.10.1 Type

init_contin is a subroutine used by the main program `aaaa.f90`.
init_contin is referenced in the library `aaaa.a`.

1.2.1.10.2 Function

The subroutine *init_contin* has been made to initialize the O_2, N_2 and H_2O continua computation. It reads the required coefficients in the files *name_h2o, name_n2* and *name_o2* (directory `input`).

1.2.1.10.3 Subordinates

The subroutine *init_contin* uses the module *error* and *constants*.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.10.4 Dependencies

It depends on the existence of the continuum coefficient files *name_h2o, name_n2* and *name_o2*.

1.2.1.10.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
name_h2o	Filename for H_2O continuum	string (len=400 max)	-	i	Ini: makefile	foreignself_h2o.coef
name_n2	Filename for N_2 continuum	string (len=400 max)	-	i	Ini: makefile	foreign_n2.coef
name_o2	Filename for O_2 continuum	string (len=400 max)	-	i	Ini: makefile	foreign_o2.coef
maxpth2o	Maximun number of continuum coefficients for H2O	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpth2o
nValO2	Number of tabulated values for O_2	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nValO2
nValN2	Number of tabulated values for N_2	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nValN2
Output						
freqch2o	Wave numbers corresponding to the H_2O continuum coefficients	r[maxpth2o]	cm^{-1}	o	Ini: file name_h2o	
coselfh2o	Continuum coefficients for H_2O at 296 K	r[maxpth2o]	-	o	Ini: file name_h2o	
ratih2o	Ratio Self/broadening (Absorption coefficient)	r[maxpth2o]	-	o	Ini: file name_h2o	
frach2o	Correction Coefficients of the water Vapour continuum at 296K due to the differences between the RT model used by clough and the 4A RT model	r[maxpth2o]	-	o	Ini: file name_h2o	
SigRefN2	Wavenumbers for N_2	r[nValN2]	cm^{-1}	o	Ini: file name_n2	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
B0N2	Density normalized absorption coefficient B at 296 K for pure N_2	r[nValN2]	cm^{-1} $Amagat^{-2}$	o	Ini: file name_n2	
Beta0N2	Parameter Beta of the temperature dependence	r[nValN2]	K	o	Ini: file name_n2	
SigRefO2	Wavenumbers for O_2	r[nValO2]	cm^{-1}	o	Ini: file name2_o2	
B0O2	Density normalized absorption coefficient B at 296 K for pure O_2	r[nValO2]	cm^{-1} $Amagat^{-2}$	o	Ini: file name2_o2	
Beta0O2	Parameter Beta of the temperature dependence	r[nValO2]	K	o	Ini: file name2_o2	
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if reading is successful Err_error otherwise

1.2.1.10.6 Processing

init_contin reads the data required for the computation of the collision-induced absorption by pure N_2 in the fundamental band near 4.3 microns. These data can then be used by the function *contN2*. It also reads the data required for the computation of the collision-induced absorption by pure O_2 in the fundamental band near 6.3 microns. These data can then be used by the function *contO2*. The following articles describes the continua computation for:

- O_2:
  - Thibault *et al*, Applied Optics, Vol. 36, pp. 563-567, 1997 ([Ref. 14]);
- N_2:
  - Lafferty *et al*, Applied Optics, Vol. 35, N. 30, pp. 5911-5917, 1996 ([Ref. 13]);
- H_2O:
  - Clough, CKD version 2.1 reajusted with HIS or ARIES measured spectra ([Ref. 11], [Ref. 12])

The programs dedicated to the N_2 and O_2 continua computation have been developped by J.M. Hartmann, LPMA, Orsay (hartmann@lpma.u-psud.fr).

1.2.1.11. peicom

1.2.1.11.1 Type

peicom is a subroutine used by the main program `aaaa.f90`.

peicom is referenced in the library `aaaa.a`.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.11.2 Function

The subroutine *peicom* has been made to define the location of the computation points in the Atlas through a boolean array.

1.2.1.11.3 Subordinates

The subroutine *peicom* uses the module *precision*.

1.2.1.11.4 Dependencies

It depends on the existence of the ASCII Atlas index file atl$(RESOL)index.dsf and the binary Atlas files. Before this component is called for the second time, the parameters listed in the file atl$(RESOL)index.dsf must have been already read as well as the header of the Atlas files via the subroutines *liread*.

1.2.1.11.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
init	Initialization flag	int	-	i	Main program aaaa	1 or 0
iabsc	Abscissa array in the non regular frames corresponding to the indices of the points that are computed in the Atlas	int[nmax]	-	i	Ini: header of Atlas files	
iadf	Total number of computed points (samples) in the frame characterizing the spectral domain in progress	int	-	i	Ini: header of Atlas files	
nmax	Maximum number of computed points (wave numbers) in a spectral domain	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Input/Output						
idgcom	Boolean array defining the computation points in the Atlas	l[nmax]	-	i/o		true or false

1.2.1.11.6 Processing

The Atlases are built as non regular discretization frames (in wave numbers). Two different cases are possible:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- if init = 1, idgcom is initialized to "false"
- if init = 0, idgcom is initialized to "true" for all values of the index array iabsc

iabsc is defined during the Atlas creation and it takes into account every atmospheric layer and the 12 temperature profiles. The sub-routine *peicom* is called for every molecule then all common points are obtained at the end of the molecule loop. idgcom is then made up at each molecule loop.

1.2.1.11.7 Variable classification

Symbol	Description	Type	Unit	Classification
iabsc	Abscissa array in the non regular frames corresponding to the indices of the points that are computed in the Atlas	int[nmax]	-	double: iabcom in aaaa
iadf	Total number of computed points (samples) in the frame characterizing the spectral domain in progress	int	-	double: iadcom in aaaa

1.2.1.12. liread

1.2.1.12.1 Type

liread is a subroutine used by the main program `aaaa.f90`, and by the subroutine `calcsig.f90`.

liread is referenced in the library `aaaa.a`.

1.2.1.12.2 Function

The subroutine *liread* has been made to read the direct access (binary) files of the optical thickness Atlases (directory `datatl`) and arrange the optical thicknesses.

1.2.1.12.3 Subordinates

The subroutine *liread* calls the subroutines *readatlas_header* and *readatlas*. It uses the modules *error*, *precision* and *modatlas*.

1.2.1.12.4 Dependencies

It depends on the existence of the files of the optical thickness Atlases.

1.2.1.12.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
init	Initialization flag	int	-	i	Main program aaaa	1 or 0
nds	Index number of the spectral domain in progress	int	-	i	Main program aaaa	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
nl	Index number of the pressure level in the reference vertical discretization	int	-	i	Subroutine calcsig	
nc	Identification of the molecule corresponding to the Atlas in progress	int	-	i	Main program aaaa	
np	Index number of the temperature profile among the 12 reference temperature profiles	int	-	i	Subroutine calcsig	
maxpt	Maximum number of computed points (wave numbers) in the whole spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
nprof	Number of reference temperature profiles	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nprof
maxpt2	Maximum number of abscissa points ; abscissa points = computed points + transitions	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt2
fatl	Name of the input file to read	string (len = 400 max)	-	i	Main program aaaa	Defined in aaaa
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Input/Output						
Atlas_Header	File header of the Atlas file to read	type(type_ Atlas_Header)	-	i/o		defined in module modatlas
Output						
freq1	Lower limit (starting wave number) of the spectral domain	d	cm^{-1}	o	Ini: Atlas file of the molecule nc	
freq2	Upper limit (stopping wave number) of the spectral domain	d	cm^{-1}	o	Ini: Atlas file of the molecule nc	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
nmax	Number of computed points in the spectral domain in progress that must be the same for all the molecules in order to optimize the interpolation (same discretization grid for all the molecules)	int	-	o	Ini: Atlas file of the molecule nc	≤ maxpt It depends on the minimum alpha Doppler of the reference molecule in the spectral domain at freq1 and reference minimum temperature
ikeep	Number of computed points	int	-	o	Ini: Atlas file of the molecule nc	
iabsc	Abscissa array of the points that are computed in the Atlas in progress for the level nl, a temperature profile np and the molecule nc	int[maxpt2]	-	o	Ini: Atlas file of the molecule nc	
sfu	Decoded optical thicknesses	r[maxpt]	-	o	Ini: Atlas file of the molecule nc	
iadcom	Total number of computed points (samples) defining the frame that characterizes the spectral domain in progress and corresponding to the points shared by all involved molecules	int	-	o	Ini: Atlas file of the molecule nc	
iabcom	Abscissa array in the non regular frames corresponding to the indices of the points that are computed in the Atlas	int[maxpt]	-	o	Ini: Atlas file of the molecule nc	
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if reading is successful Err_error otherwise

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.12.6 Processing

Two different read cases are possible:

- if init = 1, the header file is read;
- else, optical thicknesses are read.

The various steps of *liread* are:

- If init = 1:
  - Read the Atlas header (subroutine *readatlas_header*),
  - Specify the wave number limits of the involved spectral domain freq1 and freq2,
  - Specify the abscissa of the computed points in the non regular frames that is shared by both the profiles and levels for the current Atlas molecule;
- Else
  - Read the Atlas for each level nl and each temperature profile np (subroutine *readatlas*),
  - Arrangement of optical thicknesses sfu:
    - Restoring peak spectral lines and between the spectral lines,
    - Remove the transition abscissa from iabsc then iabsc contains the abscissa of the computed points for the present Atlas only: iabsc = iabsc - transitions ; useful for very small values of optical thicknesses that are stored as the mean value on a wave number range where optical thicknesses are below a given threshold.

1.2.1.12.7 Variable classification

Symbol	Description	Type	Unit	Classification
freq1	Lower limit (starting wave number) of the spectral domain	r	cm^{-1}	double: f1 in calcsig
freq2	Upper limit (stopping wave number) of the spectral domain	r	cm^{-1}	double: f2 in calcsig
nl	Index number of the pressure level in the reference vertical discretization	int	-	double: nlatl in calcsig
ikeep	Number of computed points	int	-	double: iadloc2 or iadloc1 in calcsig
iabsc	Abscissa array of the points that are computed in the Atlas in progress for the level nl, a temperature profile np and the molecule nc	int[maxpt2]	-	double: iabsc2 or iabsc1 in calcsig
sfu	Decoded optical thicknesses	r[maxpt]	-	double: tab3 in calcsig

1.2.1.13. readatlas_header

1.2.1.13.1 Type

readatlas_header is a subroutine used by the subroutine `liread.f90`.
readatlas_header is defined within the module `modatlas`.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.13.2 Function

The subroutine *readatlas_header* has been made to read the header of the direct access (binary) files of the optical thickness Atlases.

1.2.1.13.3 Subordinates

The subroutine *readatlas_header* uses the modules *precision, model* and *modinouttools*. It calls the subroutines *open_file_r, read_field* and *close_file* written in C and defined in the module *modinouttools.*

1.2.1.13.4 Dependencies

It depends on the existence of the files of the optical thickness Atlases.

1.2.1.13.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
file	Name of the input file to read	string (len = 400 max)	-	i	Main program aaaa	Defined in aaaa
Output						
Atlas_Header	File header of the file to read	type(type_ Atlas_Header)	-	o		defined in module modatlas
iostatus	Error status	int	-	o		0 if reading is successful 1 otherwise

1.2.1.13.6 Processing

This subroutine is dedicated to the reading of the atlas file header.

1.2.1.14. readatlas

1.2.1.14.1 Type

readatlas is a subroutine used by the subroutine `liread.f90`.
readatlas is defined within the module `modatlas`.

1.2.1.14.2 Function

The subroutine *readatlas* has been made to read the body of the direct access (binary) files of the optical thickness Atlases.

1.2.1.14.3 Subordinates

The subroutine *readatlas* uses the modules *precision, model* and *modinouttools*. It calls the subroutines *open_file_r, read_field* and *close_file* written in C and defined in the module *modinouttools.*

1.2.1.14.4 Dependencies

It depends on the existence of the files of the optical thickness Atlases.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.14.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
file	Name of the input file to read	string (len = 400 max)	-	i	Main program aaaa	Defined in aaaa
Atlas_Header	File header of the file to read	type(type_ Atlas_Header)	-	i	Main program aaaa	defined in module modatlas
Nl	Index number of the pressure level in the reference vertical discretization	int	-	i	Subroutine calcsig	
Np	Index number of the temperature profile among the 12 reference temperature profiles	int	-	i	Subroutine calcsig	
Output						
Atlas_Body	File body of the file to read	type(type_ Atlas_Body)	-	o		defined in module modatlas
iostatus	Error status	int	-	o		0 if reading is successful 1 otherwise

1.2.1.14.6 Processing

This subroutine is dedicated to the reading of the atlas file body.

1.2.1.15. inouttools

1.2.1.15.1 Type

The program inouttools is a C program. It contains reading and writing subroutines. It is called by the module *modatlas* (see section 1.2.11.9).

1.2.1.15.2 Function

The program inouttools contains subroutines for direct access to read or/and write in binary files. These binary files are the optical thickness Atlas files.

1.2.1.15.3 Subordinates

The program inouttools contains the subroutines: *swap_endianess*, *array_swap_endianess*, *open_file_ad*, *open_file_rw*, *open_file_w*, *write_tab*, *write_field*, *open_file_r*, *read_tab*, *read_field* and *close_field*.

1.2.1.15.4 Dependencies

None.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.15.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
open_file_r						
Input						
Filename	The name of the file to read	string (len = 400 max)	-	i	modatlas	
Output						
fid_ptr	File identification (pointeur)	int	-	o	modatlas	
read_field						
Input						
fid_ptr	File identification (pointeur)	int	-	i	modatlas	
fieldtype	Parameter for the size of the block	int	-	i	modinouttools	Defined in modinouttools
size	Size of the element to be read	int	-	i	modatlas	
offset	Offset	int	-	i	modatlas	
Output						
field	File body of the file to read	type_Atlas_Body	-	o	modatlas	
close_file						
Input						
filename	File identification (pointeur)	string (len = 400 max)	-	i	modatlas	
Output						
fid_ptr	File identification (pointeur)	Int	-	o	modatlas	

Processing

None.

1.2.1.16. calcsig

1.2.1.16.1 Type

calcsig is a subroutine used by the main program `aaaa.f90`.

calcsig is referenced in the library `aaaa.a`.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.16.2 Function

The subroutine *calcsig* has been made to extract the coded optical thicknesses from the Atlas files and to calculate (interpolate) the optical thicknesses for the meteorological state that is simulated.

1.2.1.16.3 Subordinates

The subroutine *calcsig* calls the subroutines *dichotom*, *liread*, *inttab2* and *continua*. It uses the modules *error*, modatlas and *constants*.

1.2.1.16.4 Dependencies

It depends on the existence of the files of the optical thickness Atlases.

1.2.1.16.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxpt	Maximum number of computed points (wave numbers) in the whole spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
maxpt2	Maximum number of abscissa points ; abscissa points = computed points + transitions	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt2
nlevmax	Maximum number of vertical levels	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nlevelmax
ncmax	Maximum number of molecules allowed in a spectral domain	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncmax
ncjacmax	Maximum number of molecules allowed in a spectral domain to calculate thier Jacobians	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncjacmax
iabcom	Abscissa array in the non regular frames corresponding to the indices of the iadcom points that are computed in the Atlas	int[maxpt]	-	i	Main program aaaa	
iadcom	Number of computed common points in the the spectral domain in progress corresponding to the points shared by all	int	-	i	Main program aaaa	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	involved molecules					
nmax	Number of computed points in the spectral domain in progress that must be the same for all the molecules (<or= maxpt) in order to optimize the interpolation (same discretization grid)					≤ maxpt It depends on the minimum alpha Doppler of the reference molecule in the spectral domain at freq1 and reference minimum temperature
nu	Wave numbers common to the extraction of the optical thicknesses	d[maxpt]	cm^{-1}	i	Main program aaaa	
pdnu	Calculation resolution	d	cm^{-1}	i	Main program aaaa	
nds	Index number of the spectral domain in progress	int	-	i	Main program aaaa	
maxpth2o	Maximun number of continuum coefficients for H_2O	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpth2o
idsel	Array of indexes to select molecules among the 42 gases available in the GEISA and 3 CFCS	int[ncorps]	-	i	Ini: parameter file para4a.dtp	1 or 0
nselds	Number of selected molecules in the Atlas in progress	int	-	i	Main program aaaa	
numselds	Indexes of these selected molecules	int	-	i	Main program aaaa	
jselds	Number of selected molecules in the Atlas in progress for which the Jacobians are calculated	int	-	i	Main program aaaa	
jacselds	Indexes of these selected molecules	int[jselds]	-	i	Main program aaaa	
level1	First level (up) of the atmosphere	int	-	i	Ini: parameter file para4a.dtp	
level2	Last level (down) of the atmosphere	int	-	i	Ini: parameter file para4a.dtp	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
nlevref	Number of reference pressure levels	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nlevref
nprof	Number of reference temperature profiles	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nprof
ncorps	Maximum number of atmospheric molecules	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncorps
pl	Pressure levels of the simulated atmospheric profiles	r[nlevmax]	hPa	i	Ini: file atm4a$(ATM) .ddb	
plref	Reference pression levels	r[nlevref]	hPa	i	Ini: file atmref.dsf	
pc	Pressure values at the middles of the layers delimited by the levels pl	r[0:nlevmax]	hPa	i		
pcref	Reference pression values for each layer delimited by the pression levels	r[nlevref]	hPa	i	Ini: file atmref.dsf	
tc	Temperature values at the layer middles of the atmospheric state to simulate	r[0:nlevmax]	K	i		
tcref	Reference temperature values for each layer	r[nlevref, nprof]	K	i	Ini: file atmref.dsf	
roc	Molecule concentration values at the layer middles of the atmospheric state to simulate, weighted with sectta1lev	r[nlevmax, ncorps]	g/g	i		
rocref	Reference molecule mixing ratio for each layer	r[nlevref, ncorps]	g/g	i	Ini: file atmref.dsf	
gc	gravity acceleration at geometric layer centre	r[0:nlevmax]	m s^{-2}	i	Subroutine atmsph	
dzc	geometric layer depth	r[0:nlevmax]	km	i	Subroutine atmsph	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
f1	Lower limit (starting wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	
f2	Upper limit (stopping wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	
freqch2o	Wave numbers corresponding to the H_2O continuum coefficients	r[maxpth2o]	cm^{-1}	i	Ini: file name_h2o	
coselfh2o	Continuum coefficients for H_2O at 296 K	r[maxpth2o]		i	Ini: file name_h2o	
ratih2o	Ratio Self/broadening (Absorption coefficient)	r[maxpth2o]		i	Ini: file name_h2o	
frach2o	Correction Coefficients of the water Vapour continuum at 296K due to the differences between the RT model used by clough and the 4A RT model	r[maxpth2o]		i	Ini: file name_h2o	
sectta1lev	Secant value of the primary path angle per layer	r	-	i	Ini: parameter file para4a.dtp	
contin	Continua (H_2O, N_2, O_2) contribution flag	string (len=6 max)	-	i	Ini: parameter file para4a.dtp	YES or NO
jacob	Jacobian calculation flag	string (len=6 max)	-	i	Ini: parameter file para4a.dtp	YES or NO
intopt	Indicator for interpolation in temperature of the optical thicknesses	string (len=6 max)	-	i	Ini: parameter file para4a.dtp	YES or NO
nValO2	Number of tabulated values for O_2	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nValO2
nValN2	Number of tabulated values for N_2	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nValN2
StpSigO2	Tabulation step of the O_2 parameters	r	cm^{-1}	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ StpSigO2

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
StpSigN2	Tabulation step of the N_2 parameters	r	cm^{-1}	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ StpSigN2
SigRefO2	Wavenumbers for O_2	r[nValO2]	cm^{-1}	i	Ini: file name2_o2	
B0O2	Density normalized absorption coefficient B at 296 K for pure O_2	r[nValO2]	cm^{-1} Amagat^{-2}	i	Ini: file name2_o2	
Beta0O2	Parameter Beta of the temperature dependence	r[nValO2]	K	i	Ini: file name2_o2	
SigRefN2	Wavenumbers for N_2	r[nValN2]	cm^{-1}	i	Ini: file name_n2	
B0N2	Density normalized absorption coefficient B at 296 K for pure N_2	r[nValN2]	cm^{-1} Amagat^{-2}	i	Ini: file name_n2	
Beta0N2	Parameter Beta of the temperature dependence	r[nValN2]	K	i	Ini: file name_n2	
fatl	Name of the input file to read	string[ncmax] (len=400max)	-	i	Main program aaaa	Defined in aaaa
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Atlas_Header	File header of the Atlas file to read	type_Atlas_ Header [ncorps]	-	i	Main program aaaa	defined in module modatlas
Output						
iabsc1	Abscissa array of the points that are computed in the Atlas in progress for tcref (nl,np)	int[maxpt2]	-	o	Ini: Atlas file of the molecule nc	
iabsc2	Abscissa array of the points that are computed in the Atlas in progress for tcref (nl,np+1)	int[maxpt2]	-	o	Ini: Atlas file of the molecule nc	
sig	Optical thickness	r[maxpt, 0:nlevmax]	-	o		
dtsig	Derivative of the optical thickness with respect to the temperature	r[maxpt, 0:nlevmax]	K^{-1}	o		
dcsig	Derivative of the optical thickness with	r[maxpt, 0:nlevmax,	$(g/g)^{-1}$	o		

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	respect to molecule concentration	ncjacmax]				
ondo	Wave numbers corresponding to the abscissa of tab3 (see below)	d[maxpt]	cm^{-1}	o		
tab1	Optical thicknesses tab3 read in the Atlas (liread) for tcref(np) and defined for ondo interpolated on the wave numbers nu	r[maxpt]	-	o		
tab2	Optical thicknesses tab3 read in the Atlas (liread) for tcref(np+1) and defined for ondo interpolated on the wave numbers nu	r[maxpt]	-	o		
tab3	Optical thicknesses read in the Atlas via liread	r[maxpt]	-	o		
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if no error occurs in called subroutines Err_error otherwise

1.2.1.16.6 Processing

The subroutine *calcsig* uses the discrete equations of the radiative transfer and Jacobians described in the description part of the model physics (Part II 3).

The final optical thickness (excluding CFC and continua contribution) in a layer nl at wave number index iad sig(nl,iad) needs to be corrected by the factor Cst_g/gc(nl). This is because the geometric layer depth (being inversely proportional to the local gravity acceleration at a given pressure) is calculated in the atlas reference layers with constant g. Thus:

- the factor alfa(1:2) has to be multiplied with Cst_g/gc(nl);
- dzc(nl) replaces the explicit layer height computation for continua optical thickness contribution.

The frame representing the abscissa common to all molecules is always the same whatever the reference profile is BUT it does not necessarily contain the same number of significative points.

The various steps of *calcsig* are:

- Locate the pressure levels pc in the reference scale pcref by using the dichotomy technique (subroutine *dichotom*):
  - Calculate the contribution alfa of the layer depth plref(nlatl+1)-plref(nlatl) on which the reference optical depths at pcref(nlatl) have been calculated:

alfa(i)=alfa(i)*[pl(nl+1)-pl(nl)]/[plref(nlatl(i)+1)-plref(nlatl(i))],

with i starting at 1 and stopping at ifin.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

nl is the index number of the simulated profile level.

ifin is the total number of reference levels that surround the level nl of the profile to simulate: 1 or 2.

nlatl and nlatl+1 correspond to the index numbers of the reference levels that surround the level nl;

- Look for the two reference temperature profiles np and np+1 (tcref(nl,np) and tcref(nl,np+1)) that surround the temperature profile to simulate, for each level nl;
- For each molecule:
  - Read the optical thicknesses for tcref(nl,np) (subroutine *liread*); the number of computed (grid) points is iadloc2; the abscissa array of the points that are computed in the Atlas is iabsc2,
  - Read the optical thicknesses for tcref(nl,np+1) (subroutine *liread*); the number of computed (grid) points is iadloc1; the abscissa array of the points that are computed in the Atlas is iabsc1,
  - Define the wave numbers ondo corresponding to the abscissa iabsc2 and iabsc1:

    ondo(iad)=sngl(pdnu)*(iabsc1(iad)-1)+f1,

    with iad starting at 1 and stopping at iadloc1,
  - Interpolate linearly (subroutine *inttab2*) the read optical thicknesses defined for the iadloc1 (respectively iadloc2) wave numbers ondo on the grid of the iadcom points common to all involved molecules. The interpolated optical thicknesses are respectively tab1 and tab2,
  - Interpolate in temperature the optical thicknesses:

    sigma=(tab1(iad)+dtsigma*dt(i)),

  considering that:

  dtsigma=[tab2(iad)-tab1(iad)]*sdtref(i),

  dt(i)=tc(nl)-tcref(nlatl(i),np(i)),

  sdtref(i)=1./[tcref(nlatl(i),np(i)+1)-tcref(nlatl(i),np(i))],

  with iad starting at 1 and stopping at iadcom and i starting at 1 and stopping at ifin,
  - Correct the mixing ratio:

    sigma=sigma*rapc(i),

    with rapc(i)=alfa(i)*roc(nl,nc)/rocref(nlatl(i),nc),
  - Then the total optical thickness sig of the layer nl and the wave number nu(iad) is the sum over the total number of involved molecules (and with the contribution of both reference layers if ifin=2):

    sig(iad,nl)=sig(iad,nl)+sigma,
  - If jacob='YES', the partial derivatives of the optical thickness with respect to the temperature dtsig and concentrations dcsig are calculated. They are required for the computation of the corresponding Jacobians:

    dtsig(iad,nl)=dtsig(iad,nl)-dtsigma*rapc(i),

    dcsig(iad,nl,nbcj)=dcsig(iad,nl,nbcj)-sigma/roc(nl,nc);
- Incorporate the continuum contribution:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Calculate the optical thicknesses for the continua cnt (subroutine *continua*) and interpolate them on the wave numbers nu (subroutine *inttab2*),
- Add cnt to sig.

1.2.1.16.7 Variable classification

Symbol	Description	Type	Unit	Classification
f1	Lower limit (starting wave number) of the spectral domain	r	cm^{-1}	double: freq1 in aaaa
f2	Upper limit (stopping wave number) of the spectral domain	r	cm^{-1}	double: freq2 in aaaa

1.2.1.17. continua

1.2.1.17.1 Type

continua is a subroutine used by the subroutine `calcsig.f90`.

continua is referenced in the library `aaaa.a`.

1.2.1.17.2 Function

The subroutine *continua* has been made to incorporate the O_2, N_2 and H_2O continua in the simulation of the radiative transfer. It computes the corresponding optical thicknesses.

1.2.1.17.3 Subordinates

The subroutine *continua* calls the subroutines *contO2* and *contN2*. It uses the modules *error* and *constants*.

1.2.1.17.4 Dependencies

It depends on the existence of the continuum coefficient files name_h2o, name_n2 and name2_o2.

1.2.1.17.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
contin	Continua (H_2O, N_2, O_2) contribution flag	string (len=6 max)	-	i	Ini: parameter file para4a.dtp	YES or NO
freq1	Lower limit (starting wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file	
freq2	Upper limit (stopping wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file	
pm	Mean pressure between two levels	r	hPa	i		
dp	Pressure variation between two levels	r	hPa	i		
ncorps	Maximum number of	int	-	i	Main program	Parameter

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	atmospheric molecules				aaaa	defined in module model used in aaaa: Cst_ncorps
idsel	Array of indexes to select molecules among the 42 gases available in the GEISA and 3 CFCS	int[ncorps]	-	i	Ini: parameter file para4a.dtp	1 or 0
maxpth2o	Maximun number of continuum coefficients for H_2O	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpth2o
freqch2o	Wave numbers corresponding to the H_2O continuum coefficients	r[maxpth2o]	cm^{-1}	i	Ini: file name_h2o	
coselfh2o	Continuum coefficients for H_2O at 296 K	r[maxpth2o]		i	Ini: file name_h2o	
ratih2o	Ratio Self/broadening (Absorption coefficient)	r[maxpth2o]		i	Ini: file name_h2o	
frach2o	Correction Coefficients of the water Vapour continuum at 296K due to the differences between the RT model used by clough and the 4A RT model	r[maxpth2o]		i	Ini: file name_h2o	
r1	Gaz concentration of the layer vertically	r	g/g	i		
tt	Inverse layer temperature	r	K^{-1}	i		
sectta	Secant value of the viewing angle	r	-	i		
nds	Index number of the spectral domain in progress	int	-	i	Main program aaaa	
dz	Altitude variation corresponding to dp	r	m	i		
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
nValO2	Number of tabulated values for O_2	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nValO2

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
nValN2	Number of tabulated values for N_2	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nValN2
StpSigO2	Tabulation step of the O_2 parameters	r	cm^{-1}	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ StpSigO2
StpSigN2	Tabulation step of the N_2 parameters	r	cm^{-1}	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ StpSigN2
SigRefO2	Wavenumbers for O_2	r[nValO2]	cm^{-1}	i	Ini: file name2_o2	
B0O2	Density normalized absorption coefficient B at 296 K for pure O_2	r[nValO2]	cm^{-1} Amagat^{-2}	i	Ini: file name2_o2	
Beta0O2	Parameter Beta of the temperature dependence	r[nValO2]	K	i	Ini: file name2_o2	
SigRefN2	Wavenumbers for N_2	r[nValN2]	cm^{-1}	i	Ini: file name_n2	
B0N2	Density normalized absorption coefficient B at 296 K for pure N_2	r[nValN2]	cm^{-1} Amagat^{-2}	i	Ini: file name_n2	
Beta0N2	Parameter Beta of the temperature dependence	r[nValN2]	K	i	Ini: file name_n2	
Output						
cnt	Optical thicknesses deduced from the continua contribution	r[2]	-	o		
dtcnt	Derivatives of cnt with respect to the temperature	r[2]	K^{-1}	o		
dch2o	Derivatives of cnt with respect to the water vapour concentration	r[2]	$(g/g)^{-1}$	o		
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if computation is successful Err_error

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
						otherwise

1.2.1.17.6 Processing

Computation of the optical thicknesses for:

- the H_2O continuum (two parts):
  - Water vapour continuum "self" and "foreign" 8 ==> 12 microns: if freq1 < 1245, the global change of the continuum at 296 K from Clough of 30% global_change_C0_H20 is equal to 0.7 and the temperature dependance coefficient of the H_20 continuum freq_depend_Co_H2O = 1800,
  - Water vapour continuum "self" and "foreign" 4.3 microns: if freq1 < 2920.0 and freq1 ≥ 1245.0 then global_change_C0_H20 = 0.7 and freq_depend_C0_H2O = 1350.

  Then a temperature correction is applied according to freq_depend_C0_H2O, and a conversion from (g/g) to mol/cm^3 and an angle correction are performed.

  The following articles describes the H_2O continua computation:

  Roberts and al, Applied Optics, Vol 15, N9, 1976,

  Report AFGL-TR-84-0128, Burch and al,

  Coefficients at 296 from Clough CKD2.1 ([Ref. 11], [Ref. 12]);
- the O_2 continuum (function *contO2*, section);
- the N_2 continuum (function *contN2*, section).

1.2.1.17.7 Variable classification

Symbol	Description	Type	Unit	Classification
freq1	Lower limit (starting wave number) of the spectral domain	r	cm^{-1}	double: f1 in calcsig
freq2	Upper limit (stopping wave number) of the spectral domain	r	cm^{-1}	double: f2 in calcsig
sectta	Secant value of the viewing angle	r	-	double: sta1=sectta1lev(nl) in calcsig

1.2.1.18. contO2

1.2.1.18.1 Type

contO2 is a function used by the subroutine `continua.f90`.

contO2 is referenced in the library `aaaa.a`.

1.2.1.18.2 Function

The function *contO2* has been made to incorporate the absorption coefficient due to collision-induced absorption by Nitrogen in the fundamental band near 6.3 µm.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.18.3 Subordinates

The function *cont02* uses the module *constants*.

1.2.1.18.4 Dependencies

It depends on the existence of the continuum coefficient files name2_o2.

1.2.1.18.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
Sigma	WaveNumber for which the O_2 absorption is to be computed	d	cm^{-1}	i		
Ptot	Total Pressure	r	hPa	i		
Temp	Temperature.	r	K	i		
nValO2	Number of tabulated values for O_2	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nValO2
StpSigO2	Tabulation step of the O_2 parameters	r	cm^{-1}	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ StpSigO2
SigRefO2	Wavenumbers for O_2	r[nValO2]	cm^{-1}	i	Ini: file name2_o2	
B0O2	Density normalized absorption coefficient B at 296 K for pure O_2	r[nValO2]	cm^{-1} $Amagat^{-2}$	i	Ini: file name2_o2	
Beta0O2	Parameter Beta of the temperature dependence	r[nValO2]	K	i	Ini: file name2_o2	
Output						
contO2	Absorption coefficient.	r	cm^{-1}	o		

1.2.1.18.6 Processing

This function is adapted from the Double Precision Version (last change 08 April 1997) of Jean-Michel Hartmann (hartmann@lpma.u-psud.fr) converted into single precision. The corresponding article is:

F. Thibault,V. Menoux, R. Le Doucen, L. Rosenmann, J.M. Hartmann, and C. Boulet, 1997: "Infrared collision-induced absorption by O2 near 6.4 microns for atmospheric applications: measurements and empirical modelling". Applied Optics, Vol. 36, pp. 563-567 ([Ref. 14]).

This Function computes the absorption coefficient at wavenumber Sigma due to collision-induced absorption by Nitrogen in the fundamental band near 6.3 μm. O_2 in atmospheric air is assumed, i.e. the collision

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

partners are (79% N_2 + 21 % O_2). The absorption coefficient *contO2* includes the O_2 partial pressure so that the transmission of a path length L (cm) would be equal to Exp(-ContO2*L).

The various steps of *contO2* are:

- Computation of O_2-O_2 absorption for unit density from tabulated data;
- Correction for effect of O_2-N_2 collisions;
- Convertion of Pressures to Amagats (ideal gas) --> Final Result contO2.

1.2.1.18.7 Variable classification

Symbol	Description	Type	Unit	Classification
Sigma	WaveNumber for which the O_2 absorption is to be computed	d	cm^{-1}	double: freq1 or freq2 in continua

1.2.1.19. contN2

1.2.1.19.1 Type

contN2 is a function used by the subroutine `continua.f90`.

contN2 is referenced in the library `aaaa.a`.

1.2.1.19.2 Function

The function *contN2* has been made to incorporate the absorption coefficient due to collision-induced absorption by Nitrogen in the fundamental band near 4.3 µm.

1.2.1.19.3 Subordinates

The function *contN2* uses the module *constants*.

1.2.1.19.4 Dependencies

It depends on the existence of the continuum coefficient files name_n2.

1.2.1.19.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
Sigma	WaveNumber for which the N_2 absorption is to be computed	d	cm^{-1}	i		
Ptot	Total Pressure	r	hPa	i		
Temp	Temperature.	r	K	i		
nValN2	Number of tabulated values for N_2	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nValN2
StpSigN2	Tabulation step of the N_2 parameters	r	cm^{-1}	i	Main program aaaa	Parameter defined in

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
						module model used in aaaa: Cst_ StpSigN2
SigRefN2	Wavenumbers for N_2	r[nValN2]	cm^{-1}	i	Ini: file name_n2	
B0N2	Density normalized absorption coefficient B at 296 K for pure N_2	r[nValN2]	cm^{-1} $Amagat^{-2}$	i	Ini: file name_n2	
Beta0N2	Parameter Beta of the temperature dependence	r[nValN2]	K	i	Ini: file name_n2	
Output						
contN2	Absorption coefficient.	r	cm^{-1}	o		

1.2.1.19.6 Processing

This function is adapted from the Double Precision Version (last change 08 April 1997) of Jean-Michel Hartmann (hartmann@lpma.u-psud.fr) converted into single precision. The corresponding article is:

W.J. Lafferty, A.M. Solodov, A. Weber, W.B. Olson, and J.M. Hartmann 1996: "Infrared collision-induced absorption by N2 near 4.3 microns for applications in the atmosphere. Measurements and empirical modelling". Applied Optics, Vol. 35, pp. 5911-5917 ([Ref. 13]).

This Function computes the absorption coefficient at wavenumber Sigma due to collision-induced absorption by Nitrogen in the fundamental band near 4.3 µm. N_2 in atmospheric air is assumed, i.e. the collision partners are (79% N_2 + 21 % O_2). The absorption coefficient *contN2* includes the N_2 partial pressure so that the transmission of a path length L (cm) would be equal to Exp(-ContN2*L).

The various steps of *contN2* are:

- Computation of N_2-N_2 absorption for unit density from tabulated data;
- Correction for effect of N_2-O_2 collisions;
- Convertion of Pressures to Amagats (ideal gas) --> Final Result contN2.

1.2.1.19.7 Variable classification

Symbol	Description	Type	Unit	Classification
Sigma	WaveNumber for which the O_2 absorption is to be computed	d	cm^{-1}	double: freq1 or freq2 in continua

1.2.1.20. calcrad

1.2.1.20.1 Type

calcrad is a subroutine used by the main program `aaaa.f90`.

calcrad is referenced in the library `aaaa.a`.

1.2.1.20.2 Function

The subroutine *calcrad* has been made to compute the vertical integral of the radiative transfer equation in order to deduce the outgoing radiances.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.20.3 Subordinates

The subroutine *calcrad* calls the subroutines *plkdir*, *plkder* and *inttab2.* It uses the module *constants*, precision and *error*.

1.2.1.20.4 Dependencies

Before this component is called, the optical thicknesses must have been already computed via the subroutine *calcsig*.

1.2.1.20.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxb	Maximum number of computed Plank functions in a spectral domain	int	-	i	Main program aaaa	Parameter defined in aaaa: 5
maxpt	Maximum number of computed points (wave numbers) in the whole spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
nlevmax	Maximum number of vertical levels	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nlevelmax
ncjacmax	Maximum number of molecules allowed in a spectral domain to calculate thier Jacobians	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncjacmax
npttabexp	Maximum number of exponential tabulations					Parameter defined in module model used in aaaa: Cst_npttabexp
level1	First level (up) of the atmosphere	int	-	i	subroutine *calcsphtraj*	
level2	Last level (down) of the atmosphere	int	-	i	subroutine *calcsphtraj*	
levobs	Observation level	int	-	i	subroutine *calcsphtraj*	level1 ≤ *levobs* ≤ level2
iadcom	Number of computed common points in the the spectral domain in progress corresponding to the points shared by all involved molecules	int	-	i	Main program aaaa	
jselds	Number of selected	int	-	i	Main program	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	molecules in the Atlas in progress for which the Jacobians are calculated				aaaa	
tabexp	Exponential tabulation (pre-calculated values)	r[npttabexp]	-	i	Main program aaaa	
pastabexp	Exponential tabulation step	r	-	i	Main program aaaa	
sectta1lev	Secant value of the primary path angle per layer	r	-	i	Ini: parameter file para4a.dtp	
sectta2lev	Secant value of the secondary path angle per layer	r	-	i	Ini: parameter file para4a.dtp	
secttaslev	Secant value of the secondary solar path angle per layer	r	-	i	Ini: parameter file para4a.dtp	
esdist	Earth-sun distance in astronomical units	r	Astronomical unit	i	Ini: parameter file para4a.dtp	
tabemup	Wavenumber and Emissivity of the top level at initial spectral resolution	r[nonde(1),2]	cm^{-1} -	i	Ini: file spemis.dat	
tabemdown	Wavenumber and Emissivity of the lower level at initial spectral resolution	r[nonde(2),2]	cm^{-1} -	i	Ini: file spemis.dat	
nonde	Spectral dimension of tabemup and tabemdown	int[2]	-	i	Ini: file spemis.dat	
freq1	Lower limit (starting wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	
freq2	Upper limit (stopping wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	
tc	Temperature values at the layer middles of the atmospheric state to simulate	r[0:nlevmax]	K	i		
nu	Wave numbers common to the extraction of the optical thicknesses	d[maxpt]	cm^{-1}	i	Main program aaaa	
trajet	Viewing configuration	string (len=6 max)	-	i	Ini: parameter file para4a.dtp	UP, DOWNUP, DOWN, UPDOWN

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
jacob	Jacobian calculation flag	string (len=6 max)	-	i	Ini: parameter file para4a.dtp	YES or NO
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Input/Output						
sig	Optical thicknesses	r[maxpt, 0:nlevmax]	-	i/o	Main program aaaa	If no Jacobian calculation
	Intermediate radiances used for the computation of rad		W/ (m^2.strd. cm^{-1})			If Jacobian calculation
dcsig	Derivatives of the optical thickness with respect to molecule concentration	r[maxpt, 0:nlevmax, ncjacmax]	$(g/g)^{-1}$	i/o	Main program aaaa	If no Jacobian calculation
	Jacobians with respect to molecule concentration		W/ (m^2.strd. cm^{-1})/ (g/g)			If Jacobian calculation
dtsig	Derivatives of the optical thickness with respect to the temperature	r[maxpt, 0:nlevmax]	K^{-1}	i/o	Main program aaaa	If no Jacobian calculation
	Jacobians with respect to the temperature		W/ (m^2.strd. cm^{-1})/ K			If Jacobian calculation
Output						
somsig	Integral over the optical thickness	r[maxpt]	-	o		
rad	Outgoing radiances	r[maxpt]	W/ (m^2.strd. cm^{-1})	o		
to1	Transmittances for the main path	r[maxpt, 0:nlevmax]	-	o		
to2	Transmittances for the secondary path	r[maxpt, 0:nlevmax]	-	o		
btab	Plank function calculated every 5 cm^{-1}	r[maxpt]	-	o		
dbtab	Corresponding derivative	r[maxpt]		o		
nutab	Corresponding wave numbers	d[maxpt]	cm^{-1}	o		
b	btab interpolated on the common wave	r[maxpt]		o		

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	numbers nu					
db	Corresponding derivative	r[maxpt]		o		
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if no error occurs Err_error otherwise

1.2.1.20.6 Processing

The subroutine *calcrad* computes the radiance at the end of the path and the Jacobians along the path if it is required.

The various steps of *calcrad* are:

- Initialization: In order to save time, the Planck Function is calculated every 5 cm^{-1} only. The number of calculated points in the spectral domain in progress it is:

  it = nint[(freq2-freq1)/5.] + 1

  An the corresponding wave numbers nutab are:

  nutab(i) = freq1 + (i-1)*5, with i starting at1 and stopping at it
- Linear interpolation of the spectral emissivities read via *lireemis* from available to required wave numbers nu (subroutine *inttab2*);
- Initialization of the viewing configuration: determination of the surface or reflexion parameters according to the viewing configuration. Determination of the:
  - Limits of the integral computation: kush1 and kush2,
  - Direction of the integral computation: kpas1 and kpas2 (1:up or -1:down),
  - Characteristics of the reflexion layer, using spectral emissivities: kurfx and sigrfx,
  - Computation of dtsig and dcsig if it is required;
- Computation of the transmittances for:
  - The main path: to1,
  - The secondary path: to2,
  - The secondary solar path: tosun;
- Computation of the vertical integral of the radiative transfer equation and computation of the Jacobians, if it is required (for the main path, reflexion layer and secondary terrestrial and solar path):
  - Computation of the Plank functions btab for temperatures tc (subroutine *plkdir*),
  - Computation of the Plank function bsuntab for temperature tsun if solar zenith angle is less than 90 degrees (subroutine *plkdir*),
  - If it is required, computation of the Plank function derivatives with respect to the temperature dbtab (subroutine *plkder*),
  - Linear interpolation of btab (and bsuntab, dbtab) on the iadcom wave numbers nu common to all involved molecules (subroutine *inttab2*),

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Radiative transfer equation application: computation of rad (and dtsig and dcsig if it is required) including the solar contribution if solar zenith angle < 90 degrees. The Jacobians with respect to the gas mixing ratio do not exist for the surface levels. The value of dcsig at the surface levels (level1-1 and level2) is set to the Jacobian value of the spectral surface emissivity.

1.2.1.20.7 Variable classification

Symbol	Description	Type	Unit	Classification
iadcom	Number of computed common points in the the spectral domain in progress corresponding to the points shared by all involved molecules	int	-	double: nmin in aaaa

1.2.1.21. contract

1.2.1.21.1 Type

contract is a subroutine used by the main program `aaaa.f90`.

contract is referenced in the library `aaaa.a`.

1.2.1.21.2 Function

The subroutine *contract* has been made to contract the wave numbers for two different cases:

- Contraction of a "pseudo-infinite" spectrum (if respec ≥ 0.0005 cm-1 else no contraction)
- Contraction of a spectrum before the convolution

1.2.1.21.3 Subordinates

The subroutine *contract* calls the subroutines *echantini*, *echantillon* and *inttab2*. It uses the module *precision*.

1.2.1.21.4 Dependencies

Before this component is called, the high-resolution radiance must have been already computed via the subroutine *calrad* and the parameters listed in the files para4a$(RSTR).dtp must have been already read via the subroutine *para4a.*

1.2.1.21.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxpt	Maximum number of computed points (wave numbers) in the whole spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
nlevmax	Maximum number of vertical levels	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nlevelmax
ncjacmax	Maximum number of molecules allowed in	int	-	i	Main program aaaa	Parameter defined in module model

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	a spectral domain to calculate thier Jacobians					used in aaaa: Cst_ncjacmax
jselds	Number of selected molecules in the Atlas in progress for which the Jacobians are calculated	int	-	i	Main program aaaa	
level1	First level (up) of the atmosphere	int	-	i	Ini: parameter file para4a.dtp	
level2	Last level (down) of the atmosphere	int	-	i	Ini: parameter file para4a.dtp	
delds	Width of each Atlas spectral band	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
freq1	Lower limit (starting wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	
jacob	Jacobian calculation flag	string (len=6max)	-	i	Ini: parameter file para4a.dtp	YES or NO
fpoid	Transmittance calculation flag	string (len=6max)	-	i	Ini: parameter file para4a.dtp	YES or NO Computation of Jacobians and transmittances is not permitted if *conv*=NO
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Input/Output						
iadcom	Number of computed common points in the the spectral domain in progress corresponding to the points shared by all involved molecules	int	-	i/o	Main program aaaa	
iabcom	Abscissa array in the non regular frames corresponding to the indices of the iadcom points that are computed in the Atlas	int[maxpt]	-	i/o	Main program aaaa	
nmin	Number of sampling points	int	-	i/o	Main program aaaa	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
lfil	Box width (if 1: no contraction else: contraction): proportionality factor between boxes	int	-	i/o	Main program aaaa	
respec	Spectral resolution: contraction of the spectra before convolution or to define the spectral discretisation with an "infinite" resolution simulation	d	cm^{-1}	i/o	Ini: parameter file para4a.dtp	
nu	Wave numbers common to the extraction of the optical thicknesses and recomputed if the contraction is effective	d[maxpt]	cm^{-1}	i/o	Main program aaaa	
rad	Outgoing radiances	r[maxpt]	W/ (m^2.strd. cm^{-1})	i/o	Main program aaaa	
to1	Transmittances for the main path	r[maxpt, 0:nlevmax]	-	i/o	Main program aaaa	
dcsig	Jacobians with respect to molecule concentration	r[maxpt, 0:nlevmax, ncjacmax]	W/ (m^2.strd. cm^{-1})/ (g/g)	i/o	Main program aaaa	
dtsig	Jacobians with respect to the temperature	r[maxpt, 0:nlevmax]	W/ (m^2.strd. cm^{-1})/ K	i/o	Main program aaaa	
pdnu	Calculation resolution	d	cm^{-1}	i/o	Main program aaaa	
Output						
pds	Weight of every point to be contracted (0.5 for the point shared by two boxes and 1 for the others)	r[maxpt]	-	o		

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
tab1	Work table: radiance or an other quantity interpolated on the maxpt wave numbers tab2	r[maxpt]	W/ (m^2.strd. cm^{-1}) or /(g/g) or /K	o		
tab2	Wave numbers at the maximum sampling maxpt	d[maxpt]	cm^{-1}	o		

1.2.1.21.6 Processing

The subroutine *contract* allows the contraction of the radiance spectrum. The step is given by the variable respec. The minimum value of respec is 0.0005 cm^{-1}. If respec > 0.0005 cm^{-1}, a system of boxes that fit into each other is used.

The various steps of *contract* are:

- Sampling at the maximum resolution maxpt ; computation of:
  - Corresponding sampling step: pdnux = dble(delds/(maxpt-1)),
  - Corresponding wave numbers tab2,
  - Linear interpolation of the radiances rad on tab2 (subroutine *inttab2*),
  - Computation of the transmittances: linear interpolation of to1 on tab2 (subroutine *inttab2*) if it is required,
  - Computation of the Jacobians: linear interpolation of dcsig and dtsig on tab2 (subroutine *inttab2*) if it is required;
- Initialization of the sampling reduction (contraction): Computation of the box width lfil and the box weight rnorm (subroutine *echantini*);
- If lfil different from 1: Contraction of the spectrum (and of the Jacobians and transmittances if it is required) (subroutine *echantillon*):
  - Computation of the wave numbers after the contraction nu.

1.2.1.21.7 Variable classification

Symbol	Description	Type	Unit	Classification
tab2	Wave numbers at the maximum sampling maxpt	d[maxpt]	cm^{-1}	double: tab4 in aaaa

1.2.1.22. echantini

1.2.1.22.1 Type

echantini is a subroutine used by the main program `aaaa.f90`.

echantini is referenced in the library `aaaa.a`.

1.2.1.22.2 Function

The subroutine *echantini* has been made to initialize the parameters used for the contraction.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.22.3 Subordinates

It uses the module *precision*..

1.2.1.22.4 Dependencies

Before this component is called, the parameters listed in the files para4a$(RSTR).dtp must have been already read via the subroutine *para4a.*

1.2.1.22.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxpt	Maximum number of computed points (wave numbers) in the whole spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
delds	Width of each Atlas spectral band	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Input/Output						
respec	Spectral resolution: spectra contraction before convolution or to define the spectral discretisation with an "infinite" resolution simulation	d	cm^{-1}	i/o	Ini: parameter file para4a.dtp	
Output						
iadcom	Number of computed common points in the the spectral domain in progress corresponding to the points shared by all involved molecules	int	-	o		
iabcom	Abscissa array in the non regular frames corresponding to the indices of the iadcom points that are computed in the Atlas	int[maxpt]	-	o		
nmin	Number of sampling points	int	-	o		
lfil	Box width (if 1: no contraction else: contraction): proportionality factor between boxes	int	-	o		
pds	Weight of every point to be contracted (0.5 for the point shared by two boxes and 1 for the others)	r[maxpt]	-	o		
rnorm	Box total weight or box norm (energy conservation)	d	cm^{-1}	o		
pdnu	Calculation resolution	d	cm^{-1}	o		

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.22.6 Processing

The various steps of *echantini* are:

- Sampling at the maximum resolution maxpt ; computation of:
  - Calculation resolution: pdnu = delds / dble(iadcom-1);
- Sampling at the resolution respec ; computation of:
  - The number of sampling points after the contraction: nmin = nint(delds/respec),
  - The Position of the spectral domain center: lfilo = int(maxpt/2) + 1,
  - Width and half width of the boxes: lfil = int((iadcom-1)/nmin) and lfilp = int(lfil/2),
  - Weight of each point of the box to be contracted (0.5 for a point shared by two boxes and 1 for others): pds,
  - Total weight of the box or box norm: rnorm.

1.2.1.23. echantillon

1.2.1.23.1 Type

echantillon is a subroutine used by the main program `aaaa.f90`.

echantillon is referenced in the library `aaaa.a`.

1.2.1.23.2 Function

The subroutine *echantillon* has been made to initialize (sampling reduction) of a quantity by using a system of boxes that fit into each other.

1.2.1.23.3 Subordinates

It uses the module *precision*..

1.2.1.23.4 Dependencies

Before this component is called, the box characteristics must have been already initialized via the subroutine *echantini.*

1.2.1.23.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxpt	Maximum number of computed points (wave numbers) in the whole spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
iadcom	Number of computed common points in the the spectral domain in progress corresponding to the points shared by all involved molecules	int	-	i	-	
nmin	Number of sampling points	int	-	i	-	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
lfil	Box width (if 1: no contraction else: contraction): proportionality factor between boxes	int	-	i		
pds	Weight of every point to be contracted (0.5 for the point shared by two boxes and 1 for the others)	r[maxpt]	-	i		
rnorm	Box total weight or box norm (energy conservation)	d	cm^{-1}	i		
Input/Output						
tab1	Contracted quantity (spectrum or Jacobians or transmittances	r[maxpt]	W/ (m^2.strd. cm^{-1}) or /(g/g) or /K or -	o		

1.2.1.23.6 Processing

The various steps of *echantillon* are:

- Initialization:
  - Half width of the box: lfilp = int(lfil/2),
  - Position of the spectral domain center: lfilo = int(maxpt/2)+1;
- Contraction: Each point of the quantity tab1 is pds weighted and is normalized by the box norm rnorm except for both limit points for which the norm is different (rn).

1.2.1.23.7 Variable classification

Symbol	Description	Type	Unit	Classification
tab1	Contracted quantity (spectrum or Jacobians or transmittances	r[maxpt]	W/ (m^2.strd. cm^{-1}) or W/ (m^2.strd. cm^{-1}) /K or /(g/g) or -	double: rad or to1 or dtsig or dcsig in contract

1.2.1.24. cnvspe

1.2.1.24.1 Type

cnvspe is a subroutine used by the main program `aaaa.f90`.

cnvspe is referenced in the library `aaaa.a` and is defined within the module `convolution`.

1.2.1.24.2 Function

The subroutine *cnvspe* has been made to perform a convolution with a function.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.24.3 Subordinates

The subroutine *cnvspe* calls the subroutines *calcisrf* and *dichotomd*. It uses the module *error*.

1.2.1.24.4 Dependencies

Before this component is called, the high-resolution radiance must have been already computed via the subroutine *calrad,* the parameters listed in the files para4a$(RSTR).dtp must have been already read via the subroutine *para4a* and the convolution parameters must have been already initialized via the subroutine *convini.*

1.2.1.24.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxbuf	Maximum size of a record block (buffer)	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxbuf
maxpt	Maximum number of computed points (wave numbers) in the whole spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
nlevmax	Maximum number of vertical levels	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nlevelmax
ncjacmax	Maximum number of molecules allowed in a spectral domain to calculate thier Jacobians	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncjacmax
ldmax	Maximum dimension of the instrumental functions resampled for the convolution with the high-resolution spectrum	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ldmax
kpmax	Maximum dimension of the initial instrumental functions	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: 15000
iadcom	Number of computed common points in the current spectral domain corresponding to the points shared by all involved molecules	int	-	i	Main program aaaa	
level1	First level (up) of the	int	-	i	Ini: parameter	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	atmosphere				file para4a.dtp	
level2	Last level (down) of the atmosphere	int	-	i	Ini: parameter file para4a.dtp	
jselds	Number of selected molecules in the Atlas in progress for which the Jacobians are calculated	int	-	i	Main program aaaa	
jacselds	Indexes of these selected molecules	int[jselds]	-	i	Main program aaaa	
nsc	Number of convolution points	int	-	i	Main program aaaa	
nknu	Number of ISRF	int	-	i	Ini: ISRF file isrf.ddb	
irec	Record number of the beginning of each ISRF	int[nknu]	-	i	Ini: ISRF file isrf.ddb	
ldimo	Initial number of description points of the ISRF (dimension)	int[nknu]	-	i	Ini: ISRF file isrf.ddb	
ncorps	Maximum number of atmospheric molecules	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncorps
jacsel	Array of indexes to select the calculation of the Jacobian for each molecule among the 42 gases available in the GEISA database and 3 CFCS	int[ncorps]	-	i	Main program aaaa	
nu	Wave numbers common to the extraction of the optical thicknesses	d[maxpt]	cm^{-1}	i	Main program aaaa	
spo	High-resolution spectrum (radiances)	r[maxpt]	W/ (m^2.strd. cm^{-1})	i	Main program aaaa	
dto	Transmittances for the main path	r[maxpt, 0:nlevmax]	-	i	Main program aaaa	
dtj	Jacobians with respect to the temperature	r[maxpt, 0:nlevmax]	W/ (m^2.strd. cm^{-1})/ K	i	Main program aaaa	
dcj	Jacobians with respect to molecule concentration	r[maxpt, 0:nlevmax, ncjacmax]	W/ (m^2.strd. cm^{-1})/ (g/g)	i	Main program aaaa	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
cnu	Central wave numbers of each ISRF in the convolution domain (nsc wave numbers)	d[allocatable]	cm^{-1}	i	Main program aaaa	Allocated in aaaa Size: number of convolution wave numbers
wnu	Width (definition domain) of each ISRF in the convolution domain	d[allocatable]	cm^{-1}	i	Main program aaaa	Allocated in aaaa Size: number of convolution wave numbers
cwnu2	First wave number limit of each ISRF involved in the convolution	d[allocatable]	cm^{-1}	i	Main program aaaa	Allocated in aaaa Size: number of convolution wave numbers
cwnu1	Last wave number limit of each ISRF involved in the convolution	d[allocatable]	cm^{-1}	i	Main program aaaa	Allocated in aaaa Size: number of convolution wave numbers
f1	Lower limit (starting wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	
f2	Upper limit (stopping wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	
delds	Width of each Atlas spectral band	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
nuco	Central wave numbers of each ISRF	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	
dnuo	Spectral shift of each ISRF	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	
pdnuo	Initial sampling step of the instrument functions	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	
rnu	Value of the global spectral shift	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
pdnu	Calculation resolution	d	cm^{-1}	i	Main program aaaa	
shift	Indicator of a spectral shift introduction due to the ISRF	string (len=6max)	-	o	Ini: parameter file para4a.dtp	YES or NO Only used if *conv*=YES If yes, a spectral shift is applied due to the ISRF that is not centred
jacob	Jacobian calculation flag	string (len=6max)	-	i	Ini: parameter file para4a.dtp	YES or NO
fpoid	Transmittance	string	-	i	Ini: parameter	YES or NO

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	calculation flag	(len=6max)			file para4a.dtp	Computation of Jacobians and transmittances is not permitted if *conv*=NO
jap	Logical unit defining the ISRF file to read	int	-	i	Main program aaaa	Defined in aaaa: 8
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Output						
spc	Convolved radiances	r[allocatable]	W/ (m^2.strd. cm^{-1})	o		Allocated in aaaa Size: number of convolution wave numbers
dtoc	Convolved transmittances	r[allocatable, allocatable]	-	o		Allocated in aaaa Size: number of convolution wave numbers, nlevmax
dtjc	Convolved Jacobians with respect to temperature	r[allocatable, allocatable]	W/ (m^2.strd. cm^{-1})/ K	o		Allocated in aaaa Size: number of convolution wave numbers, nlevmax
dcjc	Convolved Jacobians with respect to molecule concentration	r[allocatable, allocatable, allocatable]	W/ (m^2.strd. cm^{-1})/ (g/g)	o		Allocated in aaaa Size: number of convolution wave numbers, nlevmax, ncjacmax
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if no error occurs in called subroutines or during the computation Err_error otherwise

1.2.1.24.6 Processing

The high-resolution spectrum is convolved by an instrumental function (as well as the Jacobians and transmittances if it is necessary).

The principle of the convolution processing is summarized below.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

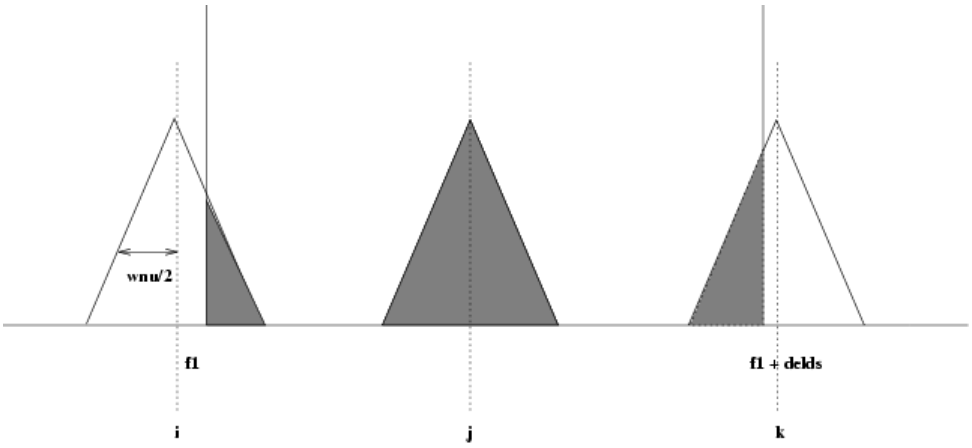


Figure 26: 4A convolution principle

For the wave number *j*, the instrument function is totally in the spectral domain. For *j*, the following value is affected:

$$crad(j) = \sum_{i=1}^{l\dim o} \left(crp(i) * rad(i) \right)$$

For the wave number *k*, a part of the instrument function belongs to the spectral domain between f1 and f1+delds. Then *crad(k)* is:

$$crad(k) = \sum_{i=k-wnu/2}^{f1+delds} \left(crp(i) * rad(i) \right)$$

The supplementary part is calculated delds farther.

By applying the same principle, *crad(i)* is obtained as follow:

$$crad(k) = crad'(i) + \sum_{i=f1}^{k+wnu/2} \left(crp(i) * rad(i) \right)$$

where *crad'(i)* is the part of the convolution that has been calculated delds before.

The various steps of *cnvspe* are:

- Definition of the points of the convolution limits nd and nf: Searching for convolution points meeting the spectral domain definition of convolution limits by using the dichotomy technique (subroutine *dichotomd*);
- For each convolution point n, n ∈ [nd,nf]:
  - Computation limits over the spectrum and the instrumental function: lcd and lcf,
  - Computation of the instrumental function from lcd to lcf for the convolution step at the wave numbers cnu (subroutine *calcisrf*),
  - Weighting of the spectral domain limits: adjustment of the limits used two times,
  - Partial convolution of the spectrum (and Jacobians and transmittances if it is required) by the instrumental function: Weighting by the function part between lcd and lcf,

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.24.7 Variable classification

Symbol	Description	Type	Unit	Classification
spo	High-resolution spectrum (radiances)	r[maxpt]	W/(m^2.strd. cm^{-1})	double: rad in aaaa
dto	Transmittances for the main path	r[maxpt, 0:nlevmax]	-	double: to1 in aaaa
dtj	Jacobians with respect to the temperature	r[maxpt, 0:nlevmax]	W/(m^2.strd. cm^{-1})/K	double: dtsig in aaaa
dcj	Jacobians with respect to molecule concentration	r[maxpt, 0:nlevmax, ncjacmax]	W/(m^2.strd. cm^{-1})/(g/g)	double: dcsig in aaaa
iadcom	Number of computed common points in the the spectral domain in progress corresponding to the points shared by all involved molecules	int	-	double: nmin in aaaa
nsc	Number of convolution points	int	-	double: iadcnv in aaaa
spc	Convolved radiances	r[allocatable]	W/ (m^2.strd. cm^{-1})	double: crad in aaaa
dtoc	Convolved transmittances	r[allocatable, allocatable]	-	double: cdtoin aaaa
dtjc	Convolved Jacobians with respect to temperature	r[allocatable, allocatable]	W/ (m^2.strd. cm^{-1})/ K	double: ctjac in aaaa
dcjc	Convolved Jacobians with respect to molecule concentration	r[allocatable, allocatable, allocatable]	W/ (m^2.strd. cm^{-1})/ (g/g)	double: ccjac in aaaa
f1	Lower limit (starting wave number) of the spectral domain	d	cm^{-1}	double: freq1 in aaaa
f2	Upper limit (stopping wave number) of the spectral domain	d	cm^{-1}	double: freq2 in aaaa
nuco	Central wave numbers of each ISRF	d[nknu]	cm^{-1}	double: nufo in aaaa
irec	Record number of the beginning of each ISRF	int[nknu]	-	double: irecf in aaaa
ldimo	Initial number of description points of the ISRF (dimension)	int[nknu]	-	double: kfmo in aaaa
pdnuo	Initial sampling step of the instrument functions	d[nknu]	cm^{-1}	double: pdnuc in aaaa

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.25. cnvspehole

1.2.1.25.1 Type

cnvspehole is a subroutine used by the main program `aaaa.f90`.

cnvspehole is referenced in the library `aaaa.a` and is defined within the module `convolution`.

1.2.1.25.2 Function

The subroutine *cnvspehole* has been made to perform a hole convolution with a function.

1.2.1.25.3 Subordinates

The subroutine *cnvspehole* calls the subroutines *calcisrf*, *dichotomd* and *dichotoml*. It uses the module *error*.

1.2.1.25.4 Dependencies

Before this component is called, the high-resolution radiance must have been already computed via the subroutine *calrad,* the parameters listed in the files para4a$(RSTR).dtp must have been already read via the subroutine *para4a* and the convolution parameters must have been already initialized via the subroutine *convini.*

1.2.1.25.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxbuf	Maximum size of a record block (buffer)	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxbuf
maxpt	Maximum number of computed points (wave numbers) in the whole spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
nlevmax	Maximum number of vertical levels	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_nlevelmax
ncjacmax	Maximum number of molecules allowed in a spectral domain to calculate thier Jacobians	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncjacmax
ldmax	Maximum dimension of the instrumental functions resampled for the convolution with the high-resolution spectrum	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ldmax
kpmax	Maximum dimension of the initial instrumental functions	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
						15000
nmin	Number of computed common points in the the spectral domain in progress corresponding to the points shared by all involved molecules	int	-	i	Main program aaaa	WARNING: this argument has the same meaning (and argument place) as iadcom in the previous routine (cnvspe: convolution with no hole)
iadcom	Number of points except hole ones. This is iadcom before interpolation	int	-	i	Main program aaaa	
iabcom	Abscissa array in the non regular frames corresponding to the indices of the points that care computed in the Atlas.	int[maxpt]	-	i	Main program aaaa	These are the indexes in nu of each spectrum (and jacobians, etc.) point.
level1	First level (up) of the atmosphere	int	-	i	Ini: parameter file para4a.dtp	
level2	Last level (down) of the atmosphere	int	-	i	Ini: parameter file para4a.dtp	
jselds	Number of selected molecules in the Atlas in progress for which the Jacobians are calculated	int	-	i	Main program aaaa	
jacselds	Indexes of these selected molecules	int[jselds]	-	i	Main program aaaa	
nsc	Number of convolution points	int	-	i	Main program aaaa	
nknu	Number of ISRF	int	-	i	Ini: ISRF file isrf.ddb	
irec	Record number of the beginning of each ISRF	int[nknu]	-	i	Ini: ISRF file isrf.ddb	
ldimo	Initial number of description points of the ISRF (dimension)	int[nknu]	-	i	Ini: ISRF file isrf.ddb	
ncorps	Maximum number of atmospheric molecules	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ncorps
jacsel	Array of indexes to select the calculation	int[ncorps]	-	i	Main program aaaa	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	of the Jacobian for each molecule among the 42 gases available in the GEISA database and 3 CFCS					
nu	Wave numbers common to the extraction of the optical thicknesses	d[maxpt]	cm^{-1}	i	Main program aaaa	
spo	High-resolution spectrum (radiances)	r[maxpt]	W/ (m^2.strd. cm^{-1})	i	Main program aaaa	
dto	Transmittances for the main path	r[maxpt, 0:nlevmax]	-	i	Main program aaaa	
dtj	Jacobians with respect to the temperature	r[maxpt, 0:nlevmax]	W/ (m^2.strd. cm^{-1})/ K	i	Main program aaaa	
dcj	Jacobians with respect to molecule concentration	r[maxpt, 0:nlevmax, ncjacmax]	W/ (m^2.strd. cm^{-1})/ (g/g)	i	Main program aaaa	
cnu	Central wave numbers of each ISRF in the convolution domain (nsc wave numbers)	d[allocatable]	cm^{-1}	i	Main program aaaa	Allocated in aaaa Size: number of convolution wave numbers
wnu	Width (definition domain) of each ISRF in the convolution domain	d[allocatable]	cm^{-1}	i	Main program aaaa	Allocated in aaaa Size: number of convolution wave numbers
cwnu2	First wave number limit of each ISRF involved in the convolution	d[allocatable]	cm^{-1}	i	Main program aaaa	Allocated in aaaa Size: number of convolution wave numbers
cwnu1	Last wave number limit of each ISRF involved in the convolution	d[allocatable]	cm^{-1}	i	Main program aaaa	Allocated in aaaa Size: number of convolution wave numbers
f1	Lower limit (starting wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	
f2	Upper limit (stopping wave number) of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	
delds	Width of each Atlas spectral band	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
nuco	Central wave numbers of each ISRF	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
dnuo	Spectral shift of each ISRF	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	
pdnuo	Initial sampling step of the instrument functions	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	
rnu	Value of the global spectral shift	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
pdnu	Calculation resolution	d	cm^{-1}	i	Main program aaaa	
shift	Indicator of a spectral shift introduction due to the ISRF	string (len=6max)	-	o	Ini: parameter file para4a.dtp	YES or NO Only used if *conv*=YES If yes, a spectral shift is applied due to the ISRF that is not centred
jacob	Jacobian calculation flag	string (len=6max)	-	i	Ini: parameter file para4a.dtp	YES or NO
fpoid	Transmittance calculation flag	string (len=6max)	-	i	Ini: parameter file para4a.dtp	YES or NO Computation of Jacobians and transmittances is not permitted if *conv*=NO
jap	Logical unit defining the ISRF file to read	int	-	i	Main program aaaa	Defined in aaaa: 8
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Output						
spc	Convolved radiances	r[allocatable]	W/ (m^2.strd. cm^{-1})	o		Allocated in aaaa Size: number of convolution wave numbers
dtoc	Convolved transmittances	r[allocatable, allocatable]	-	o		Allocated in aaaa Size: number of convolution wave numbers, nlevmax
dtjc	Convolved Jacobians with respect to temperature	r[allocatable, allocatable]	W/ (m^2.strd. cm^{-1})/ K	o		Allocated in aaaa Size: number of convolution wave numbers, nlevmax
dcjc	Convolved Jacobians with respect to molecule concentration	r[allocatable, allocatable, allocatable]	W/ (m^2.strd. cm^{-1})/ (g/g)	o		Allocated in aaaa Size: number of convolution wave numbers, nlevmax,

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
						ncjacmax
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if no error occurs in called subroutines or during the computation Err_error otherwise

1.2.1.25.6 Processing

The high-resolution spectrum is convolved by an instrumental function (as well as the Jacobians and transmittances if it is necessary).

This option speeds up convolution computing when jacobian calculation is selected. It slightly slows down it when convolution is done with the spectrum only. Note that results computed with this option are physically the same as those computed without it, even if there is a numerical difference between them (consult appendices section Part III 1.4).

The various steps of *cnvspehole* are:

- Definition of the points of the convolution limits nd and nf: Searching for convolution points meeting the spectral domain definition of convolution limits by using the dichotomy technique (subroutine *dichotomd*);
- For each convolution point n, n ∈ [nd,nf]:
  - Computation limits over the spectrum and the instrumental function: lcd and lcf,
  - Computation of the instrumental function from lcd to lcf for the convolution step at the wave numbers cnu (subroutine *calcisrf*),
  - Weighting of the spectral domain limits: adjustment of the limits used two times,
  - Partial convolution of the spectrum (and Jacobians and transmittances if it is required) by the instrumental function: Weighting by the function part between lcd and lcf. See section Part III 1.4 for the detail of the algorithm:
    - vii. Find the range of spectrum indices that contains [ncd, ncf]: first and last,
    - viii. Compute the weight for the first point: p(1),
    - ix. Compute the weight for the other points except the last: p(i), i ∈ [2, nInd-1], nInd=last-first+1,
    - x. Compute the weight for the last point: p(nInd),
    - xi. Partial convolution of the spectrum spc:

$$spc(n) = spc(n) + sum(spo(first:last)*p(1:nInd)).$$

1.2.1.25.7 Variable classification

Symbol	Description	Type	Unit	Classification
spo	High-resolution spectrum (radiances)	r[maxpt]	$W/(m^2.strd.\ cm^{-1})$	double: rad in aaaa
dto	Transmittances for the main	r[maxpt, 0:nlevmax]	-	double:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Description	Type	Unit	Classification
	path			to1 in aaaa
dtj	Jacobians with respect to the temperature	r[maxpt, 0:nlevmax]	W/(m^2.strd. cm^{-1})/K	double: dtsig in aaaa
dcj	Jacobians with respect to molecule concentration	r[maxpt, 0:nlevmax, ncjacmax]	W/(m^2.strd. cm^{-1})/(g/g)	double: dcsig in aaaa
nsc	Number of convolution points	int	-	double: iadcnv in aaaa
spc	Convolved radiances	r[allocatable]	W/ (m^2.strd. cm^{-1})	double: crad in aaaa
dtoc	Convolved transmittances	r[allocatable, allocatable]	-	double: cdtoin aaaa
dtjc	Convolved Jacobians with respect to temperature	r[allocatable, allocatable]	W/ (m^2.strd. cm^{-1})/ K	double: ctjac in aaaa
dcjc	Convolved Jacobians with respect to molecule concentration	r[allocatable, allocatable, allocatable]	W/ (m^2.strd. cm^{-1})/ (g/g)	double: ccjac in aaaa
f1	Lower limit (starting wave number) of the spectral domain	d	cm^{-1}	double: freq1 in aaaa
f2	Upper limit (stopping wave number) of the spectral domain	d	cm^{-1}	double: freq2 in aaaa
nuco	Central wave numbers of each ISRF	d[nknu]	cm^{-1}	double: nufo in aaaa
irec	Record number of the beginning of each ISRF	int[nknu]	-	double: irecf in aaaa
ldimo	Initial number of description points of the ISRF (dimension)	int[nknu]	-	double: kfmo in aaaa
pdnuo	Initial sampling step of the instrument functions	d[nknu]	cm^{-1}	double: pdnuc in aaaa

1.2.1.26. calcisrf

1.2.1.26.1 Type

calcisrf is a subroutine used by the main program `aaaa.f90`.
calcisrf is referenced in the library `aaaa.a`.

1.2.1.26.2 Function

The subroutine *calcisrf* has been made to compute the instrumental function used at the moment of the convolution.

1.2.1.26.3 Subordinates

The subroutine *calcisrf* calls the subroutines *lireisrf* and *dichotomd*. It uses the module *error*, and *precision*.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.26.4 Dependencies

Before this component is called, the parameters listed in the files para4a$(RSTR).dtp must have been already read via the subroutine *para4a* and the convolution parameters must have been already initialized via the subroutine *convini.*

1.2.1.26.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxbuf	Maximum size of a record block (buffer)	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxbuf
maxpt	Maximum number of computed points (wave numbers) in the whole spectral domains	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxpt
ldmax	Maximum dimension of the instrumental functions resampled for the convolution with the high-resolution spectrum	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ldmax
kpmax	Maximum dimension of the initial instrumental functions	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: 15000
ldim	Number of description points of an instrumental function at the spectrum calculation resolution	int	-	i		
nknu	Number of ISRF	int	-	i	Ini: ISRF file isrf.ddb	
irec	Record number of the beginning of each ISRF	int[nknu]	-	i	Ini: ISRF file isrf.ddb	
ldimo	Initial number of description points of the ISRF (dimension)	int[nknu]	-	i	Ini: ISRF file isrf.ddb	
lcd	First limit point of the function part used for the convolution in the spectral domain in progress	int	-	i		
lcf	Last limit point of the function part used for the convolution in the domain domain in progress	int	-	i		
ond	Wave number of the convolution point	d	cm^{-1}	i		
nuco	Central wave numbers of each	d[nknu]	cm^{-1}	i	Ini: ISRF file	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
	ISRF				isrf.ddb	
dnuo	Spectral shift of each ISRF	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	
pdnuo	Initial sampling step of the instrument functions	d[nknu]	cm^{-1}	i	Ini: ISRF file isrf.ddb	
rnu	Value of the global spectral shift	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
pdnu	Calculation resolution	d	cm^{-1}	i	Main program aaaa	
shift	Indicator of a spectral shift introduction due to the ISRF	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	YES or NO Only used if *conv*=YES If yes, a spectral shift is applied due to the ISRF that is not centred
jap	Logical unit defining the ISRF file to read	int	-	i	Main program aaaa	Defined in aaaa: 8
jlist	Writing logical unit (standard output)	int	-	i	Main program aaaa	Defined in aaaa: 6
Input/Output						
iisrfp1	First record block number of the first of the two read functions that surround the wave number ond	int	-	i/o		
iisrfp2	First record block number of the second read function	int	-	i/o		
rap1	First of the two read functions resampled on the wave numbers nux that surround the wave number ond	r[ldmax]	-	o		
rap2	Second of the two read functions resampled on the wave numbers nux that surround the wave number ond	r[ldmax]	-	o		
Output						
crp	Instrumental function at the wave number ond	r[ldmax]	-	o	Ini: ISRF file isrf.ddb	
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if no error occurs in called subroutines or during the

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
						computation Err_error otherwise

1.2.1.26.6 Processing

The various steps of *calcisrf* are:

- Determination of the size of the convolution: lct;
- Searching for both functions surrounding the convolution point ond by using the dichotomy technique (subroutine *dichotomd*): iisrf1 and iisrf2 correspond to the record block numbers of these functions;
- Reading of the (or both) function(s) corresponding to the convolution point ond (subroutine *lireisrf*): rap1 and rap2;
- Linear interpolation from both functions in order to deduce the instrumental function at the wave number ond (performed only if both functions are different): crp.

1.2.1.27. lireisrf

1.2.1.27.1 Type

lireisrf is a subroutine used by the main program aaaa.f90.

lireisrf is referenced in the library aaaa.a.

1.2.1.27.2 Function

The subroutine *lireisrf* has been made to read the instrumental function and its resampling in the spectral domain used for the convolution.

1.2.1.27.3 Subordinates

The subroutine *lireisrf* calls the subroutines *intspline*. It uses the module *error* and *precision*.

1.2.1.27.4 Dependencies

Before this component is called, the parameters listed in the files para4a$(RSTR).dtp must have been already read via the subroutine *para4a* and the convolution parameters must have been already initialized via the subroutine *convini.*

1.2.1.27.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxbuf	Maximum size of a record block (buffer)	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_maxbuf
ldmax	Maximum dimension of the instrumental functions resampled for the convolution with the high-resolution spectrum	int	-	i	Main program aaaa	Parameter defined in module model used in aaaa: Cst_ldmax

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
ldim	Number of description points of an instrumental function at the spectrum calculation resolution	int	-	i		
ldimo	Initial number of description points of the ISRF (dimension)	int	-	i	Ini: ISRF file isrf.ddb	
lcd	First limit point of the function part used for the convolution in the spectral domain in progress	int	-	i		
lcf	Last limit point of the function part used for the convolution in the spectral domain in progress	int	-	i		
lct	Size of the convolution	int	-	i		
iisrf	First record block number of the beginning of the read function	int	-	i		
dnuO	Spectral shift of each ISRF	d	cm^{-1}	i	Ini: ISRF file isrf.ddb	
pdnuo	Initial sampling step of the instrument functions	d	cm^{-1}	i	Ini: ISRF file isrf.ddb	
rnu	Value of the global spectral shift	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
pdnu	Calculation resolution	d	cm^{-1}	i	Main program aaaa	
shift	Indicator of a spectral shift introduction due to the ISRF	string (len=6 max)	-	o	Ini: parameter file para4a.dtp	YES or NO Only used if *conv*=YES If yes, a spectral shift is applied due to the ISRF that is not centred
jap	Logical unit defining the ISRF file to read	int	-	i	Main program aaaa	Defined in aaaa: 8
Output						
rap	Instrumental function resampled on the wave numbers nux	r[ldmax]	-	o		
ErrCode	Error code	int	-	o		Err_ok or Err_error

1.2.1.27.6 Processing

The various steps of *lireisrf* are:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Reading of the instrumental function: crp;
- Computation of the function wave number abscissa tabulated with or without the spectral shift correction: nuo;
- Re-sampling of the entire instrumental function using a spline interpolation on the recentered base nux of the high-resolution spectrum; Then normalization of the re-sampled function in double precision.
    - Computation of the recentered wave number: nux,
    - Spline interpolation of the instrumental function in the recentered wave number nux space (subroutine *intspline*): rap,
    - Normalization,
    - Keep the useful part of the re-sampled instrumental function corresponding to the high-resolution spectrum useful part: intersection between the instrumental function and the spectral domain,
    - Re-computation of the wave number abscissa nux recentered on the useful part of the high-resolution spectrum to be convolved.

1.2.1.27.7 Variable classification

Symbol	Description	Type	Unit	Classification
dnuO	Spectral shift of each ISRF	d	cm^{-1}	double: dnuo in calcisrf

1.2.1.28. parascat

1.2.1.28.1 Type

parascat is a subroutine used by the main program `aaaa.f90`.

parascat is referenced in the library `aaaa.a`.

1.2.1.28.2 Function

Similarly to the subroutine *para4a* for the general user defined run parameters, the subroutine *parascat* loads the user defined aerosol run parameters from the input file `parascat$(RSCA).dtp`.

1.2.1.28.3 Subordinates

The subroutine *parascat* uses the modules *error* and *model*.

1.2.1.28.4 Dependencies

It depends on the existence of the file `parascat$(RSCA).dtp` corresponding to the user specification of `$(RSCA)`.

1.2.1.28.5 Input/Output

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
jpars	Logical unit of parascat$(RSA).dtp	int	-	i	initialised in aaaa (parameter)	Value: 12
jlist	Logical unit of standard output	int	-	i	initialised in aaaa (parameter)	Value: 6
Output						
flag_scat	Aerosol presence flag	string (len=3)	-	o	calcrad	set to 'NO' in parascat.dtp
nlaero	Number of aerosol layers	int	dl.	o	pre_disort pre_sos	If flag_scat≠'NO'
aeromod	Identifier of the aerosol model, from top to bottom layer	string(len=6) [Cst_nlaeromax]	-	o	readaerosols	If flag_scat≠'NO'
aotref	Aerosol optical thickness at reference wave number 1000cm^{-1}, from top to bottom layer	r [Cst_nlaeromax]	dl.	o	pre_disort pre_sos	If flag_scat≠'NO': Initialisation to a spectrally constant value if positive
layaero	Vertical atmospheric layers where aerosol is present (upper and lower layer, then from top to bottom)	int[2, Cst_nlaeromax]	-	o	pre_disort pre_sos	If flag_scat≠'NO'
omeaero	Single scattering albedo, from top to bottom layer	r [Cst_nlaeromax]	dl.	o	pre_disort pre_sos	If flag_scat≠'NO': Initialisation to a spectrally constant value if positive
gaero	Asymetry parameter, from top to bottom layer	r [Cst_nlaeromax]	dl.	o	pre_disort pre_sos	If flag_scat≠'NO': Initialisation to a spectrally constant value if positive
nmom	Number of phase function Legendre polynoms	int	dl.	o	pre_disort pre_sos	If flag_scat≠'NO'
nstream	Number of streams	int	dl.	o	pre_disort	If flag_scat='DIS'
accur	DISORT convergence criterion if beam source is present	d	relative radiance unit	o	pre_disort	If flag_scat='DIS'
prnt	Header for prints, embedded in a DISORT banner	logical (kind=4) [5]	-	o	pre_disort	If flag_scat='DIS'
disheader	Array of DISORT logical print flags	string (len=127)	-	o	pre_disort	If flag_scat='DIS'
ErrCode	Error Code	int	-	o	aaaa	Err_ok or Err_error
Constants						
Cst_nlaeromax	Maximum number of aerosol layers	int	dl.	-	defined in model (parameter)	Value: 3
Local variables						

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
i	Loop index	int	-	-		
defini	Character string describing input parameters	string (len=66)	-	-		

1.2.1.28.6 Processing

The subroutine *parascat* reads from the file `parascat$(RSCA).dtp`.

The first reading statement loads the aerosol presence flag that can be

- 'DIS' : Aerosol is present and DISORT is used for integration of the radiative transfer equation;
- 'SOS' : Aerosol is present and SOS is used for integration of the radiative transfer equation;
- 'NO' : Aerosol is not present and *calcrad* is used for integration of the radiative transfer equation (no scattering is considered).

If the aerosol presence flag value is neither 'DIS' nor 'SOS' nor 'NO' it is set to 'NO'. If it is 'NO' no further read statements are executed and the program returns to `aaaa`. Otherwise the following parameters are loaded:

- The number of aerosol layers (between 1 and the number of authorised aerosol layers defined in *model*);
- The aerosol model corresponding to the available aerosol parameter input files `aersols_$(AEROMOD).dat` successively for each aerosol layer from top to bottom. If set to 'undef', spectrally constant values of optical thickness, single scattering albedo and asymmetry parameter are considered according to the specifications;
- The aerosol optical thickness at reference wavelength 1000 cm^{-1} in each vertical layer constituting an aerosol layer, successively from top to bottom. If set to zero and an aerosol model is defined, the value from the aerosol model data files are considered (see subroutine *readaerosols*). If set to a negative value and the aerosol model is defined or if set to a positive value and the aerosol model is undefined, the specified value is considered as constant over the entire spectral domain;
- The upper and lower vertical level constituting an aerosol layer (identical values if the aerosol layer comprises a single vertical layer). Value couples have to be defined successively from the top to the bottom aerosol layer, overlap between aerosol layers is forbidden;
- The single scattering albedo for each aerosol layer from top to bottom. This user specification is considered (as spectrally constant) only if the aerosol model is undefined. Otherwise, spectrally variable values from the aerosol model data files are considered (see subroutine *readaerosols*);
- The asymmetry parameter for each aerosol layer from top to bottom. This user specification is considered (as spectrally constant) only if the aerosol model is undefined. Otherwise, spectrally variable values from the aerosol model data files are considered (see subroutine *readaerosols*);
- The number of phase function Legendre coefficients supplied in the scattering codes.

If the aerosol presence flag is equal to 'SOS', the program returns to aaaa. Otherwise (aerosol presence flag equal to 'DIS'), additional DISORT input parameters are loaded:

- The number of computational polar angles to be considered (number of streams) should be even and at least 4. This specification controls the trade-off between computational time and accuracy;
- The convergence criterion (in relative radiance units) in presence of a beam source (sun) should be between 0. and 0.01. It has no impact for nighttime simulations;
- DISORT print flags and DISORT banner header without impact on the 4A/OP output files.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.28.7 Summary of aerosol property user specification options

aeromod	layaero(1:2)	aotref	omeaero, gaero
≠ 'undef'	must be > 0 if 0, read from model	if 0, read from model if > 0, overrules model and spectrally interpolated with model extinction coefficient if < 0, absolute value overrules model, spectrally constant	no function (loaded from model including spectral variation)
= 'undef'	must be > 0	must be ≥ 0 spectrally constant	must be ≥ 0 and ≤ 1 spectrally constant

Note: To run the DISORT or SOS code instead of the classical 4A/OP code without any presence of aerosol, the user specifies aeromod='undef' and aotref=0.

1.2.1.29. readaerosols

1.2.1.29.1 Type

readaerosols is a subroutine used by the main program `aaaa.f90`. It is called in a loop on aerosol layers if the aerosol presence flag *flag_scat* is different from 'NO' and if the aerosol model of the considered aerosol layer *aeromod* is different from 'undef'.

readaerosols is referenced in the library `aaaa.a`.

1.2.1.29.2 Function

The subroutine *readaerosols* loads the model dependent aerosol extinction parameters from the input files `aerosols_$(AEROMOD).dat`.

1.2.1.29.3 Subordinates

The subroutine *readaerosols* calls the subroutine *inttab2*. It uses the modules *error*, *precision* and *constants*.

1.2.1.29.4 Dependencies

It depends on the existence of the file `aerosols_$(AEROMOD).dat` corresponding to the user specification of $(AEROMOD), except if set to 'undef'.

1.2.1.29.5 Input/Output

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
jscat	Logical unit of aerosols_$(AEROMOD).dat	int	-	i	initialised in aaaa (parameter)	Value: 13
racine5	Aerosol model file name root	string (len=400max)	-	i	read in aaaa from standard input	Value: 'aerosols_'
racine4	Suffixe of the aerosol model file name	string (len=400max)	-	i	read in aaaa from standard input	Value: '.dat'
ksmax	Maximum number wave numbers for which aerosol properties are provided	int	-	i	defined in model	
aeromod	Identifier of the aerosol model	string(len=6)	-	i	read in parascat	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input/Output						
aotref	Aerosol optical thickness at reference wave number 1000cm^{-1}.	r	dl.	i/o	initialised in parascat	updated if initialised to 0.
layaero	Vertical atmospheric layers where aerosol is present (upper and lower layer)	int[2]	-	i/o	initialised in parascat	updated if initialised to 0
Output						
nwn	Number of wave numbers for which aerosol model properties are provided.	int	-	o	aaaa	
wnscat	Wave numbers for which aerosol model properties are provided.	r[ksmax]	cm^{-1}	o	aaaa	in increasing order
ome_aero	Single scattering albedo at wavenumbers wnscat	r[ksmax]	dl.	o	aaaa	
g_aero	Asymmetry parameter at wavenumbers wnscat	r[ksmax]	dl.	o	aaaa	
ot_aero	Aerosol optical thickness at wavenumbers wnscat	r[ksmax]	dl.	o	aaaa	
ErrCode	Error Code	int	-	o	aaaa	Err_ok or Err_error
Local variables (Selection)						
fscat	Filename of the aerosol model properties file	string (len=120)	-	-		
line	Character string of one line in *fscat*	string (len=120)	-	-		
wl	Wavelengths for which aerosol model properties are provided	r[ksmax]	µm	-		in increasing order
ssa	Single scattering albedo at wavelengths wl	r[ksmax]	dl.	-		
asp	Asymmetry parameter at wavelengths wl	r[ksmax]	dl.	-		
ecoef	Extinction coefficient at wavelengths wl	r[ksmax]	km^{-1}	-		
ecoef_ref	Extinction coefficient at reference wavelength	r	km^{-1}	-		

1.2.1.29.6 Processing

The subroutine *readaerosols* reads from the file fscat=`'aerosols_$(AEROMOD).dat'`. fscat is composed of the variables racine5, aeromod and racine4.

In case that the aerosol optical thickness is initialised to 0., it is read from fscat. If the user initialises the aerosol layer numbers with 0, they are read from fscat. Then, the extinction coefficient single scattering albedo and the asymmetry parameter are loaded for given wavelengths. Wavelengths are converted into wave numbers and the order of wave number, extinction coefficient single scattering albedo and asymmetry

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

parameter vectors are inversed (from increasing in wavelength to increasing in wave number). The extinction coefficient at reference wavenumber (WN_REF, defined in module constants) is interpolated within the spectral extinction coefficient vector. Finally, the aerosol optical thickness is calculated for the given wavenumber by multiplying the reference aerosol optical thickness with the ratio extinction coefficient over extinction coefficient at reference wavenumber.

*ot_aero(1:nwn) = aotref * ecoef(1:nwn)/ecoef_ref*

1.2.1.30. contract_sig

1.2.1.30.1 Type

contract_sig is a subroutine used by the main program `aaaa.f90`. It is called in the loop on spectral domains if the aerosol presence flag is either 'DIS' or 'SOS', i.e., if aerosol is present.

contract_sig is referenced in the library `aaaa.a`.

1.2.1.30.2 Function

The subroutine *contract_sig* resamples the gaseous optical thickness (output from *calcsig*) prior to the vertical integration of the radiative transfer equation by DISORT or SOS. This offers the possibility for reducing the computation time by decreasing the number of monochromatic spectral samples. The sampling step is equal to the user defined variable *respec*, specified in `para4a$(RSTR).dtp`. In presence of aerosol, this subroutine replaces the similar subroutine contract which is applied to spectral radiances (i.e., after integration of the radiative transfer equation).

1.2.1.30.3 Subordinates

The subroutine *contract_sig* calls the subroutines *echantini*, *echantillon* and *inttab2*. It uses the module precision.

1.2.1.30.4 Dependencies

High-resolution gaseous optical thickness for an entire spectral domain must have been already computed via the subroutine *calcsig* and the parameters listed in the files `para4a$(RSTR).dtp` must have been loaded via the subroutine *para4a*.

1.2.1.30.5 Input/Output

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
maxpt	Maximum number of wave numbers in the spectral domain	int	-	i	Main program aaaa	Parameter defined in model
nlevmax	Maximum number of vertical levels	int	-	i	Main program aaaa	Parameter defined in model
level1	First level (up) of the atmosphere	int	-	i	calcsphtraj	
level2	Last level (down) of the atmosphere	int	-	i	calcsphtraj	
delds	Width of the spectral domain (atlas)	d	cm^{-1}	i	Ini: parameter file para4a.dtp	
freq1	Lower wave number of the spectral domain	d	cm^{-1}	i	Ini: Atlas file of the molecule nc	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
jlist	Logical unit of standard output	int	-	i	Main program aaaa	Parameter defined in aaaa: 6
Input/Output						
iadcom	Number of computed common points in the the spectral domain (shared by all involved molecules)	int	-	i/o	Main program aaaa	
iabcom	Abscissa array in the non regular frames corresponding to the indices of the iadcom points that are computed in the Atlas	int[maxpt]	-	i/o	Main program aaaa	
nmin	Number of sampling points	int	-	i/o	Main program aaaa	after resampling
lfil	Box width (in spectral samples)	int	-	i/o	Main program aaaa	if 1: no contraction else: contraction
respec	Spectral sampling step before vertical integration of the radiative transfer equation	d	cm^{-1}	i/o	Ini: parameter file para4a.dtp	
nu	Wave numbers of spectral samples	d[maxpt]	cm^{-1}	i/o	Main program aaaa	In: at initial sampling Out: after resampling
sig	Optical thickness	r[maxpt, 0:nlevmax]]	dl.	i/o	Main program aaaa	In: at initial sampling Out: after resampling
pdnu	Calculation resolution	d	cm^{-1}	i/o	Main program aaaa	
Local (Selection)						
pds	Weight of every point to be contracted	r[maxpt]	-	o		0.5 for the point shared by two boxes and 1 for the others
tab1	Work table: optical thickness interpolated on the maxpt wave numbers tab2	r[maxpt]	dl.	o		
tab2	Wave numbers at the maximum sampling maxpt	d[maxpt]	cm^{-1}	o		

1.2.1.30.6 Processing

The subroutine *contract_sig* contracts the optical thickness spectrum. The step is given by the variable *respec*. The minimum value of *respec* is 0.0005 cm-1. If respec > 0.0005 cm-1, a system of boxes fitting into each other is used.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

The various steps of *contract_sig* are:

- Sampling at the maximum resolution *maxpt* ; computation of:
  - Sampling step: pdnux = dble(delds/(maxpt-1))
  - Wave numbers tab2
  - per layer (level1 to level2-1): conversion of optical thickness in transmittance
  - Linear interpolation of the transmittances per layer on tab2 (subroutine *inttab2*)
- Initialization of the resampling (contraction): Computation of the box width *lfil* and the box weight *rnorm* (subroutine *echantini*)
- If *lfil* different from 1:
  - Computation of the wave numbers after the contraction nu
  - Contraction of the transmittance per layer (subroutine *echantillon*)
- per layer: reconversion of transmittance into optical thickness
  - if the transmittance is zero (which is numerically the case for optical thickness of approximately 100 or higher), the optical thickness is set to the value 100.

1.2.1.31. pre_disort

1.2.1.31.1 Type

pre_disort is a subroutine used by the main program `aaaa.f90`. It is called only if the aerosol presence flag is equal to 'DIS'.

pre_disort is referenced in the library `aaaa.a`.

1.2.1.31.2 Function

The subroutine *pre_disort* constitutes the interface between 4A/OP and DISORT. It is called in a loop on spectral samples and mainly processes the monochromatic input parameters according to the required DISORT format, then calls DISORT.

1.2.1.31.3 Subordinates

The subroutine *pre_disort* calls the subroutine *disort*. The call of DISORT requires the fixed format fortran codes *disortlb.f, brdf.f, linpak.f, utility.f*. It uses the modules *error*, *precision* and *disort_runparams*.

1.2.1.31.4 Dependencies

None

1.2.1.31.5 Input/Output

Symbol	Descriptive Name	Type	Units	I / O	Source / Destinat°	References / Remarks
Input						
level1	Upper atmosphere level	int	-	i	calcsphtraj	
level2	Lower atmosphere level	int	-	i	calcsphtraj	
siggas	Gaseous optical thickness per layer (multiplied with secans of viewing angle)	r [level2-level1]	dl.	i	calcsig contract_sig	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I / O	Source / Destinat°	References / Remarks
temper	temperature profile	r [0:level2-level1]	K	i	atmsph	corresponds to tl[level1:level2]
btemp	Lower surface temperature	d	K	i	para4a	
edown_s	Lower surface emissivity	r	dl.	i	lireemis	
ttemp	Background temperature	d	K	i	para4a	
temis_s	Background emissivity	r	dl.	i	lireemis	
nlaero	Number of aerosol layers	int	-	i	parascat	
otaero	Aerosol optical thickness at current wave number , from top to bottom layer	r [nlaero]	dl.	i	parascat readaerosols	
layaero	Vertical atmospheric layers where aerosol is present (upper and lower layer, then from top to bottom)	int [2,nlaero]	-	i	parascat	
omeaero	Single scattering albedo, from top to bottom layer	r [nlaero]	dl.	i	parascat readaerosols	
gaero	Asymmetry parameter, from top to bottom layer	r [nlaero]	dl.	i	parascat readaerosols	
sectta1	Secans of the viewing angle at ground	r	dl.	i	para4a	
secttal	Secans of the viewing angle per vertical layer	r [level2-level1]	dl.	i	calcsphtraj	
umu0	Cosine of the solar zenith angle	d	dl.	i	para4a aaaa	
fbeam	Intensity of incident beam (incoming flux divided by umu0)	d	Wm^{-2} cm	i	aaaa	
wanu	wave number	d	cm^{-1}	i	aaaa	
nmom	Number of phase function Legendre polynoms	int	-	i	parascat	
nstream	Number of streams	int	-	i	parascat	
accur	DISORT convergence criterion if beam source is present	d	relative radiance unit	i	parascat	
prnt	Header for prints, embedded in a DISORT banner	logical (kind=4) [5]	-	i	parascat	
disheader	Array of DISORT logical print flags	string (len=127)	-	i	parascat	
Output						
rad_diff	Spectral radiance	r	$Wm^{-2}sr^{-1}cm$	o	aaaa	
ErrCode	Error Code	int	-	o	aaaa	Err_ok or Err_error
Constants (see module disort_runparams)						
Local variables						

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I / O	Source / Destinat°	References / Remarks
mxumu	Maximum number of user polar angles (internal DISORT parameterisation)	int	-		parameter: 20	only used for u0u declaration
mxulv	Maximum number of user levels (internal DISORT parameterisation)	int	-		parameter: 130	only used for u0u declaration
nlyr	Number of vertical layers	int	-		disort (in)	level2-level1+1
dtauc	Total optical thickness per layer	d[level2-level1]	dl.		disort (in)	
ssalb	Single scattering albedo per layer	d[level2-level1]	dl.		disort (in)	
pmom	Phase function Legendre polynomial coefficients	d[0:nmom , level2-level1]	dl.		disort (in)	
umu	Cosine of (user) polar angles	d[Cst_numu]	dl.		disort (in)	corresponds to the viewing zenith angle
temis	Background emissivity	d	dl.		disort (in)	
albedo	surface albedo	d	dl.		disort (in)	1 – edown_s
wvnmlo wvnmhi	Lower and upper waver numbers for computation of the Planck function	d	cm^{-1}		disort (in)	wanu ± Cst_deltawvn/2
rfldir	direct beam flux	d[level2-level1+1]	Wm^{-2} cm		disort (out)	
rfldn	Diffuse downward flux	d[level2-level1+1]	Wm^{-2} cm		disort (out)	
flup	Diffuse upward flux	d[level2-level1+1]	Wm^{-2} cm		disort (out)	
dfdt	Flux divergence d(net flux)/d(opt. thickness)	d[level2-level1+1]	Wm^{-2} cm		disort (out)	
uavg	Mean intensity (including direct beam)	d[level2-level1+1]	Wm^{-2} sr^{-1}cm		disort (out)	
uu	Intensity	d[Cst_numu, level2-level1+1, Cst_nphi]	Wm^{-2} sr^{-1}cm		disort (out)	
u0u	Azimuthally averaged intensity	d[mxumu, mxulv]	Wm^{-2} sr^{-1}cm		disort (out)	
albmed	Albedo of the medium as function of umu	d[Cst_numu]	dl.		disort (out)	no use as long as Cst_ibcnd = 0
trnmed	Transmissivity of the medium as function of umu	d[Cst_numu]	dl.		disort (out)	no use as long as Cst_ibcnd = 0
sigma_aero	Aerosol optical thickness	d[level2-level1]	dl.			
omega_aero g_aero	Double precision of input variables omeaero and gaero	d[nlaero]	dl.			
nla	Loop index for aerosol layers	int	-	-		

1.2.1.31.6 Processing

The subroutine *pre_disort* prepares within a loop on spectral samples (after contraction of the spectral scale in *contract_sig*) the call of the original disort program package, version 2.1. It sets (computes, initialises,

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

reformats or transports) all spectral DISORT input variables, calls the DISORT package, and reformats the DISORT output variables used by 4A/OP for further processing.

1) Initialisation of non-spectral DISORT specific input parameters

- The number of vertical atmospheric layers *nlyr* is set to *nlyr = level2 – level1 + 1*;
- The cosine of user polar angles is set as function of the observation zenith angle at ground *umu(1) = 1/sectta1*.

2) Initialisation/computation of spectral DISORT input parmaters

- Initialisation of aerosol optical thickness (all layers), single scattering albedo (all layers), phase function coefficients (all layers, all moments) to zero;
- Computation of Planck function limit wavenumbers *wvnmlo, wvnmhi* as function of the current wavenumber *wanu*;
- Computation of aerosol optical thickness *sigma_aero(nla)* per aerosol layer nla. **Attention:** If the aerosol layer consists of several atmospheric layers (*layaero(1,nla) ≠ layaero(2,nla)*), the user specified reference aerosol optical thickness applies to atmospheric layers, i.e., **the total AOT of the aerosol layer** is the **product** of the **specified AOT** value and the **number of atmospheric layers within** the specified **aerosol layer**. To do otherwise, the user can specify aerosol layers containing a single atmospheric layer. Furthermore, a plan-parallel to spherical correction is applied to the aerosol optical thickness transmitted to DISORT (plan-parallel model). To account for the decreasing secans of the observation zenith angle with height, the aerosol optical thickness transmitted to DISORT is weighted by the ratio between the spherical and the plan-parallel (height independent) secans of the viewing angle. This artificial reduction of the optical thickness balances the overestimation of the optical path length in a plan-parallel geometry.

*sigma_aero(layaero(1,nla):layaero(2,nla)) = otaero(nla) * secttal(layaero(1,nla):layaero(2,nla)) * umu(1)*

- The DISORT input parameters single scattering albedo and the asymmetry parameter are initialised. There are considered constant over atmospheric layers constituting the same aerosol layer;
- The optical thickness per vertical layer *dtauc(nl)* is computed as the sum of gaseous optical thickness and aerosol optical thickness *sigma_aero*. The gaseous optical thickness per vertical layer, weighted with the secans of the observation zenith angle, is imported via the variable *siggas(nl)*. To obtain the required DISORT input in consistency with the aerosol optical thickness, siggas has to be devided by the secans of the observation zenith angle and then corrected for the the plan-parallel model assumption in all atmospheric layers *nl*. In a second step, the aerosol optical thickness is added in all atmospheric layers being part of all aerosol layers *nla*.

*dtauc(nl) = siggas(nl) * umu(1)*

dtauc(layaero(1,nla):layaero(2,nla)) = dtauc(layaero(1,nla):layaero(2,nla)) + sigma_aero(layaero(1,nla):layaero(2,nla))

- The single scattering albedo ssalb is computed in all atmospheric layers being part of an aerosol layer as the ratio between aerosol and total optical thickness *sigma_aero/dtauc*, multiplied with the single scattering albedo omega_aero of the considered aerosol layer;

*ssalb(layaero(1,nla):layaero(2,nla)) = sigma_aero(layaero(1,nla):layaero(2,nla)) /dtauc(layaero(1,nla):layaero(2,nla)) * omega_aero(nla)*

- The zero degree phase function Legendre polynomial coefficients *pmom(0,:)* are initialised to 1 in atmospheric layers where aerosol is present. For other degrees *nm* (from 1 to *nmom*), they are approximated by the nm^{th} power of the asymmetry parameter $[g_aero(nla)]^{nm}$;
- Top emissivity temis is initialised (*temis = temis_s*), surface albedo is computed as the difference between blackbody and surface emissivity (*albedo = 1 – edown_s*).

3) DISORT call

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- DISORT is called in its unmodified version 2.1. The following table associates DISORT arguments with the corresponding variables or expressions in the 4A/OP code.

Argument in routine DISORT	Corresponding 4A/OP variable or expression (*)	Source in 4A/OP	Remark
NLYR	(*) nlyr – 1	pre_disort	level2 - level1
DTAUC	dtauc	pre_disort	
SSALB	ssalb	pre_disort	
NMOM	nmom	parascat	
PMOM	pmom	pre_disort	
TEMPER	temper	atmsph	Initial name: tl
WVNMLO	wvnmlo	pre_disort	
WVNMHI	wvnmhi	pre_disort	
USRTAU	Cst_usrtau	disort_runparams	Value: 'TRUE'
NTAU	Cst_ntau	disort_runparams	Value: 1
UTAU	Cst_utau	disort_runparams	Value: 0.0
NSTR	nstream	parascat	
USRANG	Cst_usrang	disort_runparams	Value: 'TRUE'
NUMU	Cst_numu	disort_runparams	Value: 1
UMU	umu	pre_disort	
NPHI	Cst_nphi	disort_runparams	Value: 1
PHI	Cst_phi	disort_runparams	Value: 0.0 (no use)
IBCND	Cst_ibcnd	disort_runparams	Value: 0
FBEAM	fbeam	aaaa	
UMU0	umu0	aaaa	
PHI0	Cst_phi0	disort_runparams	Value: 0.0 (no use)
FISOT	Cst_fisot	disort_runparams	Value: 0.0
LAMBER	Cst_lamber	disort_runparams	Value: 'TRUE'
ALBEDO	albedo	pre_disort	
BTEMP	btemp	para4a	
TTEMP	ttemp	para4a	
TEMIS	temis	pre_disort	
PLANK	Cst_planckind	disort_runparams	Value: 'TRUE'
ONLYFL	Cst_onlyfl	disort_runparams	Value: 'FALSE'
ACCUR	accur	parascat	
PRNT	prnt	parascat	
HEADER	disheader	parascat	
MAXCLY	(*) nlyr-1	pre_disort	Define internal DISORT array dimension equal to the actually required dimension
MAXULV	nlyr	pre_disort	
MAXUMU	Cst_numu	disort_runparams	
MAXPHI	Cst_nphi	disort_runparams	
MAXMOM	nmom	parascat	
RFLDIR	rfldir	(output disort)	not used
RFLDN	rfldn	(output disort)	not used
FLUP	flup	(output disort)	not used
DFDT	dfdt	(output disort)	not used
UAVG	uavg	(output disort)	not used

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Argument in routine DISORT	Corresponding 4A/OP variable or expression (*)	Source in 4A/OP	Remark
UU	uu	(output disort)	
U0U	u0u	(output disort)	not used
ALBMED	albmed	(output disort)	not used
TRNMED	trnmed	(output disort)	not used

4) DISORT output

- For instance, the only DISORT output used is the spectral radiance at the (only) specified user polar angle (= observation zenith angle) at the upper level of the atmosphere and at the (only) specified user azimuth angle, *uu(1,1,1)*. It is set to the single precision variable *rad_diff* and returned to the main program. To obtain the spectral radiance at any level in the atmosphere, the constant *Cst_usrtau* has to be set to .FALSE. and the second dimension of *uu* has to be set to *levobs* (additional input variable to *pre_disort*).

1.2.1.32. pre_sos

1.2.1.32.1 Type

pre_sos is a subroutine used by the main program `aaaa.f90`. It is called only if the aerosol presence flag is equal to 'SOS'.

pre_sos is referenced in the library `aaaa.a`.

1.2.1.32.2 Function

The subroutine *pre_sos* constitutes the interface between 4A/OP and SOS. It is called in a loop on spectral samples and mainly processes the monochromatic input parameters according to the required SOS format, then calls SOS.

1.2.1.32.3 Subordinates

The subroutine *pre_sos* calls the subroutine *diffnd*. The call of SOS requires the fixed format fortran codes *diffnd.f, d1mach.f*. It uses the modules *error, precision* and *sos_runparams*.

1.2.1.32.4 Dependencies

None

1.2.1.32.5 Input/Output

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
level1	Upper atmosphere level	int	-	i	calcsphtraj	
level2	Lower atmosphere level	int	-	i	calcsphtraj	
siggas	Gaseous optical thickness per layer (multiplied with secans of viewing angle)	r [level2-level1]	dl.	i	calcsig contract_sig	
t_lev	temperature profile	r [0:level2-level1]	K	i	atmsph	corresponds to tl[level1:level2]

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
btemp	Lower surface temperature	d	K	i	para4a	
bemis_s	Lower surface emissivity	r	dl.	i	lireemis	
ttemp	Background temperature	d	K	i	para4a	
temis_s	Background emissivity	r	dl.	i	lireemis	
nlaero	Number of aerosol layers	int	-	i	parascat	
otaero	Aerosol optical thickness at current wave number, from top to bottom layer	r[nlaero]	dl.	i	parascat readaerosols	
layaero	Vertical atmospheric layers where aerosol is present (upper and lower layer, then from top to bottom)	int[2,nlaero]	-	i	parascat	
omeaero	Single scattering albedo, from top to bottom layer	r[nlaero]	dl.	i	parascat readaerosols	
gaero	Asymmetry parameter, from top to bottom layer	r[nlaero]	dl.	i	parascat readaerosols	
sectta1	Secans of the viewing angle at ground	r	dl.	i	para4a	
secttal	Secans of the viewing angle per vertical layer	r [level2-level1]	dl.	i	calcsphtraj	
wanu	wave number	d	cm^{-1}	i	aaaa	
nmom	Number of phase function Legendre polynoms	int	-	i	parascat	
npttabexp_d	Dimension of the exp function tabulation	int	-	i	model	
tabexp_d	Tabulated exponential function	d[npttabexp_d]	-	i	aaaa	
pastabexp_d	Tabulation step of the exp function tabulation	d	-	i	aaaa	
Output						
rad_diff	Spectral radiance	r	$Wm^{-2}sr^{-1}cm$	o	aaaa	
ErrCode	Error Code	int	-	o	aaaa	Err_ok or Err_error
Constants (see module sos_runparams)						
External functions						
tbrillans	Brightness temperature	r	K			Arguments: wavenumber, radiance

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Local variables						
nlevel	Number of atmospheric levels	int	-			level2-level1+1
nlevel_incr	Number of atmospheric levels after split of aerosol layers	int	-		sos (in)	
nsplit	Number of additional layers per aerosol layer	int[nlaero]	-			
d_nsplit	Double precision of nsplit	d[nlaero]	-			
dtauc	Total optical thickness per layer (after split)	d[allocatable]	dl.		sos (in)	
ssalb	Single scattering albedo per layer (after split)	d[allocatable]	dl.		sos (in)	
temper	Temperature profile per layer (after split)	d[allocatable]	K		sos (in)	
pmom	Phase function Legendre polynomial coefficients, per degree and layer (after split)	d[allocatable, allocatable]	dl.		sos (in)	
umu	Cosine of (user) polar angle	d	dl.		sos (in)	corresponds to the viewing zenith angle
temis	Background emissivity	d	dl.		sos (in)	
bemis	Background emissivity	d	dl.		sos (in)	
sigma_aero	Aerosol optical thickness	d	dl.			
omega_aero g_aero	Double precision of input variables omeaero and gaero	d	dl.			
j, nla, nl	Loop indices	int	-	-		
sigma_aux alpha tbri radtop radbot raddiff	Auxiliary variables	r	dl. - K Wm^{-2} $sr^{-1}cm$			
indexp nladd	Auxiliary variables	int	-			
rad_diff_d	Double precision of rad_diff	d	Wm^{-2} $sr^{-1}cm$		sos (out)	
nordremaxb		int			sos (out)	***not used!***

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1.32.6 Processing

The subroutine *pre_sos* prepares within a loop on spectral samples (after contraction of the spectral scale in *contract_sig*) the call of the SOS program package. It sets (computes, initialises, reformats or transports) all spectral SOS input variables, calls the SOS package, and reformats the SOS output variables used by 4A/OP for further processing.

1) Initialisation of non-spectral SOS specific input parameters

- The number of vertical atmospheric layers *nlevel* is set to *nlevel = level2 – level1 + 1*;
- The cosine of user polar angles is set as function of the observation zenith angle at ground *umu = 1/sectta1* .

2) Split of aerosol layers (TBC: scientific justification)

- The gaseous optical thickness above each aerosol layer is accumulated and the corresponding index *indexp* in the tabulated exponential function is determined;
- If the exponential function is not saturated (*indexp < npttabexp_d*), i.e., if the transmission is significantly different from zero, the aerosol optical thickness of the aerosol layer is multiplied by the single scattering albedo and weighted with the transmission above the aerosol layer and set to the parameter *sigma_aux*. The integer ratio between *sigma_aux* and a reference value (*Cst_aotdiff*, defined in *sos_runparams*) is equal to the number of additional layers *nsplit(nla)* in the original aerosol layer *nla*. Otherwise the aerosol layer is considered at its original vertical extension (*nsplit(nla) = 0*). Note: This operation is carried out separately for each spectral sample. The vertical resolution of an aerosol layer is only increased, if necessary, i.e., if the scattering coefficient is considerably high and if the atmospheric transmission above the aerosol layer is significantly different from zero.
- Computation of the increased number of vertical atmospheric layers *nlevel_incr*:

$$nladd = \Sigma\, nsplit(nla)$$

$$nlevel_incr = nlevel - 1 + nladd$$

3) Initialisation/computation of spectral SOS input parameters

- Allocation of rank 1 arrays for temperature *temper*, total optical thickness *dtauc* and single scattering albedo *ssalb* to the dimension *nlevel_incr*;
- Allocation of rank 2 array for phase function Legendre polynom coefficients *pmom* to the dimension (*0:nmom,nlevel_incr*);
- Interpolation of the temperature profile from original *t_lev(0:nlevel-1)* to modified vertical resolution *temper(0:nlevel_incr)*. Within each atmospheric layer *nl* in an aerosol layer *nla* and if *nsplit(nla) > 0*, the top and bottom temperature of the layer are converted in the corresponding blackbody radiance. The blackbody radiance is linearly interpolated on *nsplit(nla)* points within the layer and reconverted into temperature;
- Initialisation of single scattering albedo (all layers, from 1 to *nlevel_incr*), phase function coefficients (all layers, all moments) to zero;
- Initialisation of aerosol optical thickness (layers above the highest and below the lowest aerosol layer) to the gaseous optical thickness. The gaseous optical thickness per vertical layer, weighted with the secans of the observation zenith angle, is imported via the variable *siggas(nl)*. To obtain the required SOS input consistently, *siggas* has to be divided by the secans of the observation zenith angle and then corrected for the plan-parallel model assumption in all atmospheric layers *nl* by multiplication with *secttal(nl)*umu*:

$$dtauc(1:layaero(1,1)-1) = siggas(1:layaero(1,1)-1) * umu$$

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

*dtauc(layaero(2,nlaero)+nladd+1:nlevel_incr) = siggas(layaero(2,nlaero)+1:nlevel-1) * umu*

- Computation of aerosol optical thickness sigma_aero per atmospheric layer nl. In case that the layer has been split (*nsplit(nla) > 0*), the optical thickness is equally distributed over the split layers. Furthermore, a plan-parallel to spherical correction is applied to the aerosol optical thickness transmitted to DISORT (plan-parallel model). To account for the decreasing secans of the observation zenith angle with height, the aerosol optical thickness transmitted to DISORT is weighted by the ratio between the spherical and the plan-parallel (height independent) secans of the viewing angle. This artificial reduction of the optical thickness balances the overestimation of the optical path length in a plan-parallel geometry:

  *sigma_aero = otaero(nla)/(nsplit(nla)+1) * secttal(nl) * umu*

  Attention: If the aerosol layer consists of several atmospheric layers (i.e., if *layaero(1,nla) ≠ layaero(2,nla)*), the user specified reference aerosol optical thickness applies to all atmospheric layers, i.e., it is multiplied by the number of atmospheric layers within the specified aerosol layer. To do otherwise, the user can specify aerosol layers containing a single atmospheric layer;

- In each aerosol layer, the aerosol single scattering albedo *omega_aero* and the asymmetry parameter *g_aero* are initialised. They are considered constant over atmospheric layers constituting the same aerosol layer;

- Within aerosol layers, the optical thickness per atmospheric layer *dtauc(nl)* is computed as the sum of gaseous optical thickness and aerosol optical thickness *sigma_aero*. The gaseous optical thickness is computed as outside aerosol layers, except for the case that the layer has been splitted (*nsplit(nla) > 0*). Then, the gaseous optical thickness is equally distributed over the split layers;

- The single scattering albedo *ssalb* is computed in all atmospheric layers being part of an aerosol layer as the ratio between aerosol and total optical thickness *sigma_aero/dtauc*, multiplied with the single scattering albedo *omega_aero* of the considered aerosol layer:

  *ssalb(nl) = sigma_aero /dtauc(nl) * omega_aero*

- The zero degree phase function Legendre polynomial coefficients *pmom(0,:)* are initialised to 1 in atmospheric layers where aerosol is present. For other degrees *nm* (from 1 to *nmom*), they are approximated by the nm^{th} power of the asymmetry parameter $[g_aero]^{nm}$.

- Top and surface emissivities *temis* and *bemis* are initialised (*temis = temis_s , bemis = bemis_s*);

4) SOS call

- SOS is called via the code `diffnd.f` with the following modifications with respect to the reference software.
  - The parameters *NPHAS* and *npttabexp* are converted into input arguments of *diffnd*,
  - The commented subroutines *QGAUSN* and *ERRMSG* are removed.

The following table associates SOS (diffnd) arguments with the corresponding variables or expressions in the 4A/OP code.

Argument in SOS (routine diffnd)	Corresponding 4A/OP variable or expression (*)	Source in 4A/OP	Remark
NLAYER	nlevel_incr	pre_sos	
SIGMA	dtauc	pre_sos	
OMEGA	ssalb	pre_sos	
NPHAS	nmom	parascat	
PHAS	pmom	pre_sos	
NU	wanu	aaaa	
TL	temper	pre_sos	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Argument in SOS (routine diffnd)	Corresponding 4A/OP variable or expression (*)	Source in 4A/OP	Remark
Tspace	ttemp	para4a	
EMISPACE	temis	pre_sos	
TSURF	btemp	para4a	
EMISURF	bemis	pre_sos	
MUOBS	umu	pre_sos	
RAD	rad_diff_d	(output diffnd)	
npttabexp	npptabexp_d	aaaa	
TABEXP	tabexp_d	aaaa	
pastabexp	pastabexp_d	aaaa	
COREC_GEO	Cst_corecgeo	sos_runparams	Value: 'TRUE'
NORDREMAXB	nordremaxb	(output diffnd)	not used

5) SOS output

- The SOS output is the spectral radiance at the specified observation zenith angle at the upper level of the atmosphere. It is set to the single precision variable *rad_diff* and returned to the main program.

6) Deallocation

- Deallocation of arrays *temper, dtauc, ssalb, pmom*.

1.2.2. External tools for radiative transfer

4AOP 2009 v1.0 contains external software routines, dedicated to the simulation of extinction effects, i.e., absorption/emission and scattering, by aerosol (and cloud droplets/crystals) (Part IV.[Ref. 24], Part IV.[Ref. 25] et Part IV.[Ref. 26]). As independent codes, they are referred to as

- **DISORT**, version 2.1, developed by NASA [Part IV.[Ref. 23]],
- **SOS**, developed at LMD [Part IV[Ref. 32]].

These codes are stored in the directory `srclib/EXTERNAL`.

Reminder of restrictions for the call of DISORT and SOS :

In scattering mode, the 4A/OP package is functional only for a part of possible 4A/OP parameterisations in `para4a($RSTR).dtp`. This is due to the limitations and/or incompatibilities between the classical 4A/OP model assumptions and those of the external (consciously unmodified) models. Some of these restrictions, summarised here below, can be removed in upcoming versions.

- Only Earth view configurations are allowed. This excludes limb view configurations (the geometry identifier *GEOMID* must be equal to 'EVIEW') and the viewing configurations 'DOWN' and 'UPDOWN'. The corresponding identifier *TRAJET* must be equal to 'UP' or 'DOWNUP'. The result is insensitive to the distinction between these two parameterisations. In all cases the computation includes the contribution from reflected radiation at the Earth surface (if any);
- It is currently not possible to define the observation level somewhere between the upper (in general top-of-atmosphere) and lower (in general Earth surface) simulation levels. The user has to specify the observation level (*PZOBS*) at or above the upper simulation limit (*PZUP*);
- Solar radiation (as beam source) is not accounted for in SOS. Any specification of the solar zenith angle (*SECTTASUN*) is ignored;

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Solar radiation (as beam source) is theoretically accounted for in DISORT and can be managed in the usual way via the user parameter *SECTTASUN*. However, inconsistent results are expected due to the plan-parallel geometry model in DISORT. As mentioned before, a correction of this shortcoming is applied to the optical thickness as input argument to DISORT, but this correction is driven and approximately valid only for the observation zenith angle. It becomes inconsistent, when solar and observation zenith angles are significantly different;
- Transmittances are not returned to the 4A/OP main program and cannot be written in the output files. The corresponding parameter *FPOID* must be equal to 'NO';
- Jacobians are not calculated. The corresponding parameter *JACOB* must be equal to 'NO'.

Moreover, we emphasize:

- The user parameter 'Reflexion/emission angle' (*SECTTA2*) is without function. The downward flux is computed independently of any angle specification, consistently with the DISORT and SOS model assumptions;
- The parameter 'Spectral resolution' *RESPEC* has a strong impact on the computation time since it determines the spectral sampling step and therefore the number of DISORT or SOS calls. Obviously, it must still be chosen consistently with the spectral sampling of the spectral response function (if any) and with the convolution step.

1.2.3. Mathematical tools

These tools are stored in the directory `scslib/smath`.

1.2.3.1. inttab

1.2.3.1.1 Type

inttab is a common subroutine that can be used by any programs and sub-programs.
inttab is referenced in the library `math.a`.

1.2.3.1.2 Function

The subroutine *inttab* has been made to perform an optimised linear interpolation.

1.2.3.1.3 Subordinates

It uses the module *precision*.

1.2.3.1.4 Dependencies

None.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.3.1.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
yi	Input function ordinate	r[ni]	any	i		
xi	Input function abscissa	r[ni]	any	i		
ni	Dimension of yi and xi	int	-	i		
xj	Function abscissa on which yi is interpolated	r[nj]	Same as xi	i		
nj	Dimension of xj	int	-	i		
Output						
yj	output function ordinate	d[nj]	-	o		

1.2.3.1.6 Processing

It interpolates on growing monotonous real abscissa (single precision).

1.2.3.2. inttab2

1.2.3.2.1 Type

inttab2 is a common subroutine that can be used by any programs and sub-programs.
inttab2 is referenced in the library `math.a`. and is included in the file *inttab.f90*.

1.2.3.2.2 Function

The subroutine *inttab2* has been made to perform an optimised linear interpolation. It is the same subroutine as *inttab* except that the abscissa are in double precision.

1.2.3.2.3 Subordinates

It uses the module *precision*.

1.2.3.2.4 Dependencies

None.

1.2.3.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
yi	Input function ordinate	r[ni]	any	i		
xi	Input function abscissa	d[ni]	any	i		
ni	Dimension of yi and xi	int	-	i		
xj	Function abscissa on which yi is interpolated	d[nj]	Same as xi	i		

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
nj	Dimension of xj	int	-	i		
Output						
yj	output function ordinate	d[nj]	-	o		

1.2.3.2.6 Processing

It interpolates on growing monotonous real abscissa (double precision).

1.2.3.3. intspline

1.2.3.3.1 Type

inspline is a common subroutine that can be used by any programs and sub-programs.
intspline is referenced in the library `math.a`.

1.2.3.3.2 Function

The subroutine *intspline* has been made to perform a cubic spline interpolation.

1.2.3.3.3 Subordinates

It uses the module *precision*.

1.2.3.3.4 Dependencies

The subroutine *intspline* calls the subroutine *getCubicSplineCoeff*.

1.2.3.3.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
ndim	Dimension of x and y	int	-	i		Must be greater than 2
kdim	Dimension of xx and yy	int	-	i		
x	Input function abscissa	d[ndim]	any	i		
y	Input function ordinate	d[ndim]	any	i		
xx	Output function abscissa	d[ndim]	Same as x	i		
Output						
yy	Output function ordinate	d[kdim]	any	o		

1.2.3.3.6 Processing

Performs the cubic spline interpolation using the coefficients calculated by the subroutine *getcubicsplinecoeff.*

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.3.4. getcubicsplinecoeff

1.2.3.4.1 Type

getcubicsplinecoeff is a common subroutine that can be used by any programs and sub-programs. *getcubicsplinecoeff* is referenced in the library `math.a` and is included in the file *intspline.f90*.

1.2.3.4.2 Function

The subroutine *Getcubicsplinecoeff* provides the coefficients for the cubic spline interpolation.

1.2.3.4.3 Subordinates

It uses the module *precision*.

1.2.3.4.4 Dependencies

None.

1.2.3.4.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
ndim	Dimension of x and y	int	-	i		Must be greater than 2
x	Input function abscissa	d[ndim]	Any	i		
y	Input function ordinate	d[ndim]	Any	i		
Input/Output						
cb	Cubic spline interpolation coefficient	d[ndim]	Funct. of x and y units	i/o		
cc						
cd						

1.2.3.4.6 Processing

Calculation of the coefficients for the cubic spline interpolation.

1.2.4. Physical tools

These tools are stored in the directory `scslib/sphysic`.

1.2.4.1. plkdir

1.2.4.1.1 Type

plkdir is a common subroutine that can be used by any programs and sub-programs. *plkdir* is referenced in the library `physic.a`.

1.2.4.1.2 Function

The subroutine *plkdir* has been made to compute the direct Plank function.

1.2.4.1.3 Subordinates

It uses the module *precision* and *constants*.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.4.1.4 Dependencies

None.

1.2.4.1.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
t	Reference brightness temperature	r	K	i		
ond	Wave number	r	cm^{-1}	i		
Output						
w	Plank function (radiances)	r	$W/(m^2.str.cm^{-1})$	o		

1.2.4.1.6 Processing

$$w = 2hc^2*ond^3 /[exp(hc/k*ond/t)-1]$$

1.2.4.2. plkder

1.2.4.2.1 Type

plkder is a common subroutine that can be used by any programs and sub-programs.
plkder is referenced in the library `physic.a`.

1.2.4.2.2 Function

The subroutine *plkder* has been made to compute the direct Plank function and its derivative with respect to the temperature.

1.2.4.2.3 Subordinates

It uses the module *precision* and *constants*

1.2.4.2.4 Dependencies

None.

1.2.4.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References/ Remarks
Input						
t	Reference brightness temperature	r	K	i		
ond	Wave number	r	cm^{-1}	i		
Output						
w	Plank function (radiances)	r	$W/(m^2.str.cm^{-1})$	o		
dwsdt	Plank function derivative	r	$W/(m^2.str.cm^{-1})/K$	o		

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.4.2.6 Processing

$$w = 2hc^2*ond^3/[exp(hc/k*ond/t)-1]$$

$$dwsdt = w/[exp(hc/k*ond/t)-1]*[exp(hc/k*ond/t)]*hc/k*ond/t^2$$

1.2.4.3. plkderdble

It is the same sub-routine as *plkder* except that it works with double precision floating point values.

1.2.4.4. tbrillans

1.2.4.4.1 Type

tbrillans is a function used by the programs `lirespc4a.f90` and `lirespi4a.f90` .
tbrillans is referenced in the library `physic.a`.

1.2.4.4.2 Function

The function *tbrillans* has been made to compute the inverse Plank function.

1.2.4.4.3 Subordinates

It uses the module *precision*.

1.2.4.4.4 Dependencies

None.

1.2.4.4.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
b	Plank function (radiance)	r	$W/(m^2.str.cm^{-1})$	i		
nu	Wave number	r	cm^{-1}	i		
Output						
tbrillans	Brightness temperature	r	K	o		

1.2.4.4.6 Processing

It computes the brightness temperature from the radiance.

1.2.5. Useful tools

These tools are stored in the directory `scslib/sutil`.

1.2.5.1. dichotom

1.2.5.1.1 Type

dichotom is a common subroutine that can be used by any programs and sub-programs.
dichotom is referenced in the library `util.a`.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.5.1.2 Function

The subroutine *dichotom* has been made to use the dichotomy technique for single precision floating point data.

1.2.5.1.3 Subordinates

It uses the module *precision*.

1.2.5.1.4 Dependencies

None.

1.2.5.1.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
tab	Available array	r[ir1:ir2]	any	i		
point	The point whose position has to be found in tab	r	any	i		
Input/Output						
ir1	As outputs they are the abscissa index of the array elements surrounding and including the point	int	-	i/o		
ir2						

1.2.5.1.6 Processing

It searches for the position of a point in an array by using the dichotomy technique. The array is monotonous growing.

1.2.5.2. dichotomd

1.2.5.2.1 Type

dichotomd is a common subroutine that can be used by any programs and sub-programs.
dichotomd is referenced in the library `util.a`.

1.2.5.2.2 Function

The subroutine *dichotomd* has been made to use the dichotomy technique for double precision floating point data. It is the same subroutine as *dichotom* except that the data are in double precision.

1.2.5.2.3 Subordinates

It uses the module *precision*.

1.2.5.2.4 Dependencies

None.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.5.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
tab	Available array	d[ir1:ir2]	any	i		
point	The point whose position has to be found in tab	d	any	i		
Input/Output						
ir1	As outputs they are the abscissa index of the array elements surrounding and including the point	int	-	i/o		
ir2						

1.2.5.2.6 Processing

It searches for the position of a point in an array by using the dichotomy technique. The array is monotonous growing.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.6. Atmospheric profile formatting

1.2.6.1. Flow chart

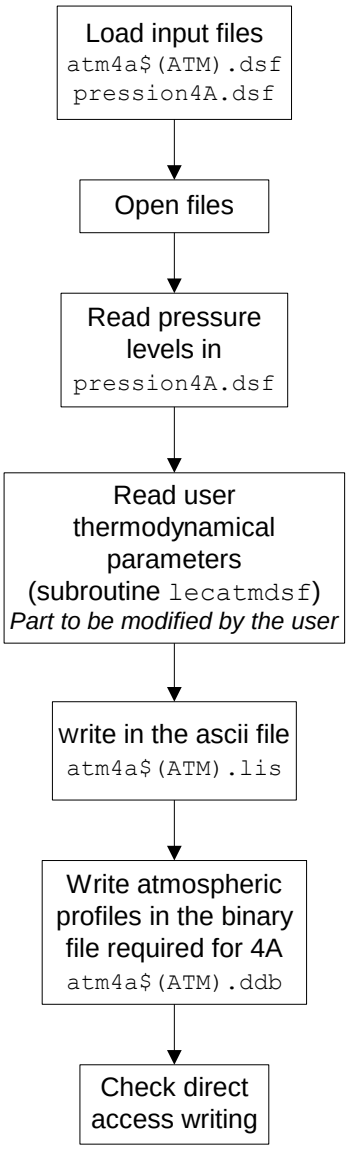


Diagram 11: Flow chart of the program for the atmospheric profile formatting `atmformat`

1.2.6.2. atmformat

1.2.6.2.1 Type

It is the main program for writing atmospheric profiles in the right format required to run 4A. It is stored in the directory `srclib`.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.6.2.2 Function

The program *atmformat* is dedicated to the reading of input user atmospheric profiles via the subroutine *lecatmdsf* and to write them in binary files with the appropriate format. The program *atmformat* has not to be modified; The user can only modify the subroutine *lecatmdsf.*

1.2.6.2.3 Subordinates

The program *atmformat* calls the subroutine *lecatmdsf*. It uses the modules *error* and *model*.

1.2.6.2.4 Dependencies

It depends on the existence of the required user input files of atmospheric profiles.

1.2.6.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Inputs						
filein1	Ascii file name of the atmospheric profile characteristics	string	-	i	Ini: makefile	datatm/atm4a$(ATM).dsf
filein2	Ascii file name of the atmospheric profile pressure levels	string	-	i	Ini: makefile	datatm/pression4A.dsf
fileout	Binary file name of the atmospheric profiles to create	string	-	i	Ini: makefile	datatm/atm4a$(ATM).ddb
filelist	Ascii file name of the atmospheric profiles to create	string	-	i	Ini: makefile	datatm/atm4a$(ATM).lis
atm	Kind name of the atmospheric profiles	string	-	i	Ini: makefile	$(ATM)
iat	Index number of the atmospheric profile	int	-	i		
Outputs						
buff	Atmospheric characteristics	r[Cst_nlevelmax*(Cst_ncorps+2)+Cst_ncorps+5]	-	o	fileout	atm4a$(ATM).ddb
iloc,ilas, nben	Addresses of each profile	int[Cst_natmax]	-	o	fileout	atm4a$(ATM).ddb
Inputs/Outputs (read in the subroutine lecatmdsf)						
p	Atmospheric pressure levels	r[nlevelmax]	-	i/o	Ini: pression4A.dsf/atm4a$(ATM).dsf	The pressure levels are defined in one of both files
t	Temperature profile at the pressure levels	r[nlevelmax]	K	i/o	Ini: atm4a$(ATM).dsf	
rolvl	Molecule concentration	r[nlevelmax,ncor	g/g	i/o	Ini:	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

	profile at the pressure levels	ps]			atm4a$(ATM).dsf	
tsol	Temperature at the surface	r	K	i/o	Ini: atm4a$(ATM).dsf	
psol	Pressure at the surface	r	-	i/o	Ini: atm4a$(ATM).dsf	
sectta	Secant value of the primary path angle	r	-	i/o	Ini: atm4a$(ATM).dsf	
idcor	Indentication numbers of the molecules (according to the GEISA notation) of which the concentration profile is defined in the input atmospheric file	int[ncorps]	-	i/o		
nlevel	Number of pressure levels	int	-	i/o		
natm	Number of atmospheric profiles	int	-	i/o		
nopro	Atmospheric profile number	int	-	i/o	Ini: atm4a$(ATM).dsf	
lat	latitude	int	°	i/o		
lon	longitude	int	°	i/o		
date	date (yymmdd)	int	-	i/o		
atm_version						
Local variables						
i, ibuff, irg, nc						
nmbuf, ipx, ipxu, ipxf, kk, nl						

1.2.6.2.6 Processing

The main functions of *atmformat* are:

- Load input files;
- Open files;
- Read pressure levels;
- Read user thermodynamical parameters (subroutine *lecatmdsf*);

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Write atmospheric profiles in binary files with the appropriate format (see section Part I. 5.2.1);
- Check direct access writing.

1.2.6.3. lecatmdsf

1.2.6.3.1 Type

lecatmdsf is a subroutine used by the program `atmformat.f90`.

1.2.6.3.2 Function

The subroutine *lecatmdsf* has been made to read the user thermodynamical parameters stored in the database file of the atmospheric profiles.

1.2.6.3.3 Subordinates

The subroutine *lecatmdsf* uses the modules *error* and *precision*.

1.2.6.3.4 Dependencies

It depends on the existence of the file of the atmospheric profile characteristics atm4a$(ATM).dsf.

1.2.6.3.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
ncorps	Maximum number of atmospheric molecules	int	-	i	Program atmformat	Parameter defined in module model used in atmformat: Cst_ncorps
nlevelmax	Maximum number of vertical levels	int	-	i	Program atmformat	Parameter defined in module model used in atmformat: Cst_nlevelmax
natmax	Maximum number of atmospheric profiles	int	-	i	Program atmformat	Parameter defined in module model used in atmformat: Cst_natmax
atm	Atmosphere kind name	string (len=30max)	-	i	Ini: makefile	
filein1	Input ascii filename of the atmospheric profile characteristics	string (len=100 max)	-	i	Ini: makefile	atm4a$(ATM).dsf
jin1	Logical unit defining the file to read filein1	int	-	i	Program atmformat	Defined in atmformat: 10
Input/Output						
p	Atmospheric pressure levels	r[nlevelmax]	-	i/o	Ini: pression4A.dsf/atm4a$(ATM).dsf	The pressure levels are defined in one of both files

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
iat	Index number of the atmospheric profile	int	-	i/o		
fin_fichier	End-of-file flag	string (len=3max)	-	i/o		oui or non
Output						
nopro	Atmospheric profile number	int	-	o	Ini: atm4a$(ATM).dsf	
tdown	Surface temperature	r	K	o	Ini: atm4a$(ATM).dsf	
pzdown	Surface pressure	r	hPa	o	Ini: atm4a$(ATM).dsf	
sectta	Secant value of the primary path angle (viewing angle)	r	-	o	Ini: atm4a$(ATM).dsf	
t	Temperature profile at the pressure levels	r[nlevelmax]	K	o	Ini: atm4a$(ATM).dsf	
rolvl	Molecule concentration profile at the pressure levels	r[nlevelmax,ncorps]	g/g	o	Ini: atm4a$(ATM).dsf	
nlevel	Number of pressure levels	int	-	o		
nbcor	Number of involved molecules of which the concentration profile is defined in the input user atmospheric file	int	-	o		
idcor	Indentication numbers of the molecules (according to the GEISA notation) of which the concentration profile is defined in the input atmospheric file	int[ncorps]	-	o		
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if reading is successful Err_error otherwise

1.2.6.3.6 Processing

The user can modify the subroutine in order to implement its format for reading its atmospheric profiles.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

The conventional numbering of the levels is (see section Part I.5.2.5.2):

- level=1: up of the atmosphere;
- level=nlevel: down of the atmosphere.

Each thermodynamical parameter has to be specified as :

- Temperature in K;
- Pressure in hPa.

And optionally:

- Gas mixing ratio in g/g;
- Secant value of the primary path angle sectta = 1/cos(viewing angle).

The pressure levels are defined either through the file `pression4A.dsf` or in the input user file `atm4a$(ATM).dsf`.
The user is free to specify any gas mixing ratio profile provided that this gas is defined in the GEISA database (see appendix Part III section 2). If the user defines any gas mixing ratio profile and if the user wants to simulate the radiative transfer by including the effect of some gases, the gas mixing ratios are defined through the default gas concentration file `gascon.dsf` (see subroutines *gascon* and *atmsph*).

The input file is necessary in an ASCII format. If you want binary data, you must close the unit "jin1" and reopen it with unformatted format as follows:

```
if (atm == 'example') then
  close(jin1)
  open(unit=jin1,file=filein1,form='unformatted')
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.7. Instrument function formatting

1.2.7.1. Flow chart

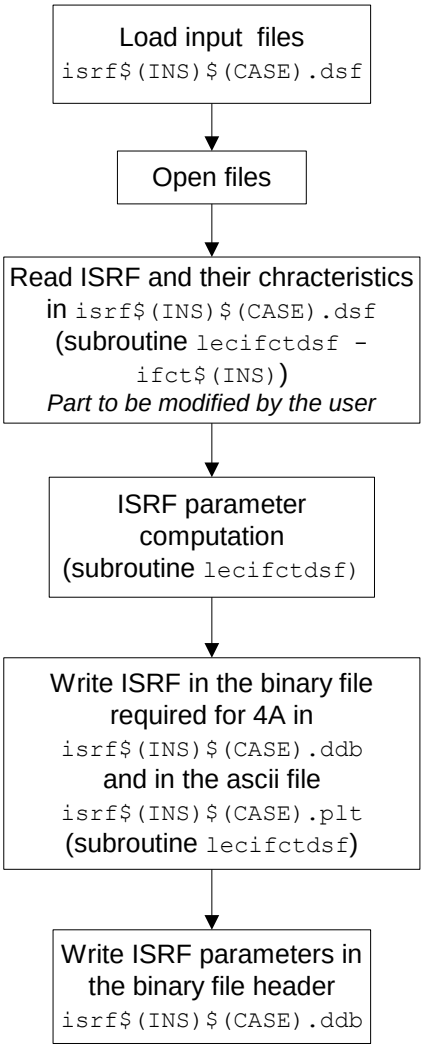


Diagram 12: Flow chart of the program for the instrument function formatting `ifct`

1.2.7.2. ifct

1.2.7.2.1 Type

It is the main program for writing instrumental spectral response functions (ISRF) in the right format required to run 4A. It is stored in the directory `srclib`.

1.2.7.2.2 Function

The program *ifct* is dedicated to the reading of input user ISRF via the subroutine *lecifctdsf* and to write them in binary files with the appropriate format. The program *ifct* has not to be modified; The user can only modify the subroutine *lecifctdsf.*

1.2.7.2.3 Subordinates

The program *ifct* calls the subroutine *lecifctdsf*. It uses the module *model*.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.7.2.4 Dependencies

It depends on the existence of the required user input files of ISRF.

1.2.7.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Inputs						
isrfin	Input ascii file describing the ISRF	string	-	i	Ini: makefile	isrf/isrf(INS)(CASE).dsf
isrfu	Name of the binary file to create	string	-	i	Ini: makefile	isrf/isrf(INS)(CASE).ddb
isrfplot	Name of the ascii file to create	string	-	i	Ini: makefile	outputascii/isrf(INS)(CASE).plt
Outputs						
isrf	ISRF	r[kpmax]	-	o	isrfu isrfplot	isrf(INS)(CASE).ddb isrf(INS)(CASE).plt
Local variables						
jisrfu	Unit number of the output binary file	int	-	-	Ini: makefile	Equal to 10
jisrfplot	Unit number of the output ascii file	int	-	-	Ini: makefile	Equal to 72
jisrfin	Unit number of the input file	int	-	-	Ini: makefile	Equal to 71

1.2.7.2.6 Processing

The main functions of *ifct* are:

- Load input files;
- Open files;
- Read ISRF and their characteristics (subroutine *lecifctdsf*);
- Write ISRF in the binary file with the appropriate format (see section Part I.5.2.3) and in an ascii file (subroutine *lecifctdsf*);
- Write ISRF characteristics in the binary file header.

1.2.7.3. lecifctdsf - ifct$(INS)

1.2.7.3.1 Type

lecifctdsf is a subroutine used by the program `ifct.f90`.

1.2.7.3.2 Function

The subroutine *lecifctdsf* has been made to read the user-defined instrument spectral response function specified through the variable INS. A subroutine is required for each instrument type INS. The user has to create the subroutine according to the subroutine examples provided with the software 4A/OP (examples: *ifcttest.f90* or *ifcttest2.f90*).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.7.3.3 Subordinates

The subroutine *lecifctdsf* uses the modules *model* and *instruments*.

1.2.7.3.4 Dependencies

It depends on the existence of the ISRF file isrf(INS)(CASE).dsf.

1.2.7.3.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
nbmax	Maximum number of bands	int	-	i	Program ifct	Parameter defined in module model used in ifct: Cst_nbmax
jisrfin	Unit number of the input file	int	-	i	Program ifct	Defined in ifct: 71
jisrfplot	Unit number of the output ascii file	int	-	i	Program ifct	Defined in ifct: 72
jisrfout	Unit number of the output binary file	int	-	i	Ini: makefile	Defined in ifct: 10
isrfin	Input filename of the ISRF	string (len=100 max)	-	i	Ini: makefile	isrf(INS)(CASE).dsf
isrfout	Output binary filename of the ISRF	string (len=100 max)	-	i	Ini: makefile	isrf(INS)(CASE).ddb
Input/Output						
irg	Record index	int	-	i/o		
Output						
titre	ISRF title (quick description of the ISRF)	string (len=100 max)	-	o	Ini: isrf(INS)(CASE).dsf	
nbk	Number of spectral bands	int	-	o	Ini: isrf(INS)(CASE).dsf	
itypeconv	Convolution type	int	-	o		See section 5.2.5.2
nufdeb	First central wave number of the ISRFs	r[Cst_nbmax]	cm^{-1}	o	Ini: isrf(INS)(CASE).dsf	
nuffin	Last central wave number of the ISRFs	r[Cst_nbmax]	cm^{-1}	o	Ini: isrf(INS)(CASE).dsf	
dnds	Spectral step of central wave numbers	r[Cst_nbmax]	cm^{-1}	o	Ini: isrf(INS)(CASE).dsf	
nknu	Number of ISRFs	int[Cst_nbmax]	-	o	Ini: isrf(INS)(CASE).	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
					dsf	
kfmo	Number of description points of the ISRF	int[Cst_knumax, Cst_nbmax]	-	o	Ini: isrf(INS)(CASE).dsf	
pdnuco	ISRF sampling step	r[Cst_nbmax]	cm^{-1}	o	Ini: isrf(INS)(CASE).dsf	
nufo	Central wave numbers of each ISRF	r[Cst_knumax, Cst_nbmax]	cm^{-1}	o	Ini: isrf(INS)(CASE).dsf	
wnuo	Width of each ISRF	r[Cst_knumax, Cst_nbmax]	cm^{-1}	o	Ini: isrf(INS)(CASE).dsf	
dnuo	Spectral shift of each ISRF	r[Cst_knumax, Cst_nbmax]	cm^{-1}	o	Ini: isrf(INS)(CASE).dsf	
irecf	Record number of the beginning of each ISRF	int[Cst_knumax, Cst_nbmax]	-	o	Ini: isrf(INS)(CASE).dsf	

1.2.7.3.6 Processing

The user can modify the subroutine in order to implement its format for reading its ISRF.

The various steps of *lecifctdsf* are:

- Part of the code the user can modify according to the user-defined ISRF type and format:
  - Defines the instrument type with the variable itypeconv (see module *instruments*):
    - itypeconv=1: Radiometer with n channels (HIRS),
    - itypeconv=2: Interferometer / spectrometer with a constant sampling step in wave number (IASI: step = 0.25 cm^{-1}),
    - itypeconv=3: Interferometer / spectrometer with a non constant sampling step and fixed wave numbers (AIRS),
  - Reads the user-defined instrument function;
- Writes the ISRF in a binary file with the appropriate format for 4A;
- Writes the ISRF in an ascii file with the corresponding wave numbers.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.8. Formatted writing of the 4A output concerning high-resolution spectra

1.2.8.1. Flow chart

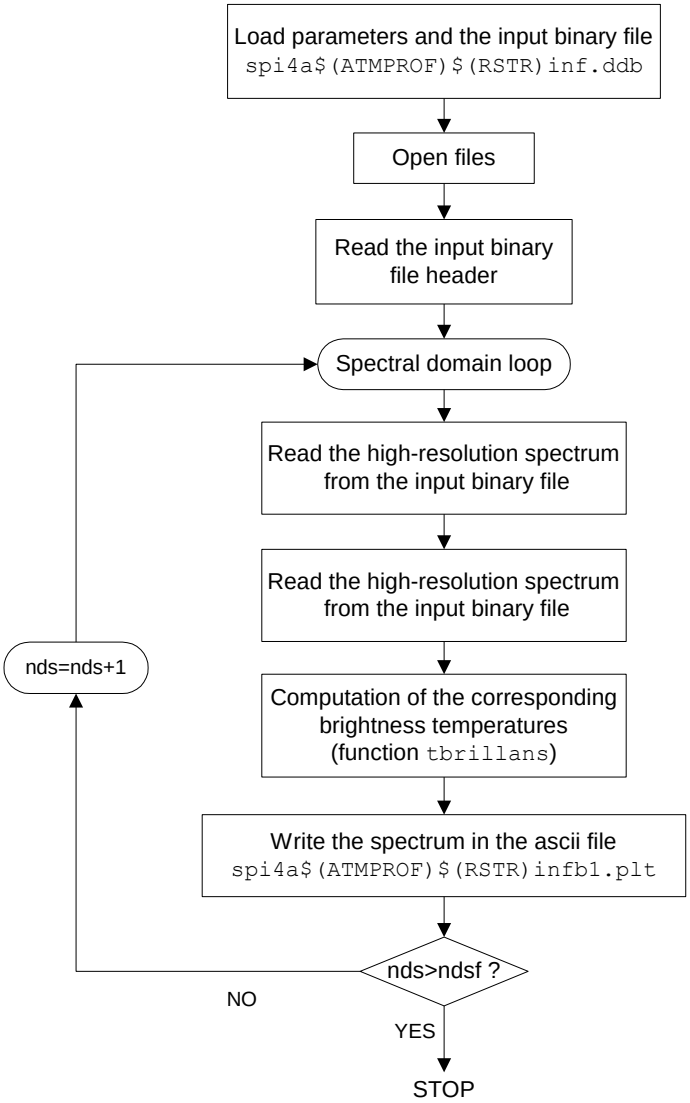


Diagram 13: Flow chart of the program for writing4A output high-resolution spectra `lirespi4a`

1.2.8.2. lirespi4a

1.2.8.2.1 Type

It is the main program for writing the 4A outputs concerning high-resolution ("infinite") spectra. It is stored in the directory `srclib`.

1.2.8.2.2 Function

The program *lirespi4a* is dedicated to the reading of the 4A high-resolution spectra stored in binary format and to write them in ASCII format.

1.2.8.2.3 Subordinates

The program *lirespi4a* calls the function *tbrillans* (see section 1.2.4.3). It uses the modules *error* and *model*.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.8.2.4 Dependencies

It depends on the existence of a 4A/OP binary file with a high-resolution spectrum.

1.2.8.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Inputs						
numin	Lower wave number limit of the extraction for the writing into ASCII format	r	cm^{-1}	i	Ini: makefile	
numax	Upper wave number limit of the extraction	r	cm^{-1}	i	Ini: makefile	
fspi	Input binary file name	string	-	i	Ini: makefile	outputbin/spi4a$(ATMPROF) $(RSTR)inf.ddb
fgnu	Output ASCII file name for the spectrum	string	-	i	Ini: makefile	outputascii/spi4a$(ATMPROF) $(RSTR)infb1.plt
Outputs						
ond	wave numbers	$d[2^{19}+1]$	cm^{-1}	o	fgnu	spi4a$(ATMPROF)$(RSTR)infb1.plt : see section Part I 5.3.1.2.1
spi	radiances	$r[2^{19}+1]$	W/ (m^2.str d. cm^{-1})			
temp	brightness temperatures	$r\ [2^{19}+1]$	K			

1.2.8.2.6 Processing

The main functions of *lirespi4a* are:

- Load the input binary file and parameters;
- Open files;
- Read the input binary file header;
- Read the high-resolution radiance for each spectral domain;
- Computation of the corresponding wave numbers;
- Computation of the brightness temperatures corresponding to the high-resolution radiances (function *tbrillans*);
- Write the spectrum in ASCII files.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.9. Formatted writing of the 4A output concerning convolved spectra and Jacobians

1.2.9.1. Flow chart

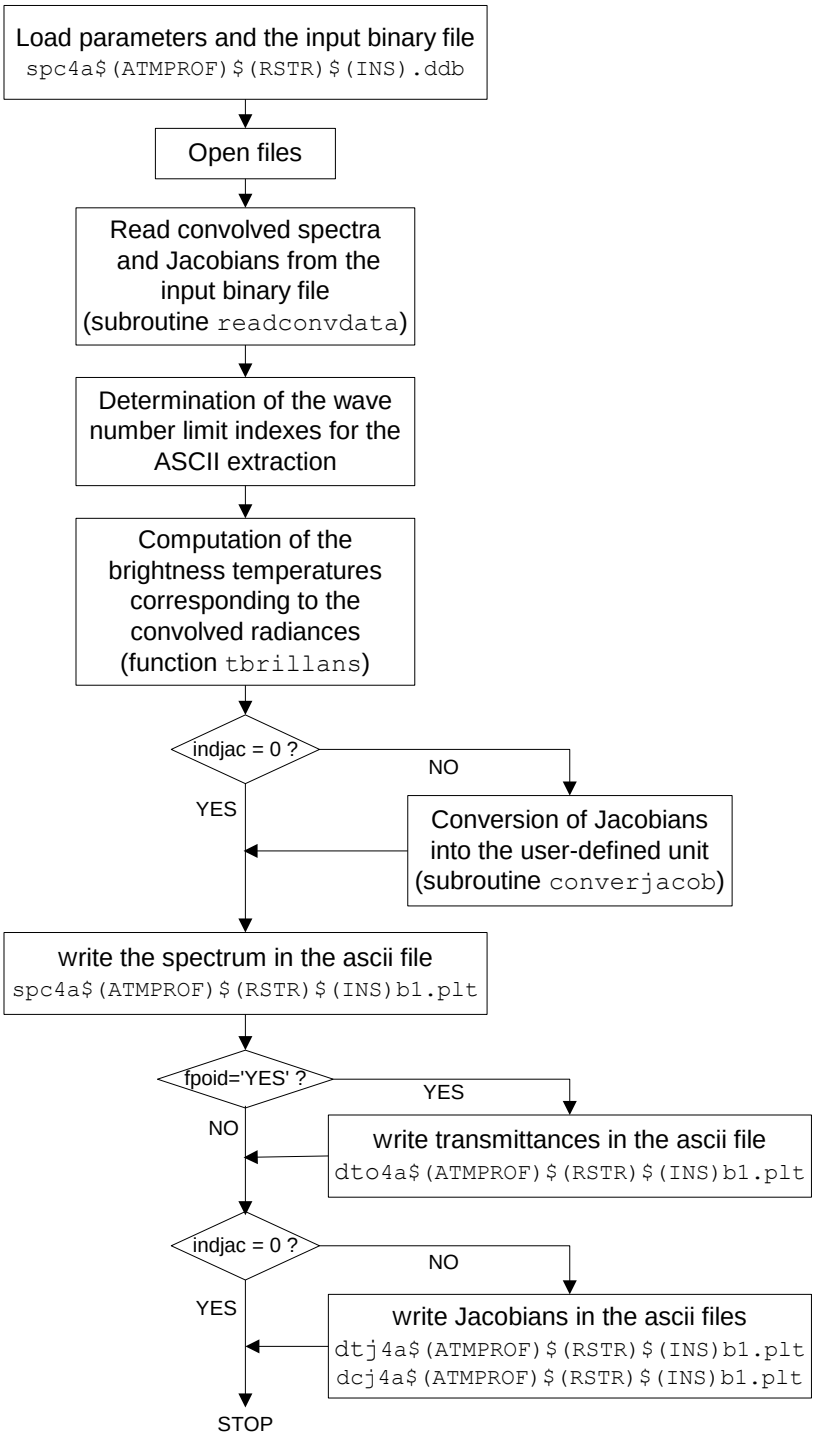


Diagram 14: Flow chart of the program for writing4A output convolved spectra and Jacobians `lirespc4a`

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.9.2. lirespc4a

1.2.9.2.1 Type

It is the main program for writing the 4A outputs concerning convolved spectra and Jacobians. It is stored in the directory `srclib`.

1.2.9.2.2 Function

The program *lirespc4a* is dedicated to the reading of the 4A convolved spectra stored in binary format via the subroutine *readconvdata* and to write them in ASCII format.

1.2.9.2.3 Subordinates

The program *lirespc4a* calls the subroutines *readconvdata* and *converjacob* and the function *tbrillans* (see section 1.2.4.3). It uses the modules *error*, *model, constants* and *spectrumio*.

1.2.9.2.4 Dependencies

It depends on the existence of a 4A/OP binary file with a convolved spectrum.

1.2.9.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Inputs						
unitjac	Unit index for Jacobian unit conversion	int	-	i	Ini: makefile	See Table 8 in section Part I 5.2.5.1
tref	Reference temperature for conversion into NedT	r	K	i	Ini: makefile	
tebr	Radiometric noise estimation for conversion into Decibel	r	K	i	Ini: makefile	
numin	Lower wave number limit of the extraction for the writing into ASCII format	d	cm^{-1}	i	Ini: makefile	
numax	Upper wave number limit of the extraction	d	cm^{-1}	i	Ini: makefile	
nopro	Atmospheric profile number	int	-	i	Ini: makefile	
fspf	Input binary file name	string	-	i	Ini: makefile	outputbin/spc4a$(ATMPROF) $(RSTR)$(INS).ddb
fgnu1	Output ASCII file name for the spectrum	string	-	i	Ini: makefile	outputascii/spc4a$(ATMPROF) $(RSTR)$(INS)b1.plt

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
fgnu2	Output ascii file name for the transmittances	string	-	i	Ini: makefile	outputascii/dto4a$(ATMPROF)$(RSTR)$(INS)b1.plt
fgnu3	Output ascii file name for the jacobians in temperature	string	-	i	Ini: makefile	outputascii/dtj4a$(ATMPROF)$(RSTR)$(INS)b1.plt
fgnu4	Output ascii file name for the jacobians in concentration	string	-	i	Ini: makefile	outputascii/dcj4a$(ATMPROF)$(RSTR)$(INS)b1.plt
Outputs						
ond	Wave numbers	d [allocatable]	cm^{-1}	o	fgnu1	spc4a$(ATMPROF)$(RSTR)$(INS)b1.plt: see section Part I 5.3.1.2.2
spf	radiances	r [allocatable]	W/(m^2.str d. cm^{-1})			
temp	corresponding brightness temperatures	r [allocatable]	K			
pc	Layer pressures	r [Cst_nlevelmax]	hPa	o	fgnu2	dto4a$(ATMPROF)$(RSTR)$(INS)b1.plt: see section Part I 5.3.1.2.6
pl	pressure levels	r [Cst_nlevelmax]	hPa			
ond	wave numbers	d [allocatable]	cm^{-1}			
cdto	transmittances	r [allocatable, allocatable]	-			
pc	Layer pressures	r [Cst_nlevelmax]	hPa	o	fgnu3	dtj4a$(ATMPROF)$(RSTR)$(INS)b1.plt: see section Part I 5.3.1.2.3
pl	pressure levels	r [Cst_nlevelmax]	hPa			
ond	wave numbers	d [allocatable]	cm^{-1}			
ctjac	jacobians in temperature	r [allocatable, allocatable]	W/(m^2.str d. cm^{-1})/K			

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
pc	Layer pressures	r [Cst_nlevelmax]	hPa	o	fgnu4	dcj4a$(ATMPROF)$(RSTR)$(INS)b1.plt: see section Part I 5.3.1.2.4
pl	pressure levels	r [Cst_nlevelmax]	hPa			
ond	wave numbers	d [allocatable]	cm^{-1}			
ccjac	jacobians in concentration	r [allocatable, allocatable, allocatable]	W/(m^2.str d. cm^{-1}) / (g/g)			

1.2.9.2.6 Processing

The main functions of *lirespc4a* are:

- Load the input binary file and parameters
- Open files (output and input)
- Read the convolved spectrum and Jacobians (subroutine *readconvdata*)
- Determination of the wave number limits for the ASCII extraction
- Computation of the brightness temperatures corresponding to the convolved radiances (function *tbrillans*)
- Conversion of Jacobians into the user-defined unit according to the variable unitjac (subroutine *converjacob*), if it is required
- Write the spectrum and Jacobians in ASCII files

1.2.9.3. readconvdata

1.2.9.3.1 Type

readconvdata is a subroutine used by the program `lirespc4a.f90`.
readconvdata is defined within the module `spectrumio`.

1.2.9.3.2 Function

The subroutine *readconvdata* has been made to read convolved spectra in the 4A/OP format.

1.2.9.3.3 Subordinates

The subroutine *readconvdata* uses the modules *error*, *precision, constants and model*.

1.2.9.3.4 Dependencies

It depends on the existence of a 4A/OP binary file with a convolved spectrum.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.9.3.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
jlist	Writing logical unit (standard output)	int	-	i	Program lirespc4a	Defined in lirespc4a: 6
jread	Logical unit defining the file to read fread	int	-	i	Program lirespc4a	Defined in statanalys: 6
maxbuf	Maximum buffer size	int	-	i	Program lirespc4a	Parameter defined in module model used in lirespc4a: Cst_maxbuf
fread	Input binary file of convolved spectra	string (len=400 max)	-	i	Ini: makefile	
Output						
nbk	Total number of spectral bands	int	-	o	Ini: fread	
iadcnv	Number of convolution points	int	-	o	Ini: fread	
itypeconv	Indentifier of the convolution	int	-	o	Ini: fread	1, 2 or 3 (radiometer, interferometer / spectrometer)
fpoid	Transmittance calculation flag	string (len=6 max)	-	o	Ini: fread	YES or NO
indjac	Jacobian flag	int	-	o	Ini: fread	1=yes, 0=no
jacsel	Array of indexes to select the calculation of the Jacobian for each molecule among the 42 gases available in the GEISA database and 3 CFCS	int[Cst_ncorps]	-	o	Ini: fread	
level1	First level (up) of the atmosphere	int	-	o	Ini: fread	
level2	Last level (down) of the atmosphere	int	-	o	Ini: fread	
ond	Wave number of output channels: Central wave numbers of each ISRF in the convolution domain (iadcnv wave numbers)	d[allocatable]	cm^{-1}	i/o	Ini: fread	Allocated in readconvdata Size: number of convolution wave numbers

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
spf	Convolved radiance per channel	r [allocatable]	W/ (m^2.str d. cm^{-1})	i/o	Ini: fread	Allocated in readconvdata Size: number of convolution wave numbers
cdto	Convolved transmittances	r [allocatable, allocatable]	-	i/o	Ini: fread	Allocated in readconvdata Size: number of convolution wave numbers, Cst_nlevelmax
ctjac	Convolved Jacobians with respect to temperature	r [allocatable, allocatable]	W/ (m^2.str d. cm^{-1})/ K	i/o	Ini: fread	Allocated in readconvdata Size: number of convolution wave numbers, Cst_nlevelmax
ccjac	Convolved Jacobians with respect to molecule concentration	r [allocatable, allocatable,allocatable]	W/ (m^2.str d. cm^{-1})/ (g/g)	o	Ini: fread	Allocated in readconvdata Size: number of convolution wave numbers, Cst_nlevelmax, Cst_ncjacmax
nopro	Atmospheric profile index number	int	-	o	Ini: fread	
trajet	Viewing configuration	string (len=6 max)	-	o	Ini: fread	UP, DOWNUP, DOWN, UPDOWN UP: nadir viewing DOWN: zenith viewing
pascon	Convolution step: step of the ISRF application	d	cm^{-1}	o	Ini: fread	
numsel	Index array of selected molecules for the radiative transfer computation	int[Cst_ncorps]	-	o	Ini: fread	
pl	Pressure levels of the simulated atmospheric profiles	r[Cst_nlevelmax]	hPa	o	Ini: fread	
tl	Temperature profile at the pressure levels	r[Cst_nlevelmax]	K	o	Ini: fread	
rol	Molecule concentration profile at the pressure levels pl	r[Cst_nlevelmax, Cst_ncorps]	g/g	o	Ini: fread	
ncsel	Total number of selected molecules	int	-	o	Ini: fread	

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
ncosel	Indentication numbers of the selected molecules according to the GEISA notation	int[Cst_ncorps]	-	o	Ini: fread	
tdown	Temperature of the lower level	r	K	o	Ini: fread	
pzdown	Pressure value at the lower level of the atmosphere	r	hPa	o	Ini: fread	
iemdown	Index for the Emissivity of the lower level	int	-	o	Ini: fread	
tup	Temperature of the top level	r	K	o	Ini: fread	
pzup	Pressure value at the upper level	r	hPa	o	Ini: fread	
iemup	Index for the Emissivity of the top level	int	-	o	Ini: fread	
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if opening is successful Err_error otherwise

1.2.9.3.6 Processing

This subroutine is dedicated to the reading of convolved spectra and Jacobians.

Remark: Output data are for one spectral band only.

1.2.9.4. converjacob

1.2.9.4.1 Type

converjacob is a subroutine used by the program `lirespc4a.f90`.

1.2.9.4.2 Function

The subroutine *converjacob* has been made to convert Jacobians into a user-defined unit.

1.2.9.4.3 Subordinates

The subroutine *converjacob* calls the subroutine *plkder*. It uses the modules *error*, *precision, constants and model*.

1.2.9.4.4 Dependencies

It depends on the existence of convolved Jacobians.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.9.4.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
dimen	Dimension of ond	int	-	i	Program lirespc4a	
level1	First level (up) of the atmosphere	int	-	i	Ini: fread	
level2	Last level (down) of the atmosphere	int	-	i	Ini: fread	
icd	First limit of the wave numbers	int	-	i	Program lirespc4a	
icf	Last limit of the wave numbers	int	-	i	Program lirespc4a	
ond	Wave numbers	r[dimen]	cm^{-1}	i	Program lirespc4a	
temp	Brightness temperatures	r[dimen]	K	i	Program lirespc4a	
dp	Thickness of the pressure layers of the simulated atmospheric profiles	r[0:Cst_nlevelmax]	hPa	i	Program lirespc4a	
roc	Molecule concentration profile at the layer pressures	r[Cst_nlevelmax, Cst_ncorps]	g/g	i	Program lirespc4a	
rog	Air concentration per pressure layer	r[Cst_nlevelmax]	Kg/m^3	i	Program lirespc4a	
jacsel	Array of indexes to select the calculation of the Jacobian for each molecule among the 42 gases available in the GEISA database and 3 CFCS	int[Cst_ncorps]	-	i	Ini: fread	
tref	Reference temperature for NedT	r	K	i	Program lirespc4a	
tebr	Radiometric noise estimation for Decibel	r	K	i	Program lirespc4a	
unitjac	Jacobian unit index for ASCII outputs	int	-	i	Program lirespc4a	0,1,2,3,4 or 5
jlist	Writing logical unit (standard output)	int	-	i	Program lirespc4a	Defined in lirespc4a: 6
Input/Output						
ctjac	Convolved Jacobians with respect to temperature	r[dimen,0 :Cst_nlevelmax]	$W/(m^2.str.cm^{-1})/K$	i/o	Program lirespc4a	The output unit depends on unitjac

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
ccjac	Convolved Jacobians with respect to gas mixing ratio	r[dimen,0 :Cst_nlevelmax, Cst_ncjacmax]	W/ (m^2.str d. cm^{-1})/ (g/g)	i/o	Program lirespc4a	The output unit depends on unitjac
	Convolved Jacobians with respect to the surface emissivity		-			
Output						
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if unitjac is correct Err_error otherwise

1.2.9.4.6 Processing

The user has the choice among six possible units:

- unit = 0:
  - Jacobians with respect to temperature: Kelvin (Nedt at tref) per Kelvin.layer,
  - Jacobians with respect to gas mixing ratio: Kelvin per layer for 1 g/g variation of the mixing ratio,
  - Jacobians with respect to the emissivity: Kelvin (Nedt at tref) for 1% variation;
- unit = 1:
  - Temperature: Kelvin (Nedt at tref) per Kelvin.Km,
  - Gas mixing ratio: Kelvin (Nedt at tref) per Km for 10% variation of the mixing ratio,
  - Emissivity: Kelvin (Nedt at tref) for 1% variation;
- unit = 2:
  - Temperature: Decibel per Kelvin.Km,
  - Gas mixing ratio: Decibel per Km for 10% variation of the mixing ratio,
  - Emissivity: Decibel for 1% variation;
- unit = 3:
  - Temperature: Kelvin (Nedt at T_b) per Kelvin.Km,
  - Gas mixing ratio: Kelvin (Nedt at T_b) per Km for 10% variation of the mixing ratio,
  - Emissivity: Kelvin (Nedt at T_b) for 1% variation;
- unit = 4:
  - Temperature: Kelvin (Nedt at T_b) per Kelvin.layer,
  - Gas mixing ratio: Kelvin (Nedt at T_b) per layer for 1 g/g variation of the mixing ratio,
  - Emissivity: Kelvin (Nedt at T_b) for 1% variation;

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- unit = 5: No change:
  - Temperature: W/(m^2 str cm^{-1}) per Kelvin.layer,
  - Gas mixing ratio: W/(m^2 str cm^{-1}) per layer for 1 g/g variation of the mixing ratio,
  - Emissivity: W/(m^2 str cm^{-1});

1.2.10. Statistics tool

1.2.10.1. statanalys

1.2.10.1.1 Type

It is the program for computing the difference between two spectra and the corresponding statistics. It is stored in the directory `srclib`.

1.2.10.1.2 Function

The program *statanalys* is dedicated to the computation of the normalised difference in NedT and the difference in radiance between two spectra. Statistical characteristics of this difference are also computed.

Warning! Both spectra must be sampled with the same sampling step.

1.2.10.1.3 Subordinates

The program *statanalys* calls the subroutines *readconvsp, plkderdble, dichotomd, statchar* and *statcharClass*. It uses the modules *error*, *spectrumio, spstats*, *precision* and *model*.

1.2.10.1.4 Dependencies

It depends on the existence of 4A/OP output binary files with convolved spectra.

1.2.10.1.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Inputs						
tref	Reference temperature in NedT	d	K	i	Ini: makefile	
wclass	Width of the class for statistical characteristics	d	cm^{-1}	i	Ini: makefile	
wnmin	Limits of the computation	d	cm^{-1}	i	Ini: makefile	
wnmax		d	cm^{-1}	i	Ini: makefile	
fcomspect	Input binary file name of the spectrum to be compared	string	-	i	Ini: makefile	outputbin/spc4a$(ATMPROF)$(RSTR)$(INS)2.ddb
frefspect	Input binary file name of the reference spectrum	string	-	i	Ini: makefile	outputbin/spc4a$(ATMPROF)$(RSTR)$(INS)1.ddb
fasc1	Output ASCII file name of the statistical characteristics	string	-	i	Ini: makefile	User-defined
fasc2	Output ASCII file name of the statistical characteristics per class	string	-	i	Ini: makefile	User-defined

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Outputs						
wnScomspect	Wave numbers	d [allocatable]	cm^{-1}	o	fasc1	User-defined Size: number of convolution wave numbers
RdScomspect	Spectrum (radiance) to be compared	r [allocatable]	W/ (m^2.str d. cm^{-1})	o	fasc1	User-defined Size: number of convolution wave numbers
RdSrefspect	Reference spectrum	r [allocatable]	W/ (m^2.str d. cm^{-1})	o	fasc1	User-defined Size: number of convolution wave numbers
diff	Normalised difference in NedT at tref between both spectra	d [allocatable]	K	o	fasc1	User-defined Size: number of convolution wave numbers
RdScomspect - RdSrefspect	Difference between both spectra	r [allocatable]	W/ (m^2.str d. cm^{-1})	o	fasc1	User-defined Size: number of convolution wave numbers
wnSclass	Wave numbers corresponding to the classes	d [allocatable]	cm^{-1}	o	fasc2	User-defined Allocated in spstats Size: number of convolution wave numbers
rmsClass	Root mean square Standard deviation of the difference per class	d [allocatable]	K	o	fasc2	User-defined Allocated in spstats Size: number of convolution wave numbers, 2
percentil	Percentiles at 90, 95 and 100 %	r [allocatable]	-	o	fasc2	User-defined Allocated in spstats Size: 3, number of convolution wave numbers

1.2.10.1.6 Processing

The main functions of *statanalys* are:

- Load parameters from standard input;
- Load input files;
- Open files;
- Read both spectra (subroutine *readconvsp*);
- Computation of the normalized difference in NedT at tref (subroutine *plkderdble*);
- Computation of the sample limits (subroutine *dichotomd*);
- Computation of the statistical characteristics of the difference: mean, quadratic mean, variance, standard deviation, root mean square (subroutine *statchar*);
- Computation of the statistical characteristics of the difference per wave number class with the width wclass: root mean square, standard deviation, percentiles at 90, 95 and 100 % (subroutine *statcharClass*);
- Write results in ASCII files (see format in section 5.3.4).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.10.2. readconvsp

1.2.10.2.1 Type

readconvsp is a subroutine used by the program `statanalys.f90`.
readconvsp is defined within the module `spectrumio`.

1.2.10.2.2 Function

The subroutine *readconvsp* has been made to read convolved spectra in the 4A/OP format.

1.2.10.2.3 Subordinates

The subroutine *readconvsp* calls the subroutine *readconvdata*. It uses the modules *error*, *precision, constants and model*.

1.2.10.2.4 Dependencies

It depends on the existence of a 4A/OP binary file with a convolved spectrum.

1.2.10.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
jlist	Writing logical unit (standard output)	int	-	i	Program statanalys	Defined in statanalys: 6
jread	Logical unit defining the file to read fread	int	-	i	Program statanalys	Defined in statanalys: 6
maxbuf	Maximum buffer size	int	-	i	Program statanalys	Parameter defined in module model used in statanalys: Cst_maxbuf
fread	Input binary file of convolved spectra	string (len=400 max)	-	i	Ini: makefile	
Output						
iadcnv	Number of convolution points	int	-	o	Ini: fread	
ond	Wave number of output channels: Central wave numbers of each ISRF in the convolution domain (iadcnv wa. numbers)	d[allocatable]	cm^{-1}	o	Ini: fread	Allocated in readconvdata
spf	Convolved radiance per channel	r [allocatable]	$W/(m^2.str\ d.\ cm^{-1})$	o	Ini: fread	Allocated in readconvdata
ErrCode	Error code	int	-	o		Err_ok or Err_error Err_ok if opening is successful Err_error otherwise

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.10.2.6 Processing

This subroutine includes the subroutine *readconvdata* with less output arguments for people that only need the spectrum and the wave numbers.

1.2.10.3. statchar

1.2.10.3.1 Type

statchar is a subroutine used by the program `statanalys.f90`.
statchar is defined within the module `spstats`.

1.2.10.3.2 Function

The subroutine *statchar* has been made to compute the statistical characteristics of the difference between two spectra. It can also be used for the computation of the statistics of any quantity.

1.2.10.3.3 Subordinates

The subroutine *statchar* uses the modules *error* and *precision*.

1.2.10.3.4 Dependencies

None.

1.2.10.3.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
nbwnS	Dimension of diff	int	-	i	Program statanalys	
diff	Difference between two spectra in NedT or any quantity	d[nbwnS]	K (*)	i	Program statanalys	
Output						
sm	Mean of the difference	d	K (*)	o		
sqm	Quadratic mean	d	K^2 (*)	o		
vs	Variance	d	K^2 (*)	o		
std	Standard deviation	d	K (*)	o		
rms	Root mean square	d	K (*)	o		

(*) It is the unit in the context of 4A/OP but it can be any other unit for a use of this subroutine in an other context

1.2.10.3.6 Processing

This subroutine computes the mean, the quadratic mean, the variance, the standard deviation and the root mean square of any quantity.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.10.4. statcharClass

1.2.10.4.1 Type

statcharClass is a subroutine used by the program `statanalys.f90`.
statcharClass is defined within the module `spstats`.

1.2.10.4.2 Function

The subroutine *statcharClass* has been made to compute the statistical characteristics of the errors per wave number class from the difference between two spectra. It can also be used for the computation of the statistics of any quantity.

1.2.10.4.3 Subordinates

The subroutine *statcharClass* calls the subroutine *dichotomd* and *inttab2*. It uses the modules *error*, *precision* and model.

1.2.10.4.4 Dependencies

None.

1.2.10.4.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References/ Remarks
Input						
nbwnS	Dimension of diff	int	-	i	Program statanalys	
diff	Difference between two spectra defined on wnS	d[nbwnS]	K (*)	i	Program statanalys	
wnS	Wave numbers corresponding to the compared spectrum	d[nbwnS]	cm^{-1} (*)	i	Program statanalys	
wnSref	Wave numbers corresponding to the reference spectrum	d[nbwnS]	cm^{-1} (*)	i	Program statanalys	
wclass	Width of the class for statistical characteristics of errors per wave number class	d	cm^{-1} (*)	i	Ini: makefile	
Output						
percent	Percentages of percentiles (90, 95 and 100 %)	d[3]	-	o		
percentil	Percentiles at percent	r[allocatable, allocatable]	-	o		Size: 3, number of convolution wave numbers
wnSclass	Wave numbers corresponding to the classes	R[allocatable]	Same as wnS	o		Size: number of convolution wave numbers
rmsClass	Array containing the root	r[allocatable,	Same	o		Size: number of

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References/ Remarks
	mean square and the standard deviation of the difference per class	allocatable]	as diff			convolution wave numbers, 2
nbClass	Number of classes	int	-	o		

(*) It is the unit in the context of 4A/OP but it can be any other unit for a use of this subroutine in an other context

1.2.10.4.6 Processing

This subroutine computes the statistical characteristics per class defined by its width in wave numbers. It can also be used in order to compute this kind of characteristics for any quantity defined on any other type of unity that defines the class.

The various steps of *statcharClass* are:

- Determination of the number of classes from the input width of the classes nbClass;
- Determination of the wave numbers corresponding to the centre of the classes wnSclass (subroutine *dichotomd*);
- Computation of the statistical characteristics per class: mean, quadratic mean, variance, the standard deviation and root mean square;
- Computation of the percentiles for the percentages 90, 95 and 100 % (subroutine *inttab2*). See appendix 4, p294 for description of percentiles.

1.2.10.5. calcStat

1.2.10.5.1 Type

It is the program for the comparison of plt files with reference plt files for non regression tests. It is stored in the directory `srclib`.

1.2.10.5.2 Function

The program calcStat is dedicated to make the difference between a plot file and a reference plot file in radiance and temperature. A test is done on the maximum relative differences for the radiance and temperature values: the test is passed if the both differences are lower than a threshold and failed if the differences are higher than the threshold.

1.2.10.5.3 Subordinates

The program calcStat uses the modules *error* and *model*.

1.2.10.5.4 Dependencies

It depends on the existence of a reference plot file and a plot file.

1.2.10.5.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
f1	Plot file name	String (len=1024 max)	-	i		
f2	Reference plot file name	String (len=1024 max)	-	i		
threshold	Relative difference permitted between the two files	d	-	i		
Output						
n	Number of data entries in plot files	int	-	o		
Statistics on radiance						
n-nDiffW	Number of same data	int	-	o		
meanW	Mean relative difference	d	-	o		
diffWMax	Max relative difference	d	-	o		
Statistics on temperature						
n-nDiffT	Number of same data	int	-	o		
meanT	Mean relative difference	d	-	o		
diffTMax	Max relative difference	d	-	o		

1.2.10.5.6 Processing

The main functions of calcStat are:

- Read the plot file;
- Read the reference plot file;
- Compute the differences of the radiance and the temperature values.

1.2.10.6. calcStatJ

1.2.10.6.1 Type

It is the program for the comparison of Jacobian plt files with reference plt files for non regression tests. It is stored in the directory `srclib`.

1.2.10.6.2 Function

The program calcStatJ is dedicated to make the difference between a Jacobian plot file and a Jacobian reference plot file. A test is done on the maximum relative differences values: the test is passed if the difference is lower than a threshold and failed if the difference is higher than the threshold.

1.2.10.6.3 Subordinates

The program calcStat uses the modules *error* and *model*.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.10.6.4 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
f1	Plot file name	String (len=1024 max)	-	i		
f2	Reference plot file name	String (len=1024 max)	-	i		
threshold	Relative difference permitted between the two files	d	-	i		
nGBloc	Number of profiles	int	-	i		
nPBloc	Number of molecules	int	-	i		
nLevel	Number of levels	int	-	i		
Output						
n	Number of data entries in plot files	int	-	o		
n-nDiff	Number of same data	int	-	o		
Mean	Mean relative difference	d	-	o		
diffMax	Max relative difference	d	-	o		

1.2.10.6.5 Processing

The main functions of calcStatJ are:

- Read the plot file
- Read the reference plot file
- Compute the differences of the jacobian values

1.2.11. Constant and common modules

These modules are stored in the directory `srclib`.

1.2.11.1. model

This module contains constants specific to the model 4A.

1.2.11.1.1 Type

model is a module used by the main program `aaaa.f90` and numerous subroutines.

1.2.11.1.2 Function

The module *model* gathers non-physical constants (i.e., array dimensions) used by 4A/OP.

1.2.11.1.3 Subordinates

The module *model* uses the module precision.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.11.1.4 Dependencies

It depends on the existence of the module precision.

1.2.11.1.5 New constants introduced with implementation of aerosol extinction

Symbol	Descriptive Name	Type	Units	Source / Destinat°	Value
Cst_npttabexp_d	Number of exponential tabulation points for use in SOS	i	dl.	aaaa pre_sos diffnd	192 002
Cst_nlaeromax	Maximum number of aerosol layers	i	dl.	aaaa	3
Cst_ksmax	Maximum number wave numbers for which aerosol properties are provided	i	dl.	aaaa readaerosols	150

1.2.11.2. constants

This module contains constants for classic physics formula and tools.

1.2.11.2.1 Type

constants is a module used by the main program `aaaa.f90` and numerous subroutines.

1.2.11.2.2 Function

The module *constants* gathers physical constants and reference values used by 4A/OP.

1.2.11.2.3 Subordinates

The module *constants* uses the module precision.

1.2.11.2.4 Dependencies

It depends on the existence of the module *precision*.

1.2.11.2.5 Constants introduced with implementation of aerosol extinction

Symbol	Descriptive Name	Type	Units	Source / Destinat°	Value
WN_REF	Reference wave number for aerosol optical thickness specification	d	cm^{-1}	readaerosols	1000.0d

1.2.11.3. instruments

This module contains the instrument identifiers required for the convolution computations.

1.2.11.4. disort_runparams

1.2.11.4.1 Type

disort_runparams is a module used by the subroutine `pre_disort.f90`, the interface between 4A/OP and DISORT.

1.2.11.4.2 Function

The module *disort_runparams* gathers fixed DISORT run parameters.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.11.4.3 Subordinates

The module *disort_runparams* uses the module *precision*.

1.2.11.4.4 Dependencies

It depends on the existence of the module *precision*.

1.2.11.4.5 Constants

All constants are declared as parameters, in analogy to the modules *model* and *constants*.

Symbol	Descriptive Name	Type	Units	Source / Destinat°	Value
Cst_deltawvn	Width of the spectral interval for which the Planck function is calculated in DISORT	d	cm^{-1}	pre_disort disortlb	1.d0
Cst_usrtau	Radiant quantities are to be returned at user specified optical depths	logical	-	pre_disort disortlb	.TRUE.
Cst_ntau	Number of user specified optical depths	i	dl.	pre_disort disortlb	1
Cst_utau	User specified optical depths	d[Cst_ntau]	dl.	pre_disort disortlb	0.d0
Cst_usrang	Radiant quantities are to be returned at user specified polar angles	logical	-	pre_disort disortlb	.TRUE.
Cst_numu	Number of user specified polar angles	i	dl.	pre_disort disortlb	1
Cst_nphi	Number of user specified azimuth angles at which to return intensity	i	dl.	pre_disort disortlb	1
Cst_phi	User specified azimuth angles	d[Cst_nphi]	deg	pre_disort disortlb	0.d0
Cst_ibcnd	Index for boundary conditions	i	-	pre_disort disortlb	0
Cst_phi0	Azimuth angle for the incident beam	d	deg	pre_disort disortlb	0.d0
Cst_fisot	Intensity of top boundary isotropic illumination	d	$Wm^{-2}sr^{-1}cm$	pre_disort disortlb	0.d0
Cst_lamber	Lambertian surface reflection index	logical	-	pre_disort disortlb	.TRUE.
Cst_planckind	Thermal emission index	logical	-	pre_disort disortlb	.TRUE.
Cst_onlyfl	Flux-only index	logical	-	pre_disort disortlb	.FALSE.

1.2.11.5. sos_runparams

1.2.11.5.1 Type

sos_runparams is a module used by the subroutine pre_sos.f90, the interface between 4A/OP and SOS.

1.2.11.5.2 Function

The module *sos_runparams* gathers fixed SOS run parameters.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.11.5.3 Subordinates

The module *sos_runparams* uses the module *precision*.

1.2.11.5.4 Dependencies

It depends on the existence of the module *precision*.

1.2.11.5.5 Constants

All constants are declared as parameters, in analogy to the modules *model* and *constants*.

Symbol	Descriptive Name	Type	Units	Source / Destinat°	Value
Cst_corecgeo	SOS geometry correction identifier (internal in *diffnd*)	logical	-	pre_sos diffnd	.TRUE.
Cst_aotdiff	SOS maximum aerosol layer depth (scattering component of optical depth)	r	dl.	pre_sos	0.26

1.2.11.6. error

1.2.11.6.1 Err_notifyError

1.2.11.6.1.1 Type

Err_notifyError is a common subroutine that can be used by any programs and sub-programs.
Err_notifyError is referenced in the library `aaaa.a` and is defined within the module `error` (`error.f90`).

1.2.11.6.1.2 Function

The subroutine *Err_notifyError* has been made to notify an error with a message and the name of the corresponding subroutine.

1.2.11.6.1.3 Subordinates

None.

1.2.11.6.1.4 Dependencies

None.

1.2.11.6.1.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
theErrorMessageStr	The string that contains the error message	string	-	i		Only the 256 first characters are used
theFunctionStr	The string that contains the name of the routine where the error occurs	string	-	i		Only the 256 first characters are used

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.11.6.2 Err_outputErrorMessage

1.2.11.6.2.1 Type

Err_outputErrorMessage is a common subroutine that can be used by any programs and sub-programs. *Err_outputErrorMessage* is referenced in the library `aaaa.a` and is defined within the module `error` (`error.f90`).

1.2.11.6.2.2 Function

The subroutine *Err_outputErrorMessage* has been made to output, to a given logical unit number, lines of text that contain the name of the subprogram, the error message and the error position.

1.2.11.6.2.3 Subordinates

None.

1.2.11.6.2.4 Dependencies

None.

1.2.11.6.2.5 Interfaces

Symbol	Descriptive Name	Type	Units	I/O	Source / Destinat°	References / Remarks
Input						
theLun	The logical unit number for the output message	int	-	i		

1.2.11.7. spectumio

1.2.11.7.1 Type

The module *spectrumio* contains spectrum in/out routines: *readconvdata* (see section 1.2.9.3) and *readconvsp* (see section 1.2.10.2).

1.2.11.7.2 Function

The module *spectrumio* is dedicated to read convolved spectra with the both sub-routines: *readconvdata* (see section 1.2.9.3) and *readconvsp* (see section 1.2.10.2).

1.2.11.7.3 Subordinates

The module *spectrumio* uses the modules *precision*, *error*, *constants* and *model*.

1.2.11.7.4 Dependencies

None.

1.2.11.7.5 Interfaces

None.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.11.8. convolution

1.2.11.8.1 Type

The module *convolution* contains sub-routines that perform partial convolution.

1.2.11.8.2 Function

The module *convolution* contains the sub-routines: *cnvspe* (see section 1.2.1.24) and *cnvspehole* (see section 1.2.1.25).

1.2.11.8.3 Subordinates

The module convolution uses the module *precision*.

1.2.11.8.4 Dependencies

None.

1.2.11.8.5 Interfaces

None.

1.2.11.9. modatlas

1.2.11.9.1 Type

The module *modatlas* contains sub-routines dedicated to read and write in the Atlas.

1.2.11.9.2 Function

The module *modatlas* contains the subroutines: *readatlas_header* (see section 1.2.1.13), *readatlas* (see section 1.2.1.14), *writeatlas_header* and *writeatlas*.

1.2.11.9.3 Subordinates

The module convolution uses the modules *precision*, *model* and *modinouttools*.

1.2.11.9.4 Dependencies

None.

1.2.11.9.5 Interfaces

None.

1.2.11.10. modinouttools

1.2.11.10.1 Type

The module *modinouttools* is used by *modatlas*. The module contains the function that make the link between fortran and C.

1.2.11.10.2 Function

The module *modinouttools* is used to read the Atlas with the C program *inouttools* (see section 1.2.1.15).

1.2.11.10.3 Subordinates

The module convolution uses the modules *precision*.

1.2.11.10.4 Dependencies

None.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.11.10.5 Interfaces

None.

1.2.11.11. spstats

1.2.11.11.1 Type

The module *spstats* contains subroutines for the statanalys program. The subroutines are *statchar* (see section 1.2.10.3) and *statcharClass* (see section 1.2.10.4).

1.2.11.11.2 Function

None.

1.2.11.11.3 Subordinates

The module *spstats* uses the modules *error* and *precision*.

1.2.11.11.4 Dependencies

None.

1.2.11.11.5 Interfaces

None.

1.2.11.12. precision

1.2.11.12.1 Type

The module *precision* is for data type precision definitions.

1.2.11.12.2 Function

The module *precision* gives the precision for the type integer (short and long), for real (simple and double) and for complex (simple and double).

1.2.11.12.3 Subordinates

None.

1.2.11.12.4 Dependencies

None.

1.2.11.12.5 Interfaces

None.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

2. Software maintenance

2.1. Version control

A revision control has been set up by the way of the control management tool CVS (Concurrent Version System) in order to administer modifications of source files. Non-regression tests are also available and are presented in appendix (section Part III 3).

A nomenclature is proposed to name the different versions. The name of the current software is 4A2000. It is considered that is the version 0.0: the current software version is named 4A2000 v0.0 (written later: *vmodel v I.r*). Conditions for incrementing *vmodel*, *I* (Issue) and *r* (review) are suggested below:

- *r*: minor changes (additional comments for example, bug fix): *4A2000 v0.1 (for example);*
- *I*: changes in the implementation with no major modification (error management addition for example, translation in Fortran90): *4A2000 v1.0;*
- *vmodel*: major modifications (improvement of the physics): *4A2002 v0.0.*

The graphical user interface and the web site must be follow this nomenclature such as:

- GUI: GUI *vmodel-I* vG.g for the 4A version *vmodel vI.r* and the GUI version vG.g;
- WEB: WEB *vmodel-I* vW.w for the 4A version *vmodel vI.r* and the WEB version vW.w.

The graphical interface and the web site must have their own version (vG.g and vW.w respectively) as well as the 4A version. They will be attached to a version of 4A (*vmodel-I*).

2.2. Error control

A component to manage errors has been set up: the module `error.f90` (section Part II1.2.11.6). It notifies an error with a message and the name of the corresponding sub-routine. Appendix (Part III 6) presents the glossary builder that produces a list of controlled errors that can occur during a 4A run.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

3. Model physics

In preparation by LMD (see [Ref. 3], [Ref. 1], [Ref. 2], [Ref. 4], [Ref. 5], [Ref. 11]).

Although most of the underlying equations or assumptions have been already widely described in the refereed literature starting in 1974, the following parts will summarize and disseminate the status of the 4A model at LMD and its operational version 4Aop in order to unambiguously present how this model performs in simulating radiances as well as transmittances and jacobians under various conditions of observations (radiometric or interferometric experiments).

1. Radiative Transfer Equation;
2. Spectroscopic Parameters;
3. Absorbers/Istopomers/Line rejections;
4. Line shapes (collision, Doppler, Voigt);
5. Line mixing;
6. Chi-factors;
7. Line wings exploration and cut-offs;
8. Optimum adaptative representation steps and integration steps;
9. Interpolation between representation and interpolation grids;
10. Atmospheric Layering;
11. Parameterization of continua (O_2, N_2, H_2O);
12. Heavy molecules (CFC's);
13. Temperature dependence of some minor gas cross sections;
14. Convolution;
15. Validation.

Physics improvements of the software are described in Appendix A (Part III 1).

3.1. Radiative transfer equation

3.1.1. General approach

The optical thickness for a temperature profile *T(j),* for a concentration profile *ρ(c,j)* and for an incidence angle *θ* are obtained by linear interpolation in the pre-calculated optical thickness atlas with respect to temperature and by multiplication with the ratio $\rho(c,j).\sec(\theta) / \rho_0(c,j).\sec(\theta_0)$.

Optical thickness in each vertical layer *j* for selected molecules *c* $\Delta\sigma(v,P(j),T_k(j),\rho_0(c,j))$ and $\Delta\sigma(v,P(j),T_{k+1}(j),\rho_0(c,j)$ are extracted such that $T_k(j) < T(j) < T_{k+1}(j)$.

The resulting optical thickness for T(j) and ρ(c,j) is:

$$\Delta\sigma(v,T(j),\rho(c,j)) = \left\{ \begin{array}{l} \Delta\sigma(v,T_k(j),\rho_0(c,j)) + \\ \left[\Delta\sigma(v,T_{k+1}(j),\rho_0(c,j)) - \Delta\sigma(v,T_k(j),\rho_0(c,j))\right] \cdot \left(\dfrac{T(j) - T_k(j)}{T_{k+1}(j) - T_k(j)}\right) \end{array} \right\} \cdot \left(\dfrac{\rho(c,j).\sec(\theta)}{\rho_0(c,j).\sec(\theta_0)}\right) \qquad (8.)$$

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

and the total optical thickness of layer *j*, and the ensemble of selected molecules *c*:

$$\Delta\sigma(\nu, j) = \sum_{c} \Delta\sigma(\nu, T(j), \rho(\nu, j)) \tag{9.}$$

with transmittance:

$$\tau(\nu, j) = \prod_{i=j_{obs}}^{i=j} e^{-\Delta\sigma(\nu, i)} \tag{10.}$$

The integrated radiative transfer equation can be written as:

$$I(\nu) = \int_{\tau_{S+1}}^{\tau_s} B(\nu_s) \cdot d\tau + \int_{\tau_s}^{(1-\varepsilon)\tau_{\lim}} B(\nu) \cdot d\tau + \int_{(1-\varepsilon)\tau_{\lim}}^{\tau_{\lim}} B(\nu_{\lim}) \cdot d\tau + \int_{\tau_{\lim}}^{\tau_{obs}} B(\nu) \cdot d\tau$$

with $\tau_{obs} = 1$ and $\tau_{s+1} = 0$.

This formulation defines two pseudo-layers, the source layer *s* (e.g., top-of-atmosphere) and the reflexion/emission layer *lim* (e.g., Earth surface). This report dealing with the implementation of the solar radiance component, we consider in the following the reflexion/emission layer as the Earth surface and the source layer as the top of atmosphere, i.e. with layer index *s=2lim*.

The optical path is separated in two parts, the path between observer (above layer *obs, obs=1* for TOA observations) and the observed surface, characterised by the observation zenith angle θ_1, and the secondary path between the source and the observed surface, characterised by the reflexion zenith angle θ_2. For spaceborne Earth observations in the infrared spectral domain, θ_2 should be set to a value of the order of 55 degrees so that it represents at best the opposite atmospheric radiation at Earth surface. If the solar contribution is considered, the angle θ_2 is not suitable to describe the position of the radiation source, i.e. the sun. The solar zenith angle θ_s must be introduced as third angle that describes the solar downward path.

It is emphasised that optical thickness and transmittance of pseudo layers *s* and *lim* are given by the emissivity of the source and the surface through

$$\Delta\sigma = -\log(1-\varepsilon) \tag{11.}$$

with ε the emissivity of the source and of the surface, respectively.

3.1.1.1. Discrete equations for the radiative transfer (terrestrial)

The above formulation leads to a simple formulation of the discrete radiative transfer equation:

$$I(\nu) = \sum_{j=obs}^{2\lim} B(\nu, T_j) \cdot (\tau_j(\nu) - \tau_{j+1}(\nu)) \tag{12.}$$

with

$$\tau_j(\nu) - \tau_{j+1}(\nu) = \tau_j(\nu) \cdot (1 - e^{-\Delta\sigma_j(\nu)}) \tag{13.}$$

j layer index

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

τ_j transmittance between observer and layer j (exclusive)

$\Delta\sigma_j$ optical thickness of layer j

3.1.1.2. Discrete equations for Jacobians (terrestrial)

Jacobians are computed under the hypothesis of a linearised radiative transfer equation. With respect to atmospheric temperature and trace gas mixing ratio in a given vertical level one obtains:

$$I(\nu) = I_0(\nu) + \int_{Plim}^{Pobs} \frac{\partial I(\nu)}{\partial T(p)} \cdot dT(p) + \sum_c \left[\int_{Plim}^{Pobs} \frac{\partial I(\nu)}{\partial \rho_c(p)} \cdot d\rho_c(p) \right] \tag{14.}$$

or in the discrete formulation:

$$I(\nu) = I_0(\nu) + \sum_{j=obs}^{2\lim} \frac{\partial I(\nu)}{\partial T_j} \cdot \Delta T_j + \sum_c \left[\sum_{j=obs}^{2\lim} \frac{\partial I(\nu)}{\partial \rho_{c,j}} \cdot \Delta\rho_{c,j} \right] \tag{15.}$$

with the Jacobian in temperature in a layer j:

$$\frac{\partial I(\nu)}{\partial T_j} = \left\{ \begin{array}{l} \frac{\partial B(\nu, T_j)}{\partial T_j} \cdot (\varepsilon_j(\nu) - \tau_{j+1}(\nu)) \\ \sum_c \left(\frac{\partial \Delta\sigma_c(\nu, T_j)}{\partial T_j} \right) \cdot \left[-B(\nu, T_j) \cdot \tau_{j+1}(\nu) + \sum_{k=j+1}^{2\lim} B(\nu, T_k) \cdot (\varepsilon_k(\nu) - \tau_{k+1}(\nu)) \right] \end{array} \right\} \tag{16.}$$

and the Jacobian in mixing ratio of a molecule c in layer j:

$$\frac{\partial I(\nu)}{\partial \rho_{c,j}} = \left\{ -\frac{\partial \Delta\sigma_c(\nu, T_j)}{\partial \rho_{c,j}} \cdot \left[-B(\nu, T_j) \cdot \tau_{j+1}(\nu) + \sum_{k=j+1}^{2\lim} B(\nu, T_k) \cdot (\varepsilon_k(\nu) - \tau_{k+1}(\nu)) \right] \right\} \tag{17.}$$

As consequence of the 4A/OP geometry, there are two contributions to the total Jacobian of a given vertical layer (except for pseudo-layers), one for each the primary and the secondary optical path. Both contributions are summed up at the end of the computation and only one Jacobian value per level and physical quantity is provided in the output.

The physical properties required for the Jacobian computation are implicitly available from the absorption atlas:

- The partial derivative of the layer optical thickness with respect to mixing ratio is obtained by dividing the actual layer optical thickness of the absorbing molecule by its mixing ratio;
- The partial derivative of the layer optical thickness with respect to temperature is available through finite differences because the optical thickness is computed for different temperatures at fixed pressure levels.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

3.1.1.3. Jacobians for surface temperature and emissivity (terrestrial)

Jacobians of surface properties, the surface temperature $T_{j=lim}$ and the spectral surface emissivity $\varepsilon(\nu)$ are easily obtained by derivation of equation (5.).

The surface temperature Jacobian can be written with equations (4.) and (6.) as:

$$\frac{\partial I(\nu)}{\partial T_{\lim}} = \frac{\partial B(\nu, T_{\lim})}{\partial T_{\lim}} \cdot \left(\tau_{\lim}(\nu) - \tau_{\lim+1}(\nu)\right) = \frac{\partial B(\nu, T_{\lim})}{\partial T_{\lim}} \cdot \tau_{\lim}(\nu) \cdot \varepsilon(\nu) \qquad (18.)$$

As expected, it is the product of the derivative of the emitted radiance of a black body at surface temperature T_{lim}, the surface emissivity and the atmospheric transmittance between observer and surface.

The Jacobian with respect to the spectral surface emissivity is:

$$\frac{\partial I(\nu)}{\partial \varepsilon(\nu)} = \left\{ -\frac{1}{1-\varepsilon(\nu)} \cdot \left[-B(\nu, T_{\lim}) \cdot \tau_{\lim+1}(\nu) + \sum_{k=\lim+1}^{2\lim} B(\nu, T_k) \cdot \left(\tau_k(\nu) - \tau_{k+1}(\nu)\right) \right] \right\} \qquad (19.)$$

The formulation of (12.) is practical in terms of computation, but feigns a dependence of the emissivity Jacobian upon surface emissivity itself. Obviously, this dependence does not exist, equation (12.) can be factorised with $\tau_{lim+1} = \tau_{lim}(1-\varepsilon(\nu))$.

On the other hand, the formulation of (12.) is not applicable in case of a black body ($\varepsilon(\nu)=1$) when all transmittances are zero while the first factor is infinite. In this special case, the formulation goes back to the layer optical thickness values:

$$\frac{\partial I(\nu)}{\partial \varepsilon(\nu)} = \left\{ -\tau_{\lim}(\nu) \cdot \left[-B(\nu, T_{\lim}) + \sum_{k=\lim+1}^{2\lim} B(\nu, T_k) \cdot \prod_{j=\lim+1}^{k} \exp(-\Delta\sigma_j) \cdot \left(1 - \exp(-\Delta\sigma_{k+1})\right) \right] \right\} \qquad (20.)$$

To avoid this lengthy computation (last term in (13.)), the transmittance differences of the secondary path (last term in (12.)) are approximated by building the limit for those of the primary path and the corresponding zenith angles. For atmospheric layers of the secondary path (lim+1 ≤ k ≤ 2lim-1) we obtain:

$$\lim(\varepsilon(\nu) \to 1) \frac{\tau_k(\nu) - \tau_{k+1}(\nu)}{1-\varepsilon(\nu)} = \tau_{\lim}(\nu)^{1+\sec\theta 2/\sec\theta 1} \left(\tau_{-k+2\lim+1}(\nu)^{-\sec\theta 2/\sec\theta 1} - \tau_{-k+2\lim}(\nu)^{-\sec\theta 2/\sec\theta 1}\right) \qquad (21.)$$

and for the source pseudo-layer (k=2lim) with emissivity ε_{top}:

$$\lim(\varepsilon(\nu) \to 1) \frac{\tau_k(\nu) - \tau_{k+1}(\nu)}{1-\varepsilon(\nu)} = \tau_{\lim}(\nu)^{1+\sec\theta 2/\sec\theta 1} \cdot \varepsilon_{top}(\nu) \qquad (22.)$$

The Jacobian in surface emissivity in case of a black body surface is then computed by:

$$\frac{\partial I(\nu)}{\partial \varepsilon(\nu)} = \tau_{\lim+1}(\nu) \cdot B(\nu, T_{\lim}) + \tau_{\lim}(\nu)^{1+\sec\theta 2/\sec\theta 1} \cdot \left[\varepsilon_{top}(\nu) + \sum_{k=\lim+1}^{2\lim-1} B(\nu, T_k) \cdot \left(\tau_{-k+2\lim+1}(\nu)^{-\sec\theta 2/\sec\theta 1} - \tau_{-k+2\lim}(\nu)^{-\sec\theta 2/\sec\theta 1}\right) \right] \qquad (23.)$$

However, this formulation is not applicable in case of an observation within the atmosphere (obs≠1, e.g. airborne measurements), because transmittances in layers *1* to *obs-1* (above the observer) are not calculated and parts of the transmittance differences in (16.) are not available. The 4A/OP surface emissivity Jacobian computation is not activated in these specific conditions.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Part III. Appendices

1. Appendix A.: Detailed description of revisions

1.1. Fortran 90 porting

The code initially developed in Fortran 77 has been ported in Fortran 90 by following the ANSI Fortran90 Standards (in order to avoid problems with different compilers). The initial single precision has been replaced by a double precision concerning the wave numbers only. The following table (Table 19) presents the run time evolution/comparison of the executable `aaaa` for different simulation conditions (on a UNIX Sun machine (E220R) at 450 MHz).

Test identification	Run time (s)		
	Initial Code F77	Single precision Code F90	Double Precision Code F90
Test1	2.55	2.37	3.07
Test2	2.65	2.44	3.19
Test3	7.71	8.00	9.69
Test4	8.78	8.53	11.29
Test5	23.07	21.12	25.43
Test6	95.91	91.11	120.72
Test7	183.41	185.69	192.72

Table 19: Run time evolution/comparison with the F90 porting of `aaaa`.

1.2. Spectral surface emissivity

4A2000 has been extended by a read and interpolation module of user-defined spectral emissivities (subroutine `lireemis`, section Part II 1.2.1.6). In its initial version, these properties were spectrally constant.

Possible input emissivities are constant surface emissivities and spectral surface emissivity functions.

The user is free to create new files with its own emissivity values.

The emissivity codes for the generic surface (no spectral variability) emissivities can be 050, 098, 100, etc that correspond to an emissivity of 0.5, 0.98, 1 respectively. The user can add other files with a constant emissivity value by following the same concept. The corresponding emissivity codes should be in the range 000 - 100.

Synthetic emissivities are used to represent spectral surface emissivity functions.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.2.1. Spectral surface emissivity

The surface type for the synthetic emissivities is derived from the IGBP classification. For each surface type classification, the corresponding infrared surface emissivity spectrum is deduced from Snyder *et al.* [Ref. 22] and presented in Table 20. Fourteen emissivity classes are defined, associated to the IGBP surface type classification. For each class the infrared surface emissivity spectra is given in the IASI spectral domain at a spectral resolution of 4 cm^{-1} (shown in Figure 27). The spectrum for water in Snyder classification refers to fresh water and is applied to ocean.

IGBP surface properties classification	Season	Snyder emissivity classes	EmissivityCode
Water (17), Wetlands (11)	all	Water	101
Snow, Ice (15)	all	Snow, Ice	102
Barren/Desert (16)	all	Arid bare soil	103
Crops (12), Tundra (18)	all	Organic bare soil	104
Open shrubs (7)	F,W	Senescent sparse shrubs	105
Open shrubs (7)	Sp,Su	Green sparse shrubs	106
Savannah (9) ; Closed Shrubs, Grasslands, Crop/Mosaic (6,10,14)	W,Sp F,W	Senescent grass savannah	107
Savannah (9) ; Closed Shrubs, Grasslands, Crop/Mosaic (6,10,14)	Su,F Sp,Su	Green Grass Savannah	108
Woody Savannah (8)	F,W	Senescent Woody Savannah	109
Woody Savannah (8)	Sp,Su	Green Woody Savannah	110
Deciduous broadleaf and Mixed forest (4,5)	F,W	Senescent Broadleaf Forest	111
Deciduous broadleaf and Mixed forest (4,5) ; Evergreen Broadleaf Forest (2)	Sp,Su all	Green Broadleaf Forest	112
Deciduous Needle Forest (3)	F,W	Senescent Needle Forest	113
Deciduous Needle Forest (3) ; Evergreen Needle Forest (1)	Sp,Su all	Green Needle Forest	114

Table 20: Emissivity class definition (adapted from Snyder et al., 1998).
Season notification: W – winter, Sp – spring, Su – summer, F – fall

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

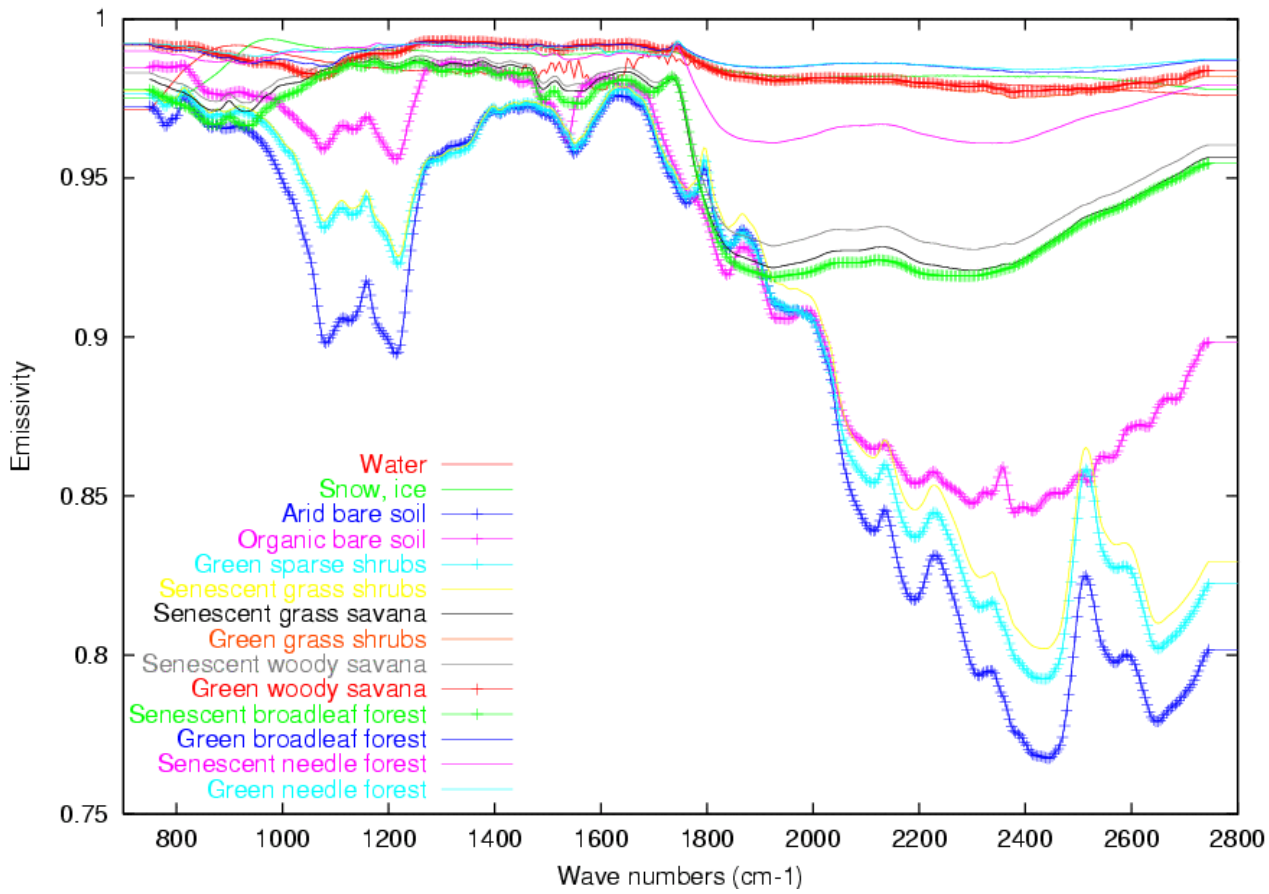


Figure 27: Spectral emissivity of 14 surface types.

1.3. Spherical atmosphere approximation

The initial version of 4A2000 considered plan-parallel atmospheric layers. The path between observer and emitter is user-defined, either by one angle, the primary path or observation zenith (nadir) angle for up-welling (down-welling) radiance simulation, or by two angles, the primary path (incidence) and secondary path (reflection) angles at the reflecting surface (i.e., the Earth surface or cloud top for up-welling radiance simulation, or the cloud base for down-welling radiance simulation). The plan-parallel atmosphere assumption has been replaced by a spherical atmosphere approximation (subroutine `calcsphtraj`, section Part II 0). Figure 28 illustrates this correction. This has been obtained by a simple vectorisation of the incidence and observation angles which are now calculated for each layer by a simple trigonometric relation. As function of altitude z, the zenith angle is given by

$$\theta(z) = \arcsin [R/(R+z) \sin\theta_0] \quad (1)$$

where R is the Earth radius and θ_0 the zenith angle at ground (z=0). In equation (1), the relation between atmospheric pressure p and altitude is approximated by

$$z(p) = H \ln(p_0/p) \quad (2)$$

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

with surface pressure p_0 and atmospheric scale height H=8km. The optical thickness δ of a layer with depth Δp and mean pressure p_m is now proportional to Δp / cos[$\theta(z(p_m))$] instead of Δp / cos[$\theta(z=0)$]. It is emphasized that the height approximation (2) does not affect the computation of Δp.

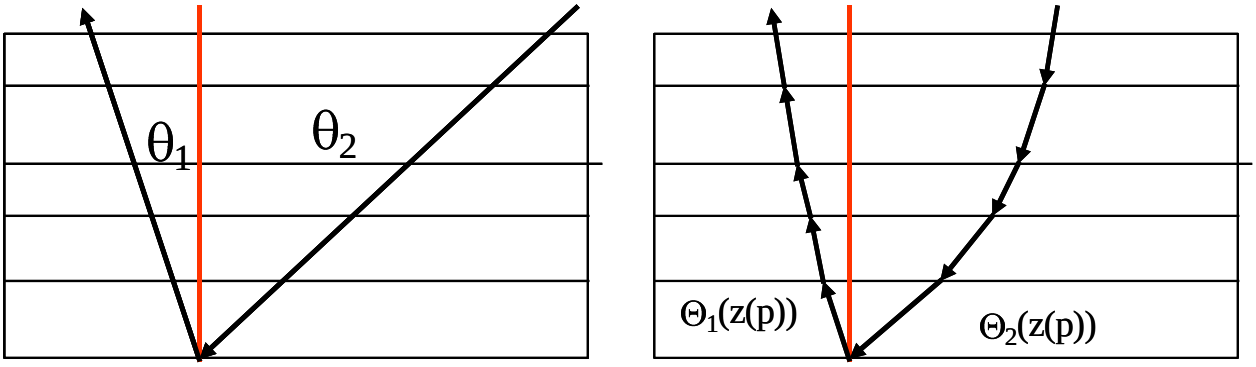


Figure 28: Correction of the plan-parallel atmosphere assumption by a spherical atmosphere approximation

Table 21 gives examples for the altitude dependence of the air mass correction for selected zenith angles. Note that a plane parallel model neglects this dependence by assuming the 1/cosθ values at ground for any altitude.

z [km] $\theta(z=0)$	0	6.53	17.15	40.76	79.33
85°	11.47	10.19	8.80	7.04	5.60
75°	3.86	3.81	3.73	3.56	3.34
45°	1.414	1.413	1.410	1.405	1.397
0°	1.000	1.000	1.000	1.000	1.000

Table 21: Values of 1/cosθ as a function of altitude z for given zenith angles at ground.

The impact of the implementation of a spherical atmosphere is illustrated in Figure 29. At an observation zenith angle 80°, differences exceed 1K in certain spectral intervals, whereas for observation zenith angles below 60°, differences are limited to within 0.1K in NedT at 280K. As expected, the strongest impact is observed in spectral domains that are sensitive to higher atmospheric levels (e.g. CO_2 bands below 800 cm^{-1} and at 2400 cm^{-1}; ozone band at 1100 cm^{-1}), where the air mass correction with respect to the plan-parallel model is more important than in lower levels.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

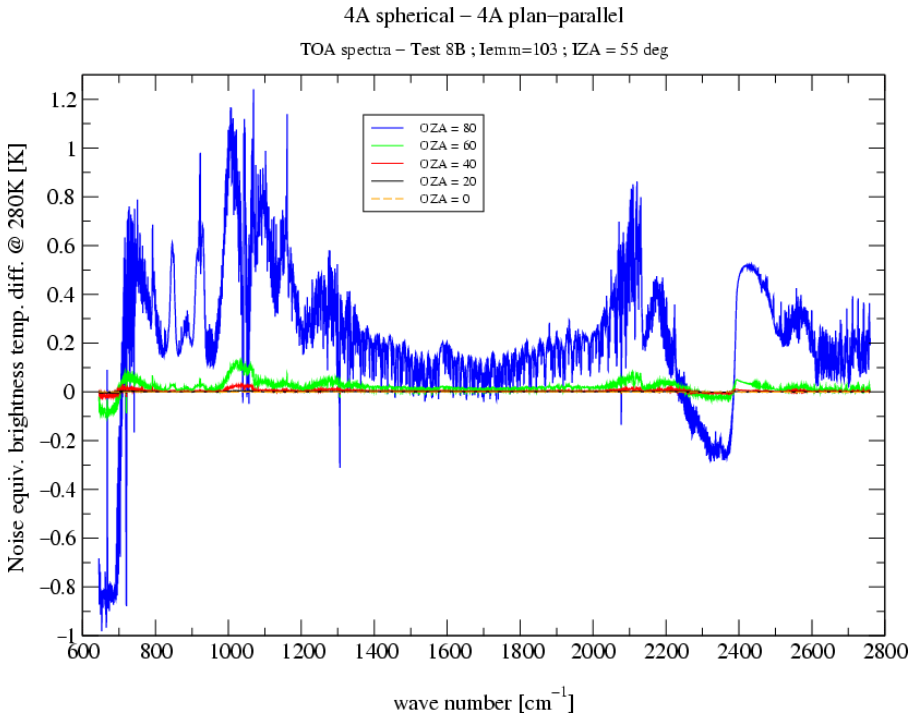


Figure 29: IASI night-time TOA spectra difference in NedT at 280K between a spherical and plan-parallel atmosphere for observation zenith angles θ_1 0°, 20°, 40°, 60°, 80° (OZA on the graph). The spectra are simulated for a temperate atmosphere (surface temperature 296K), surface emissivity 'Arid bare soil' and an incidence zenith angle θ_2 of 55° (IZA).

1.4. Hole convolution

1.4.1. Introduction

This section describes the algorithm used for the hole convolution, i.e. the convolution between a function and a signal with missing records that are replaced by interpolated values. It has been developed to reduce computing time of the convolution in the 4A model software in the case of spectra with holes.

This option speeds up convolution computing when jacobian calculation is selected. It slightly slows it when convolution is done with the spectrum only. Note that results computed with this option are physically the same as those computed without it, even if there is a numerical difference between them.

1.4.2. Algorithm description

Figure 30 and Figure 31 illustrate the notation used in the algorithm. Given a filter f defined for every j in $[b,e]$. Given a signal s defined on the increasing index set $\mathbf{I} = \{I_i,\ i \in [1,N]\}$ with $[b,e] \subseteq [I_1, I_N]$ (see Figure 30). We want to estimate the following sum of products:

$$C_b^e = \sum_{j=b}^{e} s_j f_j. \tag{1}$$

We approximate s_j for j by a linear interpolation:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

$$I_i \leq j < I_{i+1}, \quad s_j = \frac{j - I_i}{\Delta_i}(s_{I_{i+1}} - s_{I_i}) + s_{I_i},$$

with $\Delta_i = I_{i+1} - I_i$. By grouping the terms of the sum, we can estimate C_b^e directly from the values s_{I_i} as

$$C_b^e = \sum_{i=1}^{N} p_i s_{I_i}, \tag{2}$$

where p_i is the weight for s_{I_i} in the computation of C_b^e. Thus, the computation needs only N products and $N-1$ sums instead of $b-e+1$ products and $b-e$ sums. In order to estimate p_i, let us rewrite (1) as the sum of the product sums on each interval $[I_i, I_{i+1}-1]$ as

$$C_b^e = C_b^{I_2-1} + C_{I_2}^{I_3-1} + \cdots + C_{I_{i-1}}^{I_i-1} + C_{I_i}^{I_{i+1}-1} + \cdots + C_{I_{N-1}}^{e}.$$

Let us note $C_i = C_{I_i}^{I_{i+1}-1}$. The two sums

$$C_{i-1} = \sum_{j=I_{i-1}}^{I_i-1} \left(\frac{j - I_{i-1}}{\Delta_{i-1}}(s_{I_i} - s_{I_{i-1}}) + s_{I_{i-1}} \right) f_j \tag{3}$$

and

$$C_i = \sum_{j=I_i}^{I_{i+1}-1} \left(\frac{j - I_i}{\Delta_i}(s_{I_{i+1}} - s_{I_i}) + s_{I_i} \right) f_j \tag{4}$$

are the only ones that need the value of s_{I_i} and they can be written as

$$C_{i-1} = p_{i-1}^1 s_{I_{i-1}} + p_i^2 s_{I_i}$$

and

$$C_i = p_i^1 s_{I_i} + p_{i+1}^2 s_{I_{i+1}}.$$

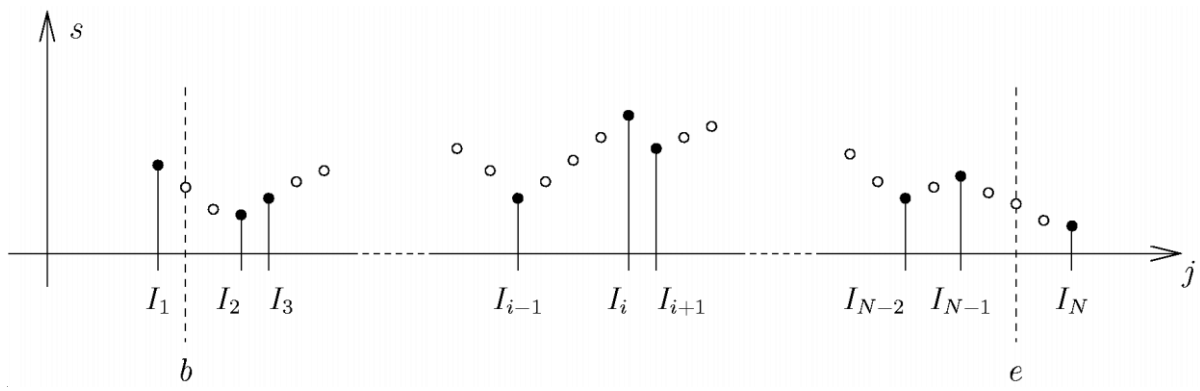


Figure 30: Example of signal s. Plain black circles represent records that define s; they correspond to indices I_i in I. Empty circles represent interpolated values on indices not in I. Note that in this example, sum limit indices b and e (equation (1)) are not in I.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

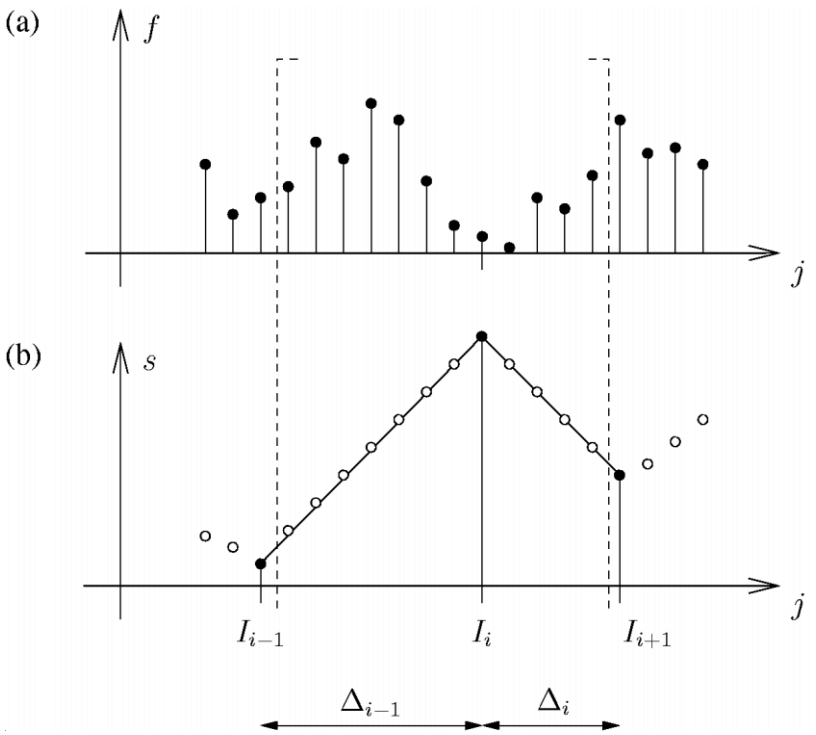


Figure 31: Example around I_i to illustrate the notations used in the text. Dashed brackets indicate the limits of indices used to compute the weight for s_{I_i}.

Thus the total weight for s_{I_i} is:

$$p_i = p_i^1 + p_i^2.$$

The equation (4) gives

$$p_i^1 = \sum_{j=I_i}^{I_{i+1}-1} \left(1 - \frac{j - I_i}{\Delta_i}\right) f_j \tag{5}$$

and the equation (3) gives

$$p_i^2 = \sum_{j=I_{i-1}}^{I_i-1} \frac{j - I_{i-1}}{\Delta_{i-1}} f_j.$$

So, by replacing $j - I_i$ by j, we can rewrite (5) as

$$p_i^1 = \sum_{j=0}^{\Delta_i-1} \left(1 - \frac{j}{\Delta_i}\right) f_{j+I_i}$$

and

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

$$p_i^1 = \sum_{j=0}^{\Delta_i - 1} f_{j+I_i} - \sum_{j=0}^{\Delta_i - 1} \frac{j}{\Delta_i} f_{j+I_i}.$$

Then

$$p_i^1 = \mathrm{sum}_1(i) - \frac{\mathrm{sum}_2(i)}{\Delta_i}$$

with

$$\mathrm{sum}_1(i) = \sum_{j=0}^{\Delta_i - 1} f_{j+I_i}$$

and

$$\mathrm{sum}_2(i) = \sum_{j=0}^{\Delta_i - 1} j f_{j+I_i}.$$

Using the same kind of development, we obtain then

$$p_i^2 = \frac{\mathrm{sum}_2(i-1)}{\Delta_{i-1}}.$$

The weights p_i are evaluated for each value of i in $[1, N]$. Then they are used in equation (2) to determine C_b^e. The computation of p_i is more complex than the computation of C_b^e directly with (1). Thus the computation time is longer. However, if a unique filter must be applied to several signals defined on exactly the same set $\mathbf{I}$, the computational cost for p_i is rapidly covered by the gain on each additional sum.

1.5. Solar contribution

1.5.1. Introduction

In the frame of developments of the 4A/OP release 4AOP-2003 version 1.0, the addition of the solar component to the 4A/OP radiative transfer simulation is investigated. Section 1.5.2 describes the approach and provides its mathematical description. Section 1.5.3 summarises the software modifications and their consequences for the user. Validation results are presented in section 1.5.4.

1.5.2. Mathematical description

1.5.2.1. Reminder of the general approach

The optical thickness for a temperature profile $T(j)$, for a concentration profile $\rho(c,j)$ and for an incidence angle θ are obtained by linear interpolation in the pre-calculated optical thickness atlas with respect to temperature and by multiplication with the ratio $\rho(c,j).\sec(\theta) / \rho_0(c,j).\sec(\theta_0)$.

Optical thickness in each vertical layer j for selected molecules c $\Delta\sigma(v,P(j),T_k(j),\rho_0(c,j))$ and $\Delta\sigma(v,P(j),T_{k+1}(j),\rho_0(c,j)$ are extracted such that $T_k(j) < T(j) < T_{k+1}(j)$.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

The resulting optical thickness for T(j) and ρ(c,j) is:

$$\Delta\sigma(c,T(j),\rho(c,j)) = \left\{ \begin{array}{l} \Delta\sigma(c,T_k(j),\rho_0(c,j)) + \\ \left[\begin{array}{l} \Delta\sigma(c,T_{k+1}(j),\rho_0(c,j)) \\ -\Delta\sigma(c,T_k(j),\rho_0(c,j)) \end{array} \right] \cdot \left(\frac{T(j) - T_k(j)}{T_{k+1}(j) - T_k(j)} \right) \end{array} \right\} \cdot \left(\frac{\rho(c,j) \cdot \sec(\theta)}{\rho_0(c,j) \cdot \sec(\theta_0)} \right) \qquad (1.)$$

and the total optical thickness of layer *j*, and the ensemble of selected molecules *c*:

$$\Delta\sigma(c,j) = \sum_c \Delta\sigma(c,T(j),\rho(c,j)) \qquad (2.)$$

with transmittance:

$$\tau(c,j) = \prod_{i=j_{obs}}^{i=j} e^{-\Delta\sigma(c,i)} \qquad (3.)$$

The integrated radiative transfer equation can be written as:

$$I(\tau) = \int_{\tau_{S+1}}^{\tau_s} B(\tau_s) \cdot d\tau + \int_{\tau_s}^{(1-\varepsilon)\tau_{\lim}} B(\tau) \cdot d\tau + \int_{(1-\varepsilon)\tau_{\lim}}^{\tau_{\lim}} B(\tau_{\lim}) \cdot d\tau + \int_{\tau_{\lim}}^{\tau_{obs}} B(\tau) \cdot d\tau$$

with $\tau_{obs} = 1$ and $\tau_{s+1} = 0$.

This formulation defines two pseudo-layers, the source layer *s* (e.g., top-of-atmosphere) and the reflexion/emission layer *lim* (e.g., Earth surface). This report dealing with the implementation of the solar radiance component, we consider in the following the reflexion/emission layer as the Earth surface and the source layer as the top of atmosphere, i.e. with layer index *s=2lim*.

The optical path is separated in two parts, the path between observer (above layer *obs, obs=1* for TOA observations) and the observed surface, characterised by the observation zenith angle θ_1, and the secondary path between the source and the observed surface, characterised by the reflexion zenith angle θ_2. For spaceborne Earth observations in the infrared spectral domain, θ_2 should be set to a value of the order of 55 degrees so that it represents at best the opposite atmospheric radiation at Earth surface. If the solar contribution is considered, the angle θ_2 is not suitable to describe the position of the radiation source, i.e. the sun. The solar zenith angle θ_s must be introduced as third angle that describes the solar downward path.

It is emphasised that optical thickness and transmittance of pseudo layers *s* and *lim* are given by the emissivity of the source and the surface through

$$\Delta\sigma = -\log(1-\varepsilon) \qquad (4.)$$

with ε the emissivity of the source and of the surface, respectively.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

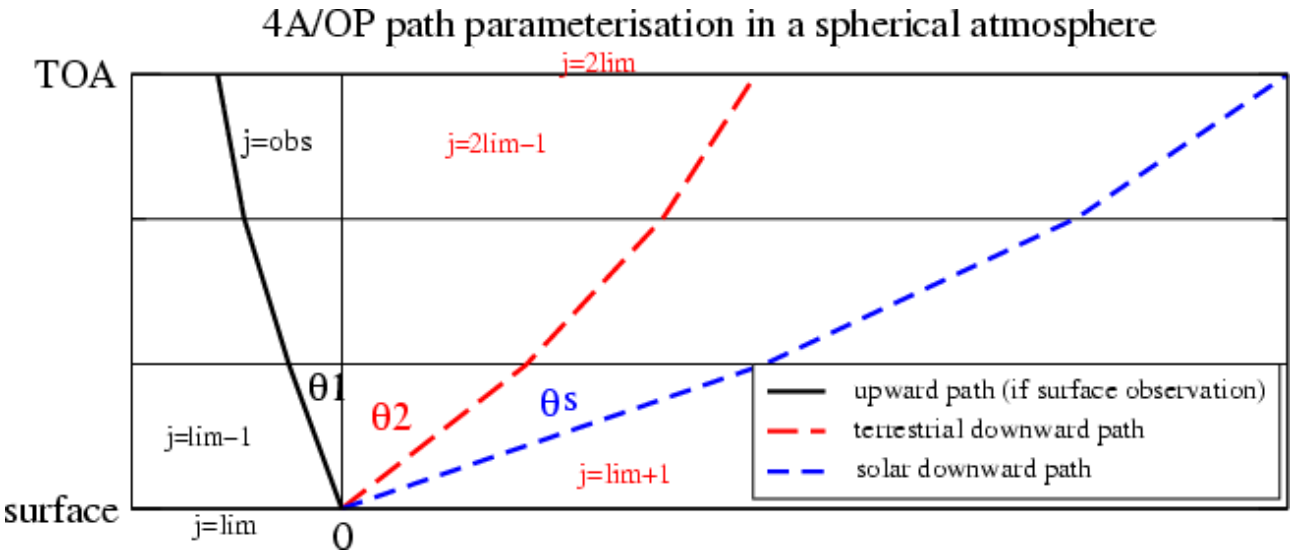


Figure 32: 4A/OP geometry in case of satellite observation of the surface '0' under zenith angle θ_1 with solar contribution at a solar zenith angle θ_s. Layer indices j are indicated for border layers. In case of TOA observations: obs=1, otherwise obs > 1 and not corresponding to layer 2lim – 1.

1.5.2.1.1 Discrete equations for the radiative transfer (terrestrial)

The above formulation leads to a simple formulation of the discrete radiative transfer equation:

$$I(\theta) = \sum_{j=obs}^{2\lim} B(\theta, T_j) \cdot (\tau_j(\theta) - \tau_{j+1}(\theta)) \tag{5.}$$

with

$$\tau_j(\theta) - \tau_{j+1}(\theta) = \tau_j(\theta) \cdot (1 - e^{-\Delta\sigma_j(\theta)}) \tag{6.}$$

j	layer index
τ_j	transmittance between observer and layer j (exclusive)
$\Delta\sigma_j$	optical thickness of layer j

1.5.2.1.2 Discrete equations for Jacobians (terrestrial)

Jacobians are computed under the hypothesis of a linearised radiative transfer equation. With respect to atmospheric temperature and trace gas mixing ratio in a given vertical level one obtains:

$$I(\theta) = I_0(\theta) + \int_{Plim}^{Pobs} \frac{\partial I(\theta)}{\partial T(p)} \cdot dT(p) + \sum_c \left[\int_{Plim}^{Pobs} \frac{\partial I(\theta)}{\partial \rho_c(p)} \cdot d\rho_c(p) \right] \tag{7.}$$

or in the discrete formulation:

$$I(\theta) = I_0(\theta) + \sum_{j=obs}^{2\lim} \frac{\partial I(\theta)}{\partial T_j} \cdot \Delta T_j + \sum_c \left[\sum_{j=obs}^{2\lim} \frac{\partial I(\theta)}{\partial \rho_{c,j}} \cdot \Delta\rho_{c,j} \right] \tag{8.}$$

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

with the Jacobian in temperature in a layer *j*:

$$\frac{\partial I(\nu)}{\partial T_j} = \begin{cases} \frac{\partial B(\nu,T_j)}{\partial T_j} \cdot \left(\varepsilon_j(\nu) - \tau_{j+1}(\nu)\right) \\ \sum_c \left(\frac{\partial \Delta\sigma_c(\nu,T_j)}{\partial T_j}\right) \cdot \left[-B(\nu,T_j) \cdot \tau_{j+1}(\nu) + \sum_{k=j+1}^{2\lim} B(\nu,T_k) \cdot \left(\varepsilon_k(\nu) - \tau_{k+1}(\nu)\right)\right] \end{cases} \quad (9.)$$

and the Jacobian in mixing ratio of a molecule *c* in layer *j*:

$$\frac{\partial I(\nu)}{\partial \rho_{c,j}} = \left\{ -\frac{\partial \Delta\sigma_c(\nu,T_j)}{\partial \rho_{c,j}} \cdot \left[-B(\nu,T_j) \cdot \tau_{j+1}(\nu) + \sum_{k=j+1}^{2\lim} B(\nu,T_k) \cdot \left(\varepsilon_k(\nu) - \tau_{k+1}(\nu)\right)\right]\right\} \quad (10.)$$

As consequence of the 4A/OP geometry (c.f. Figure 32), there are two contributions to the total Jacobian of a given vertical layer (except for pseudo-layers), one for each the primary and the secondary optical path. Both contributions are summed up at the end of the computation and only one Jacobian value per level and physical quantity is provided in the output.

The physical properties required for the Jacobian computation are implicitly available from the absorption atlas:

- The partial derivative of the layer optical thickness with respect to mixing ratio is obtained by dividing the actual layer optical thickness of the absorbing molecule by its mixing ratio;
- The partial derivative of the layer optical thickness with respect to temperature is available through finite differences because the optical thickness is computed for different temperatures at fixed pressure levels.

1.5.2.1.3 Jacobians for surface temperature and emissivity (terrestrial)

Jacobians of surface properties, the surface temperature $T_{j=lim}$ and the spectral surface emissivity $\varepsilon(\nu)$ are easily obtained by derivation of equation (5.).

The surface temperature Jacobian can be written with equations (4.) and (6.) as:

$$\frac{\partial I(\nu)}{\partial T_{\lim}} = \frac{\partial B(\nu,T_{\lim})}{\partial T_{\lim}} \cdot \left(\varepsilon_{\lim}(\nu) - \tau_{\lim+1}(\nu)\right) = \frac{\partial B(\nu,T_{\lim})}{\partial T_{\lim}} \cdot \tau_{\lim}(\nu) \cdot \varepsilon(\nu) \quad (11.)$$

As expected, it is the product of the derivative of the emitted radiance of a black body at surface temperature T_{lim}, the surface emissivity and the atmospheric transmittance between observer and surface.

The Jacobian with respect to the spectral surface emissivity is:

$$\frac{\partial I(\nu)}{\partial \varepsilon(\nu)} = \left\{ -\frac{1}{1-\varepsilon(\nu)} \cdot \left[-B(\nu,T_{\lim}) \cdot \tau_{\lim+1}(\nu) + \sum_{k=\lim+1}^{2\lim} B(\nu,T_k) \cdot \left(\varepsilon_k(\nu) - \tau_{k+1}(\nu)\right)\right]\right\} \quad (12.)$$

The formulation of (12.) is practical in terms of computation, but feigns a dependence of the emissivity Jacobian upon surface emissivity itself. Obviously, this dependence does not exist, equation (12.) can be factorised with $\tau_{lim+1} = \tau_{lim}(1-\varepsilon(\nu))$.

On the other hand, the formulation of (12.) is not applicable in case of a black body ($\varepsilon(\nu)=1$) when all transmittances are zero while the first factor is infinite. In this special case, the formulation goes back to the layer optical thickness values:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

$$\frac{\partial I(\nu)}{\partial \varepsilon(\nu)} = \left\{ -\tau_{\lim}(\nu) \cdot \left[-B(\nu, T_{\lim}) + \sum_{k=\lim+1}^{2\lim} B(\nu, T_k) \cdot \prod_{j=\lim+1}^{k} \exp(-\Delta\sigma_j) \cdot (-\exp(-\Delta\sigma_{k+1})) \right] \right\} \quad (13.)$$

To avoid this lengthy computation (last term in (13.)), the transmittance differences of the secondary path (last term in (12.)) are approximated by building the limit for those of the primary path and the corresponding zenith angles. For atmospheric layers of the secondary path (lim+1 ≤ k ≤ 2lim-1) we obtain:

$$\lim(\varepsilon(\nu) \to 1) \frac{\tau_k(\nu) - \tau_{k+1}(\nu)}{1-\varepsilon(\nu)} = \tau_{\lim}(\nu)^{1+\sec\theta 2/\sec\theta 1} \left(\tau_{-k+2\lim+1}(\nu)^{-\sec\theta 2/\sec\theta 1} - \tau_{-k+2\lim}(\nu)^{-\sec\theta 2/\sec\theta 1} \right) \quad (14.)$$

and for the source pseudo-layer (k=2lim) with emissivity ε_{top}:

$$\lim(\varepsilon(\nu) \to 1) \frac{\tau_k(\nu) - \tau_{k+1}(\nu)}{1-\varepsilon(\nu)} = \tau_{\lim}(\nu)^{1+\sec\theta 2/\sec\theta 1} \cdot \varepsilon_{top}(\nu) \quad (15.)$$

The Jacobian in surface emissivity in case of a black body surface is then computed by:

$$\frac{\partial I(\nu)}{\partial \varepsilon(\nu)} = \tau_{\lim+1}(\nu) \cdot B(\nu, T_{\lim}) + \tau_{\lim}(\nu)^{1+\sec\theta 2/\sec\theta 1} \cdot \left[\varepsilon_{top}(\nu) + \sum_{k=\lim+1}^{2\lim-1} B(\nu, T_k) \cdot \left(\tau_{-k+2\lim+1}(\nu)^{-\sec\theta 2/\sec\theta 1} - \tau_{-k+2\lim}(\nu)^{-\sec\theta 2/\sec\theta 1} \right) \right] \quad (16.)$$

However, this formulation is not applicable in case of an observation within the atmosphere (obs≠1, e.g. airborne measurements), because transmittances in layers *1* to *obs-1* (above the observer) are not calculated and parts of the transmittance differences in (16.) are not available. The 4A/OP surface emissivity Jacobian computation is not activated in these specific conditions.

1.5.2.2. The solar component

Solar radiation is considered as an external radiation source at the top of the atmosphere that does not interfere with the terrestrial component along the optical path. The solar component is thus added to the terrestrial one (equations (5) and (9-12) for radiance and Jacobians, respectively). The formulation of the solar component of radiance and Jacobian (indicated by the index 's') accounts for the attenuation of the solar radiation along the optical path, i.e. the solar downward path characterised by the solar zenith angle θ_s and the primary path characterised by the observation zenith angle θ_1. It is then sufficient to compute the total transmittance along these paths and to describe the solar irradiance.

As a radiative transfer code dedicated to the infrared spectral domain, the capacity of 4A/OP to simulate the solar component of radiance and Jacobians is limited:

- No scattering is taken into account, which limits the applicability of 4A/OP towards smaller wavelengths. Solar radiation at the surface is considered as mono-directional and then normalised with respect to an isotropic radiance field from the hemisphere above surface;
- No bi-directional effects at surface are accounted for, since 4A/OP has no parameterisation of the azimuth angle. Surface reflection is considered as isotropic, which excludes the simulation of specular reflection effects. The surface albedo (or lambertian surface reflectance) $\alpha(\nu)$ is not specifically parameterised and depends only on the surface emissivity via Kirchhoff's law for a grey body: $\alpha(\nu) = 1 - \varepsilon(\nu)$ (17.)

For instance, the sun is considered as a black body at surface temperature T_{sun} = 6000 K.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.5.2.2.1 Solar radiance component

The solar radiance at TOA $B_{S,TOA}$ is given by

$$B_{S,TOA}(\nu) = B(\nu, T_{sun}) \cdot \left(r_{sun} / R_{E-S} \right)^2 \cdot \cos\theta_s \qquad (18.)$$

with r_{sun} the solar radius, R_{E-S} the Earth-sun distance and θ_s the solar zenith angle at the observed surface (or the observer's position in case of a ground based observation).

The solar TOA irradiance is attenuated in the atmosphere. The total transmittance τ_s along the solar optical path is obtained after adaptation of the optical thickness (equation 1.) to the solar zenith angle θ_s and by interpretation of the solar TOA irradiance as the product of solar emission and emissivity of the (TOA) source pseudo layer *j=2lim*:

$$\tau_S(\nu) = \tau(\nu, j = 2\lim) = \prod_{j=obs}^{2\lim-1} e^{-\Delta\sigma(\nu, j)} \cdot \left(r_{sun} / R_{E-S} \right)^2 \cdot \cos\theta_s \qquad (19.)$$

The solar radiance component

$$I_S(\nu) = B(\nu, T_{sun}) \cdot \tau_S(\nu) \qquad (20.)$$

is added to the terrestrial component (equation 5.).

It is recalled that the computation of the solar contribution to the total radiance $I(\nu) + I_s(\nu)$ does not require a transmittance calculation for each atmospheric layer, only the total transmittance is of interest.

1.5.2.2.2 Solar Jacobian component

As for the radiance, the solar component of Jacobians is only dependent upon the total transmittance $\tau_s(\nu)$ along the solar optical path. Derivation of equation (20.) yields for the temperature of an atmospheric layer *j*:

$$\frac{\partial I_S(\nu)}{\partial T_j} = -\sum_c \left(\frac{\partial \Delta\sigma_c(\nu, T_j)}{\partial T_j} \right) \cdot \left[B(\nu, T_{sun}) \cdot \tau_S(\nu) \right] , \qquad (21.)$$

for the mixing ratio of an atmospheric absorber *c* in layer *j*:

$$\frac{\partial I_S(\nu)}{\partial \rho_{c,j}} = -\frac{\partial \Delta\sigma_c(\nu, T_j)}{\partial \rho_{c,j}} \cdot B(\nu, T_{sun}) \cdot \tau_S(\nu) , \qquad (22.)$$

for the surface temperature:

$$\frac{\partial I_S(\nu)}{\partial T_{\lim}} = 0 , \qquad (23.)$$

and for the spectral surface emissivity:

$$\frac{\partial I_S(\nu)}{\partial \varepsilon(\nu)} = -\frac{1}{1-\varepsilon(\nu)} \cdot B(\nu, T_{sun}) \cdot \tau_S(\nu) . \qquad (24.)$$

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

As for the terrestrial component, the surface emissivity Jacobian does not depend on the surface emissivity itself. The approximation for surface emissivity one (i.e., no surface reflection at all) exploits the transmittance along the primary path between observer and surface $\tau(\nu,j=\lim)$ and adjusts this transmittance with respect to the solar zenith angle:

$$\frac{\partial I_S(\nu)}{\partial \varepsilon(\nu)} = -B(\nu, T_{sun}) \cdot \left(R_{sun} / R_{E-S}\right)^2 \cdot \cos\theta_s \cdot \tau_{\lim}(\nu)^{\left(1+\frac{\cos\theta_1}{\cos\theta_s}\right)} \qquad (25.)$$

1.5.3. Modifications

Modifications of the 4A/OP software package due to the insertion of the solar component to the radiance and Jacobian computation are limited to modifications in a few routines and minor changes in user controlled input files.

1.5.3.1. Modified software routines

Modified routines are:

- the principal programme `aaaa`:

  Management and allocation of additional variable and arrays related to the geometry of the secondary solar path and the transmittance along the solar path;

- the subroutine `para4a`:

  Reading of the additional user-defined parameters solar zenith angle and Earth-sun distance;

- the sub-routine `calcsphtraj`:

  Computation of the solar zenith angle per atmospheric layer;

- the subroutine `calcrad`:

  Computation of the solar irradiance at TOA, the atmospheric transmittance along the solar path, and of radiance and Jacobians as sum of the terrestrial and solar components as described in section 1.5.2.

1.5.3.2. Modified input/output files

The user controlled input file `para4A`[$1]`.dtp` contains two additional parameters, the solar zenith angle at surface level and the Earth-sun distance.

In practice, the user suppresses all computations with respect to the solar component by specifying a solar zenith angle outside the range [0°,90°[or a secant value of the solar zenith angle inferior to 1. Obviously, the computation of the solar component can only be activated in case of the observation configurations 'DOWN' or 'DOWNUP'. Since arrays dedicated to the solar component are always allocated, RAM and swap memory are slightly increased by one per thousand.

The format of output files has not been modified.

1.5.3.3. Run time evaluation

Run time tests are applied to an atmosphere at 40 levels. Spectra are calculated at the resolution of the absorption atlas and are then convolved with the IASI level 1c instrument spectral response function at 0.25 cm^{-1} spectral sampling in the spectral range from 1890 to 2760 cm^{-1}. If selected, Jacobians are calculated for H_2O, CO_2 and O_3. The test results are summarised in Table 22.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Apart from the computation of the solar component of radiance and Jacobians, the '4A/OP – solar' release also deviates from the currently distributed version '4A/OP 2003-1-0' by addition of the surface emissivity Jacobian computation in case of a black surface after equation (16). Therefore, tests are carried out for a black body and a desert surface with spectrally variable emissivity.

Tests have been performed using the *gprof* UNIX utility. The *gprof* utility produces an execution profile of a program. Since absolute time estimations of *gprof* prove rather inaccurate (in particular, processing times with and without computation of Jacobians do not reflect reality), the results are presented relative to the cpu time of '4A/OP 2003-1-0' for each surface/Jacobian configuration.

Run			4A/OP - solar		4A/OP 2003-1-0	
surface	night/day	Jacobian	cpu total	cpu calcrad	cpu total	cpu calcrad
black	night	no	0.985	0.184	1.000	0.162
	day	no	1.014	0.205		
desert	night	no	0.990	0.231	1.000	0.219
	day	no	1.008	0.250		
black	night	yes	1.022	0.026	1.000	0.016
	day	yes	1.024	0.027		
desert	night	yes	1.025	0.019	1.000	0.018
	day	yes	1.027	0.021		

Table 22: Total cpu time and partial cpu time in subroutine *calcrad* (including children routines) for the above described eight test configurations. All times are given relative to the total cpu time of 4A/OP 2003-1-0 for each of the four surface/Jacobian configurations. Using the compilation option O4 (maximum optimisation) on a SUN Fire 880 system (4 processors Ultrasparc III, 900 MHz), the total cpu time is roughly 50s and 1400s for cases without and with Jacobian computation, respectively.

Generating numerically identical results, the night-time cases are directly comparable between the two versions. Concerning the spectra computation (without Jacobians) the total user cpu time of the solar version is reduced by 1% and 1.5% for the two surface types, respectively. This finding is unexpected, so we conclude that the accuracy of the run time evaluation is not better than about ±2%. Thus, if Jacobians are included, the total user cpu time increase of the solar version (2.2% and 2.5%, respectively) appears acceptable. Partial cpu times spent in *calcrad* and children routines can be considered as equivalent within the limits of accuracy, except for the case of a black-body surface including Jacobian computation. Here the cpu time increase of the solar version is due to the additionally implemented computation of the surface emissivity Jacobian, requiring numerous calls of the computationally costly *power* function (see section 1.5.2.1.3, equation 16).

The additional cpu time due to the computation of the solar component (comparison between night-time and daytime cases in Table 22) varies between a few percent (without Jacobians) and a few per mille (including Jacobians). The corresponding absolute additional computational effort is roughly equivalent and mainly due to the transmittance computation along the solar path, while the actual solar radiance and Jacobian computation itself is hardly significant in the cpu time budget.

It is concluded that, within the run time evaluation accuracy, a user neglecting the solar contribution to radiance and Jacobians does not suffer any set-back from the release change (4A/OP 2003-1-0 to 4A/OP-solar). On the other hand, 4A/OP-solar proposes a new and computationally efficient optional functionality to account for the impact of the solar radiation in various applications.

1.5.4. Validation

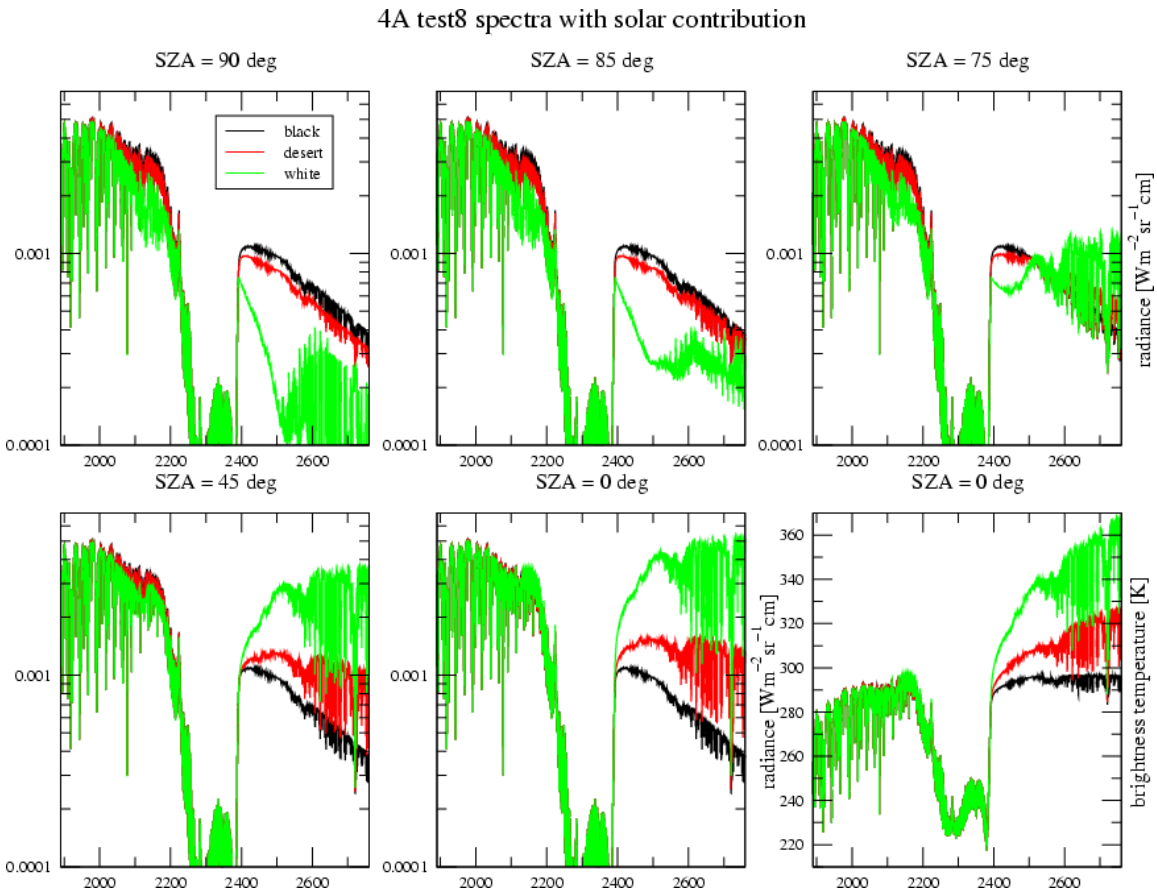
The implementation of the solar radiation in 4A/OP is validated qualitatively, i.e., we verify the impact of the solar radiation component for its reliability with expected features in different simulation conditions. In order

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

to generate some approximate quantitative validation elements, we evaluate the impact of the solar component not only for realistic grey body surfaces but also for the hypothetical case of a white surface of albedo one and zero emissivity.

The simulation conditions are summarised here below:

- The instrument spectral response function is IASI level 1c, i.e., the spectral sampling interval and the spectral resolution are 0.25 cm^{-1} and 0.5 cm^{-1}, respectively;
- The test atmosphere is a moderately warm and humid atmosphere with near-surface temperature 296 K, as used as 'test 8' in the currently distributed 4A/OP release. Additionally, for the determination of the significant spectral range of the solar radiance component, cold and dry atmospheric conditions have been considered (surface temperature 261 K, total water vapour content 3.97 kg/m^2);
- The surface is considered at temperature 296 K as white surface, as a desert surface with spectral emissivity according to emissivity index 103 defined in Table 20, and, for the purpose of comparison, as a black body;
- Spectral radiances, temperature, water vapour, carbon dioxide and ozone Jacobians are simulated for surface and TOA observations (4A/OP simulation configurations 'DOWN' and 'DOWNUP', respectively). The zenith angle of the incoming terrestrial radiance is set to 55°. In case of TOA observations, when we also compute the Jacobians of surface temperature and of surface emissivity, the satellite zenith angle is set to zero (nadir-view);
- Radiative transfer simulations are carried out for wave numbers above 1890 cm^{-1}, except for the determination of the spectral range with a potentially significant solar radiance contribution, where a full IASI spectrum is simulated.



© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Figure 33: IASI TOA radiance spectra for wave numbers above 1890 cm^{-1} for solar zenith angles 90° (equivalent to night time), 85°, 75°, 45°, 0°. The spectra are simulated for a temperate atmosphere (surface temperature 296K), and three types of surface emissivity: Black body (black), desert (red) and white surface (albedo 1, green). The observation zenith angle is 0°. The brightness temperature spectra at zero solar zenith angle are shown in the bottom right corner.

Figure 33 presents TOA radiance spectra at five solar zenith angles for a black body, a desert, and a white surface. The lower right graph shows brightness temperature spectra for a zero solar zenith angle. The upper left graph (θ_s=90°) represents nighttime spectra.

As expected, spectral domains of high atmospheric opacity (e.g. near 2300 cm^{-1}) are affected neither by surface emissivity variation nor by the solar radiance component, because the top-of–atmosphere signal is entirely emitted by the upper atmosphere. In window regions, night-time spectra decrease with decreasing surface emissivity, because the then increasing reflected component is associated to temperatures below surface temperature. The impact of the solar contribution is not negligible in the 5µm window. The night-time brightness temperature deficit of the desert surface with respect to the black body disappears when the sun is close to zenith. This effect is much more important in the 3.7 µm-window region, where the desert surface appears warmer than the black body for solar zenith angles smaller than 75° due to enhanced reflection of solar radiation.

An approximate quantitative validation of the solar impact can be obtained by comparing the white surface spectra at zero and 90° solar zenith angles. Neglecting any atmospheric absorption in the 3.7 µm window, we expect that solar radiance is fully conserved in the TOA spectrum. In Figure 33, we state a solar impact of approximately 5 mWm^{-2}sr^{-1}cm at wave numbers close to 2700 cm^{-1}. Verifying that the solar irradiance at 2700 cm^{-1} is about 1.6 10^{-6} Wcm^{-2}cm, we state with satisfaction that this indeed corresponds to a spectral radiance of 5.1 mWm^{-2}sr^{-1}cm.

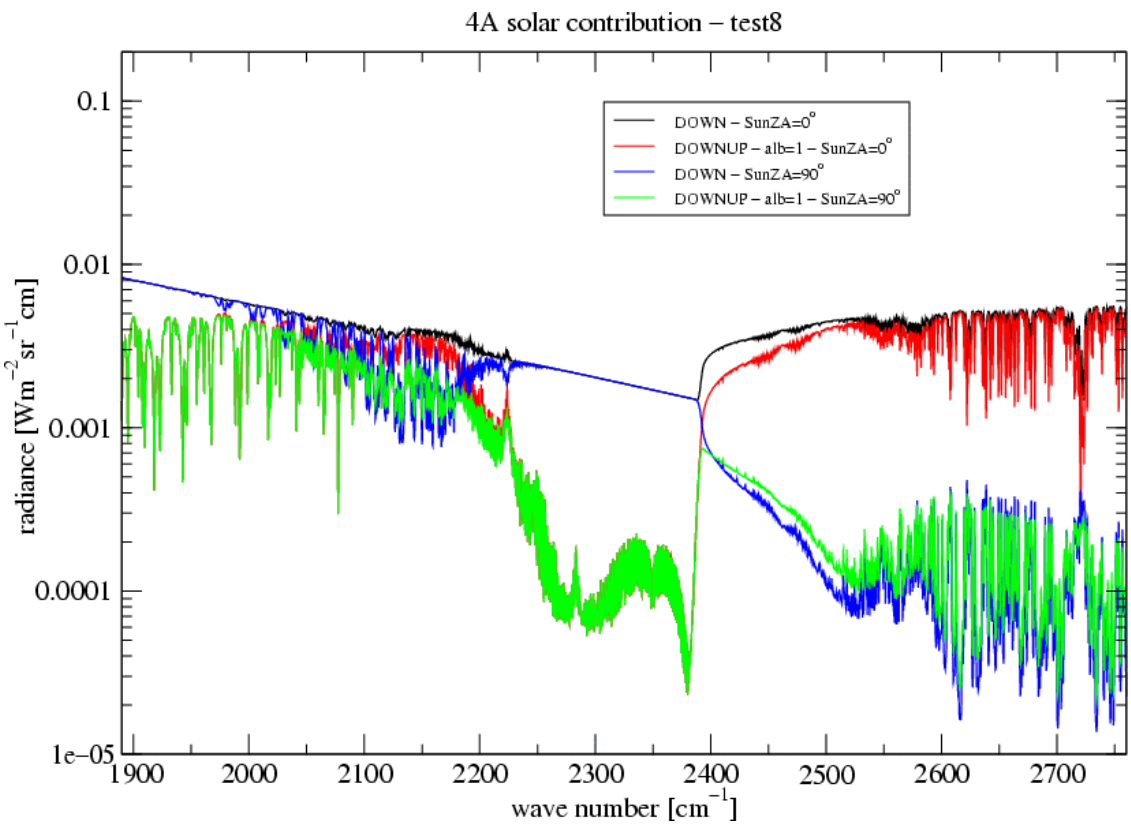


Figure 34: IASI daytime (solar zenith angle 0°) and nighttime TOA radiance spectra for a white surface (green and red, respectively) in comparison to corresponding surface daytime and nighttime spectra (black and blue, respectively) for wave numbers above 1890 cm^{-1}. The spectra are simulated for a temperate atmosphere (surface temperature 296K). The observation zenith angle is 0°.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Figure 34 compares nighttime and daytime (sun at zenith) spectra observed at surface and at TOA in case of a white surface. The expected features are observed:

- Equivalence between daytime and nighttime spectra at the observation levels in spectral domains associated with high atmospheric opacity;
- Quasi-equivalence between surface and TOA spectra in spectral domains associated with high atmospheric transparency. This equivalence becomes stronger at daytime in spectral regions where the solar component dominates the terrestrial radiance component. For nighttime spectra, the surface spectrum appears colder than the TOA spectrum when the atmospheric transmittance is close to one, because the opposite radiation observed (and fully reflected) at the surface is cold and weakly enhanced by emissions along the optical path towards the TOA observer. On the other hand, with decreasing atmospheric transmittance, the opposite radiation becomes warmer and is weakened along the optical path towards TOA by colder emissions.

TOA radiance Jacobians of temperature and of water vapour as well as of CO_2 and of ozone are presented in Figure 35 and Figure 36, respectively, for three pressure levels representing the upper, mid- and lower troposphere. The unit of temperature Jacobians is 1K NEdT@280K for a 1K temperature increase all over the layer, the unit of greenhouse gas Jacobians is 1K NEdT@280K for a mixing ratio increased by 1 g/kg in the entire layer. We distinguish the case of a black body surface, a desert surface at nighttime, and the same desert surface with the sun at zenith.

The solar impact on temperature Jacobians is small, since the impact of temperature changes on spectroscopy and thus on optical thickness in a given layer (c.f. equation (21.)) is small with respect to the direct effect due to enhanced emission.

Water vapour Jacobians in the upper troposphere are quite similar in the three simulation conditions. The strongest absorption domains (wave numbers below 2000 cm^{-1}) lead to heavily negative values without noticeable impact of solar radiation and of surface emissivity in the upper troposphere. Weak water vapour absorption in the 3.8 µm window leads to slightly negative Jacobians. The now noticeable negative impact of solar radiation is simply explained by the enhanced TOA radiance. Lower tropospheric Jacobians for a black body surface are slightly negative or zero, because either atmospheric opacity makes the lower troposphere invisible, or the surface emission is weakly reduced by slightly colder water vapour emissions. In these spectral domains, Jacobians become positive in case of a desert surface, where the positive effect of reflected opposite radiation predominates over the negative effect of weakened surface emission. Obviously, the impact of solar radiation remains negative because a water vapour mixing ratio increase leads always to an increase in optical thickness and thus to a weakening of the reflected solar radiance contribution observed at TOA. The same features are noticed for CO_2 and ozone Jacobians in the relevant spectral domains, i.e., negative values for a black body surface (due to the absence of vertical temperature inversions), positive values for the desert/nighttime case in the lower troposphere in atmospheric windows and a strongly negative impact of solar radiation, increasing with increasing wave number.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

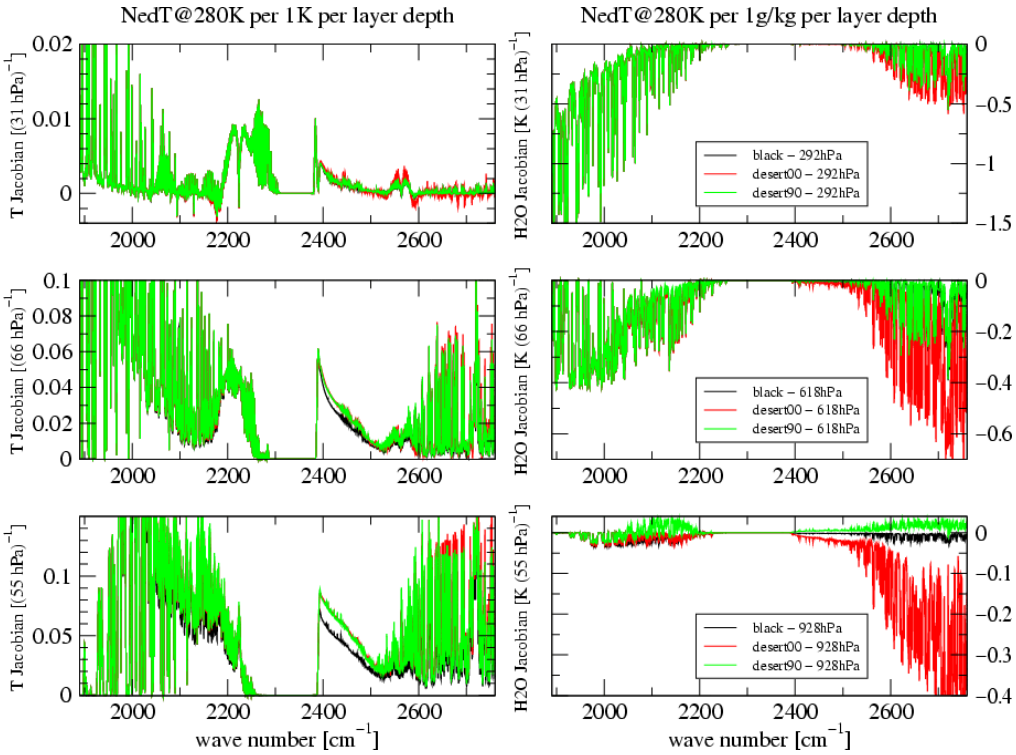


Figure 35: **Temperature (left) and water vapour mixing ratio (right) TOA radiance Jacobian simulation for three vertical layers of depth 31, 66, 55 hPa and top pressure 292, 616, 928hPa (from top to bottom), respectively. Black: black body surface. Red: Desert surface at night. Green: Desert surface with sun at zenith. Other simulation conditions as before.**

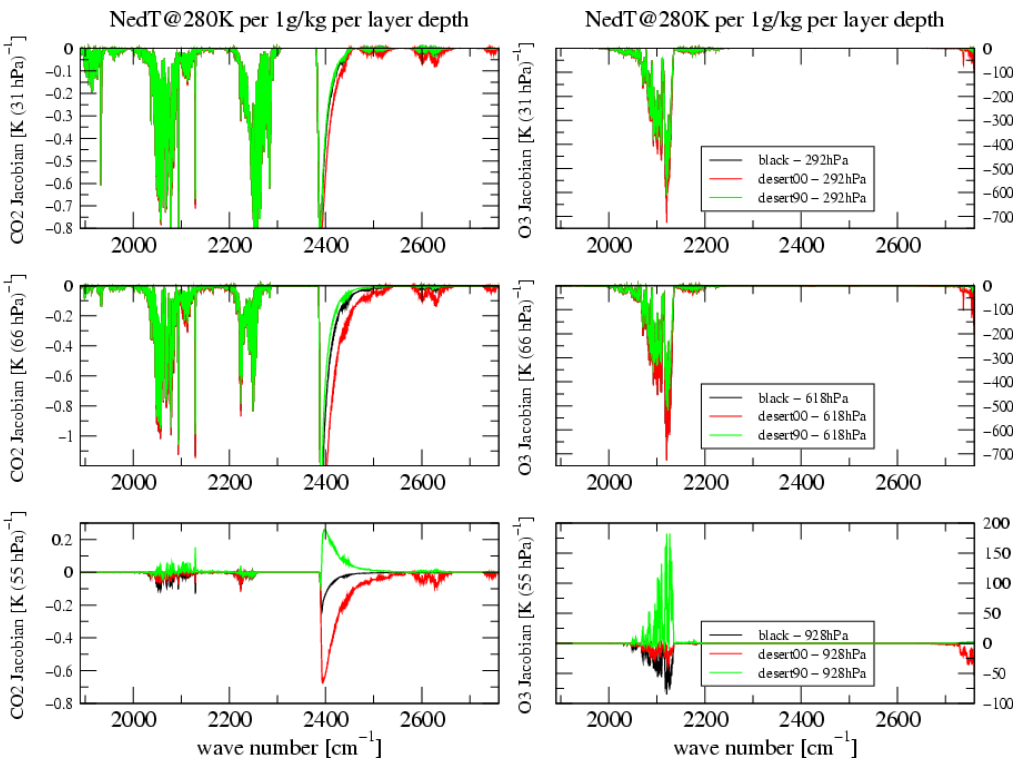


Figure 36: As Figure 35 for carbon dioxide (left) and ozone mixing ratio (right) TOA radiance Jacobians.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

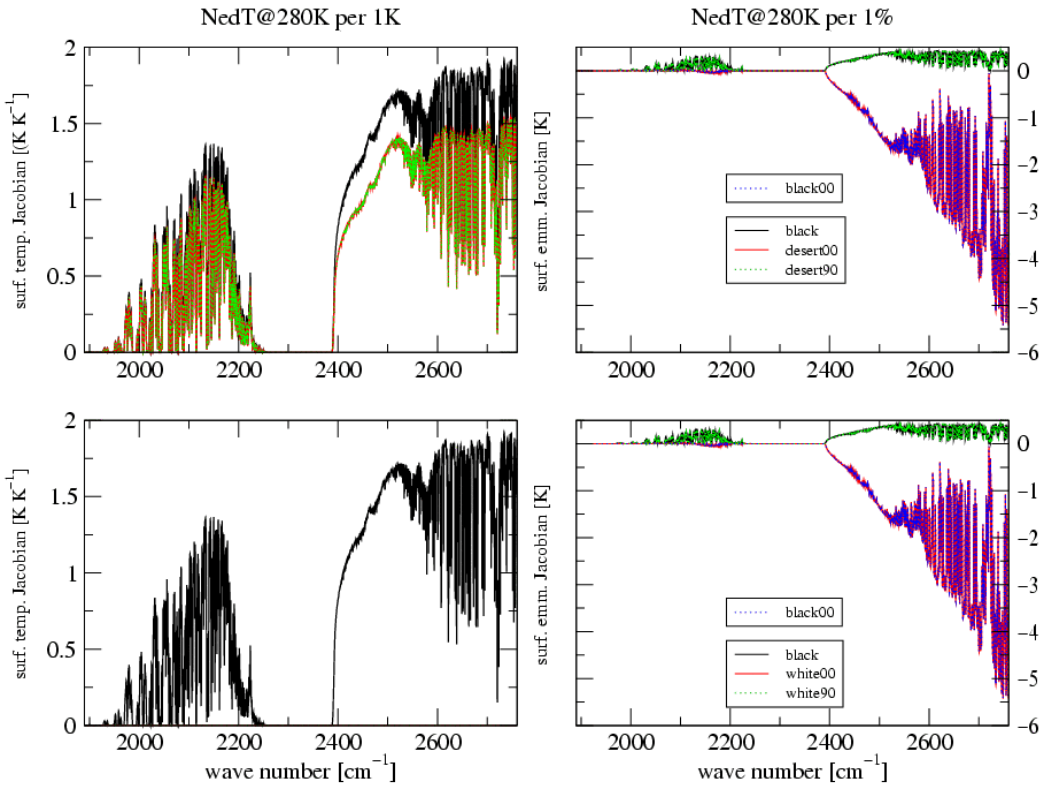


Figure 37: Surface temperature (left) and surface emissivity (right) TOA radiance Jacobian simulation. Top: Black surface vs. desert surface, day (00) and night (90). Bottom: Black surface vs. white surface, day (00) and night (90). Other simulation conditions as before.

Surface temperature Jacobians are presented on the left of Figure 37. The unit is 1K NEdT@280K for a surface temperature increase of 1K. Considering the three surface types, we state without surprise that they are proportional to the surface emissivity (in particular, zero for the white surface) and that they are insensitive to solar radiation. More interestingly, right hand side of Figure 37, surface emissivity Jacobians are shown in K NEdT@280K for a surface emissivity increase of 1%. As expected, nighttime Jacobians are positive, while daytime Jacobians become negative for wave numbers above 2400 cm^{-1}, and to a smaller extent, between 2000 and 2200 cm^{-1} (c.f. Figure 33).

According to the discussion in section 1.5.2, it is confirmed that the surface emissivity Jacobian is independent of the surface emissivity itself. In particular, we state the successful application of equations (16.) and (25.), i.e., the new implementation of the surface emissivity Jacobian calculation in case of a black body surface.

The solar TOA radiance component (sun at zenith) is shown for the test atmosphere (white and desert surface and for a dry and cold atmosphere (desert surface) in Figure 38. The red curve provides a realistic case, indicating that the solar radiance component is not negligible in the 2000 –2200 cm^{-1} wave number range and becomes dominating for wave numbers beyond 2500 cm^{-1}. Comparison with the green curve provides the impact of surface albedo. While the simulated case of albedo one does not correspond to any natural surface, we can immediately conclude that in case of quasi-specular reflection (with bi-directional reflectance values that may exceed one by far) the solar radiance contribution becomes significant all over the thermal infrared window region between 800 and 1300 cm^{-1}.

Finally, the results for the cold and dry atmosphere indicate that in extreme cases, TOA radiance spectra are not affected by solar radiation only in the inner part of the strong water vapour absorption band between 1400 and 1800 cm^{-1} and in the CO_2 absorption band between 2250 and 2375 cm^{-1}. In this context, spectral thresholds for the solar radiance contribution are currently not implemented.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

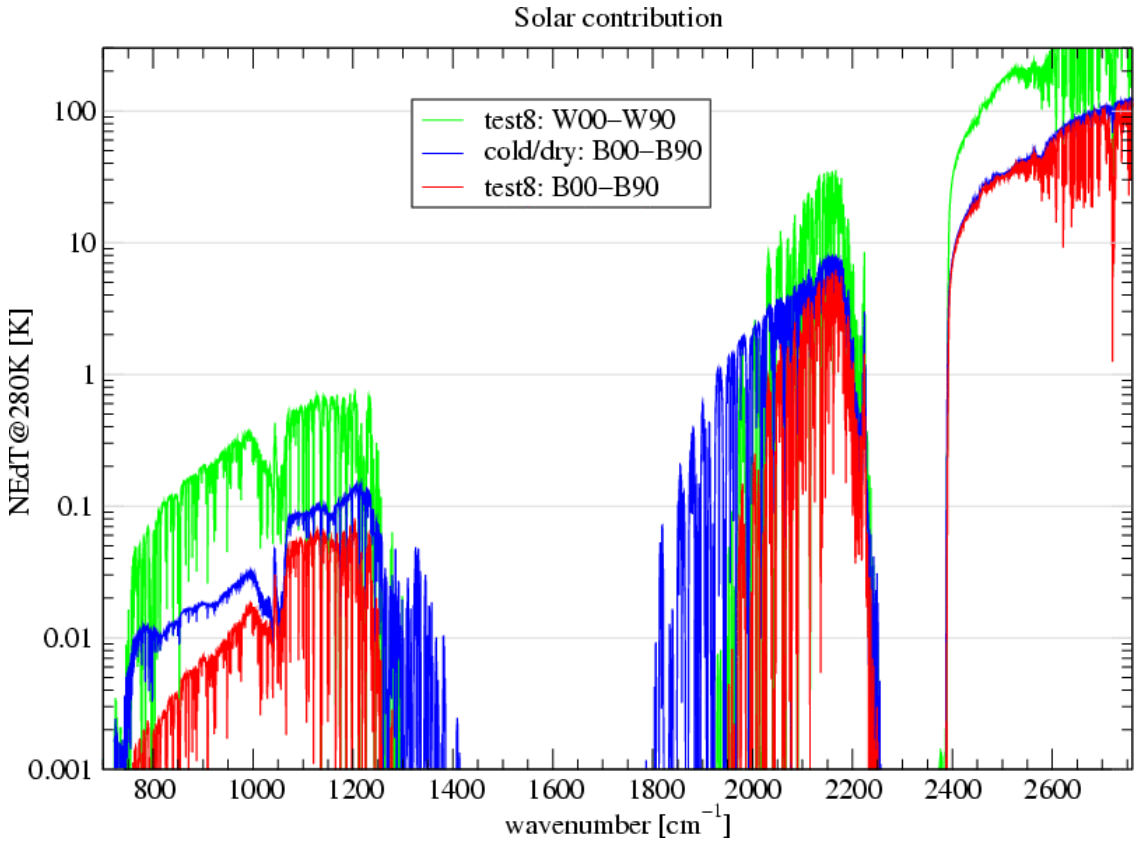


Figure 38: Solar impact on the TOA radiance for test8 atmosphere (desert (B00-B90) and white (W00-W90) surface) and a dry and cold atmosphere (desert surface). Other simulation conditions as before.

A validation of the implementation of the solar radiance component in 4A/OP upon observational data is desirable. In terms of suitability and accessibility, MODIS channels 20 to 25 (3.75µm-4.53µm) appear as a good candidate. However, the validation with observational data will always be affected by knowledge errors of surface temperature, emissivity and the vertical temperature profile.

The implementation of the solar radiance component in 4A/OP is meant to provide the option to consider its impact for applications in the IASI spectral domain. The common 4A/OP user must not be bothered by this option. It is anticipated that a simulation tool of higher precision is needed if the spectral range of 4A/OP radiative transfer simulations is extended towards the shortwave infrared. This would require especially the introduction of the azimuth angle as additional polar coordinate in order to account for anisotropic reflection, and the use of a high-precision solar irradiance spectrum in replacement of the 6000 K black body approximation.

1.6. Implementation of aerosol extinction

In the frame of developments of the 4A/OP, aerosol extinction has been implemented : in this aim, DISORT [Ref. 23] and (optionally) SOS routines, dedicated to the simulation of aerosol (and cloud) induced scattering of IR radiation, has been coupled with the 4A/OP programme package. All routines has been written in the fortran 90 applicable to 4A/OP, except the external DISORT and SOS main programmes (transformed in subroutines) and depending subroutines, which are written in fortran 90 fixed form source code.

System tests are executed for the same geophysical, instrumental and geometric conditions as model unit tests. The user script parameters are provided to enable reapplication of the system verification tests. Parameters identical for all tests are:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- INS = 'iasi1c'
- ATM = 'satigr_v4.0_moyclas'

1.6.1. General

The general test verifies the capacity to produce a convolved spectrum over the entire spectral domain using the three 4A/OP-aerosol options

- using the original RT routine (without aerosol);
- using DISORT or SOS without aerosol;
- using DISORT with aerosol;
- using SOS with aerosol.

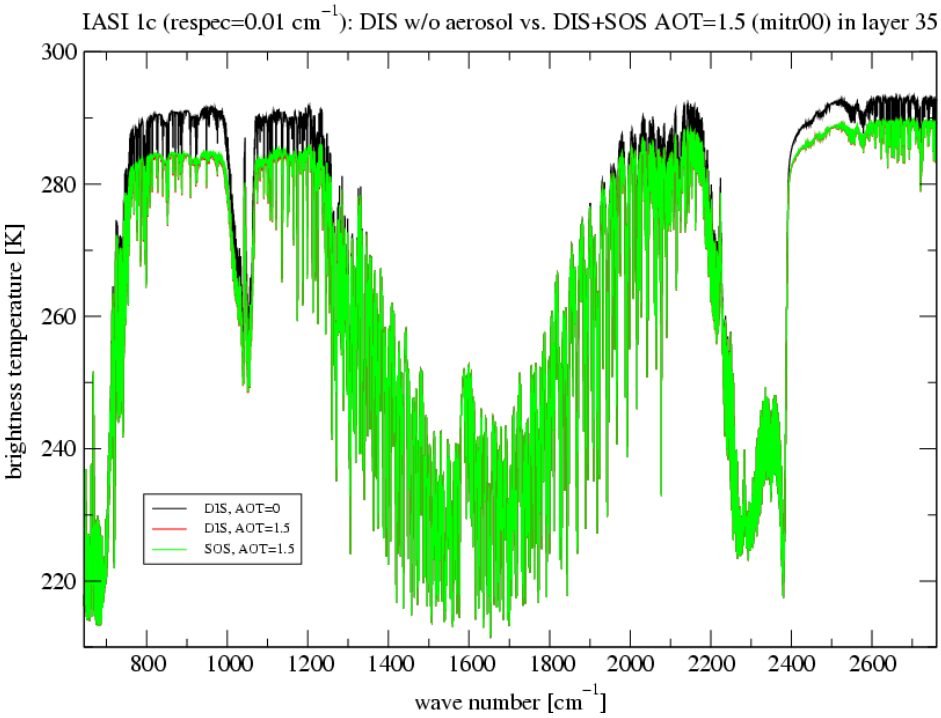
System verification **test 1** compares spectra obtained by DISORT and SOS for a single aerosol layer with the spectrum obtained by DISORT without aerosol in otherwise identical conditions.

The script parameters of test 1 are:

Sp Id	RSCA	AEROMOD	AOTREF	OMEAERO=GAERO	Remark
S11	DIS_0	undef	0.	0.5	DISORT spectrum without aerosol
S12	DIS_1_5	mitr00	-1.5	0.	DISORT spectrum with AOT=1.5
S13	SOS_1_5	mitr00	-1.5	0.	SOS spectrum with AOT=1.5

Table 23: 4A/OP-aerosol user script parameters for system verification test 1. Common parameters are ATMPROF=0001; RSTR=ref0_01, NUMIN=645, NUMAX=2760.

Spectra 11, 12, and 13 are shown in Figure 39. Test 1 confirms the functionality of the external routines DISORT and SOS over an extended spectral domain. The output spectra appear reasonable. Introduction of aerosol leads to a brightness temperature decrease in atmospheric window regions (S12 and S13 vs. S11). There are only slight differences between DISORT and SOS spectra including aerosol (S12 and S13). These differences are analysed in more detail in further tests.



© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Figure 39: System Verification test 1.

1.6.2. Aerosol layer definition and options on spectral variation

System verification **test 2** compares DISORT spectra between 800 and 805 cm^{-1} for similar conditions of the vertical aerosol distribution. The script parameters of test 2 are:

Sp Id	RSCA	AEROMOD	LAYAERO	AOTREF	OMEAERO = GAERO	Remark
S21	DIS_1_5v	mitr00	35 35	1.5	0.	layer 35: AOT=0.936@800cm^{-1}
S22	DIS_0_5v3l	mitr00	33 35	0.5	0.	same as S21 but aerosol distributed in layers 33-35
S23	DIS_0_5v3l_0_5v3l	mitr00 soot00	33 35 36 38	0.5 0.5	0. 0.	same as S22, plus underlying aerosol layer (model soot00, total reference AOT=1.5) in atm. layers 36-38
S24	DIS_0v	mitr00	35 35	0.	0.	Reference AOT loaded from model: 1.805@1000cm^{-1}, thus AOT=1.126@800cm^{-1}
S25	DIS_1_5	mitr00	35 35	-1.5	0.	AOT=1.5 constant in layer 35
S26	DIS_1u5	undef	35 35	1.5	0.5	AOT=1.5, ω=0, g=0.5, constant in layer 35
S27	DIS_3v	mitr00	35 35	3.	0.	layer 35: AOT=1.872@800cm^{-1}, twice the load as in S21

Table 24: 4A/OP-aerosol user script parameters for system verification test 2. Common parameters are ATMPROF=0001; RSTR=ref0_01_short, NUMIN=800, NUMAX=805.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Spectra S21 to S27 are shown in Figure 40. The individual spectra do not intersect within the small domain of investigation. S21 loads the specified reference AOT and interpolates correctly to 0.936 at 800 cm^{-1}. With respect to S21, S22 smears the same aerosol load over the current layer and the two overlying atmospheric layers. Since these are colder we observe S22 < S21 as expected. With respect to S22, S23 adds another aerosol layer below the first layer. This leads to further obstruction of the surface emission, S23<S22. S24 corresponds to the case S21, except that the reference AOT is loaded from the model.

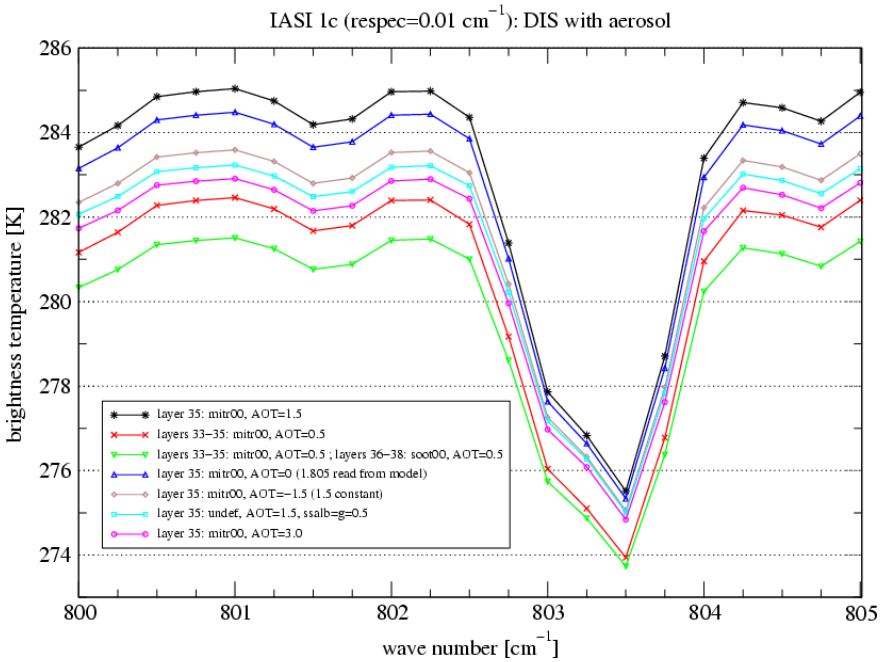


Figure 40: System Verification test 2.

It is about 20% above the reference AOT for S21. For S25 and S27, the reference AOT is further increased, by consequence S27 < S25 < S24 < S21. S26 correspond to the specific case where AOT, single scattering albedo and asymmetry parameter are spectrally constant. The two latter being different from all other cases of test2, S26 is not directly comparable.

System verification **test 3** is identical to test 2 except that spectra are obtained by SOS.

The script parameters of test 3 are the same as for test 2 except for RSCA where SOS replaces DIS:

Sp Id	RSCA	AEROMOD	LAYAERO	AOTREF	OMEAERO = GAERO	Remark
S31	SOS_1_5v	mitr00	35 35	1.5	0.	layer 35: AOT=0.936@800cm^{-1}
S32	SOS_0_5v3l	mitr00	33 35	0.5	0.	same as S21 but aerosol distributed in layers 33-35
S33	SOS_0_5v3l_0_5v3l	mitr00 soot00	33 35 36 38	0.5 0.5	0. 0.	same as S22, plus underlying aerosol layer (model soot00, total reference AOT=1.5) in atm. layers 36-38
S34	SOS_0v	mitr00	35 35	0.	0.	Reference AOT loaded from model: 1.805@1000cm^{-1}, thus AOT=1.126@800cm^{-1}
S35	SOS_1_5	mitr00	35 35	-1.5	0.	AOT=1.5 constant in layer 35
S36	SOS_1u5	undef	35 35	1.5	0.5	AOT=1.5, ω=0, g=0.5, constant in layer 35
S37	SOS_3v	mitr00	35 35	3.	0.	Layer35: AOT=1.872@800cm^{-1}, twice the load as in S21

Table 25: 4A/OP-aerosol user script parameters for system verification test 3. Common parameters are ATMPROF=0001; RSTR=ref0_01_short, NUMIN=800, NUMAX=805.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

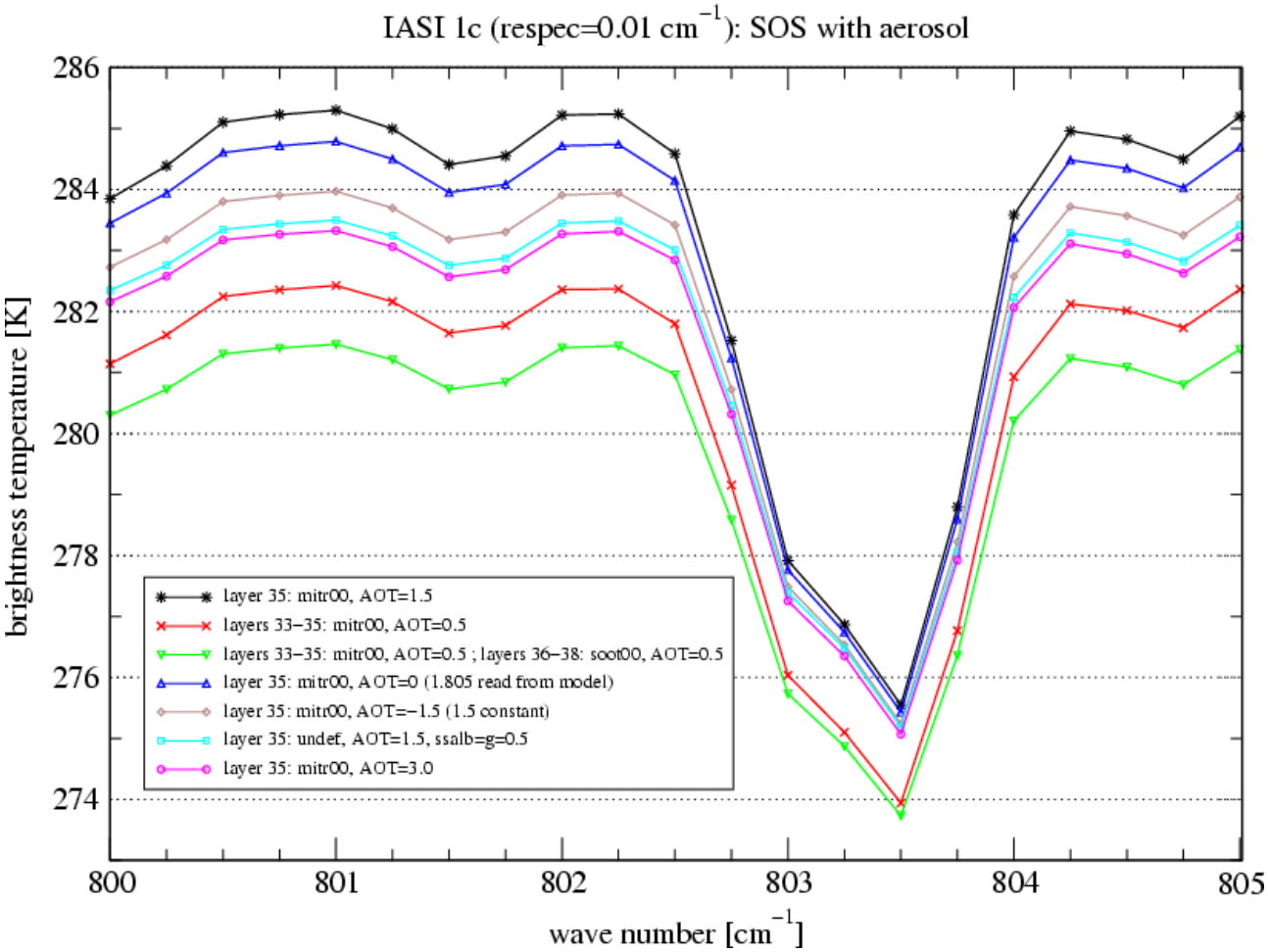


Figure 41: System Verification test 3.

Spectra S31 to S37 are shown in Figure 41 and correspond to spectra S21 to S27, respectively, in Figure 40. SOS spectra do not deviate from corresponding DISORT spectra by more than a few tenths of K in brightness temperature. These deviations are likely due to the additional approximations in SOS with respect to DISORT. Their evaluation will be subject of scientific validation in a later stage. The differences between the individual SOS spectra (S31 to S37) are qualitatively as consistent as those between DISORT spectra (S21 to S27).

1.6.3. Comparison with reference software without aerosol

System verification **test 5** is dedicated to the comparison of spectra obtained with the three options (classical, DISORT, SOS) in identical, **aerosol free** conditions. This test evaluates the consistency of the radiative transfer modules among each other.

The script parameters of test 5 are:

Spectrum Id	RSCA	Remark
S51	{empty}	calcsig, no aerosol
S52	DIS_0	DISORT, no aerosol, identical to S11
S53	SOS_0	SOS, no aerosol
S54	-	Corresponding spectrum obtained with reference software RS 1

Table 26: 4A/OP-aerosol user script parameters for system verification test 5. Common parameters are ATMPROF=0001, RSTR=ref0_01, NUMIN=645, NUMAX=2760, AEROMOD=undef, AOTREF=0.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Figure 42 presents the spectra differences of S51, S52 and S53 with respect to the reference spectrum S54. Spectra S51 and S54 are identical, which means that the classical radiative transfer simulation is not modified. DISORT and SOS spectra in aerosol free conditions are equivalent to better than 1 mK. The deviation from the classical 4A/OP spectrum is significant. A positive bias is observed is atmospheric window regions and spectral regions of extremely high opacity (typical value 0.1 K). A negative bias of the same magnitude can be stated in spectral domains of medium opacity (e.g., in the water vapour band around 7 µm. Total equivalence is not expected due to:

- the different implementation of the radiative transfer equation in 4A/OP on the hand, DISORT and SOS on the other hand;
- the approximate 'transformation' of the DISORT and SOS geometry module from plan-parallel to spherical;
- the different vertical interpolation mode of atmospheric profiles (computation of layer temperatures and mixing ratios from input profiles).

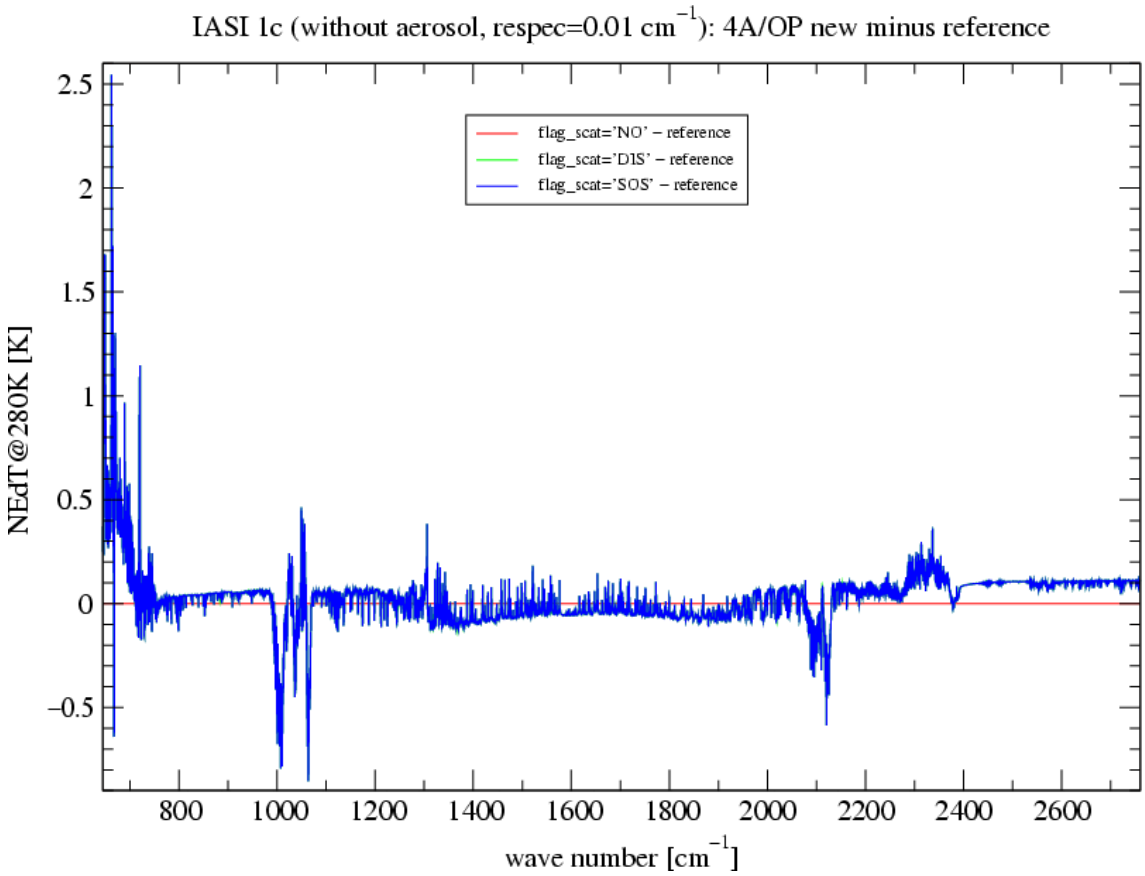


Figure 42: System Verification test 5.

Table 27 summarises the bias and standard deviations over all 8461 IASI channels.

S51-S54	S52-S54	S53-S54
0.00±0.00	0.03±0.15	0.03±0.15

Table 27: Bias and standard deviation (brightness temperature difference in [K]) over the entire IASI spectrum for system verification test 5.

Even though inconsistency between 4A/OP and DISORT/SOS spectra remains globally in reasonable limits, it becomes obvious that significant discrepancies appear in some spectral domains. In this sense the results of test 5 may be considered as a performance evaluation.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.6.4. Impact of the user parameterised spectral sampling

System verification **test 6** analyses the impact of the user parameterisation of the spectral sampling of high-resolution (HR) radiance spectra prior to the convolution process. The HR spectral sampling step has to be considered relative to the spectral resolution, i.e., 0.5 cm^{-1}. We consider HR spectral sampling at 0.001, 0.005, 0.01 and 0.05 cm^{-1} and generate DISORT and SOS spectra for the standard conditions (aerosol mitr00 in layer 35 at constant AOT=1.5). Expecting the DISORT spectrum at highest HR spectral sampling as the most accurate, we compute the difference of any other spectrum with respect to this reference. The script parameters of test 6 are:

Spectrum Id	RSCA	RSTR	Remark
S601	DIS_1_5	ref0_001	DISORT, HR spectral sampling 0.001 cm^{-1}
S602	DIS_1_5	ref0_005	DISORT, HR spectral sampling 0.005 cm^{-1}
S603	DIS_1_5	ref0_01	DISORT, HR spectral sampling 0.01 cm^{-1}, identical S12
S604	DIS_1_5	ref0_05	DISORT, HR spectral sampling 0.05 cm^{-1}
S611	SOS_1_5	ref0_001	SOS, HR spectral sampling 0.001 cm^{-1}
S612	SOS_1_5	ref0_005	SOS, HR spectral sampling 0.005 cm^{-1}
S613	SOS_1_5	ref0_01	SOS, HR spectral sampling 0.01 cm^{-1}, identical S13
S614	SOS_1_5	ref0_05	SOS, HR spectral sampling 0.05 cm^{-1}

Table 28: 4A/OP-aerosol user script parameters for system verification test 6. Common parameters are ATMPROF=0001, NUMIN=645, NUMAX=2760, AEROMOD=mitr00, AOTREF=-1.5, OMEAERO=GAERO=0.

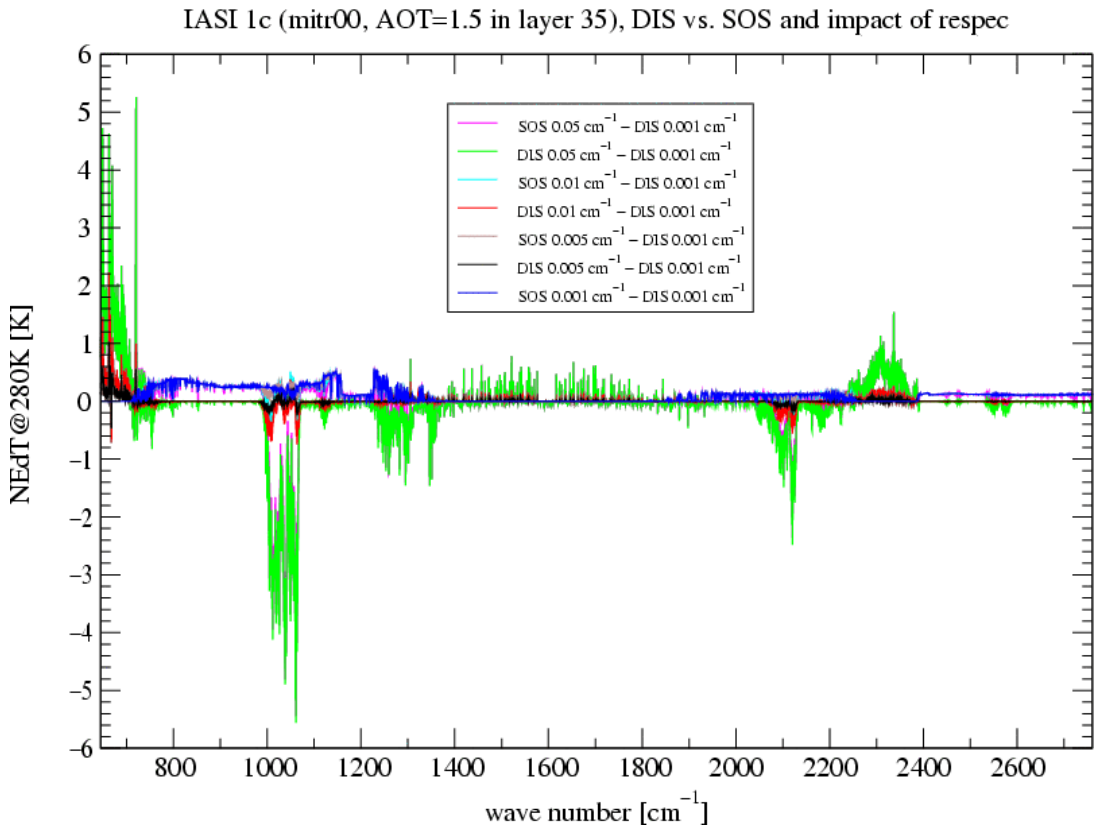


Figure 43: System Verification test 6.

Figure 43 presents these spectra differences with respect to the reference spectrum S601.

Table 29 summarises the bias and standard deviations over all 8461 IASI channels.

	-	S602-S601	S603-S601	S604-S601

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

	S611-S601 (0.001 cm^{-1})	S612-S601 (0.005 cm^{-1})	S613-S601 (0.01 cm^{-1})	S614-S601 (0.05 cm^{-1})
DISORT-DISORT (0.001 cm^{-1})	-	0.01±0.05	0.01±0.12	-0.07±0.62
SOS-DISORT(0.001 cm^{-1})	0.10±0.11	0.10±0.11	0.11±0.14	0.03±0.61

Table 29: Bias and standard deviation (brightness temperature difference in [K]) over the entire IASI spectrum for system verification test 6.

These results confirm the stability of the formerly stated positive bias of SOS with respect to DISORT. Moreover, they suggest first conclusions (in terms of performance).

- Differences between HR spectral sampling 0.001 and 0.005 cm^{-1} (i.e., 1/500 and 1/100 of spectral resolution, respectively) become negligible;
- HR spectral sampling 0.01 cm^{-1} (i.e., 1/50 of spectral resolution) might be a compromise in terms of trade-off between accuracy and computation speed;
- At HR spectral sampling 0.05 cm^{-1} (i.e., 1/10 of spectral resolution), the convolution error largely dominates the uncertainty of the radiative transfer computation. As expected, strong convolution errors appear in domains of high spectral variability.

1.6.5. Validation of variable viewing geometry

System verification **test 7** analyses the stability of the differences observed in test 5 for nadir view as function of the observation zenith angle. Therefore, we consider spectra in aerosol free conditions at observation zenith angle 45°. If consistent with test 5 results, we can conclude that the implementation of the plan-parallel DISORT and SOS geometry models in the spherical 4A/OP geometry model is successful.

The script parameters of test 7 are:

Spectrum Id	RSTR	RSCA	SECTTA1	Remark
S701	ref0_01	{empty}	0	calcsig, no aerosol, nadir, identical S51
S702	refSAT45SUN90_0_01	{empty}	45	calcsig, no aerosol, 45°
S703	refSAT45SUN90_0_01	DIS_0	45	DISORT, no aerosol, 45°
S704	refSAT45SUN90_0_01	SOS_0	45	SOS, no aerosol, 45°

Table 30: 4A/OP-aerosol user script parameters for system verification test 7. Common parameters are ATMPROF=0001, NUMIN=645, NUMAX=2760, AEROMOD=undef, AOTREF=0.

Figure 44 presents these spectra difference between the two 4A spectra at 45° and 0° observation zenith angle (S702-S703) for reference. The differences of DISORT (S703) and SOS (S704) spectra at 45° from the corresponding 4A spectrum (S702) are also illustrated.

Table 31 summarises the bias and standard deviations over all 8461 IASI channels.

S702-S701	S703-S702	S704-S702
-1.54±1.12	0.03±0.16	0.08±0.15

Table 31: Bias and standard deviation (brightness temperature difference in [K]) over the entire IASI spectrum for system verification test 7.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

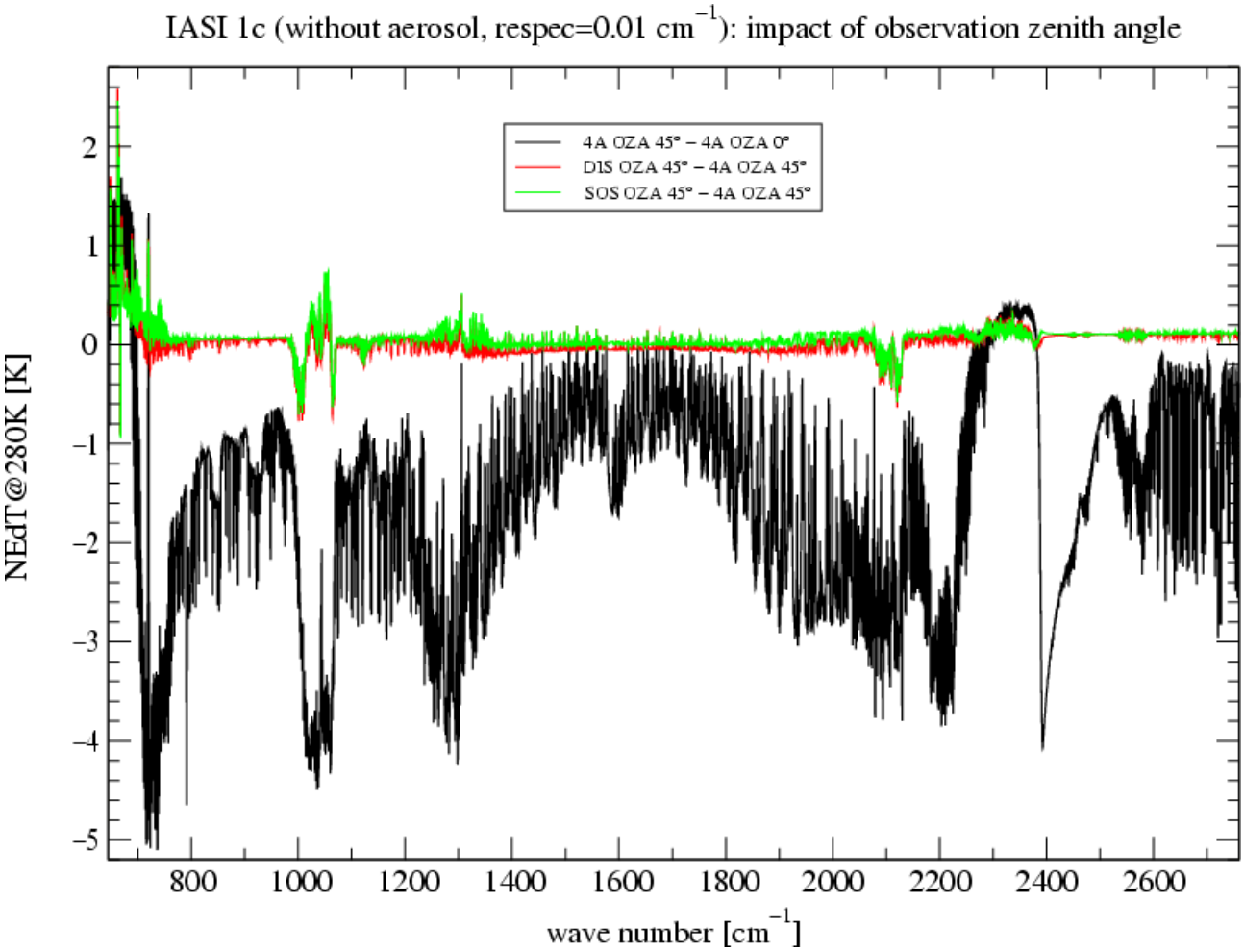


Figure 44: System Verification test 7.

The result is surprising. We observe that the deviation between DISORT and 4A is very stable between observation zenith angles 0° and 45°. However, the SOS-4A deviation, very similar to the DISORT-4A deviation at nadir view, increases (addition of a positive bias of 0.05K) at observation zenith angle 45°. Could this indicate that the SOS performance decreases with increasing observation zenith angle?

1.6.6. Validation of the simulation of solar radiation

System verification **test 8** investigates the processing of solar radiation between 4A/OP and DISORT codes (reminder: solar radiation is not implemented in SOS). We focus on the spectral range 2750-2755 cm^{-1} that contains channels of high atmospheric transmittance as well as a few channels of higher opacity. Direct comparison between 4A/OP and DISORT requires aerosol free conditions. As consequence of the different geometry models in 4A/OP and DISORT, we expect as mentioned in section 0 potentially inconsistent simulations in particular for observation geometry with strong differences between solar and observational zenith angles (referred to as SZA and OZA, respectively). Therefore we consider the cases SZA=0°/OZA=80° and vice versa, to be compared with a case of similar SZA/OZA 5°/0°.

The script parameters of test 8 are:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Sp.Id	RSTR	RSCA	SECTTA1	SECTTASUN	Remark
S801	refSAT00SUN80_0_01	{empty}	0	80	calcsig, nadir, low sun
S802	refSAT80SUN00_0_01	{empty}	80	0	calcsig, low satellite, sun at zenith
S803	refSAT05SUN00_0_01	{empty}	5	0	calcsig, heigh satellite, sun at zenith
S811	refSAT00SUN80_0_01	DIS_0	0	80	as S801 but DISORT
S812	refSAT80SUN00_0_01	DIS_0	80	0	as S802 but DISORT
S813	refSAT05SUN00_0_01	DIS_0	5	0	as S803 but DISORT

Table 32: 4A/OP-aerosol user script parameters for system verification test 8. Common parameters are ATMPROF=0001, NUMIN=2750, NUMAX=2755, AEROMOD=undef, AOTREF=0.

Figure 45 presents classical 4A/OP spectra (S801-S803) as solid lines and DISORT spectra (S811-S813) by crosses at the 21 channel positions in the selected spectral window. S801/811 intersects with S802/812 due to increased absorption at high observation zenith angle on the one hand and strong solar contribution in case of high atmospheric transmittance on the other hand. As expected, S802/812 is below S803/813 due to a higher observation zenith angle, S801/811 is below S803/S813 due to smaller solar contribution at comparable observation zenith angle. The most remarkable feature is the equivalence between 4A/OP and DISORT simulated spectra including solar radiation. A weak positive bias of DISORT spectra with respect to 4A/OP is observed. The risk of inconsistent results in case of strong OZA-SZA difference is not confirmed.

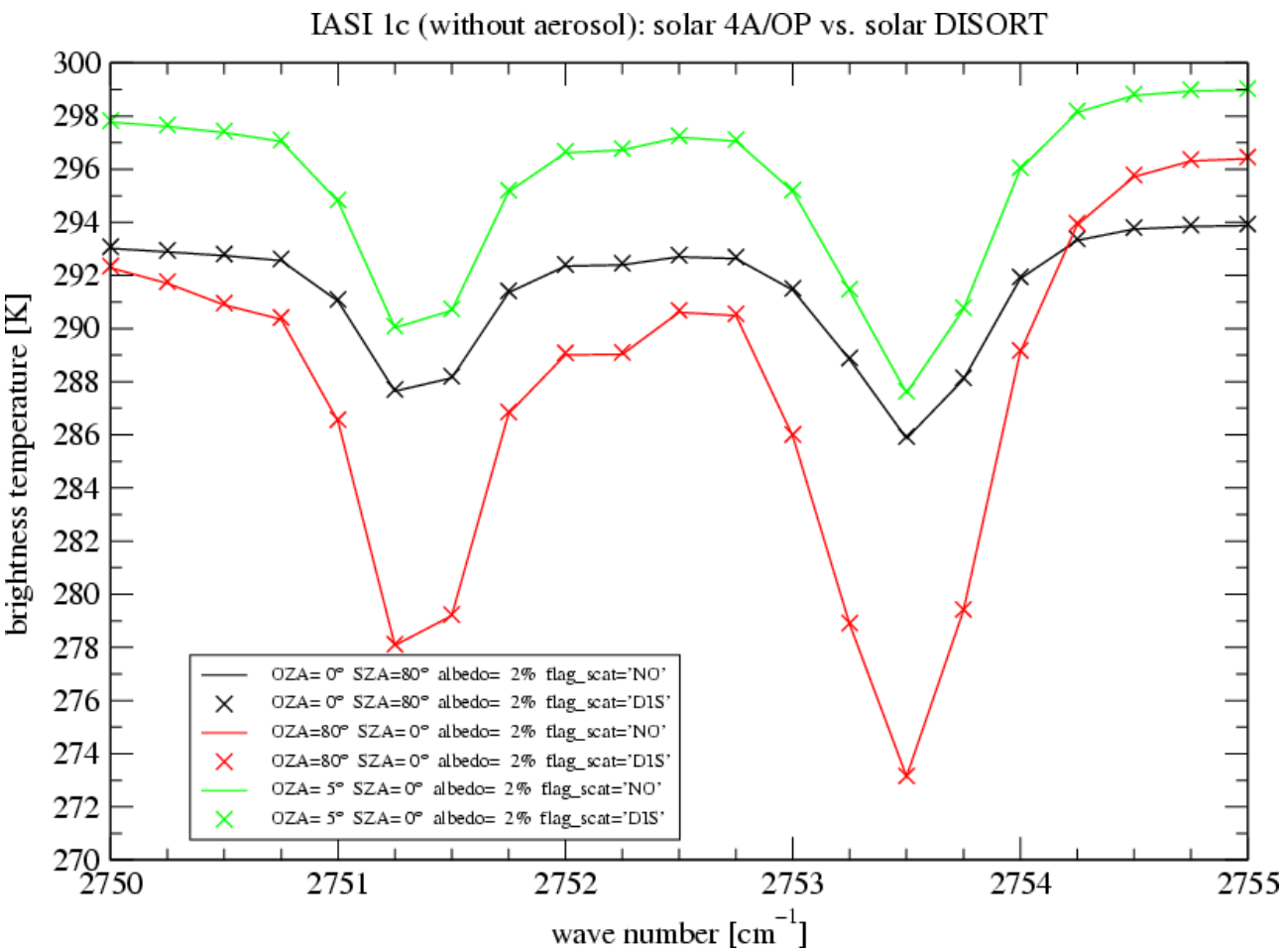


Figure 45: System Verification test 8.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Table 33 summarises the bias values over the selected 21 channels.

S811-S801	S812-S802	S813-S803
0.061	0.061	0.056

Table 33: Bias (brightness temperature difference in [K]) over 21 spectral channels for system verification test 8.

These results let us conclude not only that the implementation of solar radiation in the 4A/OP-DISORT interface routine has been successful, they present at the same time the first quantitative validation for the implementation of solar radiation in 4A/OP (this validation has been obtained so far only qualitatively).

1.6.7. Run time tests and trade-off speed vs. accuracy

Run time test are carried for standard geophysical (including one aerosol layer corresponding to a single 4A/OP layer) and geometric conditions over the full IASI spectrum. Reference is a 4A/OP spectrum (without aerosol) at spectral sampling 0.01 cm^{-1} before convolution.

The tests are run under Solaris OS on UltraSparc 4 server using the unix command 'time'.

1.6.7.1. Spectral sampling step

Tests in Table 34 investigate the increase of computation speed with decreasing spectral sampling for DISORT and SOS. In all runs (except reference 4A/OP) the number of streams is 10. As a reading example, a DISORT run at 0.001 cm^{-1} spectral sampling takes 919 times the 4A/OP reference run time at 0.01 cm^{-1} spectral sampling.

	DISORT (10 streams)			SOS (10 streams)			4A/OP		
spectral sampling	u cpu [s]	u–s cpu [s]	u–s cpu [rel]	u cpu [s]	u–s cpu [s]	u–s cpu [rel]	u cpu [s]	u–s cpu [s]	u–s cpu [rel]
0.001 cm^{-1}	134489	134210	919.2	3935	3908	26.8	-	-	-
0.005 cm^{-1}	28335	28310	193.9	927	903	6.2	-	-	-
0.01 cm^{-1}	14162	14108	96.6	536	514	3.5	152	146	1.0
0.05 cm^{-1}	2984	2955	20.2	227	205	1.4	-	-	-

Table 34: Run time test results as function of the spectral sampling (parameter respect): u=user, s=system, rel= relative to the 4A/OP reference case.

The impact of spectral sampling on accuracy has been investigated in section 1.6.4, revealing that a sampling as low as 0.01 cm^{-1} may be acceptable for certain applications and that SOS errors relative to the DISORT high-sampling spectrum are quite insensitive to the choice of the spectral sampling parameter. In contrary, the DISORT error increases continuously with degradation of the spectral sampling.

The accuracy gain of DISORT with respect to SOS has to be paid by a processing time increased by a factor 30. This factor has only a weak dependence on the spectral sampling (slightly increasing with higher sampling).

Therefore, in cases of stringent accuracy requirements (if SOS precision is insufficient), the DISORT spectral sampling should be specified as high as possible with regard to the affordable computation time. On the other hand, if accuracy requirements are moderate and if a lower spectral sampling appears acceptable, the use of SOS is recommended in view of the gain in computation time by a factor 30 with respect to DISORT, but a relatively small loss in terms of accuracy.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

1.6.7.2. Number of streams

The specified stream number in DISORT and SOS has a significant impact on run time and accuracy. However, we conclude from Table 35 that run time variations as function of the stream number are less important than those due to the spectral sampling parameterisation. This is in particular true for DISORT, where a (not recommendable) 2-stream run takes 87% of the run time of a 12-stream run. For SOS, the runtime ratio between a 4-stream run and a 12-streamm run is approximately ½.

	DIS (0.01cm^{-1} sampling)			SOS (0.01 cm^{-1} sampling)			4A/OP		
Number of streams	u cpu [s]	u–s cpu [s]	u–s cpu [rel]	u cpu [s]	u–s cpu [s]	u–s cpu [rel]	u cpu [s]	u–s cpu [s]	u–s cpu [rel]
N/A	-	-	-	-	-	-	152	146	1.0
2	12668	12636	86.5	-	-	-	-	-	-
4	12716	12696	87.0	268	250	1.7	-	-	-
6	12850	12824	87.8	341	325	2.2	-	-	-
8	13167	13124	89.9	416	399	2.7	-	-	-
10	14162	14108	96.6	536	514	3.5	-	-	-
12	14527	14457	99.0	574	535	3.7	-	-	-

Table 35: DISORT and SOS run time test results as function of the number of streams (parameter NMOM): u=user, s=system, rel= relative to the 4A/OP reference case.

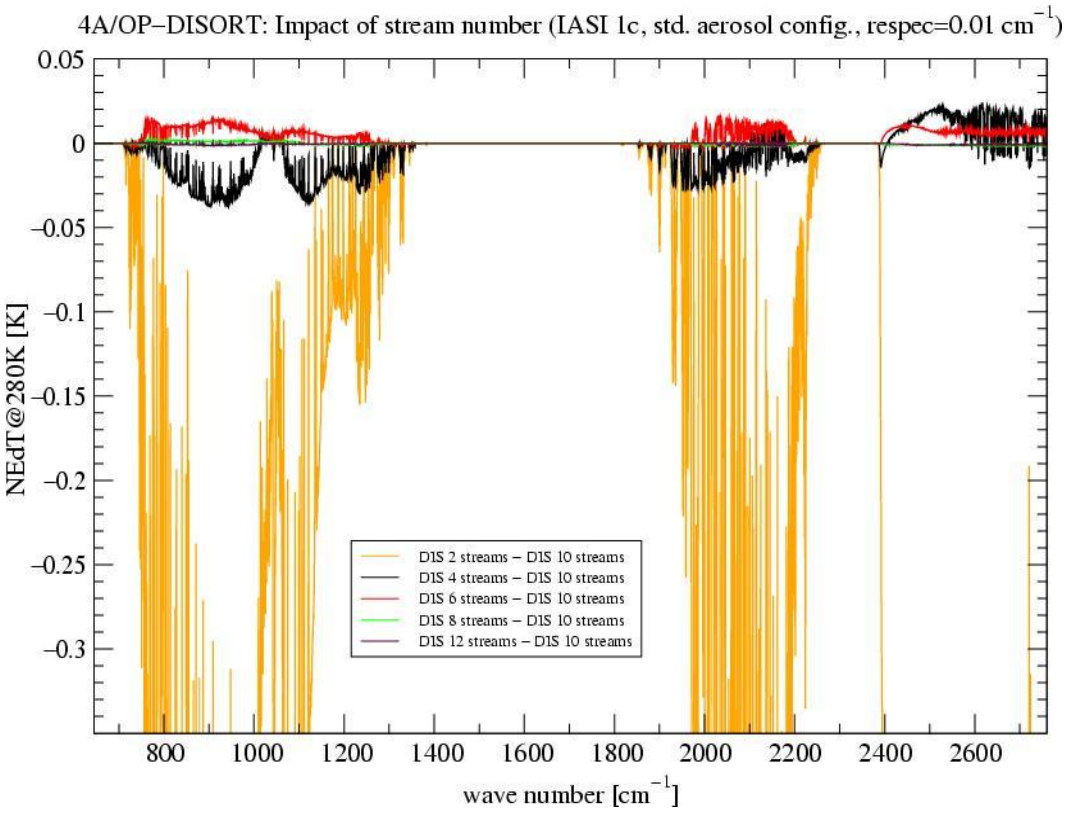


Figure 46: Noise equivalent brightness temperature differences (at 280K) of DISORT simulated spectra in standard geophysical and observation conditions as a function of the number of streams with respect to a 10-stream run. Spectral sampling (respec) before convolution is 0.01 cm^{-1} in all cases.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

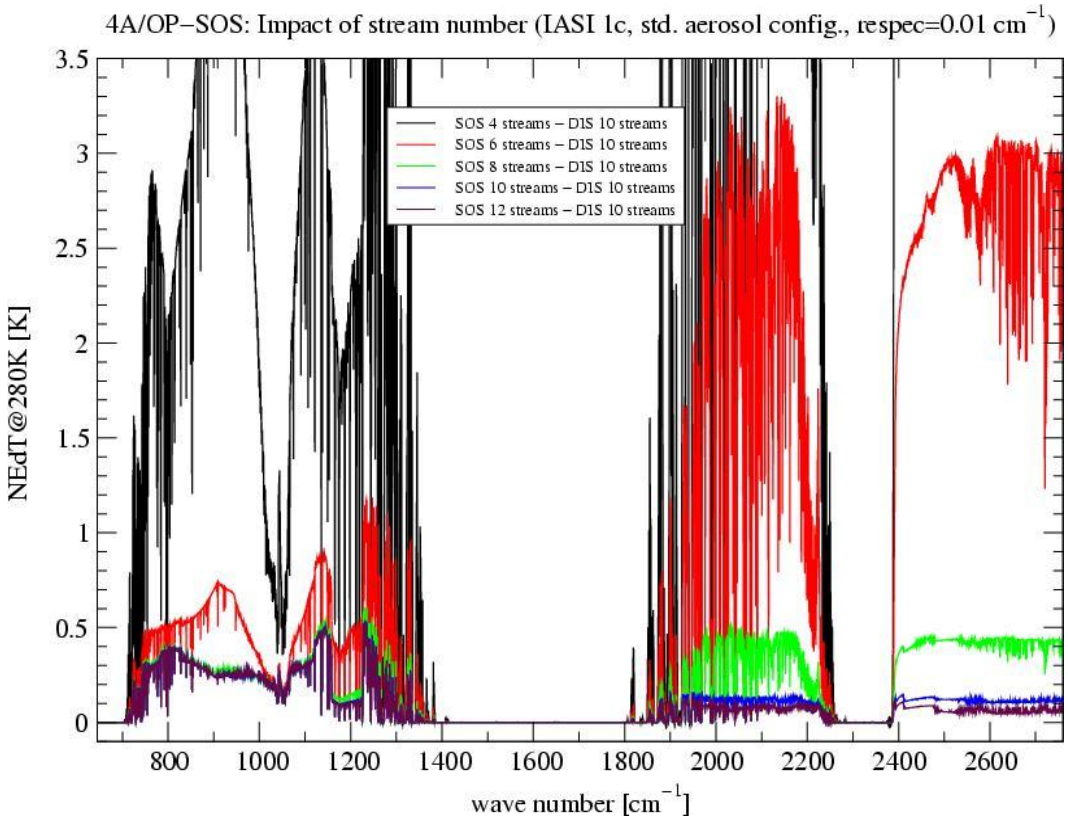


Figure 47: Noise equivalent brightness temperature differences (at 280K) of SOS simulated spectra in standard geophysical and observation conditions as a function of the number of streams with respect to a DISORT 10-stream run. Spectral sampling (respec) before convolution is 0.01 cm^{-1} in all cases.

Figure 46 and Figure 47 illustrate the impact of the choice the number of streams for DISORT and SOS, respectively. The DISORT 10-stream simulation (at 0.01 cm-1 spectral sampling before convolution) is considered as reference.

Table 36 summarises the bias and standard deviation of NEdT values in Figure 46 and Figure 47 over the entire IASI 1c spectrum (8461 spectral samples). It is emphasised that bias and standard deviation are computed over all IASI channels, including those, which are insensitive to the scattering layer (roughly in the CO_2 absorption bands 645-700 and 2250-2380 cm^{-1} and in the H_2O absorption band 1400-1800 cm^{-1}).

Streams	DISORT minus DISORT 10 streams	SOS minus DISORT 10 streams
2	-0.253 ± 0.286	-
4	+0.004 ± 0.013	+6.840 ± 9.359
6	+0.004 ± 0.004	+0.879 ± 1.108
8	+0.0002 ± 0.0008	+0.192 ± 0.181
10	-	+0.103 ± 0.112
12	+0.0004 ± 0.0005	+0.089 ± 0.112

Table 36: Bias and standard deviation of NEdT at 280K in [K] over the 8461 IASI level 1c spectral channels for DISORT and SOS simulations with variable number of streams with respect to the DISORT 10-stream simulations. All simulations consider standard geophysical and observation conditions.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

It is noticed that DISORT errors as function of the stream number is limited to a few cK even for a number of streams as low four (the lowest recommendable specification). Errors of the order mK occur for stream numbers of eight and more. The results suggest that errors do not decrease significantly if the stream number specification is higher than 8.

SOS errors are unacceptable for stream numbers below 8. The optimised SOS stream number parameterisation seems to be dependent on the spectral domain after Figure 47. While 8 streams provide almost equivalent results than 10 or 12 in the Longwave-IR, the number of streams should be at least 10 (if not 12) in the Midwave-IR.

1.6.7.3. Accuracy parameter in DISORT simulations including solar radiation

It is recalled that solar radiation can be simulated in 4A/OP either in the non-scatter mode, or in scatter mode using DISORT. Solar radiation is not taken into account by SOS.

Accounting for solar radiation in DISORT requires the simulation of an additional beam (i.e., anisotropic) source and increases the computation time considerably. The user defined parameter 'ACCUR' controls the convergence criterion for azimuthal series for intensities. The precise convergence criterion is that the largest term being added is smaller than ACCUR times the sum of the total series. Recommendation is to specify ACCUR to below 0.01 to avoid the risk of non-convergence in specific cases of parameterisations of optical depth or computation angles. For further details see [Ref. 23].

The impact of the specification of parameter ACCUR on accuracy and run time is evaluated on standard geophysical conditions by simulation of IASI level 1c spectra in the spectral range 1800-2760 cm^{-1}. Observation zenith angle and solar zenith angle are both set to 45°. For the same conditions, a 4A/OP spectrum without aerosol is simulated to provide a reference runtime. Spectral sampling before convolution of all simulations is 0.01 cm^{-1}, the number of streams in all DISORT simulations is 10.

	DISORT			**4A/OP**		
ACCUR	u cpu [s]	u–s cpu [s]	u–s cpu [rel]	u cpu [s]	u–s cpu [s]	u–s cpu [rel]
N/A	-	-	-	55	52	1.0
0.1	17355	17321	333	-	-	-
0.05	17327	17295	333	-	-	-
0.009	20227	20182	388	-	-	-
0.005	21477	21432	412	-	-	-
0.001	27980	27941	537	-	-	-

Table 37: DISORT daytime run time test results as function of the parameter ACCUR for atmospheric standard conditions, OZA=SZA=45°, IASI level 1c spectra from 1800 to 2760 cm^{-1} in comparison to a 4A/OP run in the same conditions but without aerosol. u=user, s=system, rel= relative to the 4A/OP reference case.

DISORT run times as function of ACCUR are provided as absolute values and relative to the 4A/OP reference runtime (without aerosol) in Table 37.

The impact of the ACCUR parameterisation on run time is significant, for example there is a 25% decrease for an ACCUR specification of 0.9% with respect to 0.1%.

Second, the run time ratio between DISORT and 4A/OP increases in the given conditions (sampling, stream number) from about 100 (without solar radiation) to 400 (with solar radiation). Since the 4A/OP runtime is quite independent of the inclusion of the solar radiation, we can conclude that the inclusion of solar radiation in DISORT simulations is very expensive in terms of computational cost. Obviously, the need for this time investment depends on the spectral simulation domain and the height of the scattering layer and not further discussed here.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Accuracy figures as function of the ACCUR parameterisation are quite ambiguous. Figure 48 shows spectrum differences for different ACCUR parameterisation with respect to the tightest of the considered specifications ACCUR=0.001.

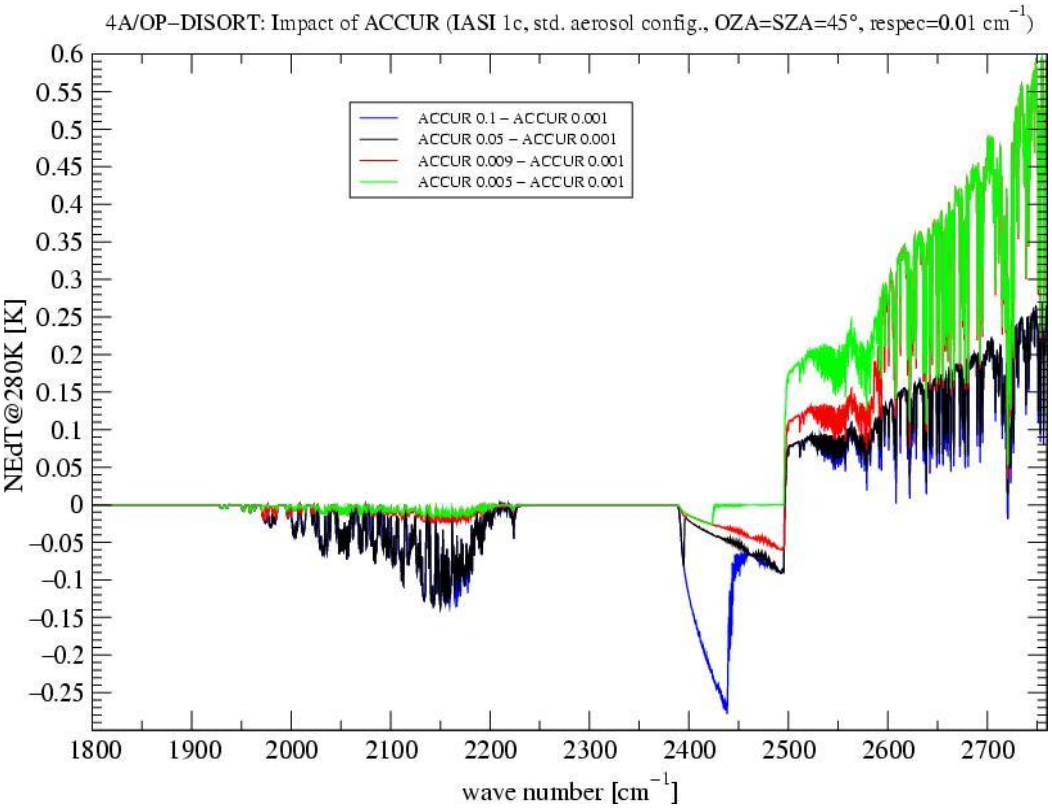


Figure 48: Noise equivalent brightness temperature differences (at 280K) of DISORT 10-stream simulated spectra as function of different ACCUR parameterisation (minus ACCUR=0.001) for atmospheric standard conditions, OZA=SZA=45°, IASI level 1c spectra from 1800 to 2760 cm^{-1}. Spectral sampling (respec) before convolution is 0.01 cm^{-1} in all cases.

If considering the spectrum obtained with the tightest ACCUR specification as truth, the error behaviour of relaxed ACCUR parameterisations is consistent with the expectation only for wave numbers below 2500 cm^{-1} sharp. Here we notice that the error increases the more the specification is relaxed. Consistent with the recommendation (ACCUR ≤ 0.01), we state that error figures for ACCUR specifications may be qualified as acceptable, while the errors for more relaxed values are potentially not.

For wave numbers above 2500 cm^{-1}, however, the "error behaviour" as function of ACCUR is not yet understood. In particular, it appears suspicious that a bias corresponding to about 0.2 K is added to all spectra with respect to the reference (ACCUR=0.001), in other words the bias between spectra associated to ACCUR parameterisations other than 0.001 stays constant at 2500 cm^{-1}. This would rather suggest that the problem is related to the reference spectrum itself.

Reinterpreting Figure 48, we state that spectra obtained for ACCUR parameterisations 0.005 and 0.009 provide the most consistent spectrum couple among all possible couples. A moderate bias between the two spectra is limited to the wave number interval 2430 – 2600 cm^{-1}. Based on this statement, we can only recommend the range 0.005 – 0.009 for user specification of ACCUR.

The error behaviour in Figure 48 hints at the occurrence of numerical errors with spectral dependence. Numerical DISORT errors are known and reported for specific user specified optical depths and angular configurations (in terms of observation zenith angle, solar zenith angle, relative azimuth angle and computation polar angles), in particular when observation and solar zenith angles are strictly identical.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

In this context, Figure 49 investigates the stability of error figures in the most critical spectral region around 2500 cm^{-1} with the viewing geometry. Here we decrease and increase the viewing zenith angle and the solar zenith angle by 1°, respectively, with respect to the configuration in Figure 48.

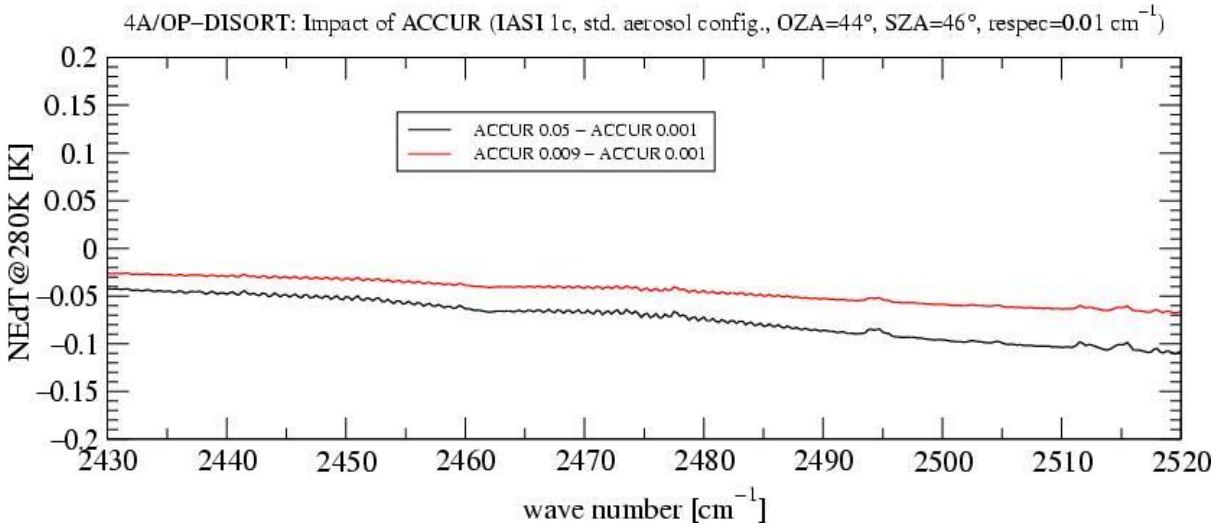


Figure 49: As Figure 48, but with OZA=44° and SZA=46° and limited to ACCUR parameterisations 0.001, 0.009, 0.05 and the spectral domain 2430 to 2520 cm^{-1}.

We can state that the spectrum differences of different ACCUR parameterisations behave quasi-identically for wave numbers below 2500 cm^{-1}, while the discontinuity of differences at 2500 cm^{-1} disappears and the differences beyond this wave number are not comparable with those in Figure 48.

The above investigation confirms that DISORT simulations may lead to (in the above case) moderate intensity discontinuities in specific configurations.

Obviously this statement would need further consolidation, which exceeds the scope of the work in progress, focused on the implementation of aerosol extinction in 4A/OP. Many users will not be concerned by the above problem. So it seems justified to simply alert future users of the problem with the above description.

1.7. Cirrus extinction

Cirrus extinction is modelled in the same way as aerosol extinction. For physical and run parameterisations please refer to sections 5.2.4 and 5.2.5.3 of the users guide (part I of this document), respectively.

1.8. Implementation of Limb Geometry

1.8.1. Introduction

In the frame of developments of the 4A/OP release 4AOP-2006 version 1.0, lim geometry has been implemented to the 4A/OP radiative transfer simulation.

In previous 4AOP versions, the user specified the optical path by the satellite (observer) local zenith angle (scaled to Earth surface) and, if applicable, the incidence angle of reflected radiation and the solar zenith angle. All three angles are scaled to the Earth surface. The Earth surface is defined as a sphere with radius 6371 km. Angles are specified in degrees or as secant values in the user controlled input file *para4A[$RSTR].dtp* or directly in the GUI. Subroutine *para4A* loads these parameters, subroutine *calcsphtraj* affects height dependent zenith angles to each vertical layer of the optical path. Since the user is required to define an observation zenith angle with respect to the Earth surface in the range [0° , 90°[, the definition of a limb viewing geometry was not possible.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

The computation of height as function of pressure is based on a crude approximation of the relationship between height z and pressure p of the geometrical layer centre:

$$z(p) = H \ln [p(z=0)/p]$$

where H is the atmospheric scale height.

The barometric height formula is valid for an isotherm atmosphere (H=8km for T=273K) and constant gravity acceleration. This approximation appears clearly too crude for stratospheric applications and must be reviewed for an implementation of limb viewing geometry.

From this discussion, two software requirements are derived:

Subroutine *para4a* shall support user controlled parameters defining any linear optical path without contact with the Earth surface.

In this early design phase, a candidate approach is to define an observation zenith angle outside the range [0° , 90°[, indicating the user's choice of a limb viewing geometry. In replacement, the tangent height or tangent point pressure is required as additional user specified parameter.

The height computation as function of pressure in subroutine *calcsphtraj* shall be improved.

This requirement implies that at least the atmospheric temperature profile (as function of pressure) is transported into the module. This information is available as output of subroutine *atmsph* (called prior to *calcsphtraj*). As mentioned above, the current z-p relation is independent of the actual atmospheric profile.

Finally, it is recalled that radiative transfer computation in subroutine *calcrad* is based on vertical layers defined by vertical levels loaded from the user defined input file *atm4a[$ATM].ddb*. In particular, it contains a surface layer of infinitesimal depth at defined emissivity and temperature. In limb geometry, this layer becomes 'invisible' (the observation zenith angle in the tangent point is exactly 90°) and has to be neutralised in the radiative transfer computation.

In limb geometry, the surface pseudo-layer in subroutine calcrad shall be neutralised.

Numerical accuracy constrains the geometric minimum depth Δz of the atmospheric layer closest to the tangent point at height H_t above Earth surface R_E=6371km. Limiting H_t to 100km, in limb geometry, the observation zenith angle θ of this atmospheric layer is given by

$$\sin\theta = (R_E+H_t)/(R_E+H_t+\Delta z)$$

Assuming a numerical accuracy of $\sin\theta$ of 2 10^{-6} (including some margin), the minimum layer depth is approximately Δz_{min} = 13m, which does not appear as a strong constraint for user applications. However, the user may define accidentally a tangent height just below a height level defined in *atm4a[$ATM].ddb*. In this case, layer definition in *calcsphtraj* has to be flexible enough to avoid numerical exceptions.

In limb geometry, the depth of the atmospheric layer closest to the tangent point, defined in subroutine calcsphtraj, must be at least 13m.

1.8.2. Theoretical Basis

1 - Gravity acceleration is a function of geodetic latitude Φ and height z

For an ellipsoidal Earth model with half axes a=6378.137km and b=6356.752km, gravity acceleration at sea level (z=0) is approximated by

$$g_0(\Phi) = 9.80616 ms^{-2} \left[1 - 0.0026373 \cos(2\Phi) + 5.9 \cdot 10^{-6} \cdot \cos^2(2\Phi) \right] \quad (26.)$$

including the centrifugal effect

$$g_c(\Phi) = -\Omega^2 \cdot \frac{f^2(\Phi)}{R(\Phi)} \cdot \cos^2(\Phi) \quad (27.)$$

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

with

$\Omega = 2\pi / 86164.091s$ angular rotation speed

f Distance between a point on Earth surface and the intersection of its local orthonormal with Earth rotation axis

R_E Distance between a point on Earth surface and geocentre.

Referring the gravitational component to $g_g = g_0 - g_c$, the height dependence is considered by

$$g(\Phi, z) = g_g(\Phi) \cdot \left(\frac{R_E(\Phi)}{R_E(\Phi) + z} \right)^2 - \Omega^2 f(\Phi) \cdot \frac{f(\Phi) + z}{R_E(\Phi)} \cdot \cos^2(\Phi) \qquad (28.)$$

4A/OP has currently no direct link with an Earth model other than spherical (geodetic latitude is not a user specified parameter). The formulation simplifies with a=b=f=R_E=6371 km and $g_g(\Phi) = g_g = 9.8065$ ms^{-2} to:

$$g(\Phi, z) = g_g \cdot \left(\frac{R_E}{R_E + z} \right)^2 - \Omega^2 \cdot (R_E + z) \cdot \cos^2(\Phi) \qquad (29.)$$

where the centrifugal component is obviously still dependent on latitude.

	Ellipsoid model			Spherical model		
	g(Φ,z)	gravitational	centrifugal	g(Φ,z)	gravitational	centrifugal
Φ=0°, z=0	9.7804	9.8143	-0.0339	9.7726	9.8065	-0.0339
Φ=90°, z=0	9.8321	9.8321	0	9.8065	9.8065	0
Φ=0°, z=100km	9.4792	9.5136	-0.0344	9.4713	9.5057	-0.0344
Φ=90°, z=100km	9.5299	9.5299	0	9.5057	9.5057	0

Table 38: Gravity acceleration and its gravitational and centrifugal components in m s^{-2} for an ellipsoidal and a spherical Earth surface at altitudes 0 and 100 km and at equator and pole.

Table 38 presents gravity acceleration values and those of its components according to the two discussed models. The spherical and the ellipsoidal model agree within 0.26%. Neglecting the centrifugal component generates a maximum error of 0.36%. These errors are small against the 3% decrease of the gravity acceleration with altitude increasing from sea level to 100 km.

It is therefore not of first priority to make the gravity acceleration dependent on latitude (and ask the user to specify another parameter).

The simplified, only height dependent gravitational model for a spherical Earth surface

$$g(z) = g_g \cdot \left(\frac{R_E}{R_E + z} \right)^2 \qquad (30.)$$

accounts for the major part of relevant variations of the gravity acceleration and is implemented in routine *atmsph*.

2 – Altitude as function of pressure

Explicit computation of altitude zl for each pressure level pl is required for a subsequent definition of the optical path.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

The basic assumption is hydrostatic equilibrium

$$dz = \frac{-R \cdot T}{M_d \cdot g(z)} \frac{dp}{p} \qquad (31.)$$

with

R	Universal gas constant
M_d	Mol mass of dry air
T	Temperature

At the lowest atmospheric pressure level nlevatm defined in atm4a$(ATM).ddb, the altitude is set to the user specified value *zatmbot* or, if not specified, set to a value approximated with the barometric height formula for an isotherm atmosphere

$$zatmbot = \max\left(ATM_SHEIGHT \cdot \ln \frac{P_REF}{pl(nlevatm)}, 0 \right) \qquad (32.)$$

with the atmospheric scale height ATM_SHEIGHT=8km and P_REF=1013.25hPa. So the user is not required to define the atmosphere down to surface.

The gravity acceleration gc at level nlevatm is calculated by equation (5).

The height of upper pressure levels is calculated in a loop from the lower to the upper levels (nl=nlevatm-1,1,-1) by computing:

- Layer temperature tc(nl) as arithmetic average of level temperatures tl(nl+1) and tl(nl);
- Geometric layer depth dzc(nl) after equation (6) with T=tc(nl), g(z) = gc(nl+1) (approximation since gc(nl) is not yet known);
- Gravity acceleration gc(nl) at layer centre after equation (5) with z=zl(nl+1) + dzc(nl)/2 (approximation since dzc(nl) is approximated);
- Correction of the geometric layer depth dzc(nl) by the factor gc(nl+1)/gc(nl);
- Level height z(nl) = z(nl+1) + dzc(nl).

3 – Refractive index

Atmospheric refraction provokes a bending of the optical path towards the Earth, because the refraction index increases with pressure.

In general, the refractive index is proportional to the air density. For molecules with a permanent dipole momentum, however, an additional component appears. Therefore atmospheric refraction can be described as function of temperature T, pressure p and partial water vapour pressure e. The impact of water vapour on the refractive index is considerably different in the sub-mm and mm spectral domain and the infrared.

A general formulation of the atmospheric refractive index n (after [Ref. 27], Essen and Froome, 1951) is

$$n - 1 = a_1 \cdot \frac{p}{T} + a_2 \cdot \frac{e}{T} + a_3 \cdot \frac{e}{T^2} \qquad (33.)$$

where the last term applies only to wavelengths above 100µm.

Spectral dispersion of atmospheric refraction is relevant in the visible and near-infrared spectrum.

[Ref. 28] (Barrel and Sears; 1939) found for the refractive index as function of wavelength λ:

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

$$(n-1)\cdot 10^6 = \left(77.48\frac{K}{hPa} + 0.44\frac{K\mu m^2}{hPa}\lambda^{-2} + 0.007\frac{K\mu m^4}{hPa}\lambda^{-4}\right)\cdot\frac{p}{T} - \left(12.79\frac{K}{hPa} - 0.14\frac{K\mu m^2}{hPa}\lambda^{-2}\right)\cdot\frac{e}{T} \quad (34.)$$

The above formula is recommended by IUGG since 1966. [Ref. 29] (Rüeger, 1999) proposes an adjustment of this recommendation to

$$(n-1)\cdot 10^6 = \left(77.535\frac{K}{hPa} + 1.317\frac{K\mu m^2}{hPa}\lambda^{-2} + 0.018\frac{K\mu m^4}{hPa}\lambda^{-4}\right)\cdot\frac{p}{T} - \left(11.27\frac{K}{hPa}\right)\cdot\frac{e}{T} \quad (35.)$$

In practice, spectral dispersion is negligible in the thermal infrared (e.g., λ=10µm). At this reference wavelength we compare the models

Barrel and Sears: $$(n-1)\cdot 10^6 = 77.48\frac{K}{hPa}\cdot\frac{p}{T} - 12.79\frac{K}{hPa}\cdot\frac{e}{T} \quad (36.)$$

Rueger : $$(n-1)\cdot 10^6 = 77.54\frac{K}{hPa}\cdot\frac{p}{T} - 11.27\frac{K}{hPa}\cdot\frac{e}{T} \quad (37.)$$

with the MIPAS model ([Ref. 30]) based on [Ref. 31] (Edlén, 1966). As a limb sounding instrument of the stratosphere, the MIPAS processing does not consider the water vapour effect on the refractive index.

MIPAS : $$(n-1)\cdot 10^6 = 77.534\frac{K}{hPa}\cdot\frac{p}{T} \quad (38.)$$

The refractivity (n-1)10^6 is presented in Table 39 for the three models and typical conditions representing surface, mid-troposphere, tropopause and stratopause.

T [K]	p [hPa]	e [hPa]	Barrel/Sears	Rueger	MIPAS
300	1000	20	257.41	257.72	258.45
300	1000	5	258.05	258.28	258.45
250	500	3	154.81	154.95	155.07
200	200	0.1	77.47	77.53	77.53
250	10	0	3.10	3.10	3.10

Table 39: Refractivity (n-1)10^6 after the models Barrel and Sears, Rueger and MIPAS for typical conditions in terms of temperature T, pressure p and partial water vapour pressure e.

As expected, the MIPAS model deviates from the other models near the surface since the water vapour effect is not considered. However, relative deviations of the refractive index do not exceed 10^{-6}. Relative deviations between Barrel-Sears and Rueger stay within 3 10^{-7}. For a limb viewing geometry, these values correspond to tangent height errors of 6.5 and 2m, respectively. Thus, the three models can be considered as equivalent.

For the sake of modularity of the code, we implement a refraction model dependent on the partial water vapour partial pressure and implement the Barrel-Sears model by introducing the constants:

- RI1_0 = 77.48 K hPa^{-1},
- RI2_0 = -12.79 K hPa^{-1},

in the module constants.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

The refractive index *refil(nl)* is computed per pressure level nl=nlevatm,1,-1 if the refraction computation identifier *refra* is set to 'YES'. The partial water vapour pressure is computed in a local variable *pel*:

$$pel = rol(nl1) \cdot rocoef(1)/0.622 \cdot pl(nl) \qquad (39.)$$

leading to:

$$refil(nl) = 1 + \left(Cst_RI1 \cdot \frac{pl(nl)}{tl(nl)} + Cst_RI2 \cdot \frac{pel}{tl(nl)} \right) \cdot 10^{-6} \qquad (40.)$$

If *refra* is equal to 'NO', the refractive index is set to 1 at all levels, which neutralises the implementation of refraction in the optical path computation. In practice, it is recommended to consider refraction only in limb viewing geometry and in Earth viewing geometry at observation zenith angles close to 90°.

4 – Tangent height correction (limb view)

Due to local spherical geometry, all theory concerning tangent height correction and computation of the effective layer zenith angle is based on a simple formulation of the ray tracing

$$n(r) \cdot r \cdot \sin\theta(r) = const. \qquad (41.)$$

with

r	Distance from Earth centre R_E + z
n	refractive index
θ	Local zenith angle

It is derived from the Snellius law

$$n_1 \cdot \sin\theta_1 = n_2 \cdot \sin\theta_2 = const. \qquad (42.)$$

In spherical geometry, using the sinus theorem, we obtain at different height levels:

$$n(r_1) \cdot \sin\theta(r_1) = n(r_2) \cdot \frac{r_2}{r_1} \sin\theta(r_2) \qquad (43.)$$

Equation (23) applies to all viewing configurations and geometries (Earth view or limb view, including refraction or not).

In limb view geometry, the user specified geometric tangent height z_{gtan} may considerably deviate from the refraction corrected tangent height z_{tan} (by roughly 1km at 8km altitude, 2km at sea level). The correction formular is directly derived from equation (23) by simultaneous application to the refracted ($n(z_{tan})>1$) and unrefracted ($n(z_{gtan})=1$) optical path with θ=90° at the respective tangent heights. Both paths are superposed outside the atmosphere, e.g. at satellite height z_{sat} (where $n(z_{sat})=1$) with identical local zenith angle θ_{sat}. Thus:

$$(R_E + z_{sat}) \cdot \sin\theta_{sat} = n(z_{tan}) \cdot (R_E + z_{tan}) = (R_E + z_{g\tan}) \qquad (44.)$$

and

$$z_{\tan} = \frac{R_E + z_{g\tan}}{n(z_{\tan})} - R_E \qquad (45.)$$

The corrected tangent height is obtained by a double-iteration of z_{tan} as function of $n(z_{tan})$.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

It is stated that the refraction corrected tangent height is independent of the vertical profile of the refractive index, which has impact only on the position of the tangent point but not on its height.

5 – Observation zenith angle correction (Earth view)

In Earth view geometry, a correction in analogy to the geometric tangent height correction would affect the observation zenith angle at Earth surface level. However, such correction is not applied for consistency considerations. In Earth view geometry, the user specifies still the observation zenith angle at Earth surface level (whether refraction is considered or not). This avoids specification of zenith angles above 90°.

6 - Computation of effective local zenith angles

Local zenith angles θ are calculated at every level nl through equation (25):

$$\theta(z(nl)) = a\sin\left(\frac{n(z_{ref})}{n(z(nl))}\cdot\frac{R_E + z_{ref}}{R_E + z(nl)}\cdot\sin\left(\theta(z_{ref}\right)\right) \qquad (46.)$$

- in Earth view geometry: z_{ref}=0 ; $\theta(z_{ref})$=1/acos(sectta) (user specified) for all angles θ_1, θ_2, θ_s;
- in limb view geometry for θ_1: $z_{ref}=z_{tan}$; $\sin\theta(z_{ref})=1$;
- in limb view geometry for θ_2: either $z_{ref}=z_{rfx}$; $\theta(z_{ref})$=55° with reflection layer height z_{rfx} or $\theta_2=\theta_1$;
- in limb view geometry for θ_s: $z_{ref}=z_{rfx}$; $\theta(z_{ref})$= 1/acos(sectts) (user specified) with reflection layer height z_{rfx}.

The effective local zenith layer angle qeff in layer nl is defined such that the ratio between the geometric layer depth dzc(nl) and the length of the optical path dlc(nl) in layer nl is equal to its cosine value:

$$\cos\left(\theta_{eff}(nl)\right) = \frac{dlc(nl)}{dzc(nl)} \qquad (47.)$$

It is calculated by

$$\theta_{eff}(nl) = a\cos\left(\frac{\cos\frac{\theta(nl)+\theta(nl+1)}{2}}{\cos\frac{\theta(nl)-\theta(nl+1)}{2}}\right) \qquad (48.)$$

The approximation $\theta_{eff}(nl) = \left(\theta(nl)+\theta(nl+1)\right)/2$ is not valid at zenith angles close to 90° or in case of a very crude vertical resolution.

The secant values of θ_{eff} are used as weighting factors of the optical depth interpolation in routine *calcsig* and the transmission computation in routine *calcrad*.

1.8.3. Validation

We try to simulate real measurements in best-knowledge (i.e., rather limited) geophysical/instrumental conditions.

- instrument: MIPAS;
- merged level1/level2 product (processing level 4.61);
- date: 2003 Jan 17, original granule index: 32 of 74;
- LOS index selection: 10 and 16 (approximate tangent heights: 28 and 10 km);
- selected spectral domains: 716 – 755 cm^{-1}, and 2035 – 2075 cm^{-1};
- spectral sampling 0.025 cm^{-1};

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

- Instrument spectral response functions: as elaborated in ESA/MIPAS wind retrieval project (level 1b);
- atmospheric information: MIPAS level 2 retrieval at 17 tangent heights (temperature, pressure, mixing ratio H_2O, O_3, N_2O, CH_4, N_2O, HNO_3). Mixing ratio for CO_2, CO, NO, SO_2, NH_3 are taken from *gascon*.

For many reasons, the acceptance tests cannot be judged solely upon the degree of agreement between simulation and measurement.

- The measurements are affected by radiometric noise;
- The instrument spectral response functions are theoretical and not extensively validated;
- The resolution of the absorption atlas (>5 10^{-4} cm^{-1}) is rather low with respect to the spectral resolution of the measurement (roughly 5 10^{-2} cm^{-1});
- The vertical sampling of the atmosphere is rather crude (one level per tangent height, i.e., 17 levels between approximately 7 and 67 km). Interpolation errors may be important;
- Geophysical parameters are MIPAS retrieval products and thus affected by retrieval errors;
- Only crude information is available for certain trace gases, in particular CO_2;
- 4A/OP radiative transfer is mainly validated for IASI applications. Radiative transfer simulation errors for typical geophysical conditions associated to limb viewing geometry are not well known.

The objective is then to identify discrepancies between measurement and simulation and to link them either to the above-mentioned items or to an erroneous implementation of the limb view geometry in 4A/OP.

Below, we present four couples of figures (Figure 50 to Figure 57). Each upper figure contains the measured and the simulated radiance spectrum for the two selected tangent heights and the two selected spectral domains. The accompanying lower figures present a spectral zoom on an interval of width 5 cm^{-1} within each spectral domain.

Figure 50, Figure 51:

The window is dominated by a regular CO_2 absorption pattern. The agreement between simulation and measurement is remarkable, except for:

- beyond 745 cm^{-1}, an overestimation of the simulation in spectral regions of maximum transparancy, e.g., near 752 cm^{-1} (possibly due to erroneous continuum simulation of CCl_4);
- a small overestimation of the simulation in the middle of two adjacent CO_2 absorption lines (possibly explained by ISRF knowledge errors).

Figure 52, Figure 53:

At higher altitude, the above-mentioned systematic deviations between simulation and measurement disappear.

Figure 54, Figure 55, Figure 56, Figure 57:

The window is again characterised by a regular CO_2 absortion pattern. The overall agreement between mmeasurement and simulation is good, discrepancies seem to be dominated by a relatively high radiometric noise level.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

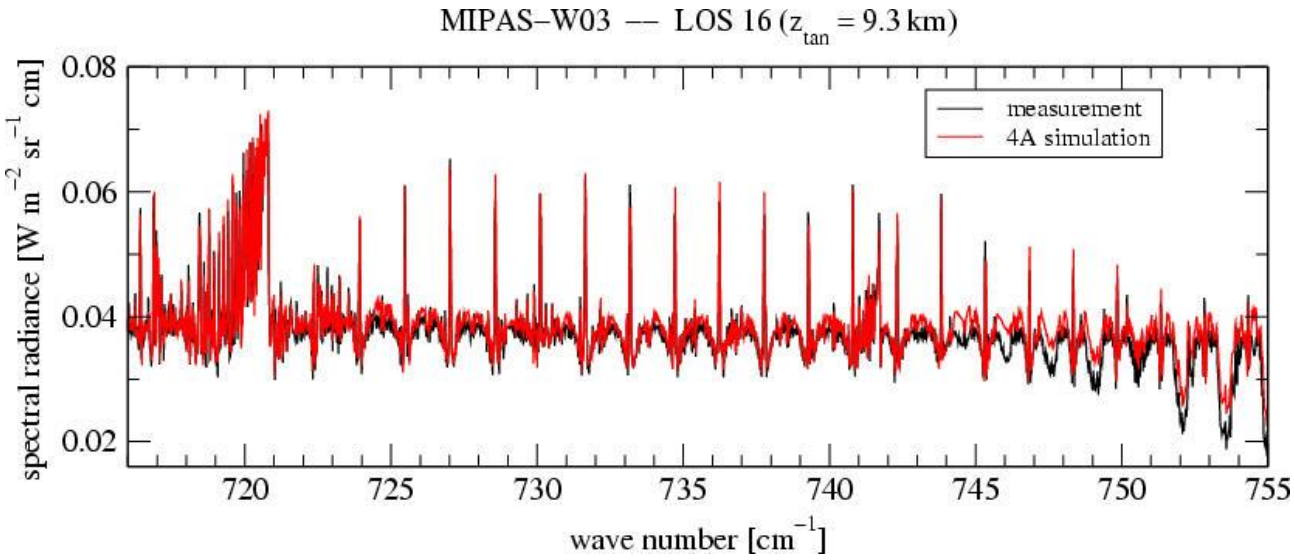


Figure 50: 4A/OP-limb simulation (red) over MIPAS level 1b spectrum (black). LOS number 16. Spectral domain: 716-755 cm^{-1}.

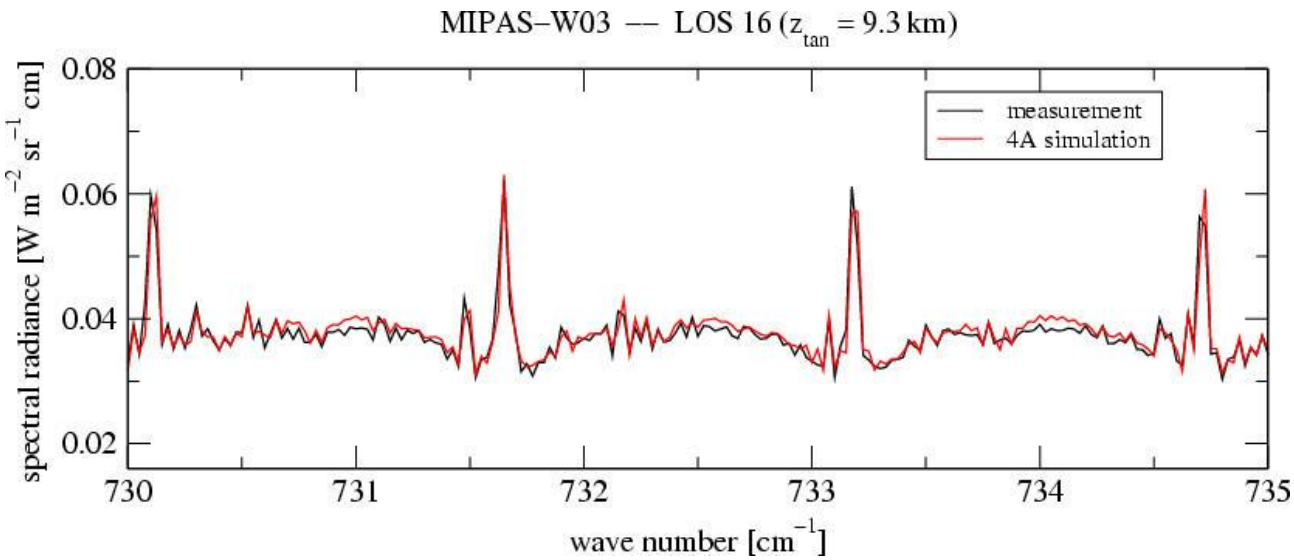


Figure 51: 4A/OP-limb simulation (red) over MIPAS level 1b spectrum (black). LOS number 16. Spectral domain: 730-735 cm^{-1} (zoom of upper figure).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

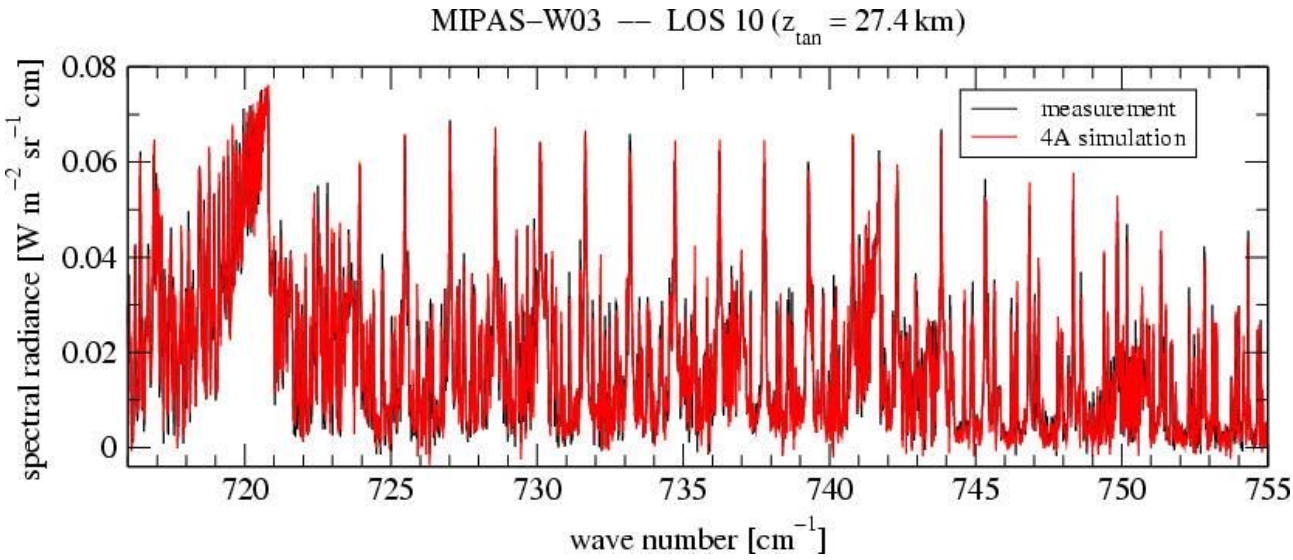


Figure 52: 4A/OP-limb simulation (red) over MIPAS level 1b spectrum (black). LOS number 10. Spectral domain: 716-755 cm^{-1}.

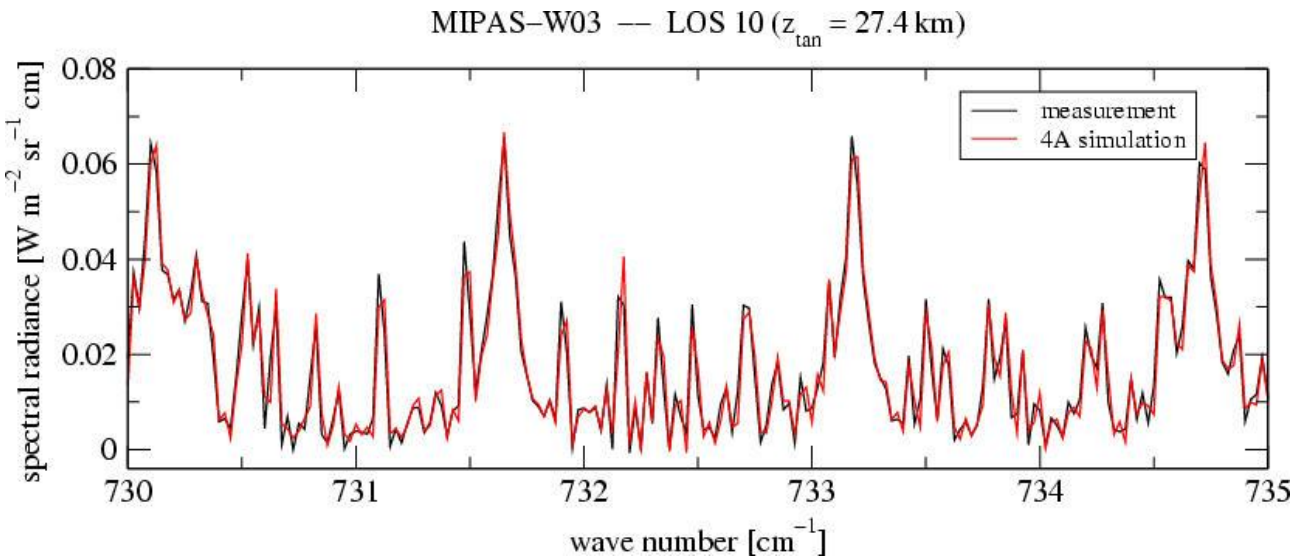


Figure 53: 4A/OP-limb simulation (red) over MIPAS level 1b spectrum (black). LOS number 10. Spectral domain: 730-735 cm^{-1} (zoom of upper figure).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

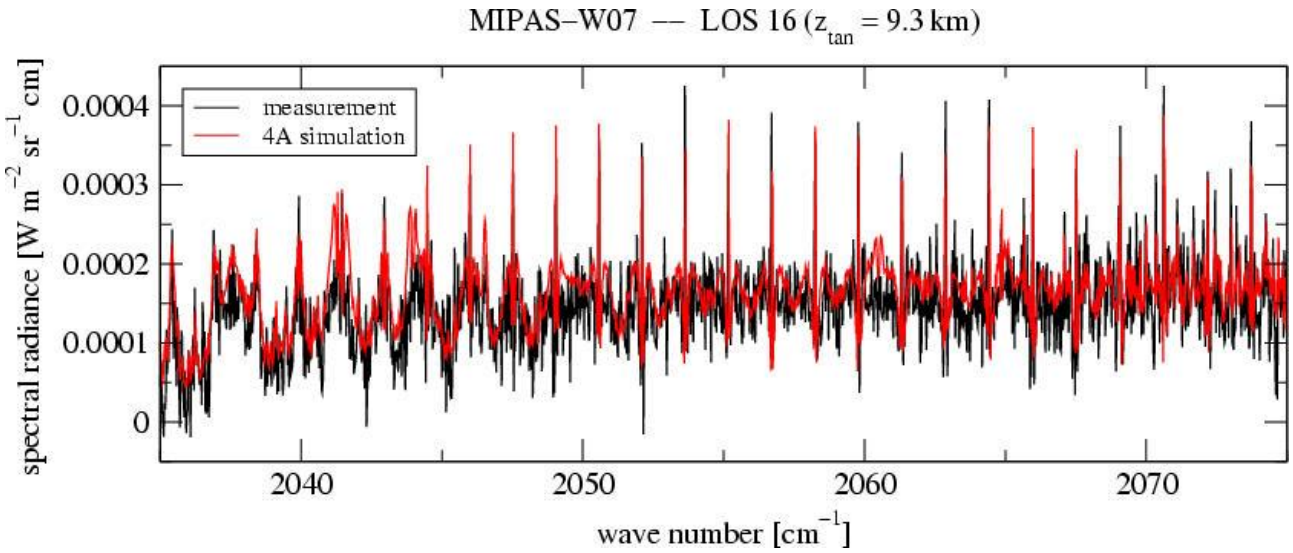


Figure 54: 4A/OP-limb simulation (red) over MIPAS level 1b spectrum (black). LOS number 16. Spectral domain: 2035-2075 cm^{-1}.

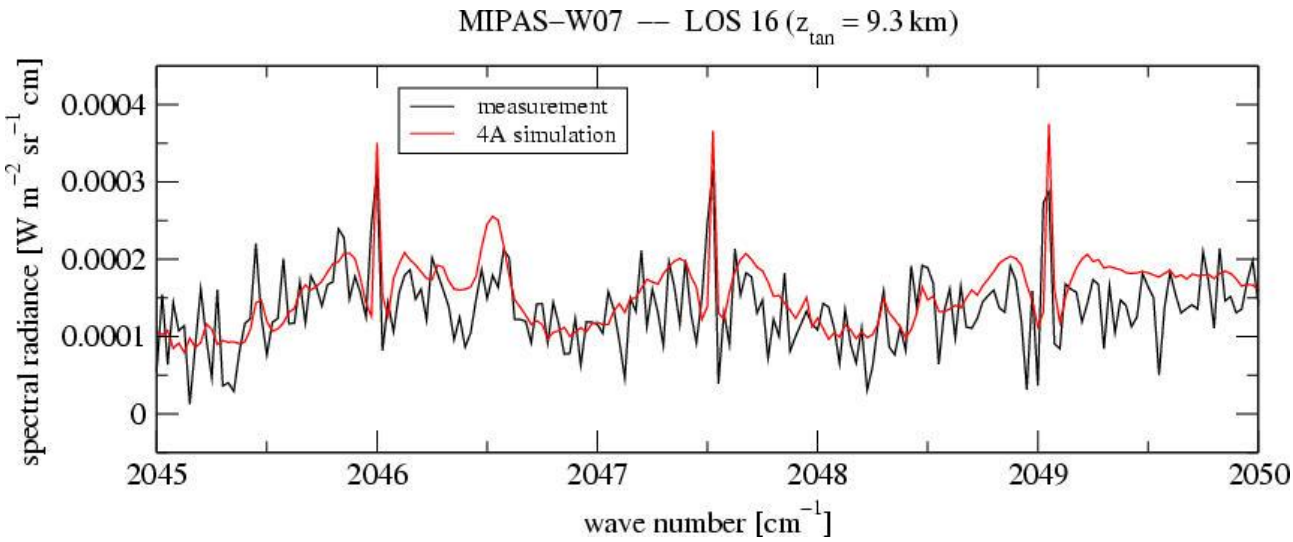


Figure 55: 4A/OP-limb simulation (red) over MIPAS level 1b spectrum (black). LOS number 16. Spectral domain: 2045-2050 cm^{-1} (zoom of upper figure).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

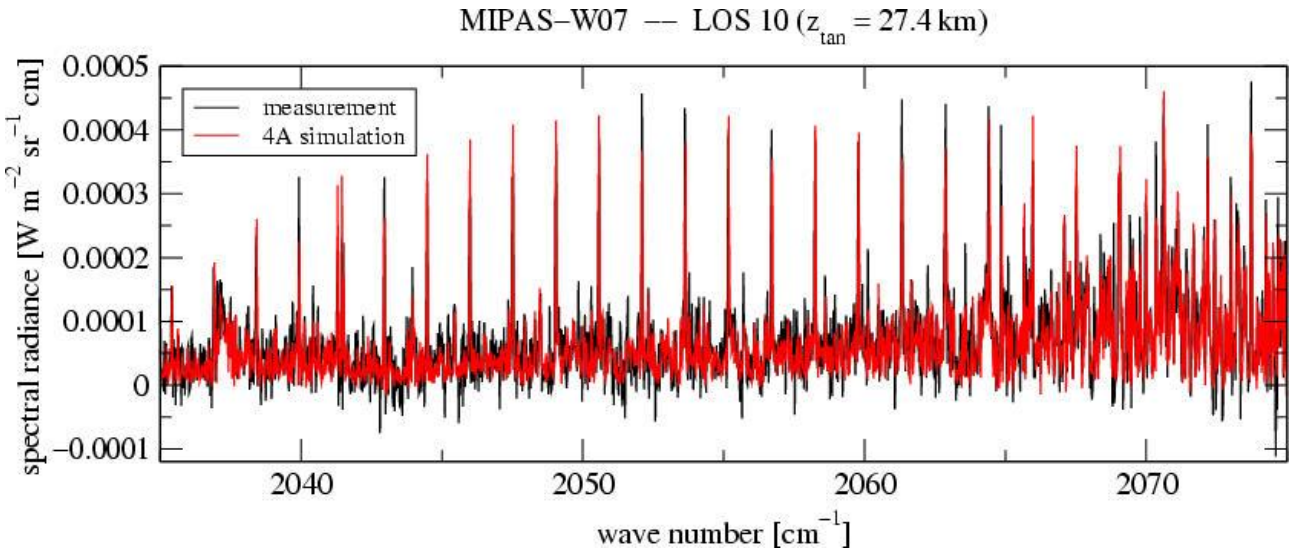


Figure 56: 4A/OP-limb simulation (red) over MIPAS level 1b spectrum (black). LOS number 10. Spectral domain: 716-755 cm^{-1}.

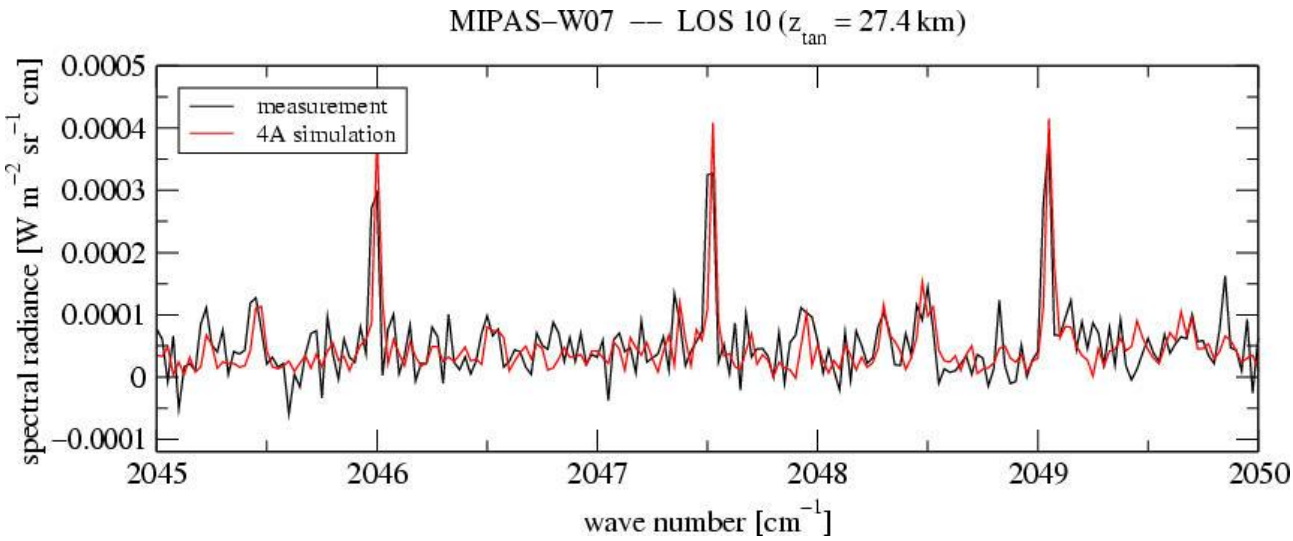


Figure 57: 4A/OP-limb simulation (red) over MIPAS level 1b spectrum (black). LOS number 10. Spectral domain: 2045-2050 cm^{-1} (zoom of upper figure).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

2. Appendix B.: Spectroscopic database GEISA

See [Ref. 5] and [Ref. 6].

The identification numbers of the molecules according to GEISA is given by:

H_2O 1	CO_2 2	O_3 3	N_2O 4	CO 5	CH_4 6	O_2 7	NO 8	SO_2 9	NO_2 10
NH_3 11	PH_3 12	HNO_3 13	OH 14	HF 15	HCl 16	HBr 17	HI 18	ClO 19	OCS 20
H_2CO 21	C_2H_6 22	CH_3D 23	C_2H_2 24	C_2H_4 25	GeH_4 26	HCN 27	C_3H_8 28	C_2N_2 29	C_4H_2 30
HC_3N 31	HOCl 32	N_2 33	CH_3Cl 34	H_2O_2 35	H_2S 36	HCOOH 37	COF_2 38	SF_6 39	C_3H_6 40
HO_2 41	$ClONO_2$ 42								

Table 40: Molecule identification according to GEISA.

The identification numbers of the CFCS is given by:

CFC11 61	CFC12 62	CCl4 63							

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

3. Appendix C.: Test cases

A script file `CasTest` (in directory `model/srclib`) in order to run non-regression tests has been implemented from the existing tests of the LMD completed with other cases. A brief description of the 4A simulation tests is described after the file `CasTest`.

```
#!/bin/csh -f

# CasTest --
#
#
#           Project : 4A-OP
#           Version : $Revision: 1.16 $
#     Creation date : December 2001
# Last modification : $Date: 2008/12/02 13:13:59 $
#
# (c) CNES - LMD (CNRS/ENS) - NOVELTIS
#
# ---
#
#
#
# Non regression tests provided by LMD and completed by Noveltis
# command line : csh CasTest (from directory srclib)
# Test output is stored in CasTest.log .

set DateExe = `date`
echo " 4a test running : $DateExe" > CasTest.log


# Thermodynamic profiles

echo
echo ' Atmospheric profile building test'
echo ' --------------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' Atmospheric profile building test'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../datatm/atm4atest.ddb
echo 'make -f makefile runatmformat ATM=test'
make -f makefile runatmformat ATM=test  >> CasTest.log
echo
echo 'Atmospheric profile building test validation'
echo '--------------------------------------------'
echo 'diff ../datatm/atm4atest.lis_ref ../datatm/atm4atest.lis'
diff ../datatm/atm4atest.lis_ref ../datatm/atm4atest.lis
if ( $status == 0 ) then
   echo ' Atmospheric profile building validation: Test OK'
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
else
   echo ' Atmospheric profile building validation: Test failed'
   exit
endif
echo '-------------------------------------'


# Instrument function tests

echo
echo
echo 'Instrument function building test : Test 1'
echo '-----------------------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' Instrument function building test : Test 1'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../isrf/isrftest.ddb
echo 'make -f makefile runifct INS=test'
make -f makefile runifct INS=test >> CasTest.log
echo
echo 'Instrument function building test validation : Test 1 '
echo '-----------------------------------------------------'
echo 'diff ../outputascii/isrftest.plt ../outputascii/isrftest.plt_ref'
diff ../outputascii/isrftest.plt ../outputascii/isrftest.plt_ref
if ( $status == 0 ) then
   echo ' Instrument function building validation: Test 1 OK'
else
   echo ' Instrument function building validation: Test 1 failed'
   exit
endif
echo '---------------------------------------'

echo
echo
echo 'Instrument function building test  : Test 2'
echo '-----------------------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' Instrument function building test : Test 2'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../isrf/isrftest2.ddb
echo 'make -f makefile runifct INS=test2'
make -f makefile runifct INS=test2 >> CasTest.log
echo
echo 'Instrument function building test validation : Test 2'
echo '-----------------------------------------------------'
echo 'diff ../outputascii/isrftest2.plt ../outputascii/isrftest2.plt_ref'
diff ../outputascii/isrftest2.plt ../outputascii/isrftest2.plt_ref
if ( $status == 0 ) then
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
   echo ' Instrument function building validation: Test 2 OK'
else
   echo ' Instrument function building validation: Test 2 failed'
   exit
endif
echo '---------------------------------------'


# Test1

echo
echo
echo ' 4A simulation : Test 1'
echo '-----------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 1'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spi4a0001testtest1inf.ddb
echo make -f makefile runlirespi4a NUMIN=719 NUMAX=721 RSTR=test1 ATM=test
make -f makefile runlirespi4a NUMIN=719 NUMAX=721 RSTR=test1 ATM=test >>
CasTest.log

# Verification
set file1 = ../outputascii/spi4a0001testtest1infb1.plt
set file2 = ../outputascii/spi4a0001testtest1infb1.plt_ref
echo
echo '4A simulation validation : Test 1 '
echo '----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test2

echo
echo
echo '4A simulation : Test 2'
echo '----------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 2'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spi4a0001testtest2inf.ddb
make -f makefile runlirespi4a NUMIN=719 NUMAX=721 RSTR=test2 INS=inf ATM=test >>
CasTest.log
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
echo make -f makefile runlirespi4a NUMIN=719 NUMAX=721 RSTR=test2 INS=inf
ATM=test

# Verification
set file1 = ../outputascii/spi4a0001testtest2infb1.plt
set file2 = ../outputascii/spi4a0001testtest2infb1.plt_ref
echo
echo '4A simulation validation : Test 2 '
echo '----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test3

echo
echo
echo 'Simulation 4A : Test 3'
echo '----------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 3'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spc4a0001testtest3test.ddb
echo make -f makefile runlirespc4a NUMIN=700 NUMAX=1000 RSTR=test3 INS=test
ATM=test
make -f makefile runlirespc4a NUMIN=700 NUMAX=1000 RSTR=test3 INS=test ATM=test
>> CasTest.log
echo
# Verification
set file1 = ../outputascii/spc4a0001testtest3testb1.plt
set file2 = ../outputascii/spc4a0001testtest3testb1.plt_ref
echo '4A simulation validation  : Test 3 '
echo '----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test4

echo
echo
echo '4A Simulation : Test 4'
echo '----------------------'
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 4'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spc4a0001testtest4test.ddb
echo make -f makefile runlirespc4a NUMIN=700 NUMAX=1000 RSTR=test4 INS=test
ATM=test
make -f makefile runlirespc4a NUMIN=700 NUMAX=1000 RSTR=test4 INS=test ATM=test
>> CasTest.log

# Verification
set file1 = ../outputascii/spc4a0001testtest4testb1.plt
set file2 = ../outputascii/spc4a0001testtest4testb1.plt_ref
echo
echo '4A simulation validation  : Test 4 '
echo '-----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test5

echo
echo
echo 'Simulation 4A : Test 5'
echo '----------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 5'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spc4a0002testtest5test.ddb
echo 'make -f makefile runlirespc4a NUMIN=700 NUMAX=1000 RSTR=test5 ATMPROF=0002
INS=test ATM=test'
make -f makefile runlirespc4a NUMIN=700 NUMAX=1000 RSTR=test5 ATMPROF=0002
INS=test ATM=test >> CasTest.log

# Validation
echo
echo '4A simulation : Test 5 '
echo '-----------------------'
set file1 = ../outputascii/dcj4a0002testtest5testb1.plt
set file2 = ../outputascii/dcj4a0002testtest5testb1.plt_ref
echo "Comparing $file1 and $file2"
../libexe/calcStatJ <<EOF
'$file1'
'$file2'
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
2
2
40
1E-3
EOF

#echo 'diff ../outputascii/dtj4a0002testtest5testb1.plt
../outputascii/dtj4a0002testtest5testb1.plt_ref'
#diff ../outputascii/dtj4a0002testtest5testb1.plt
../outputascii/dtj4a0002testtest5testb1.plt_ref
#if ( $status == 0 ) then
#   echo '4A simulation validation : Test 5 : OK'
#else
#   echo '4A simulation validation : Test 5 : failed'
#endif
set file1 = ../outputascii/dtj4a0002testtest5testb1.plt
set file2 = ../outputascii/dtj4a0002testtest5testb1.plt_ref
echo "Comparing $file1 and $file2"
../libexe/calcStatJ <<EOF
'$file1'
'$file2'
1
2
40
1E-3
EOF

set file1 = ../outputascii/dto4a0002testtest5testb1.plt
set file2 = ../outputascii/dto4a0002testtest5testb1.plt_ref
echo "Comparing $file1 and $file2"
../libexe/calcStatJ <<EOF
'$file1'
'$file2'
1
2
40
1E-3
EOF

set file1 = ../outputascii/spc4a0002testtest5testb1.plt
set file2 = ../outputascii/spc4a0002testtest5testb1.plt_ref
echo "Comparing $file1 and $file2"
../libexe/calcStat <<EOF
'$file1'
'$file2'
1E-3
EOF


# Test6

echo
echo
echo 'Simulation 4A : Test 6 '
echo '-----------------------'
echo
'*****************************************************************************' >>
CasTest.log
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
echo ' 4A simulation : Test 6'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spc4a0001testtest6test2.ddb
echo 'make -f makefile runlirespc4a NUMIN=700 NUMAX=2600 RSTR=test6 INS=test2
ATM=test'
make -f makefile runlirespc4a NUMIN=700 NUMAX=2600 RSTR=test6 INS=test2 ATM=test
>> CasTest.log

# Verification
set file1 = ../outputascii/spc4a0001testtest6test2b1.plt
set file2 = ../outputascii/spc4a0001testtest6test2b1.plt_ref
echo
echo '4A simulation validation  : Test 6 '
echo '-----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test7

echo
echo
echo 'Simulation 4A : Test 7 '
echo '------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 7'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spc4a0001testtest7test2.ddb
echo 'make -f makefile runlirespc4a NUMIN=700 NUMAX=2600 RSTR=test7 INS=test2
ATM=test'
make -f makefile runlirespc4a NUMIN=700 NUMAX=2600 RSTR=test7 INS=test2 ATM=test
>> CasTest.log

# Verification
set file1 = ../outputascii/spc4a0001testtest7test2b1.plt
set file2 = ../outputascii/spc4a0001testtest7test2b1.plt_ref
echo
echo '4A simulation validation  : Test 7 '
echo '-----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
# Test8

echo
echo
echo 'Simulation 4A : Test 8 '
echo '------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 8'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../isrf/isrfiasi1c.ddb
echo 'make -f makefile runifct INS=iasi1c'
make -f makefile runifct INS=iasi1c >> CasTest.log

rm -f ../outputbin/spc4a0001testtest8iasi1c.ddb
echo 'make -f makefile runlirespc4a NUMIN=645 NUMAX=2760 RSTR=test8 INS=iasi1c
ATM=test'
make -f makefile runlirespc4a NUMIN=645 NUMAX=2760 RSTR=test8 INS=iasi1c
ATM=test >> CasTest.log

# Verification
set file1 = ../outputascii/spc4a0001testtest8iasi1cb1.plt
set file2 = ../outputascii/spc4a0001testtest8iasi1cb1.plt_ref
echo
echo '4A simulation validation : Test 8 '
echo '----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test9

echo
echo
echo 'Simulation 4A : Test 9 '
echo '------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 9'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../isrf/isrfiasi1c.ddb
echo 'make -f makefile runifct INS=iasi1c'
make -f makefile runifct INS=iasi1c >> CasTest.log

rm -f ../outputbin/spc4a0001testtest9iasi1c.ddb
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
echo 'make -f makefile runlirespc4a NUMIN=1700 NUMAX=2760 RSTR=test9 INS=iasi1c
ATM=test'
make -f makefile runlirespc4a NUMIN=1700 NUMAX=2760 RSTR=test9 INS=iasi1c
ATM=test >> CasTest.log

# Verification
set file1 = ../outputascii/spc4a0001testtest9iasi1cb1.plt
set file2 = ../outputascii/spc4a0001testtest9iasi1cb1.plt_ref
echo
echo '4A simulation validation : Test 9 '
echo '----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test10

echo
echo
echo 'Simulation 4A : Test 10 '
echo '------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 10'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spi4a0001testtest10inf.ddb
echo make -f makefile runlirespi4a NUMIN=2300 NUMAX=2500 RSTR=test10 ATM=test
make -f makefile runlirespi4a NUMIN=2300 NUMAX=2500 RSTR=test10 ATM=test >>
CasTest.log

# Verification
set file1 = ../outputascii/spi4a0001testtest10infb1.plt
set file2 = ../outputascii/spi4a0001testtest10infb1.plt_ref
echo
echo '4A simulation validation : Test 10 '
echo '-----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test11

echo
echo
echo 'Simulation 4A : Test 11 '
echo '------------------------'
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 11'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spi4a0001testtest11inf.ddb
echo make -f makefile runlirespi4a NUMIN=2045 NUMAX=2050 RSTR=test11 ATM=test
make -f makefile runlirespi4a NUMIN=2045 NUMAX=2050 RSTR=test11 ATM=test >>
CasTest.log

# Verification
set file1 = ../outputascii/spi4a0001testtest11infb1.plt
set file2 = ../outputascii/spi4a0001testtest11infb1.plt_ref
echo
echo '4A simulation validation : Test 11 '
echo '-----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test12

echo
echo
echo 'Simulation 4A : Test 12 '
echo '------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 12'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spi4a0001testtest12inf.ddb
echo make -f makefile runlirespi4a NUMIN=2045 NUMAX=2050 RSTR=test12 ATM=test
make -f makefile runlirespi4a NUMIN=2045 NUMAX=2050 RSTR=test12 ATM=test >>
CasTest.log

# Verification
set file1 = ../outputascii/spi4a0001testtest12infb1.plt
set file2 = ../outputascii/spi4a0001testtest12infb1.plt_ref
echo
echo '4A simulation validation : Test 12 '
echo '-----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
# Test13

echo
echo
echo 'Simulation 4A : Test 13 '
echo '------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 13'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spc4a0001testtest13DIS_0test.ddb
echo make -f makefile runlirespc4a NUMIN=700 NUMAX=800 RSTR=test13 RSCA=DIS_0
INS=test ATM=test
make -f makefile runlirespc4a NUMIN=700 NUMAX=800 RSTR=test13 RSCA=DIS_0
INS=test ATM=test >> CasTest.log

# Verification
set file1 = ../outputascii/spc4a0001testtest13DIS_0testb1.plt
set file2 = ../outputascii/spc4a0001testtest13DIS_0testb1.plt_ref
echo
echo '4A simulation validation : Test 13 '
echo '-----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF


# Test14

echo
echo
echo 'Simulation 4A : Test 14 '
echo '------------------------'
echo
'*************************************************************************' >>
CasTest.log
echo ' 4A simulation : Test 14'
>> CasTest.log
echo
'*************************************************************************' >>
CasTest.log
rm -f ../outputbin/spc4a0001satigr_v4.0_moyclastest14DIS_0_5v3liasi1c.ddb
echo make -f makefile runlirespc4a NUMIN=820 NUMAX=825 RSTR=test14
RSCA=DIS_0_5v3l INS=iasi1c ATM=satigr_v4.0_moyclas
make -f makefile runlirespc4a NUMIN=820 NUMAX=825 RSTR=test14 RSCA=DIS_0_5v3l
INS=iasi1c ATM=satigr_v4.0_moyclas >> CasTest.log

# Verification
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
set file1 =
../outputascii/spc4a0001satigr_v4.0_moyclastest14DIS_0_5v3liasi1cb1.plt
set file2 =
../outputascii/spc4a0001satigr_v4.0_moyclastest14DIS_0_5v3liasi1cb1.plt_ref
echo
echo '4A simulation validation : Test 14 '
echo '-----------------------------------'
echo "Comparing $file1 and $file2"
../libexe/calcStat << EOF
'$file1'
'$file2'
1E-3
EOF
```

3.1. Description of the 4A simulation tests available in the file CasTest

4A/OP variables / Test identification	INS	ATM	ATMPROF	RSTR/ RSCA	TRAJET / GEOMID	FORIG-FSTOP (cm^{-1})	SECTTASUN (degree) EMDOWN / EMUP	JACOB	RESPEC	Number of molecules involved
Test1	inf	test	0001	test1 / -	UP / EVIEW	719-721	- / 100	NO	0	6
Test2	inf	test	0001	test2/ -	DOWN / EVIEW	719-721	- / 100	NO	0	6
Test3	test	test	0001	test3/ -	UP / EVIEW	700-1000	- / 100	NO	0	6
Test4	test	test	0001	test4/ -	DOWNUP / EVIEW	700-1000	90 / 98	NO	0	6
Test5	test	test	0002	test5/ -	UP / EVIEW	700-1000	- / 100	YES	0.1	6
Test6	test2	test	0001	test6/ -	UP / EVIEW	700-2600	- / 100	NO	0	6
Test7	test2	test	0001	test7/ -	UPDOWN / EVIEW	700-2600	90 / 97	NO	0	6
Test8	iasi1c	test	0001	test8/ -	UP / EVIEW	645-2760	90 / 100	NO	0	8
Test9	iasi1c	test	0001	test9/ -	DOWNUP / EVIEW	1700-2760	0 / 103	NO	0	8
Test10	inf	test	0001	test10/ -	UP / EVIEW	2300-2500	- / 100	NO	0.01	8
Test11	inf	test	0001	test11/ -	DOWNUP/LIMB (LOS 10 km)	2045-2050	90 / 100	NO	0.01	8
Test12	inf	test	0001	test12/ -	DOWNUP/LIMB (LOS 30 km)	2045-2050	90 / 100	NO	0.01	8

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

4A/OP variables / Test identification	INS	ATM	ATMPROF	RSTR/ RSCA	TRAJET / GEOMID	FORIG-FSTOP (cm^{-1})	SECTTASUN (degree) / EMDOWN/ EMUP	JACOB	RESPEC	Number of molecules involved
Test13	test	test	0001	test13/ DIS_0	UP / EVIEW	700-800	90 / 100	NO	0	6
Test14	iasi1c	test	0001	test14/ DIS_0_5v3l	DOWNUP / EVIEW	820-825	90 / 098	NO	0.01	9

Table 41: description of the 4A simulation tests

Only significant variables have been extracted for the description of test cases. These variables and others that vary are in bold in the following sub-sections. The first sub-section describes global variables that are initialized in the run script and passed to the execution makefile. The second sub-section describes the variables defined in the model parameter file `para4a$(RSTR).dtp` (`model/input`), where $RSTR represents the test identification 1 to 14.

For all tests, some input user-defined parameters vary and others are fixed. It is indicated in the following description.

3.1.1. Simulation definition parameters: global variables

Fixed:

JOB = « 4a »

ATM = « test »

RESOL = « hr03 »; this corresponds to the version 2003 of the GEISA spectroscopic databank

UNIT = « 4 »

TREF = « 280 » K

TEBR = « 0.28 » K

WCLASS = « 50 » cm^{-1}

WNMIN= « 645 » cm^{-1}

WNMAX = « 2760 » cm^{-1}

Variable:

NUMIN varies from FORIG to FSTOP (see following section)

NUMAX varies from FORIG to FSTOP

ATMPROF = « 0001 » (tropical atmosphere) or ATMPROF = « 0002 » (polar atmosphere)

RSTR varies from « test1 » to « test14 »

RSCA = « » (empty string: no scattering) or RSCA = « DIS_0 » (no pre-defined aerosol model) or RSCA = « DIS_0_5v3l » (pre-defined aerosol model).

INS = « inf » ("pseudo-infinite" spectrum) or INS = « iasi1c » (spectrum convolved with the instrument function of IASI level 1c) or INS = « test », « test2 » (test instrument functions)

CASE = « »

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

3.1.2. Simulation definition parameters: run characterization parameters

Fourteen files (`para4atest1.dtp`, `para4atest2.dtp`… `para4atest14.dtp`) are used. These files contain parameters that are the same for all tests and the other parameters vary:

Parameters identical for all tests:

PZDOWN = « 1013.25 » hPa (« 0 » km)

TDOWN = « 0 » K

TYP_ANG = « ANGLE »

SECTTA2 = « 55 » degree

ZATMBOT = « 0 » km

ESDIST = « 1 » astronomical unit

CONTIN = « YES »

DELDS = « 15 » cm^{-1}

INTOPT = « YES »

SHIFT = « NO »

RNU = « 0 » cm^{-1}

PASCON = « 0.25 » cm^{-1}

IDSEL = « 1 » for H_2O, CO_2, O_3, N_2O, CO and CH_4 and IDSEL = « 0 » for the other molecules except for O_2 and N_2.

ROCOEF = « 1 » pour H_2O, CO_2, O_3, N_2O, CO, CH_4, O_2, SO_2 and N_2 and ROCOEF = « 0 » for the other molecules.

IDJAC = « 1 » for H_2O and O_3 and IDJAC = « 0 » for the other molecules.

Variable parameters:

GEOMID = « EVIEW » or « LIMB »

TRAJET = « UP », « DOWN » or « DOWNUP »

PZUP = « 0.05 » hPa or « 525 » hPa or « 100 » km

EMUP = « 100 » or « 97 »

TUP = « 4 » K (for TRAJET = « UP » or « DOWNUP ») or « 265 » (for TRAJET = « UPDOWN »)

PZOBS = « 0.05 » hPa / « 100 » km (for TRAJET = « UP » or « DOWNUP ») or « 1013.25 » hPa / « 0 » km (for TRAJET = « DOWN »)

EMDOWN = « 100 » or « 103 »

ZGTAN = « 0 » km or « 10 » km or « 30 » km (only for GEOMID = « EVIEW »)

SECTTA1 = « 0 » or « 45 » degrees

SECTTASUN = « 0 » or « 90 » degrees. The solar contribution is suppressed (night case) by specifying a solar zenith angle outside the range [0°, 90°[(or a secant value of this angle inferior to 1).

REFRA = « YES » or « NO »

RESPEC = « 0 » or « 0.1 » or « 0.01 ».

FORIG varies from « 645 » cm^{-1} to « 2760 » cm^{-1}

FSTOP varies from « 645 » cm^{-1} to « 2760 » cm^{-1}

CONV = « YES » or « NO »

HOLE = « NO » or « YES » (used for JACOB = « YES »)

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

JACOB = « YES » or « NO »

FPOID = « YES » or « NO »

IDSEL = « 1 » or « 0 » for O_2 and N_2.

3.1.3. Atmospheric profiles

Only one file `atm4atest.ddb` is used where two profiles (ATMPROF) are available. For these test cases, two atmospheres (tropical and polar) are defined through their vertical profile in temperature and mixing ratio for H_2O and O_3. The corresponding lines are displayed on Figure 58, Figure 59 and Figure 60.

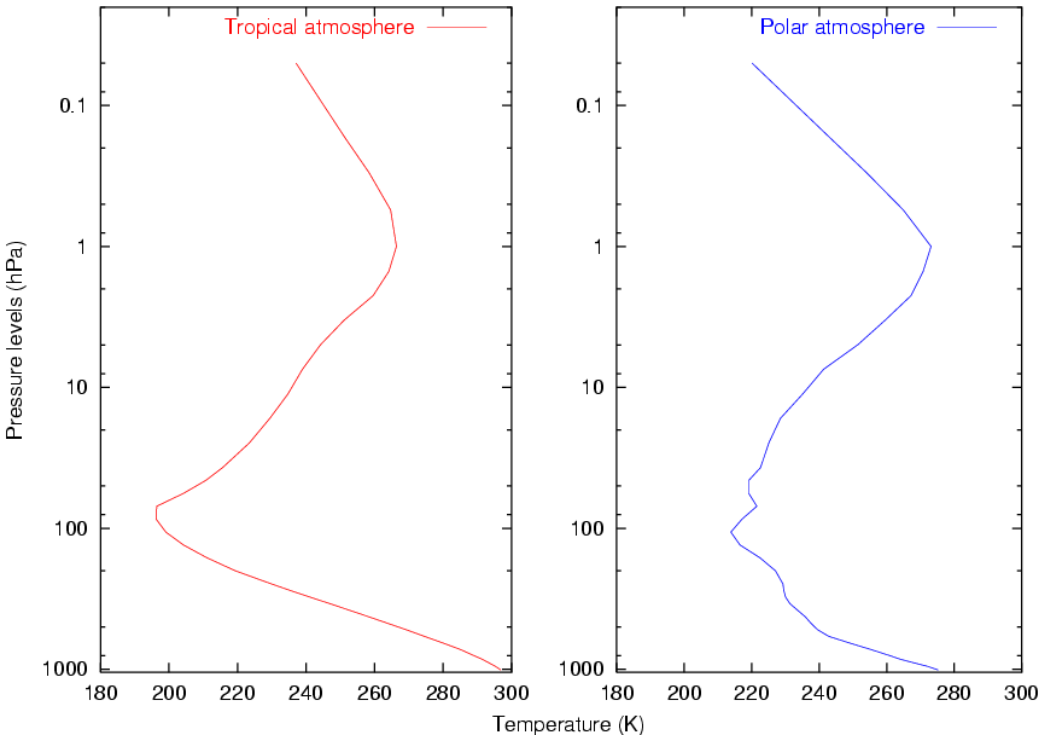


Figure 58: Vertical profiles in temperature used for 4A/OP test cases (left side: ATMPROF=0001 ; right side: ATMPROF=0002).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

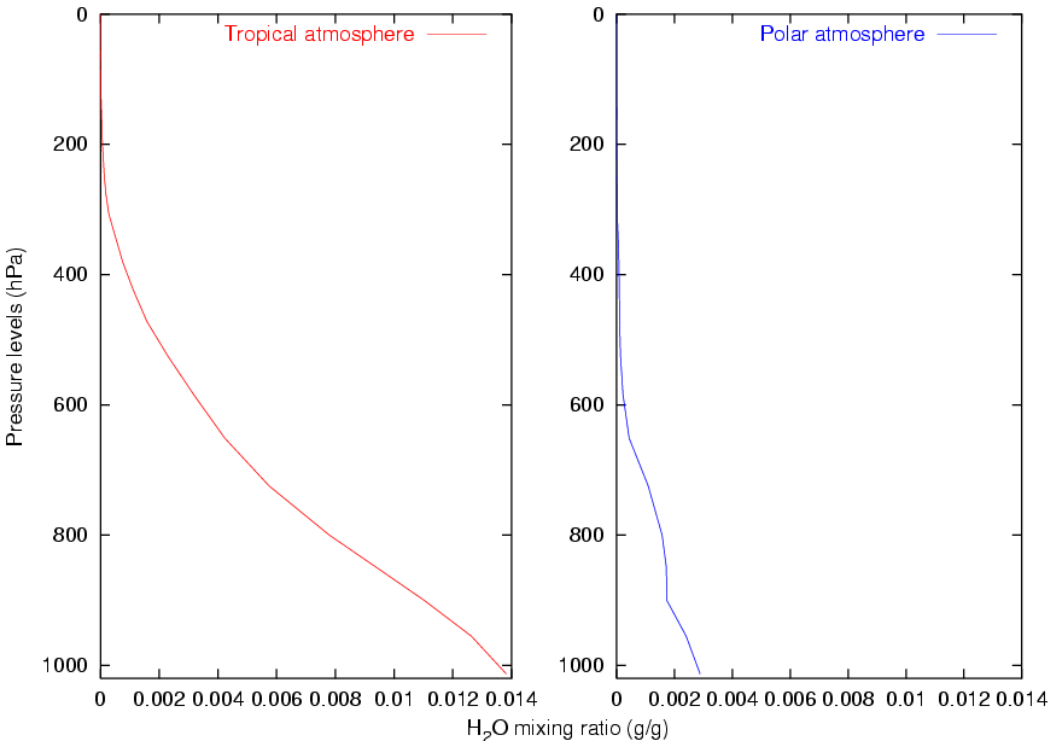


Figure 59: Vertical profiles in mixing ratio for H_2O used for 4A/OP test cases.

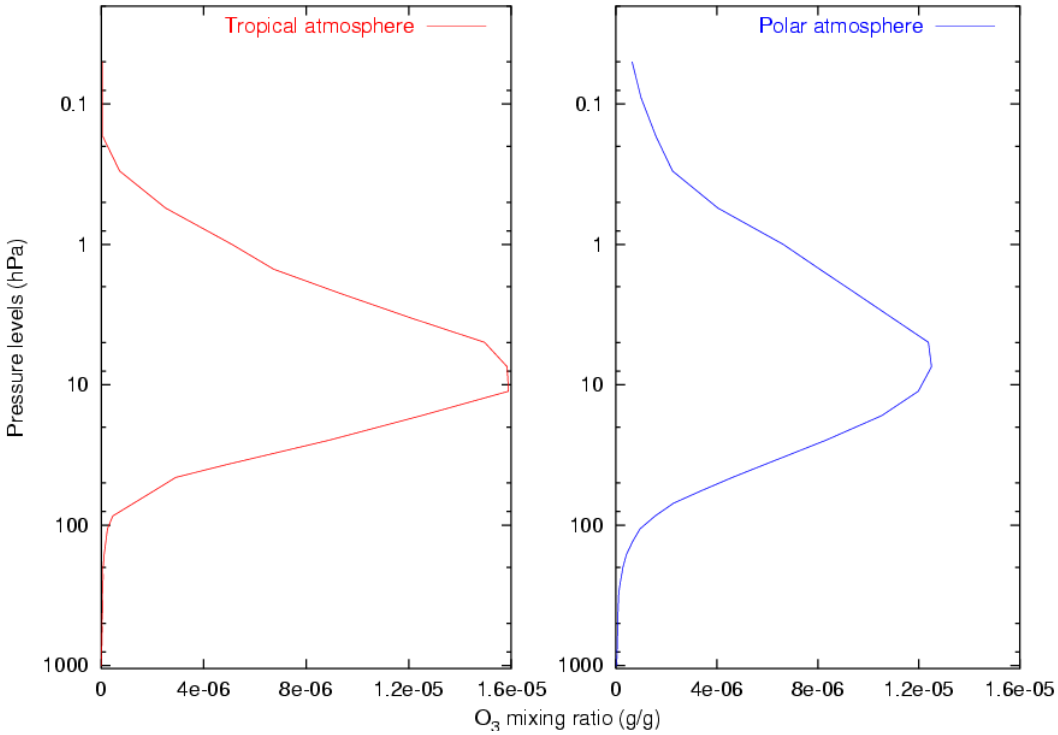


Figure 60: Same as Figure 59 for O_3.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

3.1.4. Surface spectral emissivity

Files `spemis100.dat` and `spemis103.dat` are used. The file `spemis100.dat` corresponds to a constant emissivity equal to 1 (see also appendix, Part III 1.2). The file `spemis103.dat` corresponds to an arid bare soil emissivity that varies with the wave number (Figure 61).

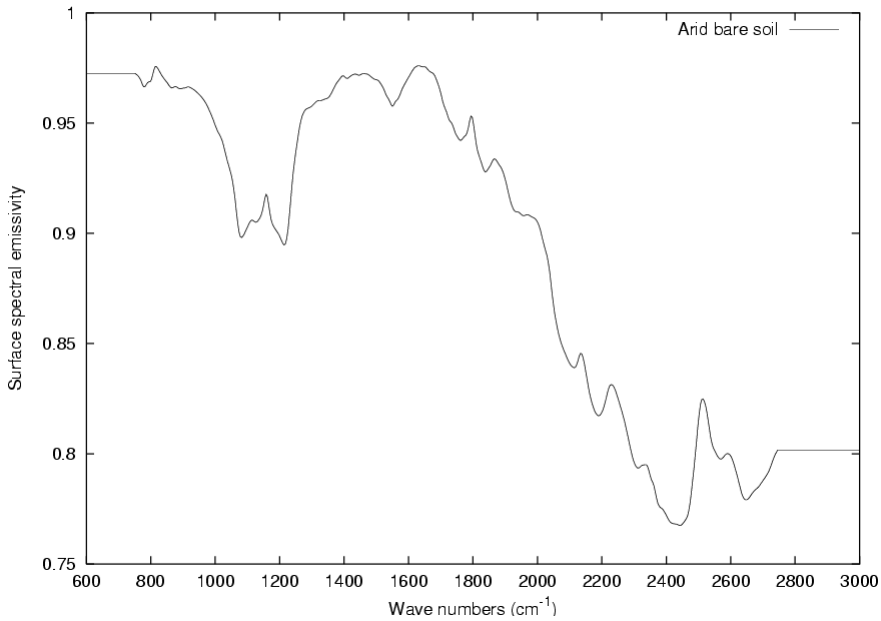


Figure 61: Surface spectral emissivity used for 4A/OP test cases.

3.1.5. Instrument functions

Three instrument functions are used for running test cases: Two test functions (files `isrftest.ddb` and `isrftest2.ddb`) and the IASI level 1c function (file `isrfiasi1c.ddb`). They are represented by Figure 62 to Figure 64.

The first test function corresponds to an equivalent radiometer instrument (itypeconv=1). Two chanels are defined (rectangle function): one at 750 cm^{-1} (CO_2 absorption) and one at 900 cm^{-1} (window).

The second test function corresponds to an equivalent interferometer function (itypeconv=2). This instrument function is defined at 750 and at 2550 cm^{-1}. For interferometer instruments, the function can vary with wave numbers (it is not the case for this second test function). That is the reason why it is defined at minimum two points. Outside the so defined spectral range, the convolution is not possible in order to avoid bad extrapolations.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

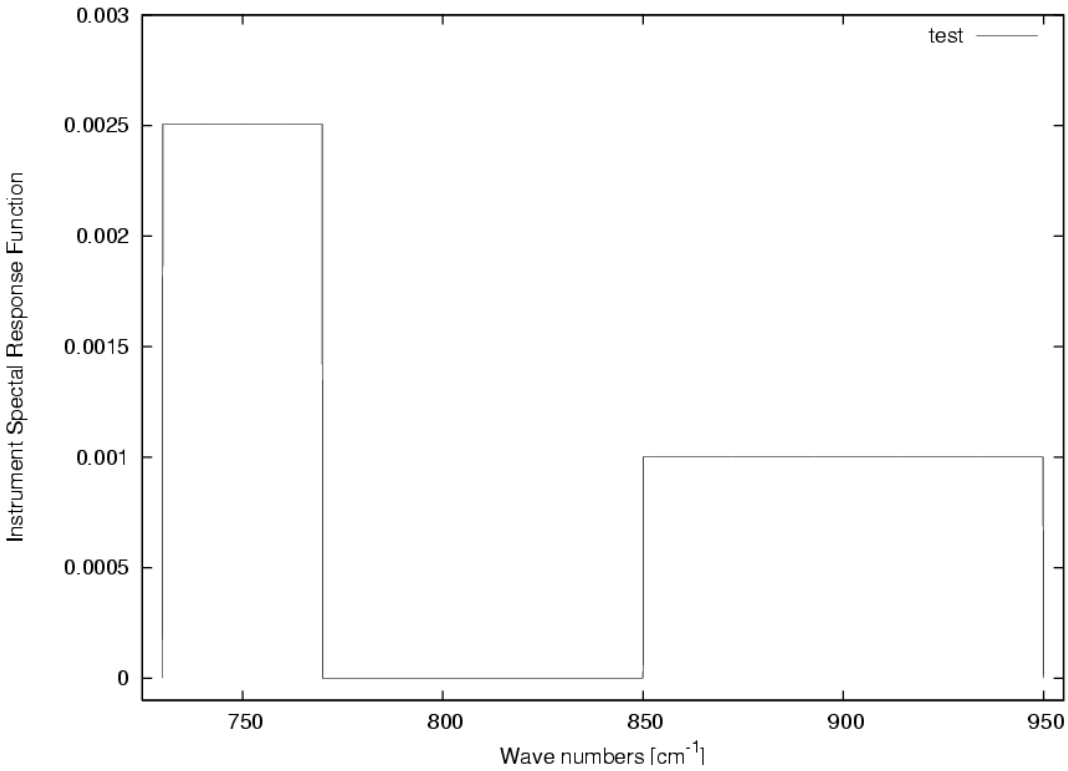


Figure 62: Instrument function "test" (radiometer equivalent)

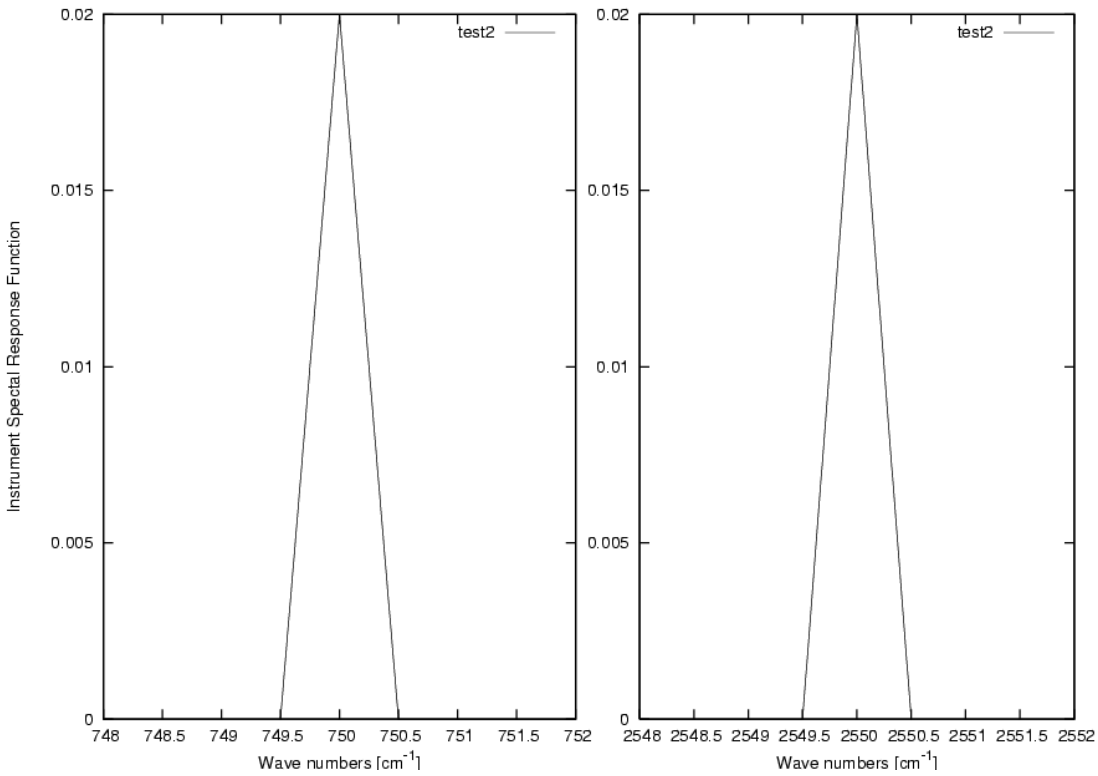


Figure 63: Instrument function "test2" (interferometer equivalent)

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

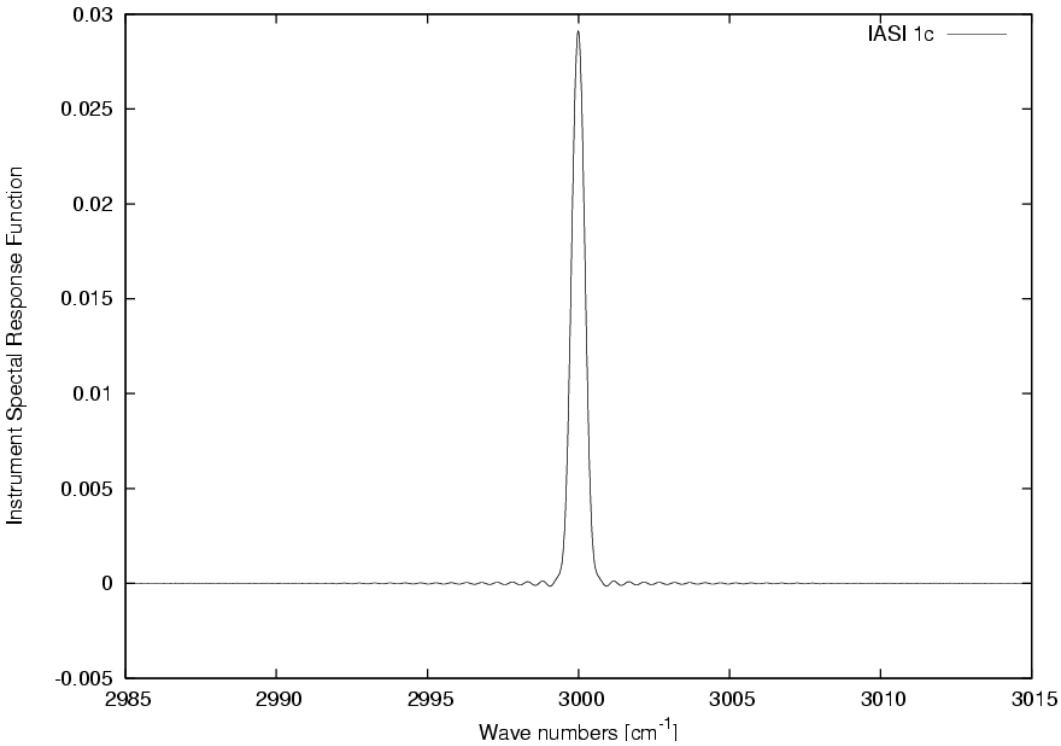


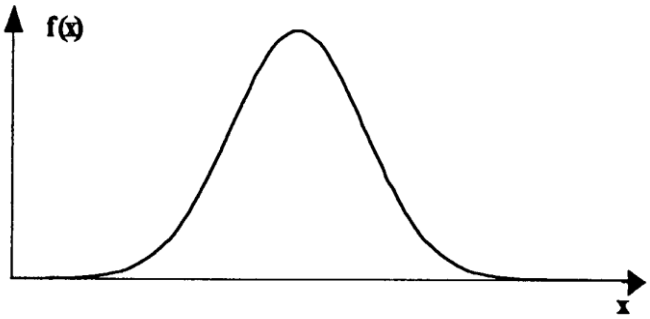
Figure 64: Instrument function "IASI 1c" (interferometer) for a given wave number.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

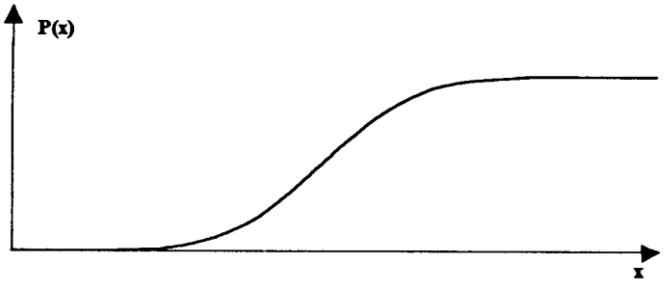
4. Appendix D.: Percentiles

A percentile is a value on a scale of one hundred that indicates the percent of a distribution that is equal to or below it.

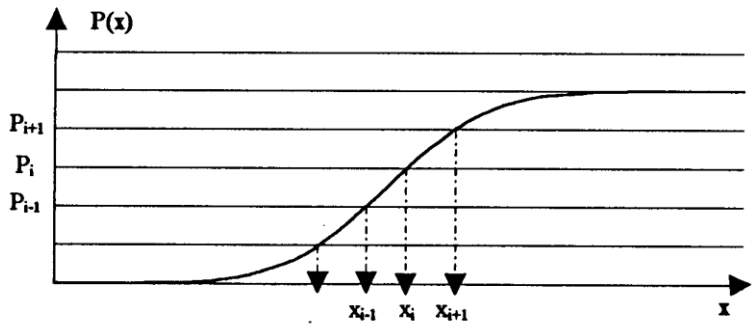
Let *f(x)* be a density function of the variable *x*:



Let $P(x) = \int_0^x f(x')dx'$ be the cumulative density:



Then this cumulative density is divided into *n* equal slices that correspond to P_i, $i \in \{1, n\}$, for which a value x_i of the variable is related:



For a division every hundredth of *P*, x_i is called a percentile. For example, at the 20th percentile, a particular value of *x* is obtained such that 20% of the total distribution of elements have values of *x* below this particular value.

An other way of describing percentiles is as follows.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

For a series of measurements Y_1, ..., Y_N, denote the data ordered in increasing order of magnitude by $Y_{[1]}$, ..., $Y_{[N]}$. These ordered data are called order statistics. If $Y_{[j]}$ is the order statistic that corresponds to the measurement Y_i, then the rank for Y_i is j; i.e.,

$$Y_{[j]} \sim Y_i \quad \Rightarrow r_i = j$$

Order statistics provide a way of estimating proportions of the data that should fall above and below a given value, called a *percentile*. The pth percentile is a value, $Y_{(p)}$, such that at most (100p)% of the measurements are less than this value and at most 100(1- p)% are greater. The 50th percentile is called the *median*.

Percentiles split a set of ordered data into hundredths. For example, 70% of the data should fall below the 70th percentile.

<u>Estimation of percentiles</u>:
Percentiles can be estimated from N measurements as follows: for the pth percentile, set $p(N+1)$ equal to $k + d$ for k an integer, and d, a fraction greater than or equal to 0 and less than 1.

1. For $0 < k < N$, $Y(p) = Y_{[k]} + d\left(Y_{[k+1]} - Y_{[k]}\right)$
2. For $k = 0$, $Y(p) = Y_{[1]}$
3. For $k = N$, $Y(p) = Y_{[N]}$

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

5. Appendix E.: On-line help of the GUI

This section is an extract of the on-line help available with the Graphical User Interface (GUI).

5.1. Overview

This documentation is the online help on the Graphical User Interface (GUI) for the 4A radiative transfer model.

Additional help is available for every parameter and is displayed when the mouse pointer stops over the parameter label as a balloon-help (a frame appears).

We place emphasis on the fact that the GUI is a layer on the model itself. It is designed to assist new users in understanding the 4A/OP model handling. With minimum knowledge, one can set, run and visualize a 4A/OP simulation with this GUI.

We refer the reader to the document "4A/OP - Reference documentation" (NOV-3049-NT-1178) for accurate information on the model itself. It contains directives to run the model without this GUI and details some aspects of the physics of the model.

This GUI allows the user to do three main operations:

1. manage sets of parameters;
2. run the model with these sets;
3. visualize results of the model.

5.2. Graphical User Interface description

This section contains specific information on this GUI. Detailed description of classic functionalities (open/save/help menus, toolbar, etc.) is intentionally left aside.

5.2.1. File menu

The GUI manages sets of parameters using files with a specific format (extension .4a). This format allows the user to keep a set of parameters between sessions of the GUI.

Some parameters take their values according to the content of some directories of the 4A/OP model program installation. To avoid problem with this parameters, be sure that the root directory for the 4A/OP model is properly specified. This can be done using the menu **File/Preferences....** The preferences are saved in a file called `.4Aguirc` in home directory of the user.

5.2.2. Run menu

The menu **Run/Run 4A** launches the 4A/OP model execution. This action requires that the current set of parameters has been saved on disk (see the **File** menu). The user can monitor the 4A/OP model computation standard output by using the menu **Run/View output log**.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

5.2.2.1.1 Note to advanced 4A/OP users

In order to run the model, the GUI creates a parameter file and a script that contains the macro definitions and the call to `make` utility that is used to actually run the model. The root name of the `.4a` file is used to name them. For example, if one creates a set of parameters that is saved under the name `test.4a`, the GUI creates the required 4A/OP parameter file called `input/para4atest.dtp` and an other file called `scripts/test` in the model installation directory (set with menu **File/Preferences**). Then `scripts/test` is executed.

5.2.3. Visualization menu

This menu allows visualization and printing of graphs of 4A/OP model products (spectra, jacobians, transmittances, atmospheric profiles and difference between two spectra). Those graphs are governed by Gnuplot scripts.

Note: You can select a new visualization range by zooming on the graph with the right mouse button. To go back to the previous range, simply type "p" on the Gnuplot window.

5.2.3.1.1 User specific plots

You can personalize the graphs produced by the GUI. To do that:

1. Copy the default Gnuplot scripts which are installed in the directory `gnuplot` of the GUI installation. Put them on your own directory. Be sure that these scripts keep their original names.

2. Edit these scripts to fit your needs using standard Gnuplot commands.

3. Modify the field **Gnuplot scripts directory** in the **Preferences** window (menu **File/Preferences**) to point at the directory of your Gnuplot scripts.

Original Gnuplot scripts contain some special codes used by the GUI which are names of variables that are substituted when the graph is actually plot. They begin with the sign `$` (dollar). For example, `$dataFile` is replaced by the name of the 4A/OP output file to plot.

The same way, you can also access to the values of the parameters. For example in a Gnuplot script, `$param_ATM` represents the **Atmospheric database indicator** entry in the **Atmosphere** section of the **General** page which is identified by the code `ATM`. The code for each parameter can be found either in the section Parameter description or, for most parameters, in their balloon-help.

5.3. Guide

This section is a step by step help in the use of this GUI from parameter definition to 4A/OP model outputs visualization. It considers that you have a minimum knowledge on the physics underlying the 4A/OP model. This guide only focuses on important parameters required to set a particular simulation.

1. Create a new parameter set. Menu **File/New**;

2. In the **General** page, choose the **Atmospheric profile indexes** you want to use. Each index corresponds to a specific type of profile in the atmospheric database. See section Atmospheric profile indexes for a description of the profiles for the default atmospheric databases;

3. In the **Spectral conditions** page, choose the spectral limits for the computation in the section **Spectral limit (radiative transfer computation)** and those that will be used only for visualization in the section **ASCII extraction spectral limits (visualization)**;

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

4. Choose the spectral sampling case: check **Pseudo-infinite** to produce a high resolution spectrum and check **Instrument** case to simulate a radiance measurement. For the last one choose the **Instrument function (ISRF)** you want to use;

5. In the **Geometric configuration** section of the **Observation configuration** page, choose the **Viewing configuration** you want to use. Each of the four possibilities corresponds to a specific configuration of the radiative transfer geometry. See section Viewing configuration for an illustration of each of them;

6. Set the upper and the lower levels;

7. In the **Molecules** page, select the molecules you want to use;

8. Save your set of parameters on a `.4a` file. Menu **File/Save** ;

9. Open the log window. Menu **Run/View output log**;

10. Run the 4A/OP model. Menu **Run/Run 4A model**;

11. Wait for the end of the simulation. The time needed depends on the kind of simulation you choose. Check the log of 4A/OP model to see if the simulation goes well;

12. Visualize your results using menu **Visualization/Spectrum in radiance**. Select the profile index (see point 2.) you want to visualize and click **View**;

13. Print your results using menu **Visualization/Spectrum in radiance**. Select the profile index (see point 2.) you want to print, click **Print** and choose the name of the graphical file (postscript, PNG or TGIF).

We refer the reader to the document "4A/OP - Reference documentation" (NOV-3049-NT-1178) for information on the model and its parameters that can not be found in this help.

5.4. Specific help for some parameters

This section gives information to complete the balloon-help description for several parameters. This helps new users to understand the meaning of tricky settings.

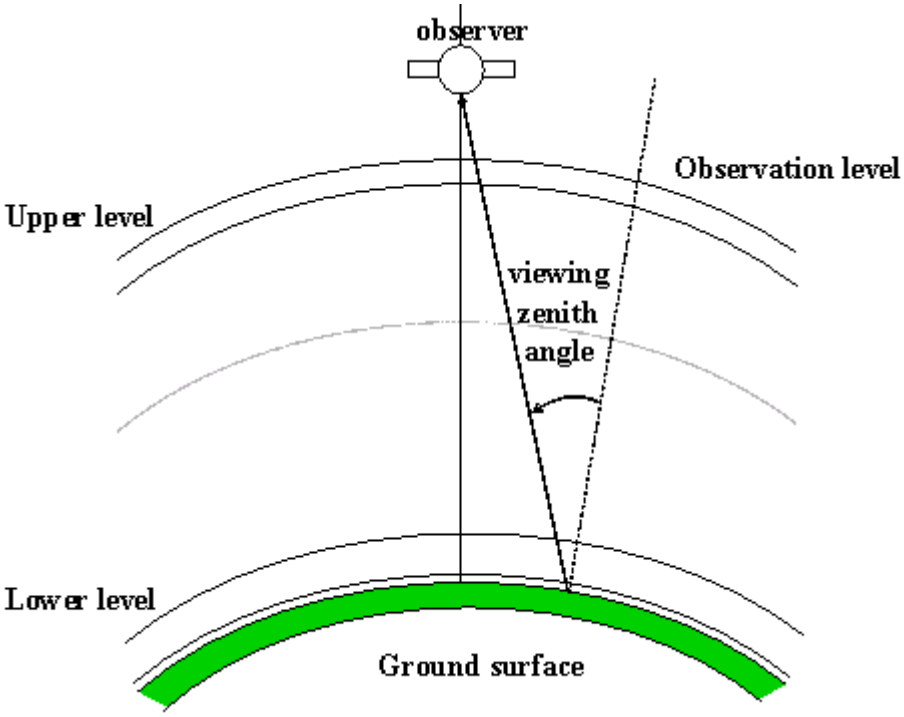
5.4.1. Viewing configuration

The parameter defining viewing configuration can be found in the **Geometric configuration** section of the **Observation configuration** page.

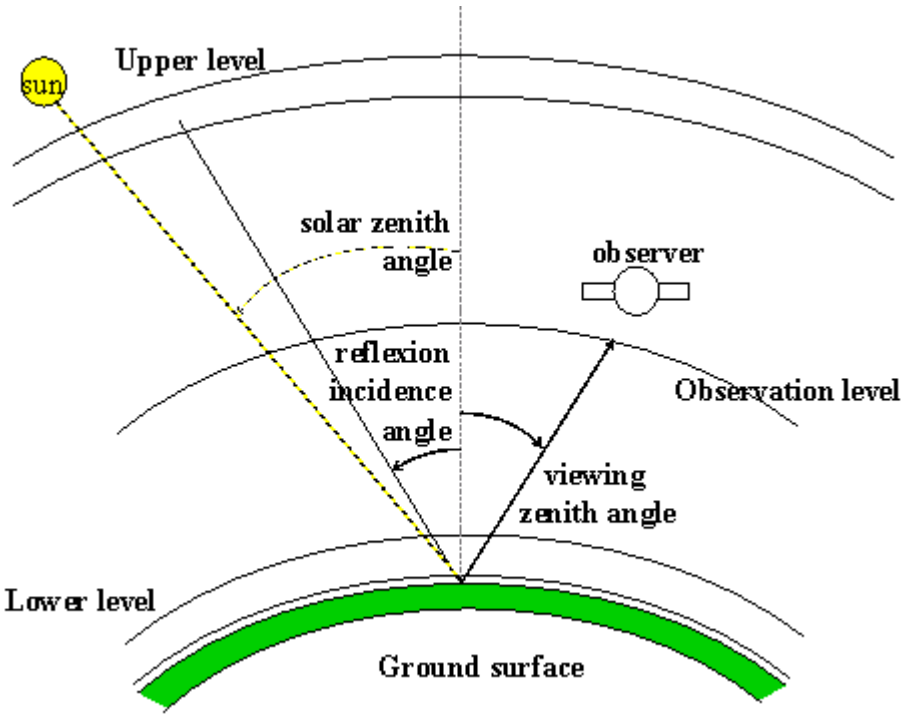
This parameter defines the configuration of the optical path for the radiative transfer computation. Its value determines the configuration of the source and the observer. It can be either UP, DOWNUP, DOWN or UPDOWN. The figures below show the configuration for each value as well as what are the upper, lower and the observation levels (see **Atmospheric vertical limits** section of the **Observation configuration** page).

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

5.4.1.1.1 UP configuration

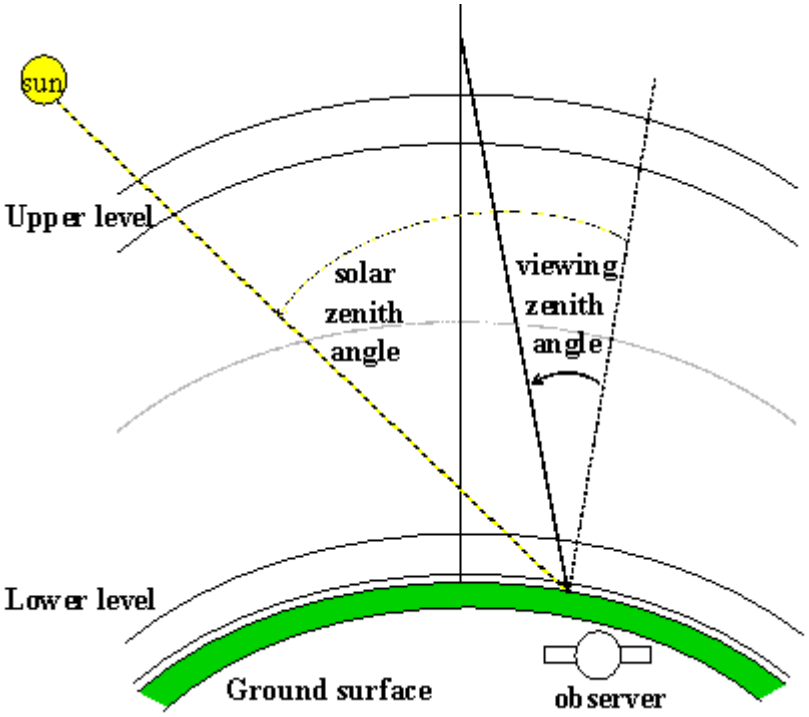


5.4.1.1.2 DOWNUP configuration

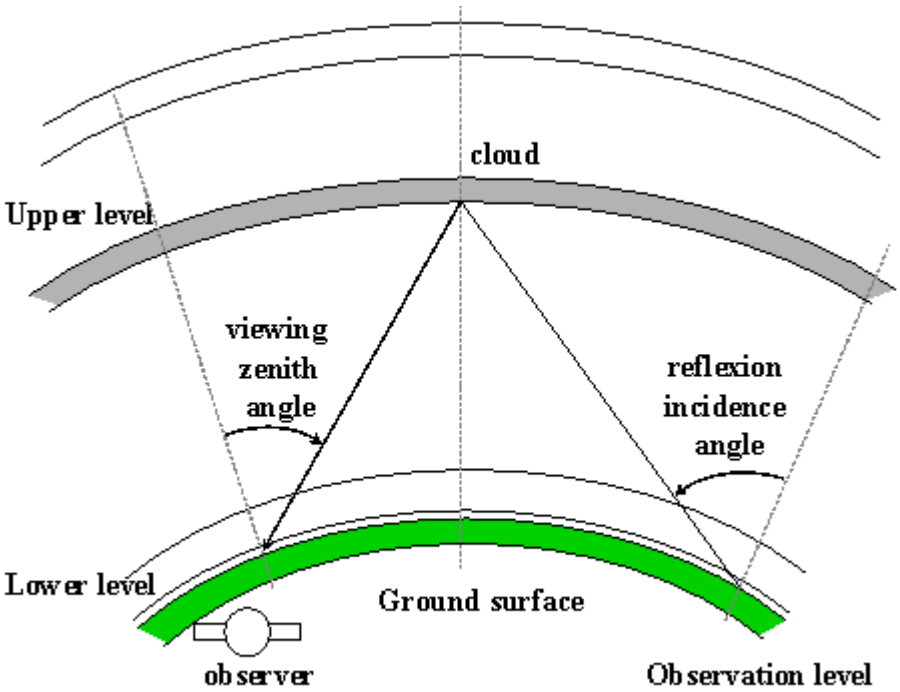


© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

5.4.1.1.3 DOWN configuration



5.4.1.1.4 UPDOWN configuration



© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

5.4.2. Atmospheric profile indexes

The range of profile indexes and their meaning depend on the atmospheric database used which is selected by the parameter **Atlas database indicator**. See your 4A/OP administrator if you need a new database to be installed.

Here is the description of the profile indexes for some databases.

5.4.2.1.1 Default databases

This section concerns the databases that come along the default installation of 4A/OP which are arbitrary test database. The possible profile indexes for them are 1 and 2.

5.4.2.1.2 TIGR database

Here are the profile indexes available for the TIGR2000 database (available at LMD):

- 1 to 872: tropical class;
- 873 to 1260: warm temperate class;
- 1261 to 1614: cold temperate class;
- 1615 to 2311: arctic class.

This database is not part of the default installation of 4A/OP.

5.4.3. Emissivity indexes

The index description should be different from an emissivity database to another. The database is selected by **Emissivity database directory**. See documentation of the database, which is, in most of the case, a file called `index.txt` in the **Emissivity database directory**.

Here are the emissivity indexes for the standard installation of 4A/OP:

- 050 - constant emissivity 0.50;
- 090 to 100 - constant emissivity from 0.9 to 1.00;
- 101 – water;
- 102 - snow/ice;
- 103 - arid bare soil;
- 104 - organic bare soil;
- 105 - senescent sparse shrubs;
- 106 - green sparse shrubs;
- 107 - senescent grass savanna;
- 108 - green grass savanna;
- 109 - senescent woody savanna;
- 110 - green woody savanna;
- 111 - senescent broadleaf forest;
- 112 - green broadleaf forest;
- 113 - senescent needle forest;
- 114 - green needle forest.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

We refer the reader who wants to define new emissivity to the document "4A/OP - Reference documentation" (NOV-3049-NT-1178). In particular, if you want to create your own constant emissivity, you can copy one of the constant emissivity file in **Emissivity database directory** and change its second column to the constant emissivity value. Be sure the name of the new file is spemisXXX.dat where XXX is the new emissivity (i.e. 057 for a value of 0.57 for example).

5.5. Resources

This section gathers resource considerations on disk usage and processing time of the simulation.

5.5.1.1.1 Disk usage

Jacobian calculations can use a lot of disk space. The amount is governed by the number of different wave numbers, the number of molecules and the number of levels. So setting large values for these parameters leads to large disk space use.

5.5.1.1.2 Processing time

The most time consuming step of the simulation is the convolution. When **Jacobian calculation** is selected a convolution is done for each molecule and each level and then the computation is significantly longer. The computation time is a function of the number of levels, the number of implied molecular species, the spectral resolution and the instrumental function. So, for Jacobians, it can take a long time to get results. The **hole convolution** option allows to speed up convolution computations when Jacobian calculation is selected.

5.6. Bug Reports

If the Reference Documentation does not address your question or problem, then you should send a message detailing the issue. If it is a problem with compiling the program, make sure you include

information on your compiler and computer (mainly the type of operating system and brand of compiler). If the program crashes, you should also provide any relevant input files.

Bug reports and development suggestions will be actively pursued.

Please report bugs in 4A/OP to 4AOP@noveltis.fr.

5.7. Annex: Acronyms

- **4A**: Automatized Atmospheric Absorptions Atlas;
- **CNES**: Centre National d'Etudes Spatiales (http://www.cnes.fr/);
- **CNRS**: Centre National de la Recherche Scientifique (http://www.cnrs.fr/);
- **GEISA**: Gestion et Etude des Informations Spectroscopiques Atmosphériques (Management and Study of Atmospheric Spectroscopic Information);
- **ISRF**: Instrument Spectral Response Function;
- **IASI**: Infrared Atmospheric Sounding Interferometer;
- **IGBP**: International Geosphere-Biosphere Program;
- **LMD**: Laboratoire de Météorologie Dynamique (http://lmd.polytechnique.fr/);
- **RTM**: Radiative Transfer Model;
- **TIGR**: Thermodynamic Initial Guess Retrieval: Climatological library of about 1800 representative atmospheric situations from radiosonde reports.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

6. Appendix F.: Error management

The shell script `errglossarybuilder` allows an automatic creation of the list of the controlled errors in the software 4A. It uses the file `errorpatterns.txt` that contains profiles used for the error locating. This script generates the file `errorglossary.txt` that contains the list of error messages produced by the source code files `*.f90` and corresponding to the profiles in `errorpatterns.txt`. The script and the resulting list of errors are presented below.

The command line is: `csh errglossarybuilder <dir>`

`<dir>` is the directory containing the files to be considered.

- Errglossarybuilder

```
#!/bin/csh

# errglossarybuilder --
#
#
#           Project : 4A-OP
#           Version : $Revision: 1.1 $
#     Creation date : January 2002
# Last modification : $Date: 2002/07/03 15:29:47 $
#
# (c) CNES - CNRS/ENS - NOVELTIS
#
# ---
#
#
#
# This script allows to build a text file containing 4A/OP error messages
# Command line : csh errglossarybuilder dir
#                where dir contains the path of the 4A source files
#                if dir is empty errglossarybuilder uses ../
# An error message is a string stored in a variable called ErrMess.

# errglossarybuilder looks for fortran lines containing the following
# patterns : 'ErrMess = ' and ' //'
# Errors should be coded the following way in F90
#        ErrMess = '......'
#             // ' .......'
#             // ' .......'
# errglossarybuilder uses the function /usr/xpg4/bin/grep,
# and the file errorpatterns.txt which should contain the patterns described
# above

if ( $1 == '') then
 set rep = '../'
else
 set rep = $1
endif

echo                                > errorglossary.txt
echo "----------------------------" >> errorglossary.txt
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
echo "4A/OP error message glossary" >> errorglossary.txt
echo "----------------------------" >> errorglossary.txt
echo                                >> errorglossary.txt
echo                                >> errorglossary.txt

foreach fich (`find $rep -name '*.f90'`)
  set LigneErrMsg = ''
  set LigneErrMsg = ` grep 'ErrMess =' $fich `
  if ( $status == 0 ) then
      echo                      >> errorglossary.txt
      echo $fich                >> errorglossary.txt
      echo                      >> errorglossary.txt
      /usr/xpg4/bin/grep -f errorpatterns.txt $fich >> errorglossary.txt
  endif
end
      echo                      >> errorglossary.txt

sed s/'\/\/'/' '/ errorglossary.txt > errorglossary.tmp
sed s/'ErrMess ='/' - '/ errorglossary.tmp > errorglossary.txt
sed s/'&'//g errorglossary.txt > errorglossary.tmp
sed s/"\'"//g errorglossary.tmp > errorglossary.txt
sed s/'\\n'//g errorglossary.txt > errorglossary.tmp
mv errorglossary.tmp errorglossary.txt
cat errorglossary.txt
```

- Errorglossary.txt

```
----------------------------
4A/OP error message glossary
----------------------------



./aaaa.f90

      -  I/O error: bad or missing instrument name
      -  INTOPT parameter must be set to "YES" for making
             Jacobian simulations  (JACOB="YES")
             =>  See file para4a$(RSTR).dtp
      -  Instrument = inf (infinite resolution)
                Convolution = YES  ==> The simulation is
             not possible: see file para4a$(RSTR).dtp
      -  The number of Jacobians is too high
            -  The spectral limits are not consistent with the instrumental
function limits
            -  The number of molecules in a spectral
                   domain is too high  (see the Atlas index
                   file) OR the parameter Cst_ncmax is too small
              -  Bad atlas resolution for the molecule in progress: nmax(nds,nb)
                            The number of computed points can vary from one
spectral domain to the next BUT it must be the same for all the molecules in one
specific spectral domain (all the molecules must be on the same discretization
in order to optimize the interpolation)
   -  Reading error
   -  Unexpected end of file
   -  File open error
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
./atmformat.f90

       -  The instructions for reading the atmospheres
              "//atm(1:len_trim(atm))//"
               are not available in the  subroutine lecatmdsf
       -  The number of levels of your atmospheric profile (nlevel)
              is higher than the  maximum number of levels available in
              atmformat (Cst_nlevelmax)
   -  Reading error

./atmref.f90

      -  Bad number of molecules or of levels
   -  Reading error
   -  Unexpected end of file

./atmsph.f90

      -  The index number of the selected atmospheric profile is not valid
      -  The number of atmospheric profiles is too high
      -  Error : secant=1/Cos(Viewing Angle)=0
             => You have to define this parameter in the file
             atm4a$(ATM).ddb OR in the file para4a$(RSTR).dtp
   -  Reading error

./calcisrf.f90

      -  Error in the convolution size
      -  Bad size of the functions
      -  The function size is too high
         -  Error in the subroutine lireisrf
            -  Error in the subroutine lireisrf

./calcrad.f90

      -  The viewing configuration  trajet(1:len_trim(trajet))// is not valid

./calcsig.f90

                  -  Error in the subroutine liread
               -  Error in the subroutine liread
            -  Error in the subroutine continua

./calcsphtraj.f90

      -  Error : Secant of incidence zenith angle (sectta1) < 1
      -  Error : Secant of zenith angle of outgoing path (sectta2) < 1

./continua.f90

            -  The temperature is out of the 150-350 K tolerable range.
            -  The temperature is out of the 150-350 K tolerable range.

./convini.f90

         -  Bad wave number limits for the convolution
         -  Bad convolution step PASCON
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
        -  No spectrometer ISRF in the spectral domain
      -  Instrument convolution identifier not recognized

./convolution.f90

          -  The shift between the spectrum wave numbers and
                the corresponding instrumental function ones is
                is too high compared to the spectrum computation
                resolution
          -  Error in the subroutine calcisrf
          -  The shift between the spectrum wave numbers and
                the corresponding instrumental function ones is
                is too high compared to the spectrum computation
                resolution
          -  Error in the subroutine calcisrf

./gascon.f90

      -  Bad number of molecules or of levels
   -  Reading error
   -  Unexpected end of file

./init_contin.f90

   -  Reading error
   -  Unexpected end of file
   -  File open error

./lecatmdsf.f90

   -  The total number of profiles is too high

./liread.f90

        -  Error in the Atlas file header
              -  Error in the optical thickness coding
                 -  Error in the Atlas targets
   -  Reading error

./lireemis.f90

   -  Reading error
   -  Unexpected end of file
   -  File open error

./lireisrf.f90

   -  Reading error

./lirespc4a.f90

       -   Error on the wave number boundaries: numin > numax
           -   Error on the wave number boundaries
   -  Reading error

./lirespi4a.f90

      -  The number of bands indicated in the header is less than 1
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

```
   -  Reading error
   -  Reading error on the output file

./para4a.f90

      -  Error : Angle type (TYP_ANG) = SECANT, and the
            primary path secant value (SECTTA1) < 1 in the parameter
             file para4a$(RSTR).dtp => Incompatibility !
      -  Error : Angle type (TYP_ANG) = SECANT, and the
            secondary path secant value (SECTTA2) < 1 in the parameter
             file para4a$(RSTR).dtp => Incompatibility !
   -  Reading error
   -  Unexpected end of file

./spectrumio.f90

     -  File open error

./converjacob.f90

            -  Bad Jacobian unit index (0,1,2,3,4 or 5)
         -  Bad Jacobian unit index (0,1,2,3,4 or 5)
                  -  Bad Jacobian unit index (0,1,2,3,4 or 5)
               -  Bad Jacobian unit index (0,1,2,3,4 or 5)
```

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

Part IV. References

[Ref. 1] Scott, N. A., 1974: A direct method of computation of transmission function of an inhomogeneous gaseous medium: description of the method and influence of various factors. J. Quant. Spectrosc. Radiat. Transfer, 14, 691-707.

[Ref. 2] Scott, N. A. and A. Chedin, 1981: A fast line-by-line method for atmospheric absorption computations: The Automatized Atmospheric Absorption Atlas. J. Appl. Meteor., 20, 802-812.

[Ref. 3] Tournier, B., R. Armante, and N. A. Scott, 1995: STRANSAC-93 et 4A-93: Développement et validation des nouvelles version des codes de transfert radiatif pour application au projet IASI. Internal Rep. LMD, No. 201, LMD/CNRS, Ecole Polytechnique, PALAISEAU, France.

[Ref. 4] Chéruy, F., N. A. Scott, R. Armante, B. Tournier, and A. Chédin, 1995: Contribution to the development of radiative transfer models for high spectral resolution observations in the infrared. J. Quant. Spectrosc. Radiat. Transfer, 53, 597-611.

[Ref. 5] Jacquinet-Husson, N. *et al.*, 1999: The 1997 spectroscopic GEISA data bank. J. Quant. Spectrosc. Radiat. Transfer, 62, 205-254.

[Ref. 6] Jacquinet-Husson, N., N. A. Scott, A. Chedin, and A. A. Chursin, 2003: The GEISA spectroscopic database system revisited for IASI direct radiative transfer modelling. Atmospheric and Oceanic Optics, 16, N°3, 256-282.

[Ref. 7] Jacquinet-Husson, N *et al.*, 2008: The GEISA spectroscopic database: Current and future archive for Earth and planetary atmosphere studies. J. Quant. Spectrosc. Radiat. Transfer, 109, 1043–1059

[Ref. 8] Jacquinet-Husson, N. *et al.*, 2011: The 2009 edition of the GEISA spectroscopic database. J. Quant. Spectrosc. Radiat. Transfer, 112, 2395-2445.

[Ref. 9] Chédin, A., *et al.*, 1985: The Improved Initialization Inversion Method: A High Resolution Physical Method for Temperature Retrievals from Satellites of the TIROS-N Series. J. Appl. Meteor., 24, 128-144.

[Ref. 10] Perrin, M. Y., and J. M. Hartmann, 1989: Temperature-dependent measurements and modeling of absorption by CO2-N2 mixtures in the far line wings of the 4.3 microns CO2 band. . J. Quant. Spectrosc. Radiat. Transfer, 42, 311-317.

[Ref. 11] Clough, S. A., F. X. Kneizys, and R. W. Davies, 1989: Line shape and the water vapor continuum. Atmos. Res., 23, 229-241.

[Ref. 12] Clough, S. A., 1995: The water vapor continuum and its role in remote sensing. Optical Remote Sensing of the Atmosphere, Vol. 2, OSA Technical Digest Series, (Optical Society of America, Washington, DC, 1995), 76-78.

[Ref. 13] Lafferty W. J., A.M. Solodov, A. Weber, W.B. Olson, and J.M. Hartmann 1996: Infrared collision-induced absorption by N2 near 4.3 microns for applications in the atmosphere. Measurements and empirical modelling. Appl. Opt., 35, 5911-5917.

[Ref. 14] Thibault F., V. Menoux, R. Le Doucen, L. Rosenmann, J.M. Hartmann, and C. Boulet, 1997: Infrared collision-induced absorption by O2 near 6.4 microns for atmospheric applications: measurements and empirical modelling. Appl. Opt., 36, 563-567.

[Ref. 15] Gamache R. R., S. Kennedy, R. Hawkins, and L. S. Rothman, 2000: Total Internal Partition Sums for Molecules in the Terrestrial Atmosphere. J. Mol. Struct., 517-518, 413-431.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

[Ref. 16] Rodrigues, R., 1998: Interferences entre raies et relaxation du moment angulaire de rotation dans des spectres infrarouges de CO2. Mesures, modélisations à l'aide d'une loi d'échelle et application à des branches Q atmosphériques. Thèse de Doctorat, Université Paris XI, France.

[Ref. 17] Niro, F., C. Boulet, J. -M. Hartmann, 2004: Spectra calculations in central and wing regions of CO_2 IR bands between 10 and 20 μm. I: model and laboratory measurements. Journal of Quantitative Spectroscopy and Radiative Transfer, Volume 88, Issue 4, 483-498.

[Ref. 18] Niro, F., T. von Clarmann, K. Jucks, J. -M. Hartmann, 2005: Spectra calculations in central and wing regions of CO_2 IR bands between 10 and 20 μm. III: atmospheric emission spectra. Journal of Quantitative Spectroscopy and Radiative Transfer, Volume 90, Issue 1, 61-76.

[Ref. 19] Lamouroux J., H. Tran, A.L. Laraia, R.R. Gamache, L.S. Rothman, I.E. Gordon, J.-M. Hartmann, 2010: Updated database plus software for line-mixing in CO_2 infrared spectra and their test using laboratory spectra in the 1.5–2.3 μm region. J. Quant. Spectrosc. Radiat. Transfer, 111, 2321-2331.

[Ref. 20] Hartmann, J.-M., 2009: A simple empirical model for the collisional spectral shift of air-broadened CO_2 lines. J. Quant. Spectrosc. Radiat. Transfer, 110, 2019–2026.

[Ref. 21] Garand, L., *et al.*, 2001: Radiance and Jacobian intercomparison of radiative transfer models applied to HIRS and AMSU channels. J. Geophys.Res., 106, N° D20, 24017-24032.

[Ref. 22] Snyder, W.C., *et al.*,1998: Classification-based emissivity for land surface temperature measurement from space. Int. J. Remote Sensing, vol. 19, 2753-2774.

[Ref. 23] Documentation File for 'DISORT', A Discrete Ordinates Radiative Transfer Fortran-90 Program for a Multi-Layered Plane-Parallel Medium (Version 2.0) - ftp://climate1.gsfc.nasa.gov/wiscombe/Multiple_Scatt/DISORT2.0beta/

[Ref. 24] Introduction to the GEISA spectroscopic database http://ara.lmd.polytechnique.fr/htdocs-public/products/GEISA/HTML-GEISA/index.html

[Ref. 25] The software package OPAC (Optical Properties of Aerosols and Clouds) http://www.lrz-muenchen.de/~uh234an/www/radaer/opac.html

[Ref. 26] Global Aerosol Data Set (GADS) http://www.lrz-muenchen.de/~uh234an/www/radaer/gads.html

[Ref. 27] Essen, L. and K. D. Froome, 1951: The Refractive Indices and Dielectric Constants of Air and its Principal Constituents at 24 GHz. Proc. of the Physical Society (London), Section B, 64:862-875.

[Ref. 28] Barrel, H and J. E.Sears, 1939: The refraction and dispersion of air for the visible spectrum. Phil. Trans. Royal Soc. London, Bd. A 238, 6-62.

[Ref. 29] Rüeger, J. M., 1999. Presented at the 22nd General Assembly of IUGG, 18-30 July 1999: Report of the Ad-Hoc Working Party on Refractive Indices of Light, Infrared and Radio Waves in the Atmosphere of the IAG Special Commission SC3 - Fundamental Constants (SCFC) for the period 1993 - 1999. Birmingham, UK, 25 pages.

[Ref. 30] TN-IROE-RSA9601, issue 3, 26 June 2001: High level algorithm definition and physical and mathematical optimisations (MIPAS Level 2 Algorithm Theoretical Baseline Document).

[Ref. 31] Edlen, B., 1966: Equation for the Refractive Index of Air. Metrologia 2 (2), 71-80.

[Ref. 32] Heilliette S., C. Pierangelo, C. Crevoisier, and A. Chédin, 2004: A fast solver to compute atmospheric radiative transfer in a scattering atmosphere using the successive order of scattering method. LMD internal note - http://ara.lmd.polytechnique.fr/htdocs-public/abstracts/2004/internal_note_LMD_20041008.pdf, Ecole Polytechnique, PALAISEAU, France.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS

[Ref. 33] Mitchell, D. L., A. Macke, and Y. Liu, 1996: Modelling Cirrus Clouds: II) Treatment of radiative properties. J. Atmos. Sci., 53, 2967-2988.

[Ref. 34] Guignard, A., Stubenrauch, C. J., Baran, A. J., and Armante, R.: Bulk microphysical properties of semi transparent cirrus from AIRS: a six years global climatology and statistical analysis in synergy with CALIPSO and CloudSat, Atmos. Chem. Phys. Discuss., 11, 24671-24725, doi:10.5194/acpd-11-24671-2011, 2011.

© CNES – LMD (CNRS/ENS) - NOVELTIS
This document is the property of CNES, LMD and NOVELTIS, no part of it shall be reproduced or transmitted without the express prior written authorisation of CNES, LMD and NOVELTIS