European Organisation for Astronomical Research in the Southern Hemisphere

Programme: GEN

Project/WP: Science Operation Software Department

MOLECFIT Pipeline User Manual

Document Number: VLT-MAN-ESO-19550-5772

Document Version: 4.3.1

Document Type: Manual (MAN)

Released on: 2023-05-31

Document Classification: Public

Prepared by: MOLECFIT Pipeline Team

Validated by:

Approved by:

Name

ESO – Karl-Schwarzschild-Str. 2 – 85748 Garching bei München – Germany
www.eso.org

Authors

Name	Affiliation
Nuwanthika Fernando Nuria P. F. Lorente Michael N. Birchall Brent Miszalski	AAO Software Pipelines Team Macquarie University & The Australian National University 105 Delhi Rd, North Ryde, NSW 2113 Australia
Alain Smette Andrea Mehner	European Southern Observatory Alonso de Cordova 3107 Vitacura, Casilla 19001 Santiago de Chile Chile
Lodovico Coccato John Pritchard Sabine Möhler Gurvan Bazin Jose A. Escartín	European Southern Observatory, ESO Headquarters Karl-Schwarzschild-Str. 2, 85748 Garching bei München Germany
Stefan Noll Wolfgang Kausch Marco Barden Cezary Szyszka Amy M. Jones Stefan Kimeswenger	Austrian ESO In-Kind Team, Innsbruck Institute for Astro- and Particle Physics, University of Innsbruck Technikerstr. 25/8, A-6020 Innsbruck Austria

Change Record

Version	Date	Affected Section	Remarks
4.2	11/02/2022	Sect. 4.5, 5.1	Mentions of FORS and X-Shooter MOLECFIT recipes
4.1	08/02/2021	All	Major revision of the code (v4.0)
4.0	20/05/2020	All	Major revision of the code (v3.0) and overhaul of manual
3.11	31/10/2018	Sect. 4.9	Known issues
3.10	13/04/2018	Sect. 4.	Environment variables
3.9	16/01/2018	Sect. 4.9	Known issues
3.8	21/12/2017	All	Major revision of the code
3.7	01/04/2014	Sect. 3, 7	Updated installation instructions, documented expert mode
3.6	31/10/2013	Sect. 4.4, 4.5, 5.1, 6.1, 6.2, and 7.1.4	Profiles, FITS keywords, and test updates
3.5	26/09/2013	All	RIXes for final review considered
3.4	25/07/2013	All	Changes related to code-testing workshop, added appendix
3.3	27/05/2013	Sect. 4.5	Changes in parameter description
3.2	30/04/2013	Sect. 5.5.3	Change in input parameter restrictions
3.1	28/03/2013	All	Changes related to midterm review and more
3.0	02/12/2012	All	Major revision of the code
2.5	17/05/2012	Sect. 4.4, 6.3, and 6.4	Discussing RFM problems and modified output files
2.4	03/05/2012	Sect. 1, 2.2, 4.4.1, 6.5, and 7	Discussing telluric absorption correction, Figure 1 modified
2.3	21/03/2012	Sect. 4 and 5.5.3	Voigt profile approximation added
2.2	16/03/2012	Sect. 5	Updating information concerning GDAS profiles Comment about width zero components (convolution)
2.1	23/11/2011	Sect. 4 and 5.5.3	Adding comments of Alain Smette
2.0	06/04/2011	All	Second version
1.0	28/02/2011	All	First version

Contents

1 Introduction

MOLECFIT is a software tool for correcting atmospheric absorption features in astronomical observations, based on fitting synthetic transmission spectra to the astronomical data. It can also estimate molecular abundances, especially the water vapour content of the Earth's atmosphere.

Correcting for these telluric absorption features usually requires supplementary calibration data which are very expensive to obtain in terms of telescope time. In addition, the observation scheduling flexibility is restricted as these data usually have to be taken immediately before or after the science observations to which the correction will be applied, due to the high variability of the telluric absorption which depends on the state and chemical composition of the atmosphere at the time of observations. A tool for telluric absorption correction such as MOLECFIT, that is based on synthetic transmission spectra calculated by a radiative transfer code does not require these supplementary calibration data if the science data can be fitted directly. This can save a significant amount of valuable telescope time and as such increase the instrumental efficiency.

As useful as MOLECFIT is, it has some limitations and observation of a telluric standard star may be better suited in some conditions. For instance, MOLECFIT might perform poorly with:

- large number of intrinsic features,
- little or no continuum emission,
- low signal-to-noise ratio,
- large airmass observations with high water vapor content.

The tool can be run from a command-line or more conveniently through a GUI (see the MOLECFIT Reflex Tutorial [RD3]).

1.1 Scope

This document describes the MOLECFIT package version 4.3. The most recent version of this document may be found on the VLT Instrument Pipelines MOLECFIT Downloads page, at `https://www.eso.org/sci/software/pipelines/molecfit/molecfit-pipe-recipes.html`.

1.2 Acknowledgements

MOLECFIT was developed for ESO by a team of astronomers at the Institute for Astro- and Particle Physics at the University of Innsbruck on the base of an IDL prototype developed by Alain Smette, Hannes Horst, Hugues Sana, Alexandre Gallenne. It is one of three sky modeling and correction tools provided in the framework of the Austrian in-kind contribution to ESO.

More recently, with version 3, it has been refactored by the ESO Pipelines team, and divided into the user interface component, and its underlying `TELLURICCORR` library with the goal to be easily integrated into the various instrument workflows. Development and maintenance of MOLECFIT is now carried out by the AAO Software Pipelines Team.

If you use MOLECFIT to produce a scientific publication, we ask that you please use the following references:

- Smette, A. and Sana, H. and Noll, S. and Horst, H. and Kausch, W. and Kimeswenger, S. and Barden, M. and Szyszka, C. and Jones, A. M. and Gallenne, A. and Vinther, J. and Ballester, P. and Taylor, J. 2015, A&A 576, A77[1]
- Kausch, W. and Noll, S. and Smette, A. and Kimeswenger, S. and Barden, M. and Szyszka, C. and Jones, A. M. and Sana, H. and Horst, H. and Kerber, F. 2015, A&A 576, A78[2]
- Molecfit Pipeline Team, MOLECFIT Pipeline User Manual v4.3, 2023. ESO VLT-MAN-ESO-19550-5772

1.3 Stylistic conventions

Throughout this document the following stylistic conventions are used:

bold in text sections for commands and other user input which has to be typed as shown

italics in the text and example sections for parts of the user input which have to be replaced with real contents

`teletype` in the text for FITS keywords, program names, file paths, and terminal output, and as the general style for examples, commands, code, etc

In example sections expected user input is indicated by a leading shell prompt.

In the text **bold** and *italics* may also be used to highlight words.

1.4 Notational Conventions

Hierarchical FITS keyword names, appearing in the document, are given using the dot–notation to improve readability. This means, that the prefix “HIERARCH ESO” is left out, and the spaces separating the keyword name constituents in the actual FITS header are replaced by a single dot.

[1] Bibtex reference: https://ui.adsabs.harvard.edu/abs/2015A%26A...576A..77S/exportcitation

[2] BibTex reference: https://ui.adsabs.harvard.edu/abs/2015A%26A...576A..78K/exportcitation

2 Related Documents

2.1 Applicable Documents

[AD1]	VLT Data Flow System Specifications for Pipeline and Quality Control	VLT–SPE–ESO–19600–1233
[AD2]	Data Flow for VLT Instruments Requirement Specification	VLT–SPE–ESO–19000–1618/2.0
[AD3]	DFS Pipeline & Quality Control — User Manual	VLT–MAN–ESO–19500-1619
[AD4]	ESO DICB — Data Interface Control Document	GEN–SPE–ESO–19400–0794
[AD5]	Common Pipeline Library User Manual	VLT–MAN–ESO–19500–2720

2.2 Reference Documents

[RD01]	Detailed Specification Document	VLT-SPE-ESO-19550-5769
[RD02]	DR06 User Manual	VLT-MAN-ESO-19550-5286
[RD03]	Reflex MOLECFIT Tutorial	VLT-MAN-ESO-19550-5928
[RD04]	Moré, J.J., Garbow, B.S., & Hillstrom, K.E. 1980, User Guide for MINPACK-1, Argonne National Laboratory Report ANL-80-74, Argonne, Ill.	
[RD05]	Clough, S.A, Shephard, M.W., Mlawer. E.J., et al. 2005, J. Quant. Spectrosc. Radiat. Transfer, 91, 233	
[RD06]	Noll, S., Kausch, W., Barden, M., et al. 2012, A&A, 543, A92	
[RD07]	VLT Astronomical Site Monitor, ASM Data, User Manual	VLT-MAN-ESO-17440-1773
[RD08]	Anu Dudhia, private communication, 2009.	
[RD09]	SM-01 User Manual,	VLT-MAN-ESO-19550-5770
[RD10]	Niro, F., Jucks, K., & Hartmann, J.-M. 2005, J. Quant. Spectrosc. Radiat. Transfer, 95, 469	
[RD11]	Clough, S.A., Iacono, M.J., & Moncet, J.-L. 1992, J. Geophys. Res., 97, 15761	
[RD12]	Masuda, K., Takashima, T., & Takayama, Y. 1988, Remote Sens. Environ., 24, 313	
[RD13]	Wu, X. & Smith, L. 1997, Appl. Opt., 36, 2609	
[RD14]	Patat, F., Moehler, S., O'Brien, K., at al. 2011, A&A, 527, A91	
[RD15]	SM-03 Science Report,	VLT-TRE-ESO-19550-5774
[RD16]	Modigliani. A., Goldoni, P., Royer, F., et al. 2010, Proc. SPIE, 7737, 773728	
[RD17]	LBLRTM, http://rtweb.aer.com/lblrtm_frame.html	
[RD18]	CMPFIT, http://www.physics.wisc.edu/~craigm/idl/cmpfit.html	
[RD19]	AER, http://www.aer.com/	
[RD20]	HITRAN, http://www.cfa.harvard.edu/HITRAN/	
[RD21]	NOAA, http://www.arl.noaa.gov/READYamet.php	

[RD22] GDAS, http://ready.arl.noaa.gov/gdas1.php
[RD23] GRIB, http://nomads.ncep.noaa.gov/txt_descriptions/fast_downloading_grib.shtml
[RD24] RFM, http://www.atm.ox.ac.uk/RFM/
[RD25] LBLRTM_instructions, http://irina.eas.gatech.edu/Lab_5560/lblrtm/lblrtm_inst.html
[RD26] PWV, http://www.eso.org/observing/dfo/quality/GENERAL/PWV/HEALTH/
[RD27] CDIAC, http://cdiac.ornl.gov/
[RD28] LBLRTM FAQ, http://rtweb.aer.com/docs/aqFAQ_LBLRTM.pdf

3 Definitions, Acronyms and Abbreviations

AER	Atmospheric and Environmental Research Inc.
AOPP	Atmospheric, Oceanic and Planetary Physics, Oxford University, UK
ARL	Air Resources Laboratory
ASCII	American Standard Code for Information Interchange
CalibDB	Calibration Database
CPL	Common Pipeline Library
CCD	Charge Coupled Device
CRIRES	CRyogenic high-resolution InfraRed Echelle Spectrograph
DFS	Data Flow System
DRS	Data Reduction System
EMM	ESO Meteo Monitor
Envisat	Environmental Satellite
ESO	European Southern Observatory
EsoRex	ESO Recipe Execution Tool
FITS	Flexible Image Transport System
FOV	Field Of View
FORS	Focal Reducer/low dispersion Spectrograph
FWHM	full width at half-maximum
GDAS	Global Data Assimilation System
GRIB	GRIdded Binary
GUI	Graphical User Interface
HITRAN	HIgh-resolution TRANsmission molecular absorption database
HTML	Hypertext Markup Language
IDL	Interactive Data Language
KMOS	K-band Multi Object Spectrograph
LBLRTM	Line-by-line Radiative Transfer Model
LNFL	Line File
LSF	Line Spread Function
MIPAS	Michelson Interferometer for Passive Atmospheric Sounding
MIR	mid-infrared
NCEP	National Centers for Environmental Prediction
NIR	near-infrared
NOAA	National Oceanic and Atmospheric Administration
OB	Observation Block
pixel	picture element (of a raster image)
PSF	Point Spread Function
PWV	precipitable water vapour
QC	Quality Control
RFM	Reference Forward Model
RMS	root mean square
RPG	Radiometer Physics GmbH
SDP	Science Data Product
SOF	Set Of Frames

TBD	To be defined
TBC	To be confirmed
VISIR	VLT Imager and Spectrometer for mid InfraRed
VLT	Very Large Telescope]
WCS	World Coordinate System

4 Overview

This document is organised as follows:

Section 4 gives a brief overview of the project and the incorporated algorithms. Section 5 contains a description on how to run the code, the required input parameter file, and the output files. Users of MOLECFIT version 1.59 may wish to read Section 7, which describes how the user interaction with the package has changed since the v3.0 overhaul. In Section 8 the atmospheric model and its adaptation to the observed spectrum are described in detail. Section 9 provides information on the installation procedures, and tips and tricks are outlined in Section 11. Finally, the appendices provide a reference, and information for the advanced user.

4.1 The Project

Ground-based astronomical observations suffer from emission and absorption processes in the atmosphere which deteriorate the quality of the obtained data. At wavelengths longer than $2 - 2.5\,\mu$m, where thermal radiation from molecules in the lower atmosphere dominates, the amount of this perturbing background radiation can determine whether scientific observations are feasible at all. For this wavelength regime, it is crucial to be able to estimate the intensity of the atmospheric emission in advance. Such a prediction requires good knowledge of the column densities of atmospheric constituents that significantly contribute to the greenhouse effect. The most important molecules are water (H_2O), carbon dioxide (CO_2), methane (CH_4), nitrous oxide (N_2O), and ozone (O_3). In particular, a good knowledge of the water abundance, i.e. air humidity, is essential, since H_2O is the main contributor to the IR atmospheric spectra. Moreover, it is much more variable than the other important species. Due to this variability of molecular abundances, the removal of telluric absorption features from astronomical spectra depends on observations of telluric standard stars with relatively smooth continua, at similar time and airmass as the scientific target. Even if those observations are available, the telluric absorption correction becomes tricky in the near- and mid-IR, where wavelength regions with negligible atmospheric absorption are hardly present. In this case, a reliable determination of the shape of the unabsorbed continuum is very difficult if an interpolation approach is used. Therefore, a realistic model of the atmospheric absorption for given observing conditions makes the telluric absorption correction more reliable, and reduces the number of required observations of telluric standard stars.

For a periodic monitoring of the abundances of crucial atmospheric constituents as well as a high-quality correction of telluric absorption features in astronomical spectra a fast, user-friendly, and reliable software tool is needed. MOLECFIT provides such a tool. Thanks to the advances made in modelling the Earth's atmosphere, it is now possible to compute realistic atmospheric emission and absorption spectra. Such model spectra can be fitted to observed atmospheric spectra with relative deviations of only a few percent. The MOLECFIT pipeline is a tool designed to fit such atmospheric models and derive reliable estimates of the atmosphere's convolution impact on observed spectra. The incorporated model spectra are generated using the radiative transfer code LBLRTM [RD17]. The basic input for the radiative transfer code is an atmospheric profile, which is created from a standard atmosphere (containing information on height, pressure, temperature, and chemical composition for general tropical environments up to 120 km), modelled meteorological GDAS data (containing pressure, temperature, and relative humidity for the region of the observing site for elevations up to $\approx$ 25 km), and on-site meteorological measurements by the ESO Meteo Monitor (EMM). Using iterative techniques, the input atmospheric profile is varied to obtain a model spectrum that fits the scientific input spectrum (see Section 4.2 for more details). In this process, it is also necessary to optimise the scaling, wavelength grid, and resolution of the model. To reliably compute radiance spectra, grey body radiation from the telescope itself also has to be taken into account.

The use of MOLECFIT as part of the data reduction process can significantly improve the efficiency of astronomical instruments working in the thermal IR such as CRIRES or VISIR. Since strong molecular absorption affects all observable wavelength regimes except for the visual, many ESO instruments could benefit from an improved correction of telluric features in spectra of astronomical targets.

The MOLECFIT package relies on the following third-party code and input files:

- CMPFIT: The C version of the least-squares fitting library `mpfit` by C. Markwardt [RD18] based on the FORTRAN fitting routine MINPACK-1 by Moré et al. [RD04]
- The radiative transfer code LBLRTM v12.11. This publicly available software is developed within the Radiative Transfer Working Group of Atmospheric and Environmental Research Inc. (AER); see also Clough et al. [RD05], [RD19], and [RD17]) for more details. LBLRTM can handle all molecules incorporated in a line parameter database, e.g. HITRAN [RD20], and offers a wide range of possibilities for adjusting input parameters. An additional part of this software is the LNFL, which provides the required line information based on the line parameter database.
- A line database of molecular parameters: Currently, the line parameter list aer_v_3.8.1.2 delivered with the LBLRTM v12.11 package is included in the MOLECFIT package. This line parameter list is built from HITRAN 2016 [RD20] and contains some updates (see [RD17] and Appendix D.1.3 for more details).
- An atmospheric profile valid for the time of observation: The profile is created using standard atmospheric profiles, on-site measurements by the EMM [RD07], and GDAS data (a product of the NCEP (National Centers for Environmental Prediction) model, created by the Air Resources Laboratory (ARL) of the National Oceanic and Atmospheric Administration (NOAA), see [RD19], [RD21], [RD22]). Such atmospheric profiles include pressure, temperature, and humidity for a number of layers of the atmosphere, at any point on the Earth for all recent dates (back to December 2004), on a 3h-grid basis.
- Perl scripts `get_inv.pl` and `get_grib.pl` from the NCEP NOMAD server [RD23] for fast download of GDAS data in the GRIB2 (GRIdded Binary 2) format.

4.2 Algorithm

Block diagrams of the basic functionality of the software are shown in Figures 4.1 and 4.2 providing, respectively, a general overview of the workflow, and a more detailed view of the `molecfit_model` component.

First, the code reads the science spectrum from a FITS table or FITS image, and an ASCII driver file containing the user input parameters. If available, ESO keywords including EMM data are directly taken from FITS header information. A single atmospheric profile is compiled from data provided by three sources: a standard atmospheric profile for a given climate zone, an appropriate GDAS model profile for the time of the observation and telescope site, and the corresponding ground-based EMM measurements (see Section 8.1 for more details). Input for the radiative transfer model is the resulting merged atmospheric profile (with a possible pre-selection of relevant molecules) and the target airmass at the time of observation (see Section 8.2). To match the observed spectrum, the code adapts the atmospheric spectrum (either transmission or radiation; see Section 8.3) further

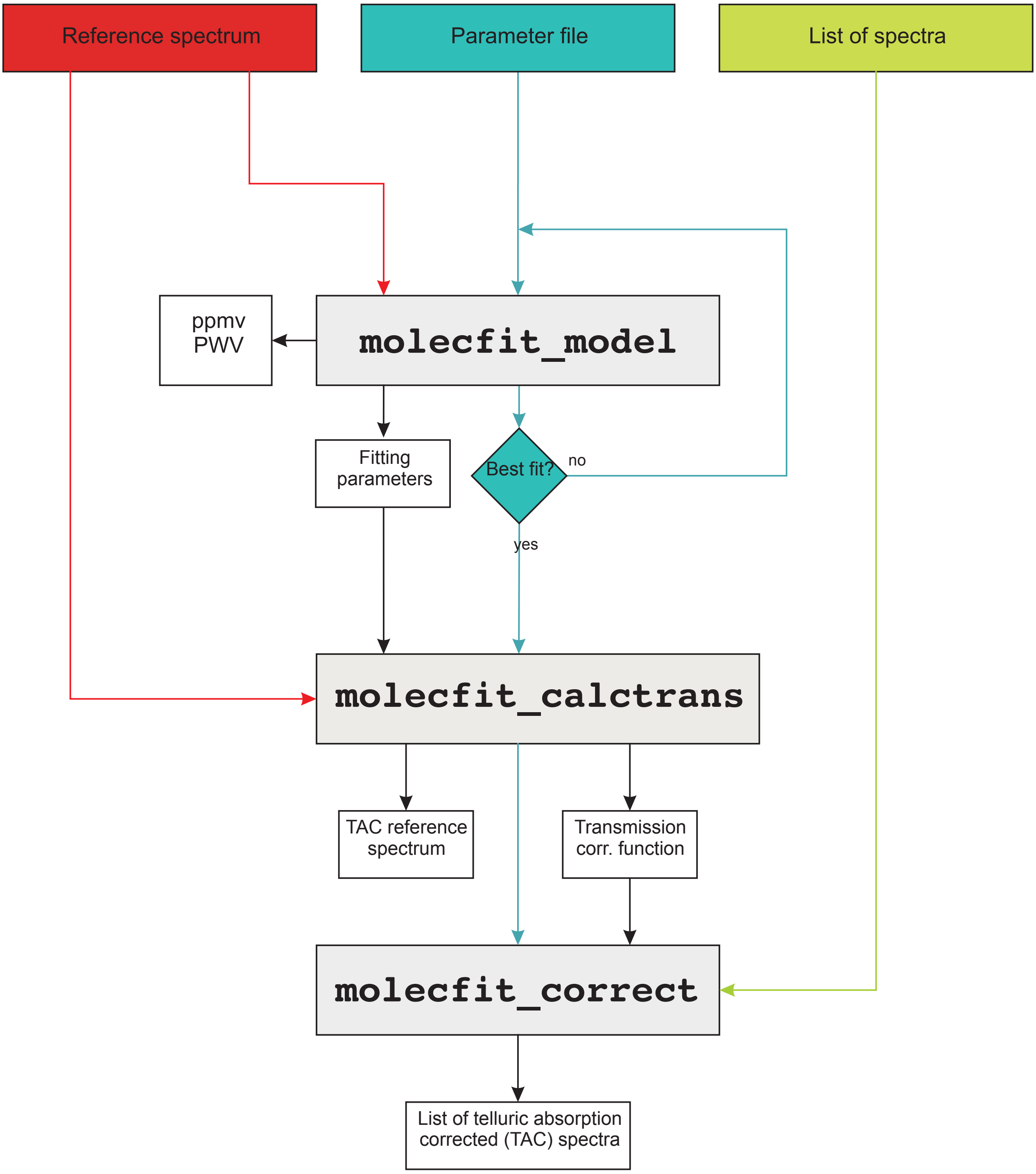

Figure 4.1: Overview of the software workflow. It shows the input and output for the three executables `molecfit_model`, `molecfit_calctrans`, and `molecfit_correct` and the connection between them.

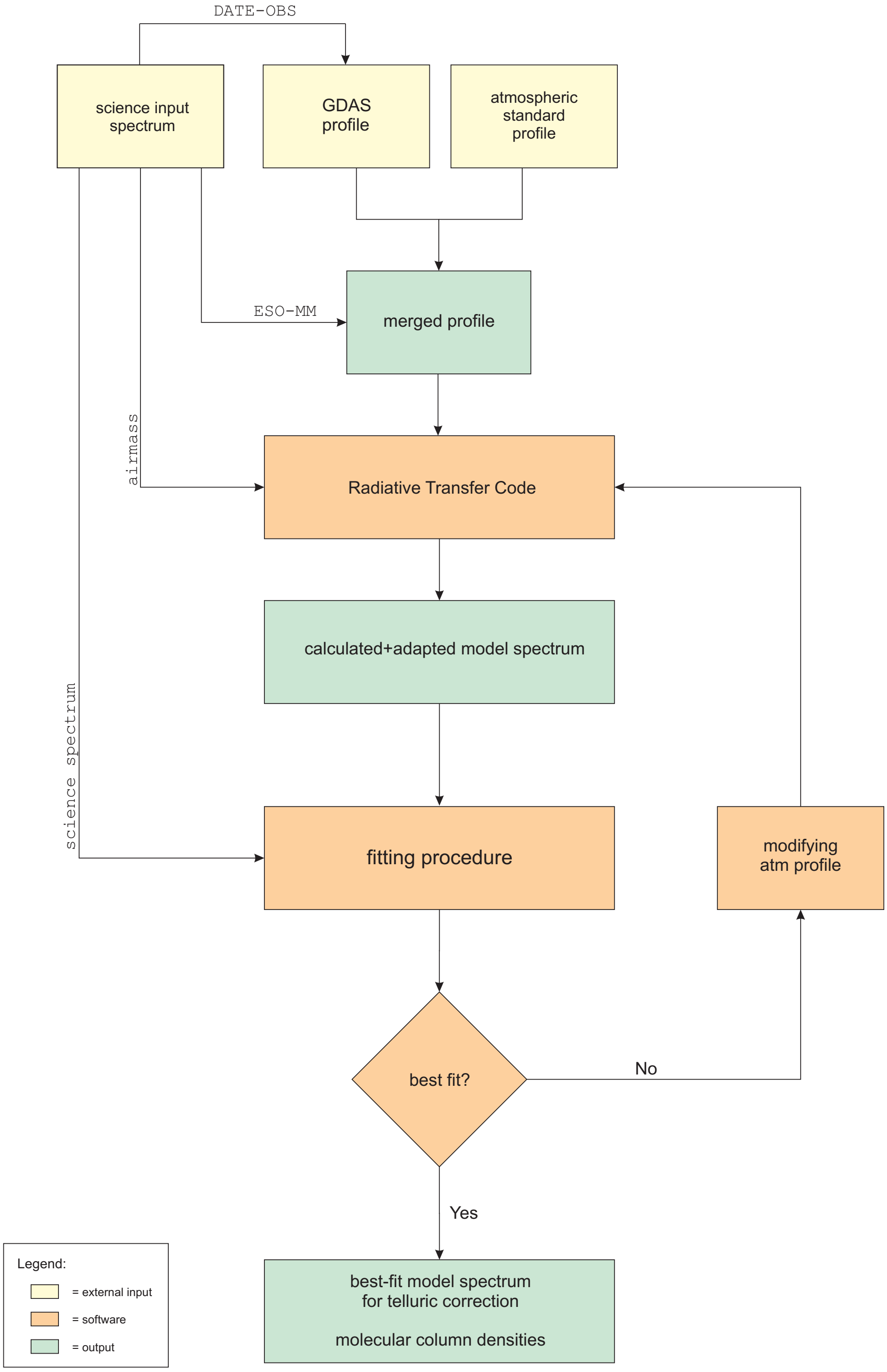

Figure 4.2: Workflow of the `molecfit_model` routine.

by flux scaling, wavelength grid correction, and convolution with a suitable instrument profile (see Section 8.5). For a radiance spectrum, the contribution of thermal emission from the telescope can also be taken into account (see Section 8.4).

The central component of the algorithm is the comparison/fitting of the calculated and the input science spectrum by means of `mpfit` [RD18]. The χ^2 minimisation procedure of this routine is based on the Levenberg-Marquardt technique (see Moré et al., [RD04]), an iterative search algorithm characterised by gradient-controlled jumps in parameter space. Since this technique is prone to finding local minima, reasonable starting values and constraints for the fit parameters are required. `mpfit` checks whether the desired fit quality is reached. If this is not the case, it changes fit parameters in an appropriate way to search for a better χ^2. Each function call causes a new calculation of the sky model. For a change of the molecular abundances, the profiles of molecules can be scaled by simple factors, which represent a subset of the fit parameters provided to `mpfit`. The other parameters are coefficients of polynomials for continuum scaling, coefficients of Chebyshev polynomials for the correction of the wavelength solution, the FWHM of boxcar, Gaussian, and Lorentzian kernels that are used to build a realistic instrumental profile, and the telescope emissivity if a radiance spectrum is computed. When a satisfactory fit is reached after several iterations of the χ^2 minimisation procedure, the code writes the best-fit spectrum, atmospheric profile, and fit parameters to output files. The best-fit molecular profiles are integrated to get molecular columns in ppmv. The `molecfit_model` recipe also computes the water vapour content of the atmosphere in mm. The atmospheric abundances are part of a special output summary file (see Section 6.5).

The fit parameters are defined in the input `.rc` file (see Section 6.2). Dynamically setting fit flags to include and exclude parameters from the fitting procedure allows the code to find a reasonable solution in a relatively fast and robust way. In detail, the `molecfit_model` recipe follows a plan consisting of six steps by default:

- Step 1: scaling of the continuum
- Step 2: wavelength and resolution fit
- Step 3: rescaling of the continuum
- Step 4: fitting of the column density for each molecule
- Step 5: joint continuum, wavelength, and resolution fit
- Step 5: fit of all components (molecules column density, continuum, wavelength, and resolution)

For the telluric absorption correction, `molecfit_calctrans` must be used with an input set of frames (SOF) which includes the name of the file with the best-fit parameters obtained by `molecfit_model`. The `molecfit_calctrans` recipe calculates the atmospheric transmission for the full wavelength range of the input spectrum and corrects this spectrum using this function. These calculations are separated from the fitting procedure, since the run of a radiative transfer code for a wide wavelength range is very time consuming. This is particularly critical if the fit is optimised by several code runs with different input parameters. For this reason, the fitting procedure should be performed for several well-defined narrow wavelength ranges that can be provided either as part of the input .rc file or in an independent FITS file (see Section 6.3). Very wide fit ranges are also not recommended due to the probable failure of the polynomial continuum fit. The narrow wavelength range of CRIRES allows one to fit the full spectrum at once. The `molecfit_calctrans` recipe writes a FITS table with the model transmission function, the telluric absorption corrected spectrum, and a quality flag.

Finally, `molecfit_correct` corrects the spectra provided as a list in its input file, preserving their format.

4.3 Major Changes between MOLECFIT v1.5.9 and v3.x

A major overhaul and refactoring of the MOLECFIT package was carried out in 2018-2020, which included changes which are backward incompatible to v1.5.9. We encourage users to move to the latest supported version, currently v4.3.

For a detailed listing of parameter name and other user input changes between the two versions, please see Section 7.

4.3.1 Driver

MOLECFIT 1.5.9 consisted of a set of standalone executables. These have been replaced with a set of *EsoRex* recipes to be processed from the *EsoRex* environment. As a consequence changes have been made to certain parameter names, parameter value formats, and how input files are declared in order to be compliant with *EsoRex* pipeline standards. Further details of the differences of the parameter formats are covered in Section 7.

4.3.2 Plot Files

MOLECFIT versions 3.x and later do not support the 1.5.9 plot file creation options, as *EsoRex* and *Reflex* offer other support methods to graph results.

4.3.3 `ASCII` Data Files

All data files must be in FITS format – i.e. `ASCII` files are not supported. This is in line with *EsoRex* pipeline standards.

4.3.4 OpenMP

MOLECFIT 3.x and later support OpenMP and will try to utilise multi cores by performing the fitting process on multiple independent ranges in parallel. The maximum number of threads to use in the executions is by default set to 1 (i.e. no parallel processing), but can be overridden by the value of the environment variable `OMP_NUM_THREAD` if that has been set.

4.4 Major Changes between MOLECFIT v3.x and v4.0

4.4.1 Removed Parameters

The following parameters have been removed:

- `CONT_BARY_RV` (not yet implemented)
- `FIT_CHIP[1234]` (no longer used)

- FIT_RANGE[1234] (no longer used)
- LBLRTM_REF_LAT (use LATITUDE_VALUE or value in LATITUDE_KEYWORD)
- LBLRTM_H1 (use ELEVATION_VALUE or value in ELEVATION_KEYWORD)
- MASK_BINARY (no longer used)
- MASK_THRESHOLD (no longer used)
- OPENMP_THREADS (no longer used)
- OUTPUT_NAME (no longer used)
- OUTPUT_PATH (no longer used)
- WLC_REF (no longer used)

4.4.2 Changed Parameters

The name or meaning of the following parameters have changed:

- CONTINUUM_N (see Tab. 6.1)
- CONTINUUM_CONST (see Tab. 6.1)
- FIT_CONTINUUM (see Tab. 6.1)
- FIT_WLC (see Tab. 6.1)
- LNFL_LINE_DB=aer_v_3.8.1.2 (updated from aer_v_3.8)
- MAP_REGIONS_TO_CHIP is new (see Tab. 6.1)
- WAVELENGTHS_FRAME is now WAVELENGTH_FRAME
- WL_GTO_MICRON is now WLG_TO_MICRON
- USE_ONLY_INPUT_PRIMARY_EXTENSION is now USE_ONLY_INPUT_PRIMARY_DATA

4.5 Instrument Specific MOLECFIT Recipes

The following pipelines now have their own independent MOLECFIT recipes:

- FORS
- KMOS
- X-Shooter

These streamlined recipes have been designed to make it easier to run MOLECFIT on the respective pipeline data products with several instrument defaults already configured.

Additional instruments will be added as they are incorporated into their pipelines.

For more information on these recipes and how to run them, please see the respective pipeline user manuals.

5 Quick Start

This chapter provides a brief description of the MOLECFIT pipeline recipes, their usage and execution.

5.1 The MOLECFIT Pipeline Recipes

The MOLECFIT pipeline consists of the following three recipes:

1. `molecfit_model`: determines the parameters that best fit the data in user selected include regions;
2. `molecfit_calctrans`: determines the telluric transmission spectrum over the full spectral range of the input spectrum;
3. `molecfit_correct`: corrects the list of input spectra using the transmission spectrum provided by `molecfit_calctrans`.

Since 2022, similar MOLECFIT recipes are now available for some specific instruments. Please see Section 4.5 for more details.

5.2 Running the MOLECFIT Pipeline Recipes

5.2.1 Getting Started with *EsoRex*

EsoRex is a command-line tool which can be used to execute the recipes of all standard VLT/VLTI instrument pipelines. With *EsoRex* in your path, the general structure of an *EsoRex* command line is

```
1> esorex [esorex options] [recipe [recipe options] [sof [sof]...]]
```

where options appearing before the recipe name are options for *EsoRex* itself, and options given after the recipe name are options which affect the recipe.

All available *EsoRex* options can be listed with the command

```
1> esorex --help
```

and the full list of available parameters of a specific recipe can be obtained with the command

```
1> esorex --help <recipe name>
```

The output of this command shows as parameter values the current setting, i.e. all modifications from a configuration file or the command line are already applied.

The listing of all recipes known to *EsoRex* can be obtained with the command

```
1> esorex --recipes
```

The last arguments of an *EsoRex* command are the so-called *set-of-frames*. A *set-of-frames* is a simple ASCII file containing a list of input data files for the recipe. Each input file is followed by an unique identifier (frame classification or frame tag), indicating the contents of this file. The input files can be given as a relative or absolute path; however, *EsoRex* allows the use of environment variables so that a common directory prefix can be abreviated. Individual lines may be commented out by using a hash character (#) in the first column. An example of a *set-of-frames* is shown in the following:

```
1> cat molecfit_model.sof
$SOF_DATA/raw/XSHOOTER_NIR_Pipeline_R71_input.fits SCIENCE
$SOF_DATA/static_calib/XSHOOTER_NIR_Pipeline_R71_molecules.fits MOLECULES
$SOF_DATA/static_calib/XSHOOTER_NIR_Pipeline_R71_range_wl_inc.fits WAVE_EXCLUDE
$SOF_DATA/static_calib/XSHOOTER_NIR_Pipeline_R71_range_pix_exc.fits PIXEL_EXCLUDE
```

These *set-of-frames* files will be created by the user using a text editor, for instance. Finally, if more than one *set-of-frames* is given on the command-line *EsoRex* concatenates them into a single *set-of-frames*.

5.3 Running MOLECFIT using *EsoRex*

MOLECFIT consists of three separate recipes: `molecfit_model`, `molecfit_calctrans`, and `molecfit_corre` each of which is an unique *EsoRex* recipe to be called from the *EsoRex* environment. If both *EsoRex* and the MOLECFIT recipes have been installed correctly, then by typing:

```
1> esorex -recipes
```

`molecfit_model`, `molecfit_calctrans`, and `molecfit_correct` will be listed as available recipes. Each recipe can be called via *EsoRex* as follows:

```
1> esorex --recipe-config= <Pars2Use.rc> <recipe_name> <SOF_filename>
```

where *Pars2Use.rc* contains the list of processing parameters (covered in detail in Section 6.2). The `SOF` (Set Of Files) contains an `ASCII` list of the names of the input data files (which are expected to be in `FITS` format) with their associated tags (see Section 6.1).

5.3.1 Environment Variables

Prior to any recipe execution the following environment variables can be defined:

TELLURICCORRDIR Defines the installation directory where TELLURICCORR may be found. If this is not set MOLECFIT will use the installation directory path defined at compile time.

TELLURICCORRDIR_DATA Defines the data directory. If this is not set MOLECFIT will use the path of the data directory created at compile time.

OMP_NUM_THREADS Defines the number of threads to use. If undefined one thread per CPU is used.

As mentioned earlier, SOF files can include environment variables in file pathnames – e.g. $HOME, as well as those listed above.

5.3.2 Input spectrum FITS header keywords

The CDELT1, CDELT2 and CDELT3 FITS header keywords are now ***deprecated*** in favour of the corresponding FITS header keywords CD1_1, CD2_2 and CD3_3. If your input spectrum is missing the latter keywords, a warning will be raised about the input spectrum. This warning can be resolved by adding these keywords to the FITS header.

5.3.3 Executing molecfit_model

The first stage of MOLECFIT is to derive a telluric model that best describes the given spectral data. This model is defined by three specific group of parameters:

1. A list of atmospheric molecules under consideration for their telluric impact in the spectral range of interest e.g. H_2O, O_2, O_3, CH_4, etc.
2. Atmospheric parameters relevant to the time and location that the input science spectra was captured, e.g. altitude-specific values of pressure, temperature, molecular abundances, etc.
3. A remaining set of parameters characterising the observational setup, e.g. various relative scales, background contributions, wavelength corrections, etc.

The first group is declared by the user either as a parameter in the relevant .rc file or in a table of a FITS file specified in the SOF file. The second one is derived via interpolation and combination of data from various sources packaged with the molecfit installation. Initial values for the third group of parameters are also provided in the .rc file or in FITS tables. The bulk of the molecfit_model process is to determine the molecular abundance for the molecules listed, as well as various of the parameters that characterise the observations.

The modelling recipe of MOLECFIT is called via esorex in the following way:

```
1> esorex --recipe-config= <Pars2Use.rc> molecfit_model <Files2Use.sof>
```

where *Pars2Use.rc* contains the list of processing parameters and, at a minimum, the SOF file contains the pathname of the science data file of interest and is tagged as SCIENCE — e.g. :

```
$HOME/raw/CRIRES_HighSNR_Telluric_input_AM1p472.fits SCIENCE
```

If successful, the recipe execution will deliver the following intermediate FITS files:

BEST_FIT_MODEL.fits Model of the best fit to the data.

GDAS.fits Initial atmospheric profiles (height, temperature, relative humidity) as a function of altitude, resulting from the combination of the profiles `GDAS_BEFORE.fits`, `GDAS_AFTER.fits`, and EMM.

GDAS_BEFORE.fits GDAS profile before the observation time.

GDAS_AFTER.fits GDAS profile after the observation time.

ATM_PROFILE_STANDARD.fits Initial atmospheric profiles, in particular, for all molecules available to `molecfit_model`.

ATM_PROFILE_COMBINED.fits Atmospheric profiles resulting from the combination of `ATM_PROFILE_STANDARD`, GDAS, and EMM data, used as initial value by `molecfit_model`.

and the following results files that explicitly define the model, and which are required as inputs to the `molecfit_calctrans` recipe:

ATM_PARAMETERS.fits The atmospheric profiles, including the volume mixing ratio for each of the relevant molecules as result of the fitting in `molecfit_model`.

MODEL_MOLECULES.fits The list of molecules used in the model, with boolean flags indicating if the column densities were to be fitted and the initial values of the column densities relative to the combined atmospheric profile.

BEST_FIT_PARAMETERS.fits The least-squares fitted parameter values.

5.3.4 Executing `molecfit_calctrans`

The second stage of MOLECFIT generates the telluric model from the set of parameters determined by the model stage, and derives data for a convolution process that can be used to correct spectra with the derived telluric transmission spectrum.

The calctrans process of molecfit is called via *EsoRex* in the following way:

```
1> esorex --recipe-config= <Pars2Use.rc> molecfit_calctans <Files2Use.sof>
```

Where *Pars2Use.rc* contains the list of processing parameters and, at a minimum, the `SOF` file contains the pathnames of the `FITS` files that uniquely define the model and are tagged as follows:

```
$MYDATADIR/SCIENCE_FILE.fits  SCIENCE
$MYDATADIR/ATM_PARAMETERS.fits  ATM_PARAMETERS
$MYDATADIR/MODEL_MOLECULES.fits  MODEL_MOLECULES
$MYDATADIR/BEST_FIT_PARAMETERS.fits  BEST_FIT_PARAMETERS
```

If successful, the recipe execution will deliver the following intermediate FITS files:

LBLRTM_RESULTS.fits The transmission spectrum as produced by the radiative transfer code (LBLRTM).

TELLURIC_DATA.fits The processed transmission spectrum including, in particular, the convolution by the instrumental line spread function.

and the following results file that exclusively contains the convoluted transmission spectrum, required as input to the third stage - the `molecfit_correct` recipe:

TELLURIC_CORR.fits	The convoluted transmission spectrum.

5.3.5 Executing `molecfit_correct`

The final stage of MOLECFIT is to correct input spectra with the assumed telluric contamination, using the convoluted transmission data generated by the `molecfit_calctrans` recipe. This MOLECFIT recipe is called via *EsoRex* in the following way:

```
1> esorex --recipe-config= <Pars2Use.rc>  molecfit_correct <Files2Use.sof>
```

where *Pars2Use.rc* contains the list of processing parameters and the *SOF* file which, at a minimum, contains the pathnames of the science data to be corrected and the TELLURIC_CORR.fits file - the convoluted telluric transmission spectrum produced by the `molecfit_calctrans` recipe – e.g. :

```
$HOME/raw/CRIRES_HighSNR_Telluric_input_AM1p472.fits SCIENCE
./TELLURIC_CORR.fits TELLURIC_CORR
```

If successful, the recipe execution will deliver the following output FITS file:

```
SCIENCE_TELLURIC_CORR_CRIRES_HighSNR_Telluric_input_AM1p472.fits
```

that is the input science frame with the original spectra replaced with that of the corrected spectra.

6 Recipe Reference

6.1 The SOF files

SOF files are ASCII files that contain a list of file path locations for specific input data. These are in the form:

file path name 1 *tagname 1*
file path name 2 *tagname 2*
.
.
.
file path name N *tagname N*

where file path names may contain environment variables and *tagnames* are specific keywords that declare the file's role in the process e.g. input science spectra, input best-fit parameters, etc. A full list of the SOF *tagnames* are given in Appendix B.4.

There are fundamentally two types of tagged files within MOLECFIT: those which are mandatory, and files which are used for optional parameter substitution. The latter is a convenient way of assigning a structured parameter that is better described as a binary table within an external FITS file instead of a long string declaration in the .rc file. A good example is the wavelength inclusion parameter. In the .rc file this is defined as string in the following format:

WAVE_INCLUDE = *ini1, end1, ..., iniN, endN*

which for a long set of ranges can be quite complex. Alternatively, if the parameter value is specified as NULL, i.e.

WAVE_INCLUDE = NULL

then MOLECFIT will look for a file path name in the SOF file with the tag WAVE_INCLUDE and, if listed, it will assume that these data exist in the specified file as FITS binary table with columns LOWER_LIMIT and UPPER_LIMIT. For example, the following string definition of the wavelength ranges in the molecfit_model .rc file:

```
WAVE_INCLUDE = 4.8417300, 4.8421100, 4.8408000, 4.841320, 4.837330,
4.837950, 4.838740, 4.83949, 4.844310, 4.845210
```

can be defined by the following parameter setting

```
WAVE_INCLUDE = NULL
```

and the associated line in the SOF file:

```
./wave_include.fits WAVE_INCLUDE
```

where wave_include.fits contains the following bin table:

LOWER_LIMIT 1D	UPPER_LIMIT 1D
4.841730000000E+00	4.842110000000E+00
4.840800000000E+00	4.841320000000E+00
4.837330000000E+00	4.837950000000E+00
4.838740000000E+00	4.839490000000E+00
4.844310000000E+00	4.845210000000E+00

This format is identical for the WAVE_EXCLUDE and PIX_INCLUDE parameters and SOF tags.

Another less straight forward example is the list of molecules to be included and/or fitted, and their values of relative column. In the .rc files this is defined by three parameters, as shown in this example:

```
LIST_MOLEC = H20, CO2, O3, CO, OCS
FIT_MOLEC = 1, 1, 1, 1, 0
REL_COL = 0.422, 1.012, 2.808, 0.809
```

The first line defines the list of molecules to use. The second lists if they are to be fitted (1) or not (0), and the third lists the values of the relative columns to be used, respectively. Any of these three parameters can be parameter substituted, but only by a single tagged file for all three (tagname = MOLECULES). The declared FITS file must contain a binary table that includes the required data in a column with a header of the parameter name being substituted.

For example, the above can be replaced with

```
LIST_MOLEC = NULL
FIT_MOLEC = NULL
REL_COL = NULL
```

with the associated line in the SOF file:

```
$FULL_DIR_PATH/molecules.fits MOLECULES
```

where molecules.fits contains the following bin table:

LIST_MOLEC 3A	FIT_MOLEC 1J	REL_COL 1D
H_20	1	4.220000000000E-01
CO_2	1	1.012000000000E+00
O_3	1	2.808000000000E+00
CO	1	8.090000000000E-01
OCS	0	1.000000000000E+00

Another example that mixes the two approaches:

```
LIST_MOLEC = H20, CO2, O3, CO, OCS
FIT_MOLEC = 1, 1, 1, 1, 0
REL_COL = NULL
```

where molecules.fits need only contain the single column table "REL_COL".

The most common tag names and their purpose are listed below. For a full list see Appendix B.4.

With regard to `molecfit_model`:

Tag Name	Type	Purpose
SCIENCE	Mandatory	Input science spectrum
WAVE_INCLUDE	Parameter Substitution	List of wavelength ranges to be included
WAVE_EXCLUDE	Parameter Substitution	List of wavelength ranges to be excluded
PIXEL_EXCLUDE	Parameter Substitution	List of pixel ranges to be excluded
MOLECULES	Parameter Substitution	List of molecules to include in the model

With regard to `molecfit_calctrans`:

Tag Name	Type	Purpose
ATM_PARAMETERS	Mandatory	Atmospheric parameter values as used in the model
MODEL_MOLECULES	Mandatory	List of molecules as used in the model
BEST_FIT_PARAMETERS	Mandatory	Best fit values as used in the model

With regard to `molecfit_correct`:

Tag Name	Type	Purpose
TELLURIC_CORR	Mandatory	Telluric convolution data necessary to correct telluric contaminated data
TELLURIC_DATA	Mandatory	In case WLC_REF is set to MODEL, as it includes the wavelength grid with the telluric lines used as reference

6.2 The .rc files

The *EsoRex* `.rc` files are only necessary in the event of using any parameter values other than the default, and even then only need to list the parameters with the non-default values. Generate an `.rc` file containing the default values as follows:

```
1> esorex --create-config= <filename> <recipe_name>
```

This can then be used as a template for the parameter settings required by the user.

6.2.1 molecfit_model default .rc parameters

The command

```
1> esorex --create-config= Default_model.rc molecfit_model
```

generates a file containing the following default parameters:

```
# File: Default_model.rc
#
# Note: This configuration file has been automatically
#       generated by the esorex (v3.13.6) program.
#
# Date: 04-May-2023 18:28:10
#
#

# --USE_ONLY_INPUT_PRIMARY_DATA
# Value=TRUE implies that only the fits primary contains the input science flux
# data.
# Value=FALSE implies that the fits extensions also contains input science
# flux data.
USE_ONLY_INPUT_PRIMARY_DATA=FALSE

# --USE_DATA_EXTENSION_AS_DFLUX
# Only valid if USE_ONLY_INPUT_PRIMARY_DATA=TRUE. The fits extension index that
# contains the
# errors of the science flux data (DFLUX). A value of 0 implies that there is
# no DFLUX.
USE_DATA_EXTENSION_AS_DFLUX=0

# --USE_DATA_EXTENSION_AS_MASK
# Only valid if USE_ONLY_INPUT_PRIMARY_DATA=TRUE. The fits extension index that
# contains the
# mask associated with the science flux data. A value of 0 implies that there
# is no mask data.
USE_DATA_EXTENSION_AS_MASK=0

# --USE_INPUT_KERNEL
# If TRUE, use the kernel library if it is provided.
USE_INPUT_KERNEL=TRUE

# --MODEL_MAPPING_KERNEL
# Mapping ’STD_MODEL/SCIENCE’ - ’MODEL_KERNEL_LIBRARY’ [string with ext_number
# comma separated (int)] :
# If set to NULL, check if the TAG[MODEL_MAPPING_KERNEL] FITS BINTABLE values
# is provided.
# The FITS BINTABLE have to one column [KERNEL_LIBRARY_EXT].
MODEL_MAPPING_KERNEL=NULL

# --LIST_MOLEC
# List of molecules to be included in the model. Represented as a comma
# separated
# string of molecule names, e.g. "H2O,CO2,O3".
# If set to NULL, the input TAG[MOLECULES] FITS BINTABLE values have to be
# provided
# where the FITS BINTABLE specified contains the three columns:
# LIST_MOLEC; FIT_MOLEC; and REL_COL.
LIST_MOLEC=NULL

# --FIT_MOLEC
# List of flags that specify which of the listed molecules are to be fitted for.
# Flag=1 implies yes. Flag=0 implies no. Represented as a string of comma
# separated
# integers in the same order as the listed molecules. For example: if
# LIST_MOLEC="H2O,CO2,O3", then
# FIT_MOLEC="1,0,1" implies that only H2O and O3 should be fitted for.
# If set to NULL, the input TAG[MOLECULES] FITS BINTABLE values have to be
# provided where the FITS
# BINTABLE specified contains the three columns: LIST_MOLEC; FIT_MOLEC; and
# REL_COL.
FIT_MOLEC=NULL

# --REL_COL
# List of the intial values of fitting of the molecular columns expressed
# relatively to the input
#  ATM profile columns. Represented as a comma separated list of doubles in
# the same order as the
# listed molecules. For example, if LIST_MOLEC="H2O,CO2,O3", then
# REL_COL="1.0,1.2,0.8"
# implies that H2O, CO2 and O3 have initial relative values of 1.0, 1.2 and
# 0.8 respectively.
# If set to NULL, the input TAG[MOLECULES] FITS BINTABLE values have to be
# provided where the FITS
# BINTABLE specified contains the three columns: LIST_MOLEC; FIT_MOLEC; and
# REL_COL.
REL_COL=NULL

# --WAVE_INCLUDE
# Wavelength ranges to be included. Represented as a string of comma separated
# doubles in pairs
# specifying the start and end wavelengths of a range. The wavelength units
# are always in microns.
# For example a KMOS sample data in the range of 1.11um to 1.67um may have
# WAVE_INCLUDE="1.773,1.78633,1.79098,1.80434,1.187691,1.189937" to represent
# three inclusion regions:
# [1.773,1.78633], [1.79098,1.80434] and [1.187691,1.189937].
# If set to NULL, molecfit will check if the TAG[WAVE_INCLUDE] FITS BINTABLE
# values is provided where
# the FITS BINTABLE specified has the two columns: LOWER_LIMIT; and
# UPPER_LIMIT.
WAVE_INCLUDE=NULL

# --WAVE_EXCLUDE
# Wavelength ranges to be excluded. Represented as a string of comma separated
# doubles in pairs
# specifying the start and end wavelengths of a range. The wavelength units
# are always in microns.
# as the input science data. For example a KMOS sample data in the range of
# 1.11um to 1.67um may have
# WAVE_EXCLUDE="1.773,1.78633,1.79098,1.80434,1.187691,1.189937" to represent
# three exclusion regions:
# [1.773,1.78633], [1.79098,1.80434] and [1.187691,1.189937].
# If set to NULL, molecfit will check if the TAG[WAVE_EXCLUDE] FITS BINTABLE
# values is provided where
# the FITS BINTABLE specified has the two columns: LOWER_LIMIT; and
# UPPER_LIMIT.
WAVE_EXCLUDE=NULL

# --PIXEL_EXCLUDE
# Pixel ranges to be excluded. Represented as a string of comma separated
# integers in pairs specifying the
# start and end pixel of a range. For example:
# PIXEL_EXCLUDE="54,128,512,514,1020,1024" represents three
# exclusion regions: [54,128], [512,514] and [1020,1024].
# If set to NULL, molecfit will check if the TAG[PIXEL_EXCLUDE] FITS BINTABLE
# values is provided where the
# FITS BINTABLE specified has the two columns: LOWER_LIMIT; and UPPER_LIMIT.
PIXEL_EXCLUDE=NULL

# --TELLURICCORR_PATH
# Installation directory.
TELLURICCORR_PATH=TELLURICCORR_PARAMETER_DEFAULT

# --TELLURICCORR_DATA_PATH
# Data directory.
TELLURICCORR_DATA_PATH=TELLURICCORR_PARAMETER_DEFAULT

# --TMP_PATH
# Temporary directory.
TMP_PATH=TELLURICCORR_PARAMETER_DEFAULT

# --SILENT_EXTERNAL_BINS
# Silent the output of the external binaries.
SILENT_EXTERNAL_BINS=TRUE

# --TRANSMISSION
# Type of input spectrum : 0 = Emission(radiance); 1 = Transmission.
TRANSMISSION=TRUE

# --COLUMN_LAMBDA
# Wavelength column ('NULL' can be used if the file is an image and that
# the data are in the primary
# (data are given by the FITS header keywords [CRVAL1=wave_ini, CD1_1=step])
# f CD1_1 is absent, then the DEPRECATED CDELT1 keyword will be used.
COLUMN_LAMBDA=lambda
```

```
# --COLUMN_FLUX
# Flux column.
COLUMN_FLUX=flux

# --COLUMN_DFLUX
# Flux error column (Avoided by writing ’NULL’) : 1-sigma error on the flux.
COLUMN_DFLUX=NULL

# --COLUMN_MASK
# Mask column (Avoided by writing ’NULL’) : Indicates if a pixel is invalid.
COLUMN_MASK=NULL

# --DEFAULT_ERROR
# Default error relative to mean for the case that the error column
# is not provided.
DEFAULT_ERROR=0.01

# --WLG_TO_MICRON
# Multiplicative factor applied to the wavelength to express is in micron.
# E.g.: if wavelength is given in nm, the value should be 0.001.
WLG_TO_MICRON=1.0

# --WAVELENGTH_FRAME
# Wavelength in vacuum                                      = ’VAC’.
# Wavelength in air    with the observatory reference frame = ’AIR’.
# Wavelength in vacuum with another         reference frame = ’VAC_RV’.
# Wavelength in air    with another         reference frame = ’AIR_RV’.
#   (typically the  sun or the barycenter of the solar system).
# In the latter case, the radial velocity of the observatory relative
#   to the external reference frame must be provided in the parameter obs_RV.
WAVELENGTH_FRAME=VAC

# --OBS_ERF_RV_KEY
# The radial velocity of the observatory in km/s
# relative to the external reference frame;
# It is positive if the distance between the science target and the Earth
# increases along the line-of-sight to the science target.
# It must be provided if MF_PARAMETERS_WAVELENGTH_FRAME = ’VAC_RV’ or ’AIR_RV’.
OBS_ERF_RV_KEY=NONE

# --OBS_ERF_RV_VALUE
# If OBS_ERF_RV_KEYWORD==’NONE’ take this value.
# It must be provided if MF_PARAMETERS_WAVELENGTH_FRAME = ’VAC_RV’ or ’AIR_RV’.
OBS_ERF_RV_VALUE=0.0

# --CLEAN_MODEL_FLUX
# Set model flux to 0 for non-fitted pixels.
CLEAN_MODEL_FLUX=FALSE

# --FTOL
# Relative chi-square convergence criterion.
```

```
FTOL=1e-10

# --XTOL
# Relative parameter convergence criterion.
XTOL=1e-10

# --FLUX_UNIT
# Conversion of fluxes from phot/(s*m2*mum*as2) (emission spectrum only)
#    to flux unit of observed spectrum:
# 0: phot / (s *  m^2 * mum * as^2) [no conversion]
# 1:    W / (     m^2 * mum * as^2)
# 2:  erg / (s * cm^2 *   A * as^2)
# 3:  mJy / (                 as^2)
# For other units, the conversion factor has to be considered
#  as constant term of the continuum fit.
FLUX_UNIT=0

# --FIT_TELESCOPE_BACKGROUND
# Fit of telescope background -- 1 = yes; 0 = no (emission spectrum only).
FIT_TELESCOPE_BACKGROUND=TRUE

# --TELESCOPE_BACKGROUND_CONST
# Initial value for telescope background fit.
TELESCOPE_BACKGROUND_CONST=0.1

# --FIT_CONTINUUM
# Comma deliminated string of flags (1=true, 0=false) for fitting continuum in
# specific regions.
# If set to NULL, check if the TAG[WAVE_INCLUDE] points to a FITS BINTABLE
# with column CONT_FIT_FLAG provided.
FIT_CONTINUUM=1

# --CONTINUUM_N
# Polynomial order for continuum model for each region. Presented as a comma
# deliminated string.
# If set to NULL, check if the TAG[WAVE_INCLUDE] points to a FITS BINTABLE
# with column CONT_POLY_ORDER provided.
CONTINUUM_N=0

# --CONTINUUM_CONST
# Initial constant term for continuum fit (valid for all fit ranges)
# [emission spectrum: about 1 for correct flux_unit].
CONTINUUM_CONST=1.0

# --MAP_REGIONS_TO_CHIP
# Comma deliminated string of chip indices that each range is associated with.
# If set to NULL, check if the TAG[WAVE_INCLUDE] points to a FITS BINTABLE
# with column MAPPED_TO_CHIP provided.
MAP_REGIONS_TO_CHIP=1

# --FIT_WLC
# Flags for including regions in wavelength corrections.
# If set to NULL, check if the TAG[WAVE_INCLUDE] points to
# a FITS BINTABLE with column WLC_FIT_FLAG provided.
FIT_WLC=0

# --WLC_N
# Polynomial degree of the refined wavelength solution.
WLC_N=1

# --WLC_CONST
# Initial constant term for wavelength adjustment
# (shift relative to half wavelength range).
WLC_CONST=0.0

# --FIT_RES_BOX
# Fit resolution by Boxcar LSF.
FIT_RES_BOX=TRUE

# --RES_BOX
# Initial value for FWHM of Boxcar rel. to slit width
# at the centre of the spectrum.
RES_BOX=1.0

# --FIT_RES_GAUSS
# Fit resolution by Gaussian.
FIT_RES_GAUSS=TRUE

# --RES_GAUSS
# Initial value for FWHM of the Gaussian in pixels
# at the centre of the spectrum.
RES_GAUSS=1.0

# --FIT_RES_LORENTZ
# Fit resolution by Lorentzian.
FIT_RES_LORENTZ=TRUE

# --RES_LORENTZ
# Initial value for FWHM of the Lorentz in pixels
# at the centre of the spectrum.
RES_LORENTZ=1.0

# --KERNMODE
# Voigtian profile approximation instead of independent Gaussian and
# Lorentzian?.
KERNMODE=FALSE

# --KERNFAC
# Size of Voigtian/Gaussian/Lorentzian kernel in FWHM.
KERNFAC=3.0

# --VARKERN
# Does the kernel size increase linearly with wavelength?.
VARKERN=FALSE

# --OBSERVING_DATE_KEYWORD
# Observing date in years or MJD in days (not string).
OBSERVING_DATE_KEYWORD=MJD-OBS

# --OBSERVING_DATE_VALUE
# If OBSERVING_DATE_KEYWORD==’NONE’ take this value.
OBSERVING_DATE_VALUE=-1.0

# --UTC_KEYWORD
# UTC in s.
UTC_KEYWORD=UTC

# --UTC_VALUE
# If UTC_KEYWORD==’NONE’ take this value.
UTC_VALUE=-1.0

# --TELESCOPE_ANGLE_KEYWORD
# Telescope altitude angle in deg.
TELESCOPE_ANGLE_KEYWORD=ESO TEL ALT

# --TELESCOPE_ANGLE_VALUE
# If TELESCOPE_ANGLE_KEYWORD==’NONE’ take this value.
TELESCOPE_ANGLE_VALUE=90.0

# --RELATIVE_HUMIDITY_KEYWORD
# Relative humidity in %.
RELATIVE_HUMIDITY_KEYWORD=ESO TEL AMBI RHUM

# --RELATIVE_HUMIDITY_VALUE
# If RELATIVE_HUMIDITY_KEYWORD==’NONE’ take this value.
RELATIVE_HUMIDITY_VALUE=15.0

# --PRESSURE_KEYWORD
# Pressure in hPa.
PRESSURE_KEYWORD=ESO TEL AMBI PRES START

# --PRESSURE_VALUE
# If PRESSURE_KEYWORD==’NONE’ take this value.
PRESSURE_VALUE=750.0

# --TEMPERATURE_KEYWORD
# Ambient temperature in deg C.
TEMPERATURE_KEYWORD=ESO TEL AMBI TEMP

# --TEMPERATURE_VALUE
# If TEMPERATURE_KEYWORD==’NONE’ take this value.
TEMPERATURE_VALUE=15.0
```

```
# --MIRROR_TEMPERATURE_KEYWORD
# Mirror temperature in deg C.
MIRROR_TEMPERATURE_KEYWORD=ESO TEL TH M1 TEMP

# --MIRROR_TEMPERATURE_VALUE
# If MIRROR_TEMPERATURE_KEYWORD==’NONE’ take this value.
MIRROR_TEMPERATURE_VALUE=15.0

# --ELEVATION_KEYWORD
# Elevation above sea level in m (default is Paranal: 2635. m).
ELEVATION_KEYWORD=ESO TEL GEOELEV

# --ELEVATION_VALUE
# If ELEVATION_KEYWORD==’NONE’ take this value.
ELEVATION_VALUE=2635.0

# --LONGITUDE_KEYWORD
# Longitude (default is Paranal: -70.4051 deg).
LONGITUDE_KEYWORD=ESO TEL GEOLON

# --LONGITUDE_VALUE
# If LONGITUDE_KEYWORD==’NONE’ take this value.
LONGITUDE_VALUE=-70.4051

# --LATITUDE_KEYWORD
# Latitude (default is Paranal: -24.6276 deg).
LATITUDE_KEYWORD=ESO TEL GEOLAT

# --LATITUDE_VALUE
# If LATITUDE_KEYWORD==’NONE’ take this value.
LATITUDE_VALUE=-24.6276

# --SLIT_WIDTH_KEYWORD
# Slit width in arcsec (taken from FITS header if present).
SLIT_WIDTH_KEYWORD=ESO INS SLIT1 WID

# --SLIT_WIDTH_VALUE
# If SLIT_WIDTH_KEYWORD==’NONE’ take this value.
SLIT_WIDTH_VALUE=0.4

# --PIX_SCALE_KEYWORD
# Pixel scale in arcsec (taken from this file only).
PIX_SCALE_KEYWORD=NONE

# --PIX_SCALE_VALUE
# If PIX_SCALE_KEYWORD==’NONE’ take this value.
PIX_SCALE_VALUE=0.086

# --REFERENCE_ATMOSPHERIC
# Reference atmospheric profile. Possible values:
# - equ.fits (default; equatorial atmosphere, valid for Paranal);
# - tro.fits (tropical atmosphere);
# - std.fits (standard atmosphere);
# - Other file located in :
#   ({TELLURICCORR_DATA_PATH}/profiles/mipas/).
REFERENCE_ATMOSPHERIC=equ.fits

# --GDAS_PROFILE
# Specify which GDAS profile to use. Possible values:
# - 'auto',  automatic retrieval of the GDAS profiles
# (P[hPa] HGT[m] T[K] RELHUM[%]) close in time to the
# observation and in location to the observatory. If
# the files are not on disk and there is no internet
# connection, the average profile is taken from
# share/molecfit/data/profiles/lib corresponding to
# the month of the observation (GDAS_t0_s1.fits for
# Dec/Jan, GDAS_t0_s2.fits for Feb/Mar, etc)
# See Sec. 8.1.4 of the molecfit manual for more info.
# - 'null', use the profile in the SOF with tag GDAS.
# If there is no profile in the SOF, the behaviour
# is the same as GDAS_PROF=auto.
# - 'none', use the average profile taken from
# share/molecfit/data/profiles/lib corresponding to
# the month of observation (see 'auto' description).
# - 'directory/file.fits', use the specified path and
# filename as the GDAS profile. Either an absolute path
# (starting with '/') or a relative path may be used,
# however a relative path is preferred, since only
# the first 40 char of the path and filename are
# copied to the FITS header.
GDAS_PROFILE=auto

# --LAYERS
# Grid of layer heights for merging ref_atm and GDAS profile.
# Fixed grid = CPL_TRUE and natural grid = CPL_FALSE.
LAYERS=TRUE

# --EMIX
# Upper mixing height in km for considering data of a local meteo station.
# If emix is below geoelev, rhum, pres, and temp are not used
# for modifying the corresponding profiles.
EMIX=5.0

# --PWV
# PWV value in mm for the input water vapor profile.
# The merged profile composed of ref_atm, GDAS, and local meteo data
# will be scaled to this value if pwv > 0 (default: -1 -> no scaling).
PWV=-1.0

# --LNFL_LINE_DB
# File name of the line list (must be stored in the directory :
# ({TELLURICCORR_DATA_PATH}/hitran/).
```

```
LNFL_LINE_DB=aer_v_3.8.1.2

# --LNFL_LINE_DB_FORMAT
# Format of the line file: gives the length in terms of characters per line.
LNFL_LINE_DB_FORMAT=100.0

# --LBLRTM_ICNTNM
# Continua and Rayleigh extinction [0,1,2,3,4,5].
LBLRTM_ICNTNM=5

# --LBLRTM_IAERSL
# Aerosols [0,1].
LBLRTM_IAERSL=0

# --LBLRTM_MPTS
# Number of optical depth values.
LBLRTM_MPTS=5

# --LBLRTM_NPTS
# Number of values for each panel.
LBLRTM_NPTS=5

# --LBLRTM_V1
# Beginning wavenumber value for the calculation.
LBLRTM_V1=1.9

# --LBLRTM_V2
# Ending wavenumber value for the calculation.
LBLRTM_V2=2.4

# --LBLRTM_SAMPLE
# Number of sample points per mean halfwidth [between 1 to 4, default=4].
LBLRTM_SAMPLE=4

# --LBLRTM_ALFAL0
# Average collision broadened halfwidth [cm-1/atm].
LBLRTM_ALFAL0=0.0

# --LBLRTM_AVMASS
# Average molecular mass [amu] for Doppler halfwidth.
LBLRTM_AVMASS=0.0

# --LBLRTM_DPTMIN
# Minimum molecular optical depth below which lines will be rejected.
LBLRTM_DPTMIN=0.0002

# --LBLRTM_DPTFAC
# Factor multiplying molecular continuum optical depth.
LBLRTM_DPTFAC=0.001

# --LBLRTM_TBOUND
# Temperature of boundary [K].
LBLRTM_TBOUND=0.0

# --LBLRTM_SREMIS1
# Emissivity coefficient 1.
LBLRTM_SREMIS1=0.0

# --LBLRTM_SREMIS2
# Emissivity coefficient 2.
LBLRTM_SREMIS2=0.0

# --LBLRTM_SREMIS3
# Emissivity coefficient 3.
LBLRTM_SREMIS3=0.0

# --LBLRTM_SRREFL1
# Reflectivity coefficient 1.
LBLRTM_SRREFL1=0.0

# --LBLRTM_SRREFL2
# Reflectivity coefficient 2.
LBLRTM_SRREFL2=0.0

# --LBLRTM_SRREFL3
# Reflectivity coefficient 3.
LBLRTM_SRREFL3=0.0

# --LBLRTM_MODEL
# Atmospheric profile [0,1,2,3,4,5,6].
LBLRTM_MODEL=0

# --LBLRTM_ITYPE
# Type of path [1,2,3].
LBLRTM_ITYPE=3

# --LBLRTM_NOZERO
# Zeroing of small amounts of absorbers [0,1].
LBLRTM_NOZERO=0

# --LBLRTM_NOPRNT
# Do not print output? [0,1].
LBLRTM_NOPRNT=0

# --LBLRTM_IPUNCH
# Write out layer data to TAPE7 [0,1].
LBLRTM_IPUNCH=0

# --LBLRTM_RE
# Radius of earth [km].
LBLRTM_RE=0.0
```

```
# --LBLRTM_HSPACE
# Altitude definition for space [km].
LBLRTM_HSPACE=120.0

# --LBLRTM_H2
# Upper height limit [km].
LBLRTM_H2=0.0

# --LBLRTM_RANGE
# Length of a straight path from H1 to H2 [km].
LBLRTM_RANGE=0.0

# --LBLRTM_BETA
# Earth centered angle from H1 to H2 [degrees].
LBLRTM_BETA=0.0

# --LBLRTM_LEN
# Path length [0,1].
LBLRTM_LEN=0

# --LBLRTM_HOBS
# Height of observer.
LBLRTM_HOBS=0.0

# --LBLRTM_AVTRAT
# Maximum Voigt width ratio across a layer.
LBLRTM_AVTRAT=2.0

# --LBLRTM_TDIFF1
# Maximum layer temperature difference at ALTD1 [K].
LBLRTM_TDIFF1=5.0

# --LBLRTM_TDIFF2
# Maximum layer temperature difference at ALTD2 [K].
LBLRTM_TDIFF2=8.0

# --LBLRTM_ALTD1
# Altitude of TDIFF1 [km].
LBLRTM_ALTD1=0.0

# --LBLRTM_ALTD2
# Altitude of TDIFF2 [km].
LBLRTM_ALTD2=0.0

# --LBLRTM_DELV
# Number of wavenumbers [cm-1] per major division.
LBLRTM_DELV=1.0

# --EXPERT_MODE
# If set to true, will check if TAG[INIT_FIT_PARAMETERS] points to a fits file
# with a bintable of parameter values to use as initial values for the
# fitting process.
EXPERT_MODE=FALSE

# --CHIP_EXTENSIONS
# Flag that determines if image extensions are to be treated as independent
# science data to be fitted for independently or as CHIP specific subranges
# of a single observation to be fitted for as a single combined spectrum.
# Value = TRUE implies to treat as CHIPS to be combined. Value = FALSE
# implies
# to treat as independent. [FALSE].
CHIP_EXTENSIONS=FALSE

#
# End of file
```

6.2.2 molecfit_correct default .rc parameters

The command

```
1> esorex --create-config= Default_correct.rc  molecfit_correct
```

generates a file containing the following:

```
# File: Default_correct.rc
#
# Note: This configuration file has been automatically
#       generated by the esorex (v3.13.6) program.
#
# Date: 04-May-2023 18:27:29
#
#

# --USE_ONLY_INPUT_PRIMARY_DATA
# Value=TRUE implies that only the fits primary contains the input science flux
# data.
# Value=FALSE implies that the fits extensions also contains input science
# flux data.
USE_ONLY_INPUT_PRIMARY_DATA=FALSE

# --USE_DATA_EXTENSION_AS_DFLUX
# Only valid if USE_ONLY_INPUT_PRIMARY_DATA=TRUE. The fits extension index that
# contains the
# errors of the science flux data (DFLUX). A value of 0 implies that there is
# no DFLUX.
USE_DATA_EXTENSION_AS_DFLUX=0

# --USE_DATA_EXTENSION_AS_MASK
# Only valid if USE_ONLY_INPUT_PRIMARY_DATA=TRUE. The fits extension index that
# contains the
# mask associated with the science flux data. A value of 0 implies that there
# is no mask data.
USE_DATA_EXTENSION_AS_MASK=0

# --SUPPRESS_EXTENSION
# Suppress arbitrary filename extension : TRUE (apply) or FALSE (don’t apply).
SUPPRESS_EXTENSION=FALSE

# --MAPPING_CORRECT
# Mapping ’SCIENCE’ - ’TELLURIC_CORR’ [string with ext_number comma separated
# (int)] :
# If set to NULL, check if the TAG[MAPPING_CORRECT] FITS BINTABLE value is
# provided.
# The FITS BINTABLE have to one column [TELLURIC_CORR_EXT].
MAPPING_CORRECT=NULL

# --WLC_REF
# Indicates that the reference for the wavelength calibration :
# - If it is set to ’DATA’,  is the input  data.
# - If it is set to ’MODEL’, is the output model, and TELLURIC_DATA file is
#required. This feature is currently available for FITS binary table inputs
#only.
WLC_REF=DATA

# --CHIP_EXTENSIONS
# Flag that determines if image extensions are to be treated as independent
# science data to be fitted for independently or as CHIP specific subranges
# of a single observation to be fitted for as a single combined spectrum.
# Value = TRUE implies to treat as CHIPS to be combined. Value = FALSE
# implies
# to treat as independent. [FALSE].
CHIP_EXTENSIONS=FALSE

#
# End of file
```

6.3 The Parameters

Here we consider the MOLECFIT parameters in more detail.

6.3.1 Model Recipe

Table 6.1 describes the individual model recipe parameters, in the order in which they appear in the `.rc` file. Description of the LBRTM parameters is available in

```
molecfit-kit-<version>/molecfit\_third\_party-<version>lblrtm/docs/html/
lblrtm_instructions_frame.html.
```

Table 6.1: Model recipe parameters.

Parameter	Description
USE_ONLY_INPUT_PRIMARY_DATA	Indicates where the science data are located. Type = Boolean; Default = FALSE Value=TRUE: Only the primary data array contains the input science flux data. Value=FALSE: The FITS extensions also contain input science flux.

USE_DATA_EXTENSION_AS_DFLUX	Specifies the FITS extension index that contains the 1-sigma error on the science flux (DFLUX).
	Type = Integer; Default = 0
	Only valid if USE_ONLY_INPUT_PRIMARY_DATA = TRUE.
	Value = 0: There is no DFLUX available.

USE_DATA_EXTENSION_AS_MASK	The FITS extension index that contains the mask associated with the science flux data. Type = Integer; Default = 0 Only valid if USE_ONLY_INPUT_PRIMARY_DATA = TRUE. Value = 0: There is no mask data.
USE_INPUT_KERNEL	Type = Boolean; Default = TRUE Value = TRUE: use a user-provided kernel library, in which case the FITS file with the kernel library must be given in the SOF, with the tag KERNEL_LIBRARY or MODEL_KERNEL_LIBRARY
MODEL_MAPPING_KERNEL	Type = String with comma-separated integers that represent the extension numbers; Default = NULL If USE_INPUT_KERNEL = FALSE, this parameter is unused. If USE_INPUT_KERNEL = TRUE, value is a list of extensions to map from the kernel library file (MODEL_KERNEL_LIBRARY or KERNEL_LIBRARY) to the extensions of the input file (STD_MODEL or SCIENCE). e.g. Assuming an input science file has extensions "1,2,3,4", then Value="2,2,2,2" means that extension 2 of the kernel library is used for extensions 1, 2, 3 and 4 of the input spectrum. i.e. mapping 2 to 1, 2 to 2, 2 to 3, and 2 to 4.

continuation of Table 6.1

Parameter	Description
	Similarly, Value="1,2,3,4" would map 1 to 1, 2 to 2, 3 to 3, and 4 to 4. *Description continues on the next page...*

continuation of Table 6.1

Parameter	Description
	If Value = NULL check if the TAG [MODEL_MAPPING_KERNEL] FITS BINTABLE values are provided. The FITS BINTABLE must have one column [KERNEL_LIBRARY_EXT]. See Appendix A.1 for more details.
LIST_MOLEC	List of molecules to be included in the model. Type = (N_val : nmolec). Comma-separated string of molecule names; Default = NULL. e.g. Value="H20,CO2,O3" If Value = NULL, the input TAG [MOLECULES] FITS BINTABLE must be provided and must contain the three columns: [LIST_MOLEC, FIT_MOLEC, REL_COL].
FIT_MOLEC	List of flags specifying which of the molecules in LIST_MOLEC to fit. Type = (N_val: nmolec). String containing comma-separated flags (1=fit; 0=do not fit) in the same order as the molecules listed in LIST_MOLEC; Default = NULL e.g. if LIST_MOLEC="H20,CO2,O3", then FIT_MOLEC="1,0,1" implies that only H_2O and O_3 should be fitted. If Value = NULL, the input TAG [MOLECULES] FITS BINTABLE must be provided and must contain the three columns [LIST_MOLEC, FIT_MOLEC, REL_COL].
REL_COL	List of the initial values of fitting of the molecular column densities expressed relative to the input ATM profile column densities. Type = String containing a list of comma-separated doubles, in the same order as the listed molecules; Default = NULL. e.g. if LIST_MOLEC = "H20,CO2,O3", then REL_COL = "1.0,1.2,0.8" implies that H_2O, CO_2 and O_3 have initial relative values of 1.0, 1.2, and 0.8 respectively. If Value = NULL, the input TAG[MOLECULES] FITS BINTABLE must be provided and must have three columns [LIST_MOLEC,FIT_MOLEC,REL_COL].
WAVE_INCLUDE	Wavelength ranges to be included. Type = String with comma separated wavelengths (in microns) in pairs of Double; Default = NULL. e.g. WAVE_INCLUDE="1.773,1.78633,1.79098,1.80434,1.187691, 1.189937" specifies the ranges [1.773,1.78633], [1.79098,1.80434], and [1.187691,1.189937]. If set to NULL, check if the TAG[WAVE_INCLUDE] FITS BINTABLE values are provided. The FITS BINTABLE must have two columns [LOWER_LIMIT,UPPER_LIMIT].

continuation of Table 6.1

Parameter	Description
WAVE_EXCLUDE	Wavelength ranges excluded. Type = String containing a list of comma-separated pairs of wavelengths (Doubles in microns); Default = NULL e.g. WAVE_EXCLUDE="1.773,1.78633,1.79098,1.80434,1.187691, 1.189937" specifies the ranges [1.773,1.78633], [1.79098,1.80434], [1.187691,1.189937]. If Value = NULL, check if the TAG[WAVE_EXCLUDE] FITS BINTABLE values are provided. The FITS BINTABLE must have two columns [LOWER_LIMIT,UPPER_LIMIT].
PIXEL_EXCLUDE	Pixel ranges to be excluded. Type = String containing comma-separated Integer pairs of pixel numbers; Default = NULL e.g. PIXEL_EXCLUDE="54,128,512,514,1020,1024" specifies the ranges [54,128], [512,514], [1020,1024]. If Value = NULL, check if the TAG[PIXEL_EXCLUDE] FITS BINTABLE values are provided. The FITS BINTABLE must have two columns [LOWER_LIMIT,UPPER_LIMIT].
TELLURICCORR_PATH	Installation directory. Default = TELLURICCORR_PARAMETER_DEFAULT
TELLURICCORR_DATA_PATH	Data directory. Default = TELLURICCORR_PARAMETER_DEFAULT
TMP_PATH	Temporary directory. Default = TELLURICCORR_PARAMETER_DEFAULT
SILENT_EXTERNAL_BINS	Supress the output of external binaries. Default = TRUE

TRANSMISSION	Type of input spectrum.
	Type = Boolean; Default = TRUE
	Value = TRUE: Transmission spectrum;
	Value = FALSE: Emission (radiance) spectrum.

COLUMN_LAMBDA	Wavelength column. Default = lambda Value = 'NULL' is used if the file is an image and the data are in the primary data array, in which case the wavelength scale is calculated from the FITS header keywords CRVAL1=<initial wavelength> and CD1_1=<wavelength step>. Mandatory if the parameter USE_ONLY_INPUT_PRIMARY_DATA = FALSE.
COLUMN_FLUX	Flux column. Default = flux Set Value = NULL if the file is an image and the data are in the primary data array. Mandatory if USE_ONLY_INPUT_PRIMARY_DATA = FALSE.

continuation of Table 6.1

Parameter	Description
COLUMN_DFLUX	1-sigma error on the flux. Optional parameter. Default = NULL Set Value = NULL if the file is an image.
COLUMN_MASK	Mask column; indicates if a pixel is invalid. Optional. Default = NULL Set Value = NULL if the file is an image
DEFAULT_ERROR	Default error relative to the mean. Only used if the error column is not provided. Default = 0.01
WLG_TO_MICRON	Multiplicative factor applied to the wavelength to express it in micrometres. e.g. if the wavelength is given in nm, set Value = 0.001. Default = 1.0
WAVELENGTH_FRAME	Wavelength reference frame. Default = VAC Value = ’VAC’: Wavelength in vacuum; Value = ’AIR’: Wavelength in air with the observatory reference frame; Value = ’AIR_RV’: Wavelength in air with the another reference frame; Value = ’VAC_RV’: Wavelength in vacuum with another reference frame, typically the sun or the barycenter of the solar system. In the latter case, the radial velocity of the observatory relative to the external reference frame must be provided in the parameter OBS_ERF_RV_VALUE or provided by the keyword given as a parameter to OBS_ERF_RV_KEY.

OBS_ERF_RV_KEY	The radial velocity of the observatory in km/s relative to the external reference frame. It is positive if the distance between the science target and the Earth increases along the line-of-sight to the science target.
	This parameter must be provided if
	WAVELENGTH_FRAME = ’VAC_RV’ or ’AIR_RV’.
	Default = NONE

OBS_ERF_RV_VALUE	If OBS_ERF_RV_KEY==’NONE’ use this value.
	This parameter must be provided if
	WAVELENGTH_FRAME = ’VAC_RV’ or ’AIR_RV’.
	Default = 0.0

CLEAN_MODEL_FLUX	Set model flux to 0 for non-fitted pixels. Default = FALSE
FTOL	Relative chi-square convergence criterion. Default = 1e-10
XTOL	Relative parameter convergence criterion. Default = 1e-10

continuation of Table 6.1

Parameter	Description

FLUX_UNIT	Conversion of fluxes from $phot/(s*m^2*mum*as^2)$ (emission spectrum only) to flux unit of observed spectrum:
	Value = 0: $phot/(s*m^2*mum*as^2)$ [no conversion]
	Value = 1: $W/(m^2*mum*as^2)$
	Value = 2: $erg/(s*cm^2*A*as^2)$
	Value = 3: mJy / (as^2)
	For other units, the conversion factor has to be considered as a constant term of the continuum fit.
	Default = 0

FIT_TELESCOPE_BACKGROUND	Specifies whether to include fitting for the telescope background.
	Type = Boolean; Default = TRUE.
	Value = TRUE: yes;
	Value = FALSE: no (fit emission spectrum only).

TELESCOPE_BACKGROUND_CONST	Initial value for telescope background constant (relevant for emission spectrum only). Default = 0.1
FIT_CONTINUUM	Flag to enable/disable the polynomial fit of the continuum. Type: String with comma-separated integers of value 0 or 1, with each number corresponding to a range. Default = "1" for each range - e.g. if there are four ranges, the default value is FIT_CONTINUUM="1,1,1,1" e.g. If there are 4 ranges, FIT_CONTINUUM="1,1,0,1" requests continuum to be fitted for ranges 1, 2, and 4, with no continuum fitting for range 3. If FIT_CONTINUUM="NULL", the values are read from the TAG[WAVE_INCLUDE] FITS BINTABLE from a column named CONT_FIT_FLAG.
CONTINUUM_N	Degree of the polynomial continuum fit to use per range. Type = String of comma-separated integers, each giving the order of the polynomial to be used in the fit (one per range). Default = "1" e.g. If there are 4 ranges to fit, CONTINUUM_N="3,3,3,3" means that all four ranges will use an order 3 polynomial to fit the continuum. If CONTINUUM_N="NULL", the values are instead read from the TAG[WAVE_INCLUDE] FITS BINTABLE from a column named CONT_POLY_ORDER.
CONTINUUM_CONST	Initial constant term for continuum fit (valid for all fit ranges). Must be set to the same order of magnitude as the flux. Default = 1.0

continuation of Table 6.1

Parameter	Description
FIT_WLC	Defines whether a region is used for the refinement of the wavelength calibration solution. Type = String containing a comma-separated list of integers of value 0 or 1. Default = "1" for each range. If, for example, there are four ranges, the default value is FIT_WLC="1,1,1,1" Value = 1: the region is used for the refinement; Value = 0: the region is not used. If FIT_WLC="NULL", the values are read from the TAG[WAVE_INCLUDE] FITS BINTABLE from a column named WLC_FIT_FLAG. Note that we do not explicitly specify that a chip is to be fitted with this parameter, but rather whether the ranges on the chip are to be fitted.
WLC_N	Polynomial degree of the refined wavelength solution. Default = 1
WLC_CONST	Initial constant term for wavelength adjustment – i.e. shift relative to half wavelength range. Default = 0.0
FIT_RES_BOX	Fit the width of a Boxcar LSF. Default = TRUE
RES_BOX	Initial value for FWHM of Boxcar relative to slit width at the centre of the spectrum. Default = 1.0
FIT_RES_GAUSS	Fit the FWHM of a Gaussian LSF. Default = TRUE
RES_GAUSS	Initial value for FWHM of the Gaussian in pixels, at the centre of the spectrum. Default = 1.0
FIT_RES_LORENTZ	Fit the FWHM of a Lorentzian LSF. Default = TRUE
RES_LORENTZ	Initial value for FWHM of the Lorentz in pixels at the centre of the spectrum. Default = 1.0
KERNMODE	Use a Voigtian profile approximation instead of independent Gaussian and Lorentzian. Default = FALSE
KERNFAC	Size of Voigtian / Gaussian / Lorentzian kernel in FWHM. Default = 3.0
VARKERN	Does the kernel size increase linearly with wavelength? Default = FALSE

continuation of Table 6.1

Parameter	Description
OBSERVING_DATE_KEYWORD	Observing date in years or MJD in days (not string). Default = MJD-OBS If Value == NONE, OBSERVING_DATE_VALUE will be used instead.
OBSERVING_DATE_VALUE	If OBSERVING_DATE_KEYWORD=='NONE' use this value. Default = -1.0
UTC_KEYWORD	UTC in seconds. Default = UTC If Value == NONE, UTC_VALUE will be used.
UTC_VALUE	If UTC_KEYWORD=='NONE' use this value. Default = -1.0
TELESCOPE_ANGLE_KEYWORD	Telescope altitude angle in degrees. Default = ESO TEL ALT
TELESCOPE_ANGLE_VALUE	If TELESCOPE_ANGLE_KEYWORD=='NONE' take this value. Default = 90.0
RELATIVE_HUMIDITY_KEYWORD	Relative humidity in %. Default = ESO TEL AMBI RHUM
RELATIVE_HUMIDITY_VALUE	If RELATIVE_HUMIDITY_KEYWORD=='NONE' take this value. Default = 15.0
PRESSURE_KEYWORD	Pressure in hPa. Default = ESO TEL AMBI PRES START
PRESSURE_VALUE	If PRESSURE_KEYWORD=='NONE' take this value. Default = 750.0
TEMPERATURE_KEYWORD	Ambient temperature in deg C. Default = ESO TEL AMBI TEMP
TEMPERATURE_VALUE	If TEMPERATURE_KEYWORD=='NONE' take this value. Default = 15.0
MIRROR_TEMPERATURE_KEYWORD	Mirror temperature. (Relevant for emission spectrum only) Default = ESO TEL TH M1 TEMP
MIRROR_TEMPERATURE_VALUE	If MIRROR_TEMPERATURE_KEYWORD=='NONE' take this value, specified in degrees C. Default = 15.0
ELEVATION_KEYWORD	Elevation above sea level in m (default is Paranal: 2635. m). Default = ESO TEL GEOELEV
ELEVATION_VALUE	If ELEVATION_KEYWORD=='NONE' take this value. Default = 2635.0
LONGITUDE_KEYWORD	Longitude (default is Paranal: -70.4051 deg). Default = ESO TEL GEOLON
LONGITUDE_VALUE	If LONGITUDE_KEYWORD=='NONE' take this value. Default = -70.4051
LATITUDE_KEYWORD	Latitude (default is Paranal: -24.6276 deg). Default = ESO TEL GEOLAT

continuation of Table 6.1

Parameter	Description
LATITUDE_VALUE	If LATITUDE_KEYWORD=='NONE' take this value. Default = -24.6276
SLIT_WIDTH_KEYWORD	Slit width in arcsec (taken from FITS header if present). Default = ESO INS SLIT1 WID
SLIT_WIDTH_VALUE	If SLIT_WIDTH_KEYWORD=='NONE' take this value. Default = 0.4
PIX_SCALE_KEYWORD	Pixel scale in arcsec (taken from this file only). Default = NONE
PIX_SCALE_VALUE	If PIX_SCALE_KEYWORD=='NONE' take this value. Default = 0.086
REFERENCE_ATMOSPHERIC	Reference atmospheric profile. Possible values: - equ.fits (default; equatorial atmosphere, valid for Paranal); - tro.fits (tropical atmosphere); - std.fits (standard atmosphere); - Other file located in (TELLURICCORR_DATA_PATH/profiles/mipas/). Default = equ.fits
GDAS_PROFILE	Specify which GDAS profile to use. If GDAS_PROFILE=='auto', automatic retrieval of the GDAS profiles (P[hPa] HGT[km] T[K] RELHUM[%]) close in time to the observation and in location to the observatory. If the files are not on disk and there is no internet connection, the average profile is taken from share/molecfit/data/profiles/lib corresponding to the month of the observation (GDAS_t0_s1.fits for Dec/Jan, GDAS_t0_s2.fits for Feb/Mar, etc) See Sec. 8.1.4 of the molecfit manual for more info. If GDAS_PROFILE=='null', use the profile in the SOF with tag GDAS. If there is no profile in the SOF, the behaviour is the same as GDAS_PROF=auto. If GDAS_PROFILE=='none', use the average profile taken from share/molecfit/data/profiles/lib corresponding to the month of observation (see 'auto' description). If GDAS_PROFILE=='directory/file.fits', use the specified path and filename as the GDAS profile. Either an absolute path (starting with '/') or a relative path may be used, however a relative path is preferred, since only the first 40 char of the path and filename are copied to the FITS header. The file format must be a FITS binary table with columns having the names 'press height temp relhum' and units hPa, km, K and %, respectively. Default = auto
LAYERS	Grid of layer heights for merging ref_atm and GDAS profile. Fixed grid = CPL_TRUE and natural grid = CPL_FALSE. Default = TRUE
EMIX	Upper mixing height in km for considering data of a local meteo station. If emix is below geoelev, rhum, pres, and temp are not used for modifying the corresponding profiles. Default = 5.0

continuation of Table 6.1

Parameter	Description
PWV	PWV value in mm for the input water vapor profile. The merged profile composed of ref_atm, GDAS, and local meteo data will be scaled to this value if pwv > 0 (default: -1 -> no scaling). Default = -1.0
LNFL_LINE_DB	File name of the line list (must be stored in the directory :(TELLURICCORR_DATA_PATH/hitran/). Default = aer_v_3.8.1.2
LNFL_LINE_DB_FORMAT	Format of the line file: gives the length in terms of characters per line. Default = 100.0
LBLRTM_ICNTNM	Continua and Rayleigh extinction [0,1,2,3,4,5]. Default = 5
LBLRTM_IAERSL	Aerosols [0,1]. Default = 0
LBLRTM_MPTS	Number of optical depth values. Default = 5
LBLRTM_NPTS	Number of values for each panel. Default = 5
LBLRTM_V1	Beginning wavenumber value for the calculation. Default = 1.9
LBLRTM_V2	Ending wavenumber value for the calculation. Default = 2.4
LBLRTM_SAMPLE	Number of sample points per mean halfwidth [between 1 to 4, default=4]. Default = 4
LBLRTM_ALFAL0	Average collision broadened halfwidth [cm-1/atm]. Default = 0.0
LBLRTM_AVMASS	Average molecular mass [amu] for Doppler halfwidth. Default = 0.0
LBLRTM_DPTMIN	Minimum molecular optical depth below which lines will be rejected. Default = 0.0002
LBLRTM_DPTFAC	Factor multiplying molecular continuum optical depth. Default = 0.001
LBLRTM_TBOUND	Temperature of boundary [K]. Default = 0.0
LBLRTM_SREMIS1	Emissivity coefficient 1. Default = 0.0
LBLRTM_SREMIS2	Emissivity coefficient 2. Default = 0.0
LBLRTM_SREMIS3	Emissivity coefficient 3. Default = 0.0

continuation of Table 6.1

Parameter	Description
LBLRTM_SRREFL1	Reflectivity coefficient 1. [0.0
LBLRTM_SRREFL2	Reflectivity coefficient 2. Default = 0.0
LBLRTM_SRREFL3	Reflectivity coefficient 3. Default = 0.0
LBLRTM_MODEL	Atmospheric profile [0,1,2,3,4,5,6]. Default = 0
LBLRTM_ITYPE	Type of path [1,2,3]. Default = 3
LBLRTM_NOZERO	Zeroing of small amounts of absorbers [0,1]. Default = 0
LBLRTM_NOPRNT	Do not print output? [0,1]. Default = 0
LBLRTM_IPUNCH	Write out layer data to TAPE7 [0,1]. Default = 0
LBLRTM_RE	Radius of earth [km]. Default = 0.0
LBLRTM_HSPACE	Altitude definition for space [km]. Default = 120.0
LBLRTM_H2	Upper height limit [km]. Default = 0.0
LBLRTM_RANGE	Length of a straight path from H1 to H2 [km]. Default = 0.0
LBLRTM_BETA	Earth centered angle from H1 to H2 [degrees]. Default = 0.0
LBLRTM_LEN	Path length [0,1]. Default = 0

continuation of Table 6.1

Parameter	Description
LBLRTM_HOBS	Height of observer. Default = 0.0
LBLRTM_AVTRAT	Maximum Voigt width ratio across a layer. Default = 2.0
LBLRTM_TDIFF1	Maximum layer temperature difference at ALTD1 [K]. Default = 5.0
LBLRTM_TDIFF2	Maximum layer temperature difference at ALTD2 [K]. Default = 8.0
LBLRTM_ALTD1	Altitude of TDIFF1 [km]. Default = 0.0
LBLRTM_ALTD2	Altitude of TDIFF2 [km]. Default = 0.0
LBLRTM_DELV	Number of wavenumbers [cm-1] per major division. Default = 1.0
CHIP_EXTENSIONS	If TRUE, treat image extensions as a single observation to be fitted for as a single combined spectrum. If FALSE, treat image extensions as independent science data to be fitted for independently. Default = FALSE
MAP_REGIONS_TO_CHIP	Where there are multiple ranges and multiple chips, with the possibility of wavelength overlaps, there's ambiguity as to which region is assigned to which chip. This parameter allows the user to explicitly specify this association as a string with a comma separated list of integers, where each integer is a chip index. The number of integers in the list can be either one, which then applies to all ranges, or a value for each range can be individually specified. e.g. "1" and "1,1,1,1" are equivalent if there are 4 ranges. If the value is "NULL", then it is assumed the values are read from the TAG[WAVE_INCLUDE] FITS BINTABLE from a column named MAPPED_TO_CHIP. For further details see Appendix A2 of the molecfit user manual. The default is for the ranges to be mapped to CHIP1, i.e. Default = "1"

continuation of Table 6.1

Parameter	Description
`EXPERT_MODE`	If TRUE, read in starting values from the TAG[INIT_FIT_PARAMETERS] FITS BINTABLE that is a similar format to the BEST_FIT_PARAMETERS.fits output file. See Appendix A5 of the molecfit user manual for more details. If FALSE, Do not read in the starting values. Default = FALSE

6.3.2 Calctrans Recipe

Table 6.2 explains the individual calctrans recipe parameters in more detail and in the order they appear in the `.rc` file.

Table 6.2: Calctrans recipe parameters.

Parameter	Description
`USE_ONLY_INPUT_PRIMARY_DATA`	Value=TRUE implies that only the primary data array contains the input science flux data. Value=FALSE implies that the FITS extensions also contains input science flux. Default = FALSE
`USE_DATA_EXTENSION_AS_DFLUX`	Only valid if `USE_ONLY_INPUT_PRIMARY_DATA=TRUE`. This parameter specifies the FITS extension index that contains the 1-sigma error on the science flux (DFLUX). A value of 0 implies that there is no DFLUX available. Default = 0.

continuation of Table 6.2

Parameter	Description
USE_DATA_EXTENSION_AS_MASK	Only valid if USE_ONLY_INPUT_PRIMARY_DATA=TRUE. This parameter specifies the FITS extension index that contains the mask associated with the science flux data. A value of 0 implies that there is no mask data. Default = TRUE
USE_INPUT_KERNEL	If TRUE, use the kernel library if it is provided. Default = TRUE
CALCTRANS_MAPPING_KERNEL	If USE_INPUT_KERNEL is TRUE, List of extensions to map from the kernel library file (CALCTRANS_KERNEL_LIBRARY or KERNEL_LIBRARY) to the extensions of the input file (SCIENCE_CALCTRANS or SCIENCE). [string with comma separated integers that represent the extension numbers] e.g. Assuming an input science file has extensions "1,2,3,4", then CALCTRANS_MAPPING_KERNEL="2,2,2,2" implies a mapping extension 2 of the kernel library for all the extensions in the input science file, i.e. mapping 2 to 1, 2 to 2, 2 to 3, 2 to 4. means that extension 2 of the kernel library is used for extensions 1, 2, 3 and 4. CALCTRANS_MAPPING_KERNEL="1,2,3,4" similarly maps 1 to 1, 2 to 2, 3 to 3, 4 to 4. If USE_INPUT_KERNEL is FALSE, this parameter is unused If set to NULL, check if the TAG[CALCTRANS_MAPPING_KERNEL] FITS BINTABLE values are provided. The FITS BINTABLE must have one column [KERNEL_LIBRARY_EXT]. Default = NULL

continuation of Table 6.2

Parameter	Description
MAPPING_ATMOSPHERIC	List of extensions to map from the associated file (ATM_PARAMETERS) to the extensions of the input file (SCIENCE_CALCTRANS or SCIENCE). [string with comma separated integers that represent the extension numbers] e.g. Assuming an input science file has extensions "1,2,3,4", then MAPPING_ATMOSPHERIC="2,2,2,2" implies a mapping extension 2 of the associated file for all the extensions in the input science file, i.e. mapping 2 to 1, 2 to 2, 2 to 3, 2 to 4. MAPPING_ATMOSPHERIC="1,2,3,4" similarly maps 1 to 1, 2 to 2, 3 to 3, 4 to 4. If set to NULL, check if the TAG[MAPPING_ATMOSPHERIC] FITS BINTABLE value is provided. The FITS BINTABLE must have one column [ATM_PARAMETERS_EXT]. Default = NULL
MAPPING_CONVOLVE	List of extensions to map from the associated file (LBLRTM_RESULTS) to the extensions of the output (TELLURIC_CORR). [string with comma separated integers that represent the extension numbers] e.g. Assuming an input science file has extensions "1,2,3,4", then MAPPING_CONVOLVE="2,2,2,2" implies a mapping extension 2 of the associated file for all the extensions in the input science file, i.e. mapping 2 to 1, 2 to 2, 2 to 3, 2 to 4. MAPPING_CONVOLVE="1,2,3,4" similarly maps 1 to 1, 2 to 2, 3 to 3, 4 to 4. If set to NULL, check if the TAG[MAPPING_CONVOLVE] FITS BINTABLE value is provided. The FITS BINTABLE must have one column [LBLRTM_RESULTS_EXT]. Default = NULL

continuation of Table 6.2

Parameter	Description
CHIP_EXTENSIONS	If TRUE, treat image extensions as a single observation to be fitted for as a single combined spectrum. If FALSE, treat image extensions as independent science data to be fitted for independently. Default = FALSE
SCALE_PWV	Value read from science file to scale the precipitable water vapor (PWV) of the telluric spectra. If SCALE_PWV='auto', the mean value of TEL AMBI IWV START/END is read from the header of the input science file. If SCALE_PWV= a user-provided numerical value, use this as the scale factor. If SCALE_PWV= a FITS header keyword name, read in the scale factor from the header of the input science file using this keyword. If SCALE_PWV='none', perform no scaling. Default = none. See Sec. A.7 for more details.
HDR_MJD	FITS header keyword to read the Modified Julian Date at the START of the exposure from telluric and science header. The telluric header was copied to BEST_FIT_PARAMETERS by molecfit_model and read from there. Default = MJD-OBS
HDR_EXP	FITS header keyword to read the exposure time from telluric and science header. The telluric header was copied to BEST_FIT_PARAMETERS by molecfit_model and read from there. Default = ESO OBS EXECTIME
HDR_AIR1	FITS header keyword to read the airmass at the start of the exposure from the telluric and science header. The telluric header was copied to BEST_FIT_PARAMETERS by molecfit_model and read from there. Default = ESO TEL AIRM START
HDR_AIR2	FITS header keyword to read the airmass at the end of the exposure from the telluric and science header. The telluric header was copied to BEST_FIT_PARAMETERS by molecfit_model and read from there. Default = ESO TEL AIRM END
SGWL	Savitzky-Golay filter smoothing window length in pixels. Default = 15. See Sec. A.8 for more details.
SGWL_ASMAX	Treat the Savitzky-Golay filter smoothing window length as a maximum length. Default = FALSE

6.3.3 Correct Recipe

Table 6.3 explains the individual correct recipe parameters in more detail and in the order they appear in the .rc file.

Table 6.3: Correct recipe parameters.

Parameter	Description
USE_ONLY_INPUT_PRIMARY_DATA	Value=TRUE implies that only the primary data array contains the input science flux data. Value=FALSE implies that the FITS extensions also contains input science flux. Default = FALSE
USE_DATA_EXTENSION_AS_DFLUX	Only valid if USE_ONLY_INPUT_PRIMARY_DATA=TRUE. This parameter specifies the FITS extension index that contains the 1-sigma error on the science flux (DFLUX). A value of 0 implies that there is no DFLUX. Default = 0.
USE_DATA_EXTENSION_AS_MASK	Only valid if USE_ONLY_INPUT_PRIMARY_DATA=TRUE. This parameter specifies the FITS extension index that contains the mask associated with the science flux data. A value of 0 implies that there is no mask data. Default = 0.
SUPPRESS_EXTENSION	Suppress arbitrary filename extension : TRUE (apply) or FALSE (don't apply). Default = FALSE

continuation of Table 6.3

Parameter	Description
MAPPING_CORRECT	List of extensions to map from the telluric correction file (TELLURIC_CORR) to the extensions of the intput file (SCIENCE_CALCTRANS or SCIENCE). [string with comma separated integers that represent the extension numbers] e.g. Assuming an input science file has extensions "1,2,3,4", then MAPPING_CORRECT="2,2,2,2" implies a mapping extension 2 of the telluric correction file for all the extensions in the input science file, i.e. mapping 2 to 1, 2 to 2, 2 to 3, 2 to 4. means that extension 2 of the telluric correction is used for extensions 1, 2, 3 and 4 of the input spectrum. MAPPING_CORRECT="1,2,3,4" similarly maps 1 to 1, 2 to 2, 3 to 3, 4 to 4. If set to NULL, check if the TAG[MAPPING_CORRECT] FITS BINTABLE value is provided. The FITS BINTABLE must have one column [TELLURIC_CORR_EXT]. Default = NULL
WLC_REF	Indicates that the reference for the wavelength calibration : - If it is set to "DATA", is the input data. - If it is set to "MODEL", is the output model. In the "MODEL" case, the wavelength given in the output spectrum has been corrected with the result of the fit. If set to 'MODEL', the degree of the polynomial used for the wavelength correction should be 0 or at most 1, except if the inclusion regions cover a large or well-sampled spectral range of the input spectrum. Default = DATA
CHIP_EXTENSIONS	If TRUE, treat image extensions as a single observation to be fitted for as a single combined spectrum. If FALSE, treat image extensions as independent science data to be fitted for independently. Default = FALSE

6.4 Input Spectrum

MOLECFIT 4.3 accepts the following formats:

- FITS binary table,
- 1D FITS image.

if USE_ONLY_INPUT_PRIMARY_DATA is TRUE, then the assumption is that the format is a 1D FITS image. Otherwise the input must be a FITS binary table.

The CDELT1, CDELT2 and CDELT3 FITS header keywords are now *deprecated* in favour of the corresponding FITS header keywords CD1_1, CD2_2 and CD3_3. If your input spectrum is missing the latter keywords, a warning will be raised about the input spectrum. This warning can be resolved by adding these keywords to the FITS header.

6.5 Output Products

With the exception of the ASCII log file esorex.log, all MOLECFIT output files are in the same format as the input files as described above. A comprehensive list is given in Appendix B.7.

7 Guide for MOLECFIT 1.5.9 Users

7.1 Execution Format

MOLECFIT 1.5.9 consisted of three main standalone executables:

- molecfit — Find best-fit parameters for molecules and continuum
- calctrans — Calculate transmission model given best fit parameters
- corrfilelist — Perform telluric correction for a set of spectra given the transmission model

These would run with a specified parameter file (`.par`) that would list all the non-default parameter values to use including the input and output filenames. A common practice would be to include the parameters for all three executables in a single .par file and execute as follows:

```
> molecfit VISIR_HR.par
> calctrans VISIR_HR.par
> corrfilelist VISIR_HR.par
```

where input spectra files could be of either FITS or ASCII format.

For MOLECFIT versions 3.x and later, these applications are now represented as *EsoRex* recipes and interface in compliance of *EsoRex* pipeline standards. They are executed from the command line in the following manner:

```
> esorex --recipe-config=Pars2Use.rc recipe_name SOF_filename
```

Where `Pars2Use.rc` contains the list of “processing parameters”, the `SOF` (set of files) is an ASCII list of the input and output filenames which are expected to be in FITS format. Note: ASCII data files are not supported.

It should be noted that the MOLECFIT 1.5.9 executables `calctrans_lblrtm` and `calctrans_convolution` are not represented as *EsoRex* recipes.

7.2 Parameter Format

In accordance with *EsoRex* standards, *EsoRex* calls will return an error if the `.rc` file specified contains a parameter that is not relevant to the recipe in use. Thus, unlike the original MOLECFIT, all the parameters to be used with the three molecfit steps — `model`, `correct`, and `calctrans` cannot be defined in a single parameter file. Instead, each step must be defined using an independent recipe configuration (`.rc`) file.

A recipe configuration file with all the default values can be generated in the working directory through *EsoRex* using the following command:

```
> esorex --create-config= filename recipe_name
```

For example:

```
> esorex --create-config=Default_model.rc molecfit_model
> esorex --create-config=Default_correct.rc molecfit_correct
> esorex --create-config=Default_calctrans.rc molecfit_calctrans
```

The above example will generate the default `.rc` files:

```
Default_model.rc
Default_correct.rc
Default_calctrans.rc
```

respectively in the working directory. These `.rc` files can then be used as templates, thereby easing the burden of writing an `.rc` file from scratch.

The majority of the parameters declared in these `.rc` files should be familiar to the 1.5.9 user. In many cases there are some subtle differences in the parameter name and in the value format. For example the logical parameter to specify whether fitting for the continuum is to be performed has been changed from:

```
  FIT_CONT: 1
```

to

```
  FIT_CONTINUUM = TRUE
```

7.3 Input Output Filename Specifications

The names of all input and output files involved in a recipe execution are to be specified in the `.sof` file given in the command line execution. These are ASCII files that specify one file per line in the form:

```
filename   tagname
```

where file pathnames may contain enviroment variables, and `tagnames` are specific keywords that declare the file's role in the process — e.g. input science spectra, input best-fit parameters, etc.

For example an `SOF` file for a MOLECFIT `correct` recipe execution may contain:

```
$HOME/molecfit/raw/CRIRES_HighSNR_Telluric_input_AM1p472.fits SCIENCE
$HOME/molecfit_output/MODEL_MOLECULES.fits MODEL_MOLECULES
$HOME/molecfit_output/ATM_PARAMETERS.fits ATM_PARAMETERS
```

where

`$HOME`	environment variable
`SCIENCE`	tagname for input file that contains the science spectra to use
`MODEL_MOLECULES`	tagname for input file that contains a list of air molecules to include in the model
`ATM_PARAMETERS`	tagname for the output file that contains the best fit parameters of the model

A full list of the `SOF` tag names are given in appendix A3.4

7.4 Plot Files

MOLECFIT 3.x and later versions do not support the 1.5.9 plot file commands, as *Reflex* offers other methods to graph results.

7.5 Extra LNFL and LBLRTM Control Parameters

From v3.x MOLECFIT supports extra optional parameters for additional control of the third party lnfnl and lblrtm executions. These are listed in Table 7.1: Description of the LBRTM parameters is available in following location in the kit distribution -

```
molecfit-kit-<version>/molecfit_third_party-<version>lblrtm /docs/html/
lblrtm_instructions_frame.html
```

Table 7.1: LNFL and LBLRTM control parameters.

Parameter	Description
LNFL_LINE_DB	File name of the line list. This must be stored in the directory TELLURICCORR_DATA_PATH/hitran/. Default = aer_v_3.8.1.2
LNFL_LINE_DB_FORMAT	Format of the line file: gives the length in terms of characters per line. Default = 100.0
LBLRTM_ICNTNM	Continua and Rayleigh extinction Valid range: [0 – 5] Default = 5
LBLRTM_IAERSL	Aerosols Valid range: [0, 1] Default = 0
LBLRTM_MPTS	Number of optical depth values. Default = 5
LBLRTM_NPTS	Number of values for each panel. Default = 5
LBLRTM_V1	Beginning wavenumber value for the calculation. Default = 1.9
LBLRTM_V2	Ending wavenumber value for the calculation. Default = 2.4
LBLRTM_SAMPLE	Number of sample points per mean halfwidth Valid range: [1 – 4] Default = 4
LBLRTM_ALFAL0	Average collision broadened halfwidth (cm^{-1}/atm). Default = 0.0
LBLRTM_AVMASS	Average molecular mass (amu) for Doppler halfwidth. Default = 0.0

continuation of Table 7.1

Parameter	Description
LBLRTM_DPTMIN	Minimum molecular optical depth below which lines will be rejected. Default = 0.0002
LBLRTM_DPTFAC	Factor multiplying molecular continuum optical depth. Default = 0.001
LBLRTM_TBOUND	Temperature of boundary (K). Default = 0.0
LBLRTM_SREMIS1	Emissivity coefficient 1. Default = 0.0
LBLRTM_SREMIS2	Emissivity coefficient 2. Default = 0.0
LBLRTM_SREMIS3	Emissivity coefficient 3. Default = 0.0
LBLRTM_SRREFL1	Reflectivity coefficient 1. Default = 0.0
LBLRTM_SRREFL2	Reflectivity coefficient 2. Default = 0.0
LBLRTM_SRREFL3	Reflectivity coefficient 3. Default = 0.0
LBLRTM_MODEL	Atmospheric profile Valid range: [0 – 6]. Default = 0
LBLRTM_ITYPE	Type of path Valid range: [1 – 3]. Default = 3
LBLRTM_NOZERO	Zeroing of small amounts of absorbers Valid range: [0 – 1]. Default = 0
LBLRTM_NOPRNT	Do not print output. Valid range: [0 – 1] Default = 0
LBLRTM_IPUNCH	Write out layer data to TAPE7 Valid range: 0 – 1. Default = 0
LBLRTM_RE	Radius of earth [km]. Default = 0.0
LBLRTM_HSPACE	Altitude definition for space [km]. Default = 120.0
LBLRTM_H2	Upper height limit [km]. Default = 0.0
LBLRTM_RANGE	Length of a straight path from H1 to H2 [km]. Default = 0.0
LBLRTM_BETA	Earth centered angle from H1 to H2 [degrees].

continuation of Table 7.1

Parameter	Description
	Default = 0.0

continuation of Table 7.1

Parameter	Description
LBLRTM_LEN	Path length Valid range = [0, 1]. Default = 0
LBLRTM_HOBS	Height of observer. Default = 0.0
LBLRTM_AVTRAT	Maximum Voigt width ratio across a layer. Default = 2.0
LBLRTM_TDIFF1	Maximum layer temperature difference at ALTD1 [K]. Default = 5.0
LBLRTM_TDIFF2	Maximum layer temperature difference at ALTD2 [K]. Default = 8.0
LBLRTM_ALTD1	Altitude of TDIFF1 [km]. Default = 0.0
LBLRTM_ALTD2	Altitude of TDIFF2 [km]. Default = 0.0
LBLRTM_DELV	Number of wavenumbers [cm-1] per major division. Default = 1.0

7.6 Simple Example

Assume that we want to process a science frame CRIRES_HighSNR_Telluric_input_AM1p472.fits which is located in the working directory and that all desired parameters have been defined in the associated `.rc` files

```
Model.rc
Calctrans.rc
Correct.rc
```

Step 1: invoke

```
> esorex --recipe-config=Model.rc molecfit_model Model.sof
```

where `Model.sof` contains a single line specifying the location of the science frame:

```
./CRIRES_HighSNR_Telluric_input_AM1p472.fits      SCIENCE
```

This will produce several FITS files, including:

MODEL_MOLECULES.fits	Contains the list of molecules used in the model
ATM_PARAMETERS.fits	Contains the atmospheric parameter values used in the model
BEST_FIT_PARAMETERS	Contains the best fit values derived for the model

Step 2: run

```
> esorex --recipe-config=Calctrans.rc molecfit_calctrans Calctrans.sof
```

where Calctrans.sof contains the location of the science frame and the parameter values as derived from the model recipe execution:

```
./CRIRES_HighSNR_Telluric_input_AM1p472.fits    SCIENCE
./MODEL_MOLECULES.fits                          MODEL_MOLECULES
./ATM_PARAMETERS.fits                           ATM_PARAMETERS
./BEST_FIT_PARAMETERS.fits                      BEST_FIT_PARAMETERS
```

This will produce several fits files including

```
 TELLURIC_CORR.fits
```

which contains the data to perform a convolution that will correct a spectrum of telluric contamination.

Step 3: invoke:

```
> esorex --recipe-config=Correct.rc molecfit_correct Correct.sof
```

where `Correct.sof` contains the location of the science frame and the convolution telluric correction data derived from the calctrans recipe execution:

```
./CRIRES_HighSNR_Telluric_input_AM1p472.fits    SCIENCE
./TELLURIC_CORR.fits    TELLURIC_CORR
```

This will produce several fits files including:

```
SCIENCE_TELLURIC_CORR_CRIRES_HighSNR_Telluric_input_AM1p472.fits
```

Which is the telluric corrected version of

```
CRIRES_HighSNR_Telluric_input_AM1p472.fits
```

7.7 Old Parameter Renames

We list the 1.5.9 MOLECFIT parameters in groups that users of the original MOLECFIT would be familiar with and tabulate them with their equivalence in the new MOLECFIT.

7.7.1 Input Data

MOLECFIT 1.5.9			MOLECFIT >3.x			
Name	Type	Default	Name	Type	Default	Comment
filename	String	"none"	–	–	–	Declared in SOF
listname	String	"none"	–	–	–	Not supported
trans	Integer	1	TRANSMISSION	Logical	TRUE	
col_lam	String	"undef"	COLUMN_LAMBDA	String	"lambda"	
col_flux	String	"undef"	COLUMN_FLUX	String	"flux"	
col_dflux	String	"undef"	COLUMN_DFLUX	String	"dflux"	
default_error	Double	0.01	DEFAULT_ERROR	Double	0.01	
wlgtomicron	Double	1.0	WLG_TO_MICRON	Double	1.0	
vac_air	String	"vac"	WAVELENGTH_FRAME	String	"VAC"	
wrange_include	Dbls[3]	"none"	WAVE_INCLUDE	Dbls[4]	"NULL"	
wrange_exclude	Dbls[1]	"none"	WAVE_EXCLUDE	Dbls[2]	"NULL"	
prange_exclude	Dbls[1]	"none"	PIXEL_EXCLUDE	Dbls[2]	"NULL"	

7.7.2 Results

MOLECFIT 1.5.9			MOLECFIT >3.x			
Name	Type	Default	Name	Type	Default	Comment
output_dir	String	"output"	–	–	–	Not supported
output_name	String	"none"	–	–	–	Not supported
plot_creation	String	XP	–	–	–	Not supported
plot_range	Logical	0	–	–	–	Not supported

7.7.3 Fit Precision

MOLECFIT 1.5.9			MOLECFIT >3.x			
Name	Type	Default	Name	Type	Default	Comment
ftol	Double	1×10^{-10}	FTOL	Double	1×10^{-10}	
xtol	Double	1×10^{-10}	XTOL	Double	1×10^{-10}	

[1] String representation of a space-separated list of doubles – e.g. "4.230 5.238 6.239", or the string value "none"

[2] String representation of a comma-separated list of doubles – e.g. "4.230,5.238,6.239", or the string value "NULL"

7.7.4 Molecular Columns

MOLECFIT 1.5.9			MOLECFIT >3.x			
Name	**Type**	**Default**	**Name**	**Type**	**Default**	**Comment**
`list_molec`	Strs[5]	"H2O"	`LIST_MOLEC`	Strs[6]	"NULL"	
`fit_molec`	Ints[7]	"1"	`FIT_MOLEC`	Ints[8]	"NULL"	
`rel_col`	Dbls[9]	"1.0"	`REL_COL`	Dbls[10]	"NULL"	

7.7.5 Background and Continuum

MOLECFIT 1.5.9			MOLECFIT >3.x			
Name	**Type**	**Default**	**Name**	**Type**	**Default**	**Comment**
`flux_unit`	Integer	0	`FLUX_UNIT`	Integer	0	
`fit_back`	Integer	1	`FIT_TELESCOPE_BACKGROUND`	Logical	TRUE	
`telback`	Double	0.1	`TELESCOPE_BACKGROUND_CONST`	Double	0.1	
`fit_cont`	Integer	1	`FIT_CONTINUUM`	Logical	TRUE	
`cont_n`	Integer	0	`FIT_CONTINUUM_N`	Integer	0	
`cont_const`	Double	1.0	`CONTINUUM_CONST`	Double	1.0	

7.7.6 Wavelength Solution

MOLECFIT 1.5.9			MOLECFIT >3.x			
Name	**Type**	**Default**	**Name**	**Type**	**Default**	**Comment**
`fit_wlc`	Integer	1	`FIT_WLC`	Logical	TRUE	
`wlc_n`	Integer	1	`WLC_N`	Integer	1	
`wlc_const`	Double	0.0	`WLC_CONST`	Double	0.0	

[3] String representation of a space-separated list of strings, e.g. “H2O O2 O3”, or the string value “none”

[4] String representation of a comma-separated list of strings, e.g. “H2O,O2,O3”, or the string value “NULL”

[5] String representation of a space-separated list of integers, e.g. “1 0 1”, or the string value “none”

[6] String representation of a comma-separated list of integers, e.g. “1,0,1”, or the string value “NULL”

[7] String representation of a space-separated list of doubles, e.g. “4.230 5.238 6.239”, or the string value “none”

[8] String representation of a comma-separated list of doubles, e.g. “4.230,5.238,6.239”, or the string value “NULL”

7.7.7 Line Spread Functions

MOLECFIT 1.5.9			MOLECFIT >3.x			
Name	**Type**	**Default**	**Name**	**Type**	**Default**	**Comment**
`fit_res_box`	Integer	1	`FIT_RES_BOX`	Logical	TRUE	
`relres_box`	Double	1.0	`RES_BOX`	Double	1.0	
`fit_res_gauss`	Integer	1	`FIT_RES_GAUSS`	Logical	TRUE	
`res_gauss`	Double	1.0	`RES_GAUSS`	Double	1.0	
`fit_res_lorentz`	Integer	1	`FIT_RES_LORENTZ`	Logical	TRUE	
`res_lorentz`	Double	1.0	`RES_LORENTZ`	Double	1.0	
`kernmode`	Integer	1	`KERNMODE`	Logical	FALSE	
`kernfac`	Double	3.0	`KERNFAC`	Double	3.0	
`varkern`	Integer	0	`VARKERN`	Logical	FALSE	
`kernel_file`	String	"none"	`USE_INPUT_KERNEL`	Logical	TRUE	Flag has to be true for kernel file to be used
			`MODEL_MAPPING_KERNEL`	String	"NULL"	If value is “NULL” then can be specified in SOF file

7.7.8 Ambient Parameters

MOLECFIT 1.5.9			MOLECFIT >3.x		
Name	**Type**	**Default**	**Name**	**Type**	**Default**
utc	Double	-1.0	UTC_VALUE	Double	-1.0
utc_key	String	"UTC"	UTC_KEYWORD	String	"UTC"
telalt	Double	90.0	TELESCOPE_ANGLE_VALUE	Double	90.0
telalt_key	String	"ESO TEL ALT"	TELESCOPE_ANGLE_KEYWORD	String	"ESO TEL ALT"
rhum	Double	15.0	RELATIVE_HUMIDITY_VALUE	Double	15.0
rhum_key	String	"ESO TEL AMBI RHUM"	RELATIVE_HUMIDITY_KEYWORD	String	"ESO TEL AMBI RHUM"
pres	Double	750.0	PRESSURE_VALUE	Double	750.0
pres_key	String	"ESO TEL AMBI PRES START"	PRESSURE_KEYWORD	String	"ESO TEL AMBI PRES START"
temp	Double	15.0	TEMPERATURE_VALUE	Double	15.0
temp_key	String	"ESO TEL AMBI TEMP"	TEMPERATURE_KEYWORD	String	"ESO TEL AMBI TEMP"
m1temp	Double	15.0	MIRROR_TEMPERATURE_VALUE	Double	15.0
m1temp_key	String	"ESO TEL TH M1 TEMP"	MIRROR_TEMPERATURE_KEYWORD	String	"ESO TEL TH M1 TEMP"
geoelev	Double	2635.0	ELEVATION_VALUE	Double	2635.0
geoelev_key	String	"ESO TEL GEOELEV"	ELEVATION_KEYWORD	STRING	"ESO TEL GEOELEV"
longitude	Double	-70.4051	LONGITUDE_VALUE	Double	-70.4051
longitude_key	String	"ESO TEL GEOLON"	LONGITUDE_KEYWORD	String	"ESO TEL GEOLON"
latitude	Double	-24.6276	LATITUDE_VALUE	Double	-24.6276
latitude_key	String	"ESO TEL GEOLAT"	LATITUDE_KEYWORD	String	"ESO TEL GEOLAT"

7.7.9 Instrument Parameters

MOLECFIT 1.5.9			MOLECFIT >3.x		
Name	**Type**	**Default**	**Name**	**Type**	**Default**
slitw	Double	0.4	SLIT_WIDTH_VALUE	Double	0.4
slit_key	String	"NONE"	SLIT_WIDTH_KEYWORD	String	"ESO INS SLIT1 WID"
pixsc	Double	0.086	PIX_SCALE_VALUE	Double	0.086
pixsc_none	String	"NONE"	PIX_SCALE_KEYWORD	String	"NONE"

7.7.10 Atmostpheric Profiles

MOLECFIT 1.5.9			MOLECFIT >3.x		
Name	**Type**	**Default**	**Name**	**Type**	**Default**
ref_atm	String	"equ.atm"	REFERENCE_ATMOSPHERIC	String	"equ.fits"
gdas_dir	String	"data/profiles/grib"	–	–	–
gdas_prof	String	"auto"	GDAS_PROFILE	String	"auto"
layers	Integer	1	LAYERS	Logical	TRUE
emix	Double	5.0	EMIX	Double	5.0
pwv	Double	-1.0	PWV	Double	-1.0

8 The Model

In this section, the atmospheric model used for MOLECFIT is described in more detail. First, the building and properties of the atmospheric profiles required for the calculation of emission and absorption spectra are discussed (Section 8.1). Then, we explain the properties of the radiative transfer code used (Section 8.2). The contribution of the different molecules to the resulting atmospheric spectra is discussed in Section 8.3. Moreover, we provide some information on the modelling of the telescope emission (Section 8.4). Finally, we describe how the resulting model is adapted to the input science spectrum (Section 8.5).

8.1 Atmospheric profiles and meteorological data

Information concerning the composition of the atmosphere is available at various levels. To the end of creating a uniform profile with the variables temperature, pressure, and density of various molecular species as a function of geoelevation, three sources of input are merged: standard profile (produced for MIPAS onboard the ENVISAT satellite), GDAS profile, and EMM data.

The largest amount of molecular density information is contained in the atmospheric standard profiles. However, they are only available for specific geographical latitudes and do not contain any time information whatsoever (see Section 8.1.1). To compensate the lack of time information, one can rely on the EMM (see Section 8.1.3). It provides the most frequent updates and is specific to the selected observing site. Unfortunately, it cannot provide molecular species information apart from water vapor (relative humidity measurements) and is restricted to a local on-site measurement, i.e. a single geoelevation data point only. To bridge the gap between these two data sources, GDAS provides a global grid of profile measurements (with approximate grid spacing of 110 km) to an altitude of $\sim$26 km with updates every three hours. GDAS does not contain molecular species apart from H_2O, though (see Section 8.1.2).

These three data sources and the required processing for use with MOLECFIT are described in detail below (see also Noll et al. [RD06]).

8.1.1 MIPAS profiles

The atmospheric standard profiles provide the basis for the model atmosphere used in MOLECFIT (see parameter REF_ATM in Section 6.3) including information on pressure, temperature, and molecular abundance as function of height (121 levels in the range 0-120 km). Up to now, the RFM homepage [RD24] provides standard profiles for mid-latitude (Lat = 45°, both, day and night), polar winter/summer (Lat = 75°) and equatorial day-time conditions in such a configuration (J. Remedios 2001). So far, the following molecules are included in this standard profile: N_2, O_2, CO_2, O_3, H_2O, CH_4, N_2O, HNO_3, CO, NO_2, N_2O_5, ClO, HOCl, $ClONO_2$, NO, HNO_4, HCN, NH_3, F11, F12, F14, F22, CCl_4, COF_2, H_2O_2, C_2H_2, C_2H_6, OCS, SO_2, and SF (see Table 8.3). Additional molecule profiles for F13 (CClF3), F21 ($CHCl_2F$), F113 ($C_2Cl_3F_3$), F114 ($C_2Cl_2F_4$), F115 (C_2ClF_5), and CH_3Cl are available. Apart from these data, less resolved profiles (a tropical, sub-arctic summer/winter and a US standard profile) are available with 50 geoelevation layers including the molecules H_2O, CO_2, O_3, N_2O, CO, CH_4, and O_2 only.

Comparison between atmospheric standard profiles In this section, an equatorial day-time `equ.fits` and a mid-latitude profile `ngt.atm`, corresponding to a latitude Lat = 45°, will be compared. The location of Paranal (Lat = 24.6°) is between these two profiles. In Figure 8.1 both profiles are shown. Although the distribution of several molecules (e.g. N_2, O_2, CO_2) does not vary, significant differences between the two profiles are visible. To investigate the impact of the input profile differences on the output spectra, LBLRTM was run with the same input parameters, but with varying standard profiles.

The resulting spectra are shown in Figures 8.2/8.3. These plots reveal output radiance spectra differing by at most ±10%. The same is true for the transmission spectra, although somewhat less obvious due to numerical instabilities. Calculating the broad-band ratios in the main filter ranges *UBVRI*$_c$*JHKLMN* indicates deviations of less than 2% (see Table 8.1). Hence, one can conclude that the differences between the two standard atmospheric profiles are negligible at this stage. Anu Dudhia [RD08], the author of the RFM code, recommends the equatorial profile `equ.fits` to be used for typical applications at Cerro Paranal.

Table 8.1: Broad-band comparison of the relative ratios between the `equ.fits` and the `ngt.atm` atmospheric standard profiles.

Filter	$\lambda_{\rm min}$ [μm]	$\lambda_{\rm max}$ [μm]	Radiance ratio [%]	Transmission ratio [%]
U	0.33	0.40	0.02	0.06
B	0.39	0.50	0.00	0.02
V	0.50	0.60	0.14	0.56
R	0.58	0.82	0.05	0.28
I_c	0.73	0.85	0.01	0.04
J	1.10	1.34	-0.00	-0.00
H	1.50	1.80	-0.00	-0.01
K	2.00	2.40	-0.05	-0.06
L	3.56	4.12	-0.00	-0.05
M	4.52	4.96	-0.69	1.37
N	7.40	13.60	-0.60	1.39

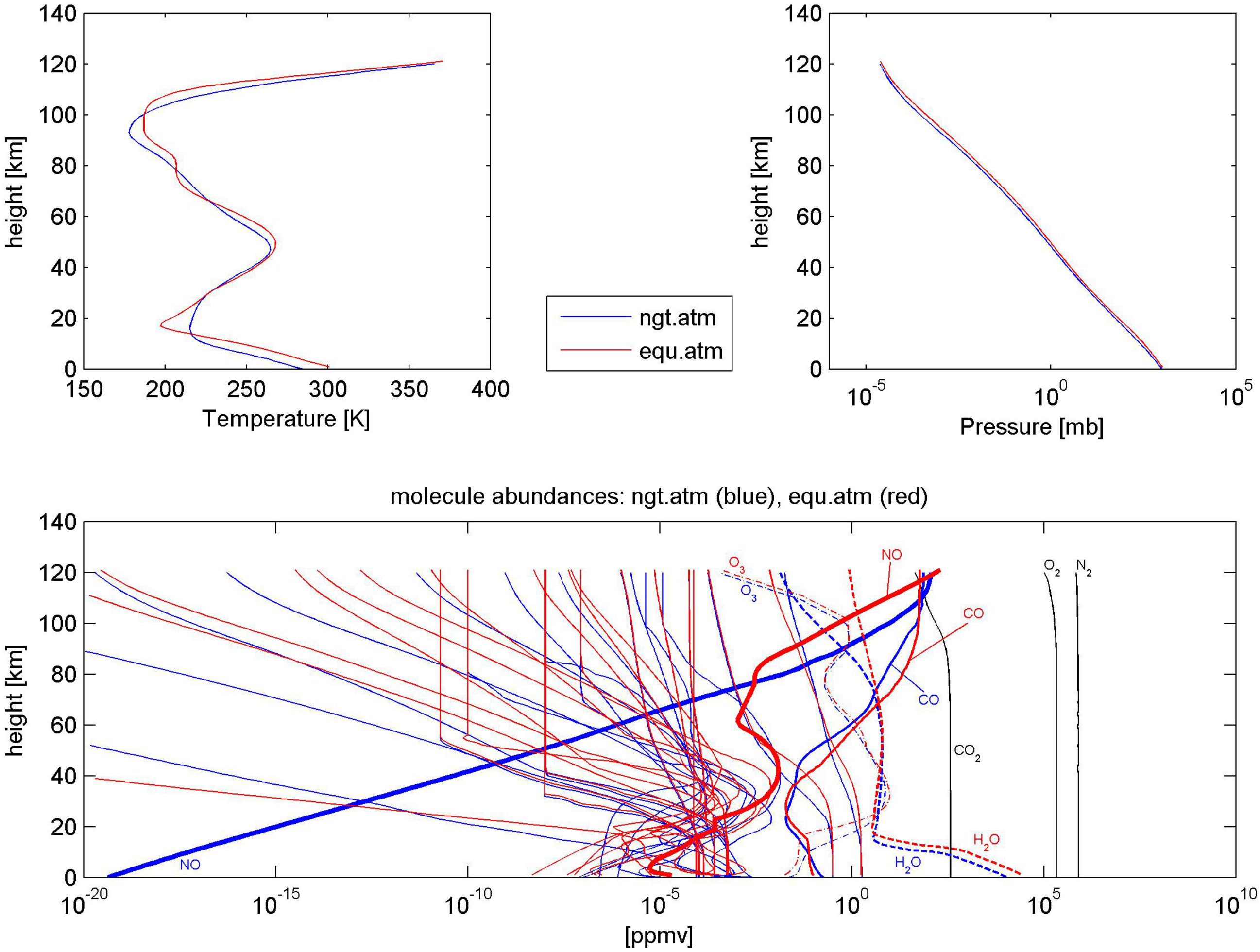

Figure 8.1: Comparison of the equatorial (`equ.fits`) and the mid-latitude night time atmospheric profile (`ngt.atm`). Red lines correspond to the equatorial, blue lines to the mid-latitude profile.

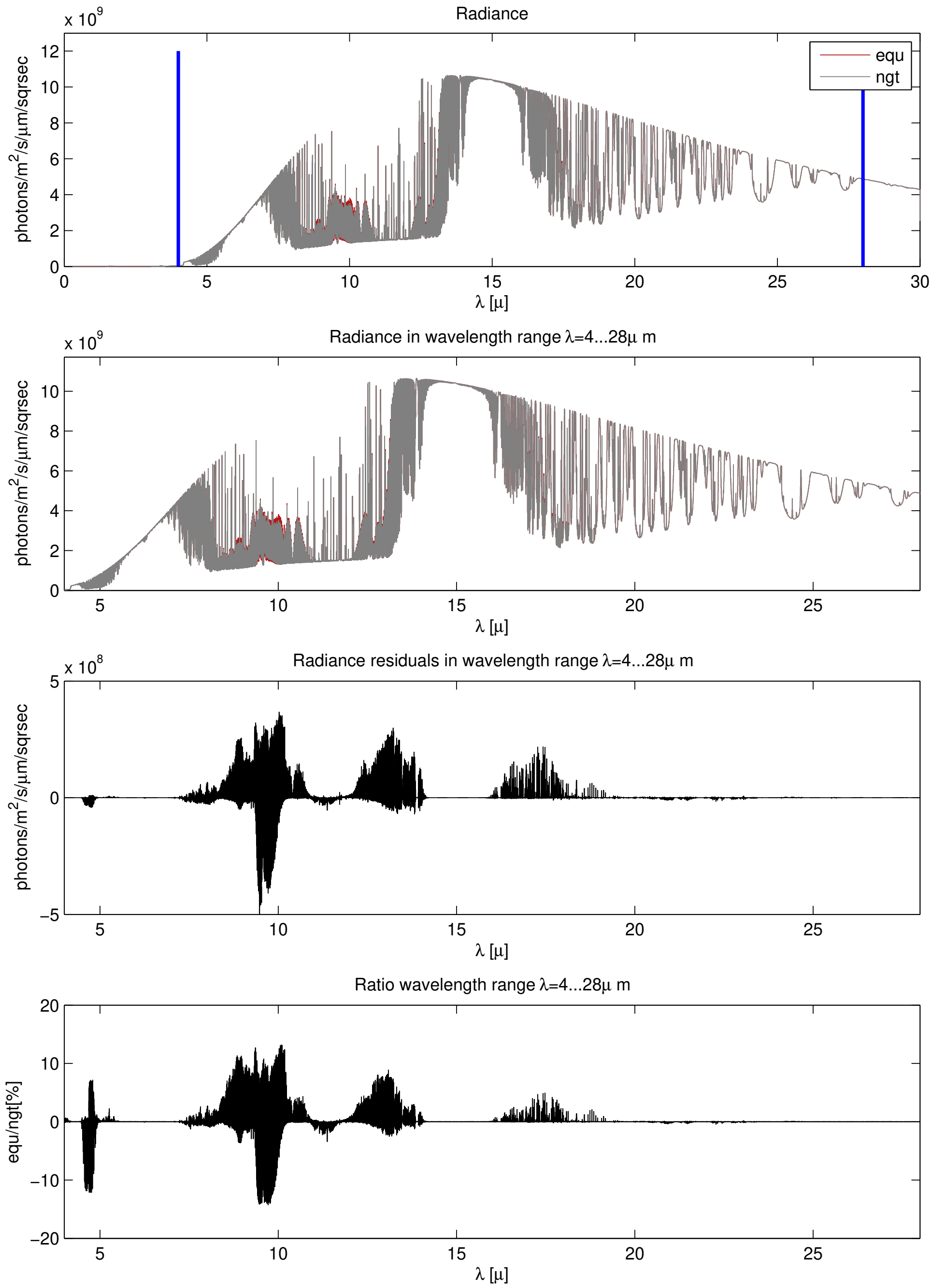

Figure 8.2: Direct comparison between the radiance spectra of equatorial day time (red line) and mid-latitude night time atmospheric standard profile (grey line) over the entire wavelength range $\lambda = 0.3 - 30\,\mu$m (top panel). Blue lines mark the wavelength range ($\lambda = 4 - 28\,\mu$m) plotted in the three panels below. *Second panel*: Same as in top panel, but for the limited wavelength range. *Third and bottom panel*: Residuals equ-ngt and ratio equ/ngt of the radiance spectra, respectively.

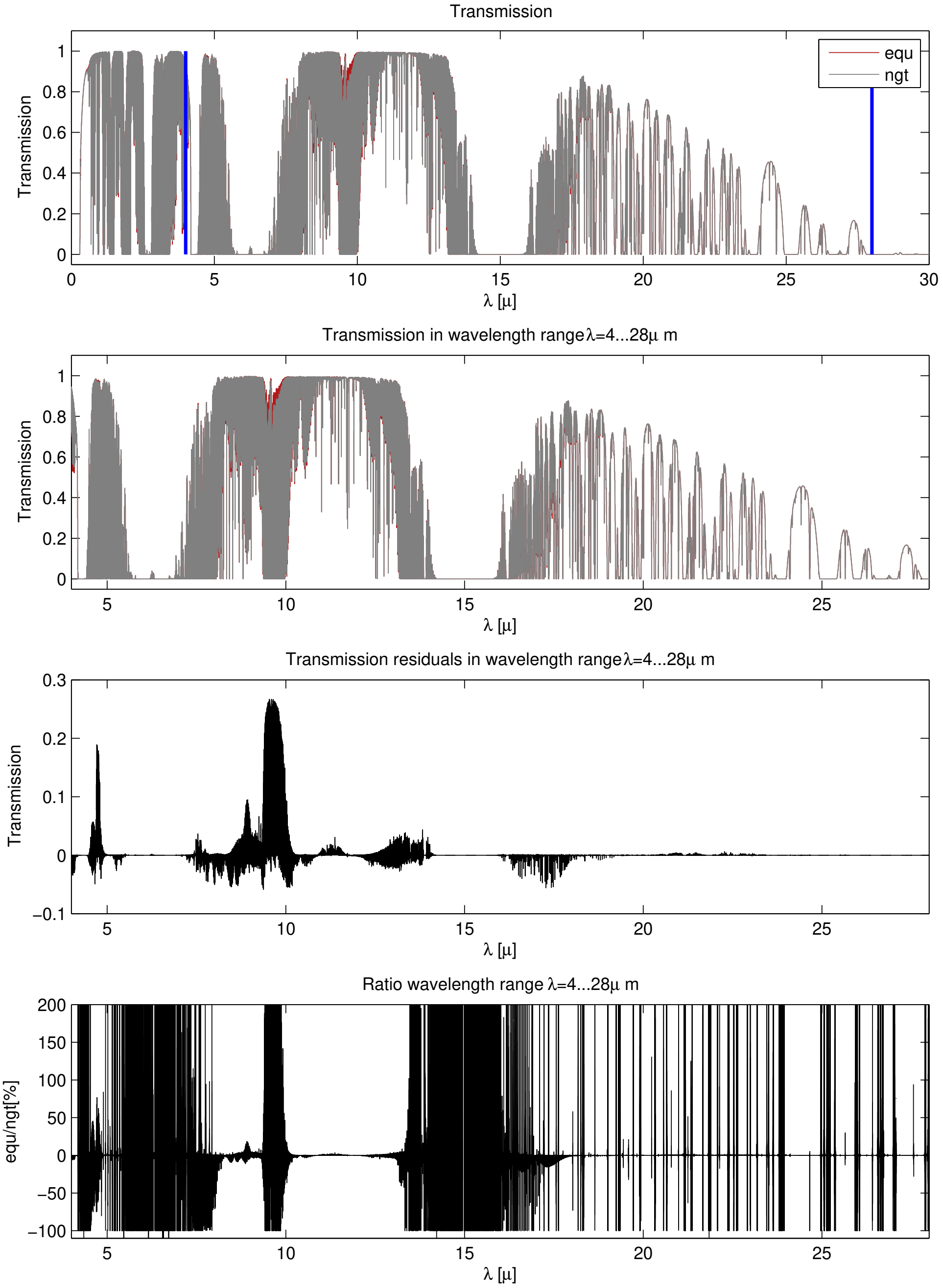

Figure 8.3: Same as Figure 8.2, but for the transmission.

Table 8.2: Example of a *GDAS profile* with columns for pressure, geoelevation, temperature, and relative humidity.

P[hPa]	HGT[km]	T[K]	RELHUM[%]
903	0.971	294.5	49.9
900	0.976	295.8	35.5
850	1.467	293.7	30.9
800	1.985	291.0	29.8
750	2.533	288.2	27.4
700	3.112	284.5	26.0
650	3.726	280.7	18.8
600	4.379	276.2	11.4
550	5.077	271.4	8.7
500	5.827	266.0	7.5
450	6.638	259.9	7.5
400	7.522	252.7	10.5
350	8.494	244.7	25.1
300	9.578	236.0	53.4
250	10.813	227.0	66.5
200	12.267	218.8	37.7
150	14.069	209.2	17.3
100	16.489	200.3	32.4
50	20.571	206.1	0.0
20	26.324	221.4	0.0

8.1.2 GDAS profiles

The GDAS data provided by NOAA are a model-based set of meteorological data dedicated to weather forecast studies. The models are archived by the ARL, as a global, 1 degree latitude/longitude data set based on pressure surfaces (starting from Dec. 2004). Apart from various meteorological parameters for the surface, vertical profiles for 23 pressure levels ranging from 0 to about 26 km are provided for the geopotential height, temperature, relative humidity, and wind components (not used in MOLECFIT) for three dimensions. An example is shown in Table 8.2.

The MOLECFIT software package provides the entire GDAS data for the location of Cerro Paranal from Dec. 2004 to Sep. 2013 on a 3 h basis taken from the NOAA archive[11]. Later dates (on a 6 h basis) or data for a different site are automatically downloaded from an online archive[12]. See also Section 8.1.4.

8.1.3 ESO Meteo Monitor

The EMM provides information on the local meteorological conditions at the ESO sites La Silla and Paranal. The data at Paranal are taken by a local meteo station mounted on a 30 m high mast installed in October 1984 [RD07]. This meteo station provides the following meteorological information on a 20 min average basis:

[11] ftp://arlftp.arlhq.noaa.gov/pub/archives/gdas1/

[12] http://nomad1.ncep.noaa.gov/pub/gdas/rotating/

```
s1,d1 = wind speed (m/s) and direction (0=North, 90=East..)
        at 10m above ground (30m from 1998 onwards)
   rh = relative humidity (%) at 2m above ground
   t1 = air temperature (Celsius) at 2m above ground
   p  = pressure (mb) at 2m above ground
   td = dew point temperature (C) computed from rh and t1
```

Starting from January 1st 1985, currently ∼ 400 000 data points are measured with the following accuracy [RD07]:

```
wind direction: ~5.63deg
wind speed:     ~2% over 10m/s
temperature:    ~0.1deg
humidity:       linearity about 1%
seeing:         better than 10% above 0.25 arcsec
```

The data can be retrieved online[13] on a daily basis, or as download provided by M. Sarazin[14] and are cumulatively shown in Figure 8.4. Thus, for any requested average time interval, like e.g. December and January, ample measurements are available. In compiling these data, care has to be taken to remove bad measurements before further processing.

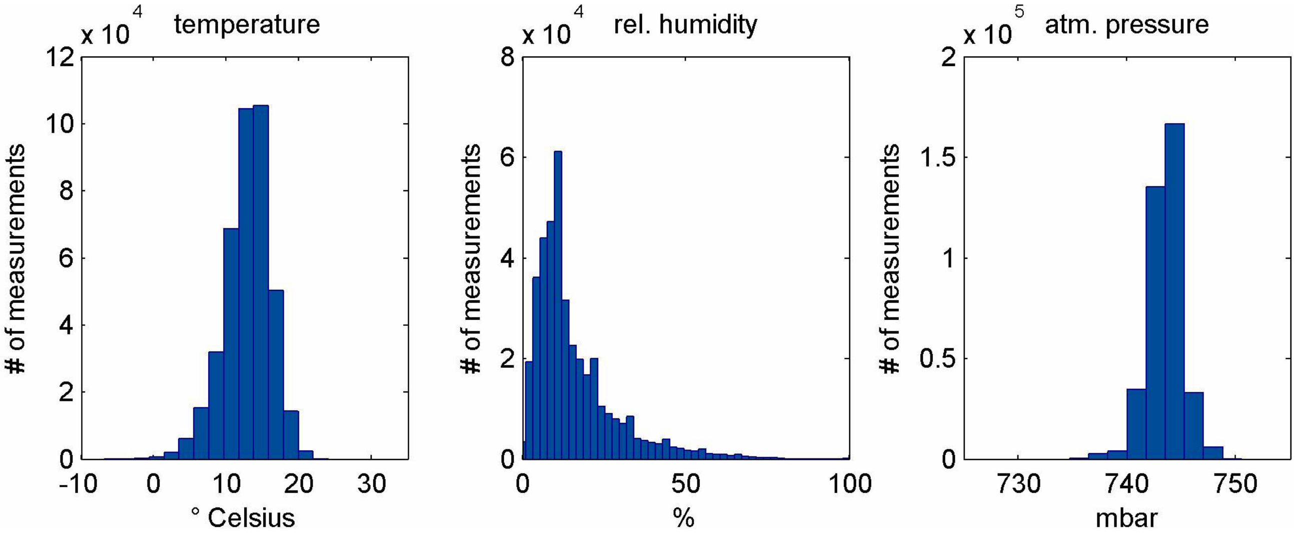

Figure 8.4: Histograms of EMM data (from Jan. 1985 to Jan. 2008). *Left panel*: temperature; *middle panel*: relative humidity; *right panel*: pressure.

[13] http://archive.eso.org/asm/ambient-server

[14] http://www.eso.org/gen-fac/pubs/astclim/paranal/database/

8.1.4 Processing of ESO Meteor Monitor data, GDAS, and MIPAS profiles

The main disadvantage of the GDAS profiles is that they do not represent the local atmospheric conditions of the geographical position and height of the observing site as accurately as provided by the EMM, and even more so for the MIPAS profiles. Therefore, one has to investigate how the three sources of information can be merged into a single profile.

GDAS profile processing The GDAS profiles originate from a server at NOAA[15]. They are retrieved via a dedicated software package GRIB (see Section 9). GRIB downloads a large data set containing the specific GDAS information for the requested point in time. As this data set contains a model for the complete globe, subsequently, the data points for the specified geolocation are extracted. Moreover, as the GDAS data are taken on a 3 hr basis only, two profiles need to be retrieved surrounding the requested point in time. The parameters OBSDATE, UTC, LONGITUDE, and LATITUDE are required for this task (see Section 6.3). They are usually provided by standard and ESO FITS keywords. In the following, we will describe how the resulting two profiles are combined to best match the date of the observations.

If the profiles exist locally, no download from the web-server is required. In addition, the MOLECFIT software distribution contains a compilation of all Cerro Paranal GDAS profiles for the dates from Dec. 01, 2004 to Sep. 30, 2013. Thus, before requesting the data from the web-server (in the case that they do not exist locally already), this database is checked for the existence of the appropriate profiles. For updates of this data set and the retrieval of data for other observing sites, see Section D.1.4.

Unfortunately, the web-server does not provide GDAS profiles for all dates or an internet connection may not always be available. Therefore, MOLECFIT incorporates a fall-back alternative to ensure availability of GDAS data in all occasions. To that end, the monthly averaged profiles from the Cerro Paranal sky model are included as well (for a detailed description see [RD09] and Noll et al. [RD06]). If after checking the local database or the web-server, a profile is still missing, the best-matching profile for the two-month bins Dec.-Jan., Feb.-Mar., ... are taken. The corresponding files, GDAS_t0_sX.fits with X=1 to 6, are located in "share/molecfit/data/profiles/lib/", with X=1 for December/January, X=2 for February/March, X=3 for April/May, X=4 for June/July, X=5 for August/September, X=6 for October/November. This is also done if the site is not Cerro Paranal.

The described procedure for the GDAS profile retrieval is performed if the parameter GDAS_PROF (see Section 6.3) is set to “auto”, which is the default. As an alternative, if GDAS_PROF is set to “null” a specific GDAS-like profile (see Table 8.2) must be provided in the SOF. Finally, it is possible to avoid the use of GDAS profiles by setting GDAS_PROF to “none”.

Time averaged profiles Typically, the requested observation date does not fall exactly onto a single GDAS time slot. Instead of simply retrieving the closest dataset the two neighbouring profiles are obtained. In order to combine the two, a time-weighted average is calculated, i.e. performing a linear interpolation.

Merging GDAS and MIPAS profiles Next, the resulting GDAS profile is merged with the MIPAS standard profile. To that end, the MIPAS profile is regridded to a new irregular height grid with 50 levels (see Figure 8.5) spanning the whole geoelevation range from 2-120 km for Cerro Paranal. The GDAS profile is regridded to the

[15] http://140.90.198.158/pub/gdas/rotating/

same grid in the range 2-26 km and then used to substitute the respective columns in the MIPAS profile. In addition, the four height levels from 20-26 km are not only a simple substitute of the MIPAS data, but a weighted mix of GDAS and MIPAS profile, in order to provide a smooth transition from one dataset to the other. The influence of the GDAS profile decreases with increasing height: 80%, 60%, 40%, 20% at 20 km, 22 km, 24 km, 26 km, respectively. Beyond 26 km, no GDAS information is available.

The discussed fixed grid of layers is used if the parameter LAYERS (see Section 6.3) is set to 1, which is the default. A value of 0 will cause the building of a natural grid consisting of all layers of the MIPAS and the GDAS profile. If local meteo data are used, the observer altitude GEOELEV is also added. The transition from GDAS to MIPAS is performed by means of a decrease of the relative difference of pressure, temperature, and water vapor concentration of both profiles at the height of the uppermost valid GDAS layer up to an altitude which is 1.2 times higher. The resulting grid of layers is (slightly) more accurate than the fixed grid, but also consists of a significantly higher number of levels.

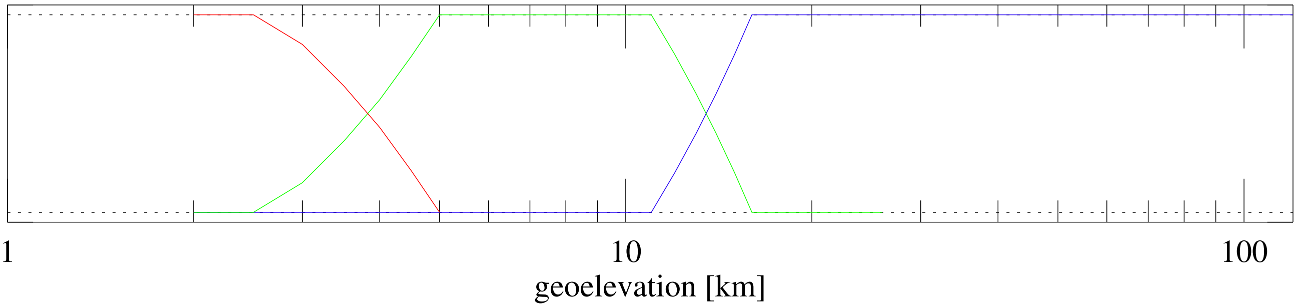

Figure 8.5: *Composition of atmospheric profile:* relative importance of EMM data (red), GDAS data (green) and MIPAS data (blue) as function of geoelevation. Note that the interface boundary of GDAS and MIPAS data varies depending on the availability of the GDAS data.

Combining GDAS/MIPAS profiles with EMM data Observed data from ESO telescopes provide on-site measurements at the ground layer for pressure, temperature, and humidity originating from the ESO meteo monitor at Cerro Paranal. A detailed study of the GDAS data (see Figure 8.6), which represent the local troposphere, including information concerning the dominant wind direction as a function of altitude reveals a gradual reversal (rotation of 180°) at a geoelevation of 5 km, the so-called mixing altitude h_{mix} (see [RD09]). Beyond this altitude, the wind direction remains constant independent of the observation date. Thus, it can safely be assumed that at this altitude the influence of the local environment (as determined from the EMM data) has diminished.

In order to smoothly integrate the EMM data, all GDAS values for pressure, temperature, and humidity below the altitude of the observatory are set to the EMM value. Values above the aforementioned mixing altitude are left untouched. Intermediate values are linearly interpolated resulting in a smooth transition. To this end, first, a logarithmically interpolated value of the GDAS data corresponding to the observatory's altitude h_{tel} is calculated for pressure, temperature, and humidity. These values describe the reference point for the linear decrease of the relative difference between EMM and GDAS data and in the interval $h_{\mathrm{tel}} - h_{\mathrm{mix}}$.

The resulting profile is a smooth combination of all input data, i.e. MIPAS, GDAS, and EMM.

The default mixing altitude of 5 km can be manipulated by changing the parameter EMIX (see Section 6.3). This could be interesting for other observing sites. Setting EMIX to a value lower than the observer altitude GEOELEV causes a profile building without local meteo data. In this case, the parameters PRES, TEMP, and RHUM are ignored.

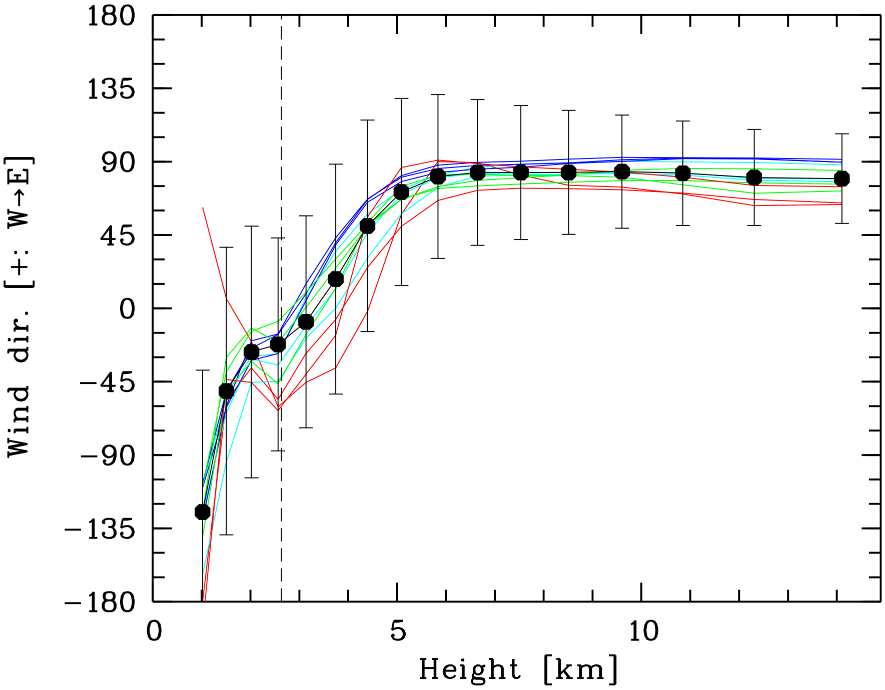

Figure 8.6: *GDAS wind direction as function of geoelevation:* red, cyan, blue, and green curves represent summer, autumn, winter, and spring, respectively. The black symbols show the all year average with the corresponding scatter. The vertical dashed line marks the geoelevation of Paranal. At ∼ 5 km height, a constant plateau is reached.

Scaling of the merged water vapour profile to a given PWV The height profiles of the molecular abundances are defined by the atmospheric standard profile. The only exception is water vapor, where the profile is a combination of modelled and observed data as described above. The GDAS and EMM data used cannot be controlled by the user. There might be cases where this is not satisfying. For example, the user might be interested to use water vapor columns derived from independent measurements for a better start profile. For this purpose, MOLECFIT allows the user to enter a PWV value (parameter PWV, see Section 6.3), which is used to scale the merged water vapor profile. In this way, the input water vapor content of the atmosphere can be fixed. Only the shape of the profile is then ruled by the GDAS and EMM data. The profile scaling factor RELCOL, which is used in the context of the fitting procedure, refers to the modified profile. By default, the PWV option is switched off as indicated by a value of -1.

8.2 Radiative transfer code

MOLECFIT uses the radiative transfer code Line-By-Line Radiative Transfer Model (LNFL v3.2 / LBLRTM v12.11), which is widely used in atmospheric and climate research studies.

8.2.1 Line File / Line-By-Line Radiative Transfer Model (LNFL/LBLRTM)

LBLRTM is developed within the Radiative Transfer Working Group of the AER (see also Clough et al. [RD05], [RD19], and [RD17]). It is publicly available. LBLRTM can handle all molecules incorporated in the `aer_v_<version>` line parameter database [RD17] and offers a wide range of possibilities to adjust input parameters (see [RD25] for more details).

The AER code package used here consists of two programmes: (a) the "Line File" code LNFL, which extracts user selected spectral lines from the line parameter database, and provides these in appropriate form as input for (b) the radiative transfer code LBLRTM. Within MOLECFIT, the most recent versions LNFL v3.2 and LBLRTM v12.11 are used.

Some LBLRTM key features are (taken from [RD17]):

- the Voigt line shape is used at all atmospheric levels with an algorithm based on a linear combination of approximating functions;
- it has been and continues to be extensively validated against atmospheric radiance spectra from the ultra-violet to the sub-millimeter;
- it incorporates the self- and foreign-broadened water vapor continuum model, MT_CKD as well as continua for carbon dioxide, and for the collision induced bands of oxygen at 1600 cm^{-1} (λ = 6.25 μm) and nitrogen at 2350 cm^{-1} (λ = 4.255 μm);
- all parameters of the line database are used including the pressure shift coefficient, the halfwidth temperature dependence, and the coefficient for the self-broadening of water vapor;
- a version of the Total Internal Partition Function (TIPS) programme is used for the temperature dependence of the line intensities;
- the effects of CO_2 line coupling are treated as first order with the coefficients for carbon dioxide generated from Niro et al. [RD10];
- temperature dependent cross section data such as those available with the `aer_v_<version>` database may be used to treat the absorption due to heavy molecules, e.g. the halocarbons;
- an algorithm is implemented for the treatment of the variation of the Planck function within a vertically inhomogeneous layer as discussed in Clough et al. [RD11];
- algorithmic accuracy of LBLRTM is approximately 0.5% and the errors associated with the computational procedures are of the order of five times less than those associated with the line parameters so that the limiting error is that attributable to the line parameters and the line shape;
- its computational efficiency mitigates the computational burden of the line-by-line flux and cooling rate calculation (Clough et al. [RD11]), for example linear algebraic operations are used extensively in the computationally intensive parts of LBLRTM so that vectorisation is particularly effective with a typical vectorised acceleration of 20;
- FFT instrument function with a choice of 9 apodisation functions;

- includes a realistic spectral sea surface emissivity model in the infrared (Masuda et al. [RD12]; Wu&Smith [RD13]);
- input atmospheric profiles in either altitude or pressure coordinates;
- interfaces with other radiative transfer models (like RRTM), and also the forward model for inversion algorithms (e.g. Tropospheric Emission Spectrometer (TES) and Infrared Atmospheric Sounding Interferometer (IASI));
- these attributes provide spectral radiance calculations with accuracies consistent with the measurements against which they are validated and with computational times that greatly facilitate the application of the line-by-line approach to current radiative transfer applications.

In principle, the user can change the setup of the LBLRTM. See the LBLRTM parameters in Table6.2. A change of the fixed LBLRTM input parameters is only recommendable for those users, which have a very good knowledge of the physics of atmospheric radiative transfer and/or LBLRTM.

8.2.2 aer line database

For calculating molecular spectra, the aer_v_<version> database [RD17] is used. It is built from HITRAN 2016 [RD20] and contains several updates. Version 3.8.1.2 covers the full spectral range from $0 - 25,232\,\mathrm{cm}^{-1}$ (i.e. down to 0.4 μm) provides spectral information for 42 molecules. In total, more than 8,900,000 spectral lines are included. The majority is based on modelled data. However, only those 30 molecules are taken into account, which are present in the atmospheric standard profile. The remaining ones are minor trace gases and do not contribute significantly neither to radiance, nor transmission spectra (see Section 8.3). Table 8.2.2 provides an overview of all molecules based on the aer database (and known by LBLRTM) and those contained in the standard atmospheres (Column 4). See further details in Appendix D.1.3

Table 8.3: List of molecules as provided by the `aer` line parameter database

Index	Molecule	Name	Std Atmosphere	LBLRTM
1	H_2O	Water	X	X
2	CO_2	Carbon dioxide	X	X
3	O_3	Ozone	X	X
4	N_2O	Nitrous oxide	X	X
5	CO	Carbon monoxide	X	X
6	CH_4	Methane	X	X
7	O_2	Oxygen	X	X
8	NO	Nitric oxide	X	X
9	SO_2	Sulfur dioxide	X	X
10	NO_2	Nitrogen dioxide	X	X
11	NH_3	Ammonia	X	X
12	HNO_3	Nitric acid	X	X
13	OH	Hydroxyl		X
14	HF	Hydrogen fluoride		X
15	HCl	Hydrogen chloride		X
16	HBr	Hydrobromic acid		X
17	HI	Hydrogen iodide		X
18	ClO	Chlorine monoxide	X	X
19	OCS	Carbonyl sulfide	X	X
20	H_2CO	Formaldehyde		X
21	HOCl	Hypochlorous acid	X	X
22	N_2	Nitrogen	X	X
23	HCN	Hydrogen cyanide	X	X
24	CH_3Cl	Chloromethane		X
25	H_2O_2	Hydrogen peroxide	X	X
26	C_2H_2	Acetylene	X	X
27	C_2H_6	Ethane	X	X
28	PH_3	Phosphine		X
29	COF_2	Carbonyl fluoride	X	X
30	SF_6	Sulfur hexafluoride	X	X
31	H_2S	Hydrogen sulfide		X
32	HCOOH	Formic acid		X
33	HO_2	Hydroperoxyl		X
34	O	Oxygen		X
35	$ClONO_2$	Chlorine Nitrate	X	X
36	NO+	Nitrosonium		X
37	HOBr	Hypobromous Acid		X
38	C_2H_4	Ethylene		X
39	CH_3OH	Methanol		X
40	CH_3Br	Methyl Bromide		X
41	C_3H_8	Propane		X
42	C_2N_2	Cyanogen		X
43	C_4H_2	Diacetylene		X
44	HC_3N	Cyanoacetylene		X
45	H_2	Hydrogen		X
46	CS	Carbon Monosulfide		X
47	SO_3	Sulfur trioxide		X

8.3 Molecular spectra

The process of fitting molecular spectra is a complex task, which requires optimised input parameters. One of the key inputs for the fitting is the number of molecules that are included in the fitting. Fewer molecules in the fitting process result in the code finding a solution in significantly shorter amounts of time. Also the fitting process is a lot more robust. However, if too few molecules are included in the fit, the results may not provide a satisfying residual for model and observation. Optimally, one should include exactly those molecules in the fit, which will significantly contribute over the wavelength range of interest.

To the end of providing some insight into what molecules are required for the fit at a specific wavelength interval, in the following, we provide guidelines with a specific focus on the test data provided with this software package for the instruments CRIRES, VISIR, and XSHOOTER.

We have computed spectra with LBLRTM covering the complete wavelength range from 0.3 $-$ 30 μm using all molecules (i.e. C_2H_2, C_2H_6, CH_4, CO_2, COF_2, CO, $ClONO_2$, ClO, F14, H_2O_2, H_2O, HCN, HNO_3, HOCl, N_2O, N_2, NH_3, NO_2, NO, O_2, O_3, OCS, SF_6, SO_2), individually, i.e. one at a time. At fixed wavelength, we have then calculated the normalised total radiance/transmission (by summing up the contributions of all molecules and subtracting the continuum) and the resulting relative importance of individual molecules.

The Figures C.1-C.5 shown in the Appendix, give all molecules that, over the displayed wavelength range, exhibit at fixed wavelength λ a relative importance of at least 5%. The molecular data have been rebinned to 3000 data points for each individual wavelength range. This results in a varying resolution and in more molecules becoming important over smaller wavelength regimes. C_2H_6, e.g. , has a few significant lines in some of the wavelength ranges shown (e.g. Figure C.2), but does not show up in the overview plot (Figure C.1). Typically, transmission (blue) and radiance (red) plots do not differ significantly. Hence, we do not show them separately. Note that these plots do not allow calculation of absolute fluxes.

These plots can be used to identify the important molecules over any wavelength range. For the range shown in Figure C.2, e.g. , the user ought to include H_2O, CH_4, and O_3. The fitting might mildly profit also from including C_2H_6.

Alternatively, a generic atlas for the molecules listed above is available at share/esopipes/datastatic/molecfit-4.3/molecule_atlas_UVB-MIR.fits, covering the wavelength range from the UVB to the MIR; alternatively, atlases of the relevant molecules for specific ESO instruments are available in share/esopipes/datastatic/molecfit-4.3, and are available in the GUI in the Reflex workflow.

8.4 Thermal emission by telescope

The telescope structure and the observing instrument cause unavoidable thermal emission in the IR. In particular, the telescope main mirror is a significant source of radiation. Hence, for radiance spectra, this background component has to be considered. A simple approximation is the calculation of a grey body spectrum, which equals a black body (BB) of temperature T times a wavelength-independent emissivity ϵ. Since the emitting source, i.e. the main mirror, also absorbs a fraction of $1 - \epsilon$ of the incoming sky radiation, the relative contribution of the telescope emission to the observed radiation is increased. Therefore, the apparent grey body radiation in flux-calibrated spectra can be derived by

$$F_{\rm tel} = \frac{\epsilon}{(1-\epsilon)} \, {\rm BB}(T). \qquad (1)$$

For T, MOLECFIT uses the temperature of the primary mirror (input parameter M1TEMP, see Section 6.3). Since the mirror temperature is close to the ambient temperature of about 280-290 K (Paranal), the grey body emission is expected to be important at wavelengths longwards of the H-band. The emissivity is a free fit parameter (see parameter TELBACK). An initial value can be provided in the parameter file. The default value is 0.1.

8.5 Adaptation of model to input spectrum

A good correspondence of the calculated model spectrum and the observed spectrum is usually prevented by the broadening of the spectral lines by the instrument, small errors in the wavelength calibration, uncertainties in the flux calibration in the case of emission spectra, or the non-flat standard star continuum in the case of transmission spectra. Hence, these unavoidable shortcomings of observed data have to be accounted for in the fitting procedure. For this reason, MOLECFIT modifies the model spectrum using a polynomial fit of the continuum and the wavelength grid. In addition, the model gets convolved with a kernel mimicking the instrumental profile. In the following, we discuss the fit parameters related to this adaptation process in detail.

8.5.1 The continuum

The model spectrum is scaled by a polynomial of degree n_c (CONT_N in the parameter file; see Section 6.3)

$$F_\mathrm{out}(\lambda) = F_\mathrm{in}(\lambda) \sum_{i=0}^{n_\mathrm{c}} a_i \lambda^i. \tag{2}$$

For deriving the $n_\mathrm{c}+1$ coefficients a_i, the zero point of the wavelength grid is shifted to the center of the fit range. For $a_0 = 1$ and all other $a_i = 0$, the model spectrum remains unchanged. This is the default configuration for the initial coefficients. In the parameter file, the initial value of the constant term of the polynomial (parameter CONT_CONST) can be set manually. The continuum correction is carried out independently for each fit range listed in the RANGE_INCLUDE file if such a file is provided. A fit range (or the full spectrum) is further split if it is distributed over more than one chip.

Before correcting the continuum, optionally a flux conversion is carried out. Details on the options selected by the parameter FLUX_UNIT are given in Section 6.3. If the required data units are not included in FLUX_UNIT, this factor must be incorporated into the a_0 coefficient of the polynomial. As a general rule, it is advisable to choose a_0 close to the mean flux (emission) or maximum flux (transmission) of the input spectrum (after consideration of FLUX_UNIT) to optimise the performance of MOLECFIT.

The expert mode (see Appendix A.5) allows the user to specify the initial values for all the polynomial coefficients of the continuum.

8.5.2 The wavelength solution

The wavelength grid of the model spectrum is adapted to that of the observed spectrum by applying a Chebyshev polynomial of degree n_w (WLC_N in the parameter file; see Section 6.3)

$$\lambda' = \sum_{i=0}^{n_\mathrm{w}} b_i t_i, \tag{3}$$

where

$$t_i = \begin{cases} 1 & \text{for } i = 0 \\ \lambda & \text{for } i = 1 \\ 2\,\lambda\, t_{i-1} - t_{i-2} & \text{for } i \geq 2 \end{cases} \tag{4}$$

and λ ranging from -1 to 1. The temporary conversion of the wavelength grid to a fixed interval results in coefficients b_i independent of the wavelength range and step size of the input spectrum. For $b_1 = 1$ and all other $b_i = 0$, the model spectrum remains unchanged. This is the default configuration for the initial coefficients. In the parameter file, the initial value of the constant term of the polynomial (parameter WLC_CONST) can be set manually. This parameter corresponds to a wavelength shift relative to half the full wavelength range. For each chip or FITS extension, the wavelength fit is carried out independently.

For checks or improvements of the input wavelength grid, the model wavelengths rebinned to the input grid are provided in the results tables of MOLECFIT and `calctrans` (column “mlambda”, see Section 6.5). The wavelengths are always given in μm and vacuum. Note that the reliability of this absolute wavelength calibration depends on the quality of the fit. Outside the selected fitting ranges, the wavelengths have to be interpolated or extrapolated by the Chebyshev polynomial. In particular, at optical and near-IR wavelengths, where strong absorption bands suitable for fitting are rare, the provided wavelengths have to be taken with care. For this reason, MOLECFIT does not provide an automatic wavelength solution correction by default.

However, it can be set by using WLC_REF=MODEL in MOLECFIT_CORRECT (see Table 6.3) .

8.5.3 The resolution

The model spectrum is convolved with up to three different line spread functions (LSFs) in order to get similar line shapes as in the observed spectrum. If a profile is not desired, it can be skipped by setting its width and fit flag in the parameter file to zero (see Section 6.3).

The first kernel is a simple boxcar

$$F_{\mathrm{box}}(i) = \begin{cases} 1 & \text{for } -w_{\mathrm{box}}/2 \leq i \leq w_{\mathrm{box}}/2 \\ 0 & \text{for } i < -w_{\mathrm{box}}/2 \cap i > w_{\mathrm{box}}/2 \end{cases} \tag{5}$$

which is adapted to the pixel scale (where the pixel number is represented by i) and normalised to an integral of 1. In the parameter file, the width w_{box} (parameter RES_BOX) has to be given as fraction of the slit width, which is determined by the parameters SLITW, the slit width in arcsec, and PIXSC, the pixel scale in arcsec (see Section 6.3). By default, RES_BOX is set to 1, i.e. the slit width. The fit parameter RES_BOX can only vary between 0 and 2. A value larger than 1 is relevant if the slit width as given in the FITS header is not accurate, which sometimes happened with CRIRES before its upgrade.

The second convolution kernel is a Gaussian

$$F_{\mathrm{gauss}}(i) = \frac{2}{\mathrm{FWHM}}\sqrt{\frac{\ln 2}{\pi}}\, e^{-4\ln 2\, i^2/\mathrm{FWHM}^2} \tag{6}$$

centered on 0. The FWHM is given by the driver file parameter RES_GAUSS in pixels (see Section 6.3). It is restricted to values below 100 pixels. The default value is 1 pixel. The number of pixels in the kernel amounts to KERNFAC (default: 3) times FWHM.

Finally, the third kernel is a Lorentzian

$$F_{\mathrm{lorentz}}(i) = \frac{1}{\pi} \frac{w_{\mathrm{lorentz}}}{i^2 + (w_{\mathrm{lorentz}}/2)^2} \tag{7}$$

centered on 0, where w_{lorentz} is the FWHM. The width is adjusted by the driver file parameter RES_LORENTZ, which has to be provided in pixels (see Section 6.3). It is restricted to values below 100 pixels. The default value is 1 pixel. The number of pixels in the kernel amounts to KERNFAC (default: 3) times w_{lorentz}. Compared to a Gaussian, the Lorentzian approaches the 0-level flux significantly slower, at much larger distances from the maximum.

Note that width zero components can occur in typical conditions, e.g. the line profile is very close to a pure Lorentzian shape. If this is not intended, the user should reduce the number of degrees of freedom by fixing individual fit components. A zero here identifies a unity convolution (i.e. no change of the input spectrum).

The combination of a Gaussian and a Lorentzian is called a Voigt profile. The flag KERNMODE (see Section 6.3) allows the user to apply only a single Voigt profile kernel, which is calculated by an approximate formula that takes the FWHM of Gaussian and Lorentzian as input. In this case (KERNMODE = 1), KERNFAC gives the kernel size in FWHM of the derived Voigt profile and not the FWHM of Gaussian and Lorentzian, as it is done for the default mode of two independent convolutions (KERNMODE = 0). For the fit results, the KERNMODE selection should be less important than the relative contributions of boxcar, Gaussian, and Lorentzian to the fitted line profile. Significant changes in the line profile can cause deviations in the water vapor column of more than 10% (cf. Section 10).

The parameter VARKERN allows the user to fit a kernel that linearly increases with wavelength (see Section 6.3). If the flag is set to 1, this option is selected. It is suitable for dispersion-dominated kernels and constant wavelength bins. In this case, the initial FWHM parameters are given for the central wavelength of the full wavelength range (considering the data of all chips). The default VARKERN = 0 assumes a constant kernel for the entire wavelength range. This option is suitable for narrow wavelength ranges and slit/object profile-dominated kernels.

Finally, the user can provide an optimised kernel via a FITS file – identified in the parameter `kernel_file` (see Section 6.3). If this option is used (i.e. the default "none" is replaced by the corresponding file name), the fixed input kernel overrules the creation of a kernel based on boxcar, Gaussian, and Lorentzian components. Since there will be no fit of the kernel shape and width, the line profile has to be known well. Note that VARKERN = 1 will have no effect, since the pixel-based input kernel is wavelength independent.

9 Installation

As the latest molecfit is a standard esorex pipeline package, downloads and instructions on installation are availble from the ESO pipeline software web pages: https://www.eso.org/sci/software/pipelines/

10 Code Performance

Discussion of the scientific performance of MOLECFIT are given in A. Smette, H. Sana, S. Noll, H. Horst, W. Kausch, S. Kimeswenger, M.Barden, C. Szyszka, A. M. Jones, A. Gallene, J. Vinther, P. Ballester and J. Taylor (2015, A&A 576, A77), and W. Kausch, S. Noll, A. Smette, S. Kimeswenger, M. Barden, C. Szyszka, A. M. Jones, H. Sana, H. Horst, and F. Kerber, (2015, A&A 576, A78).

11 Tips and tricks

In the following, we provide a summary of rules that should be taken into account for a successful application of MOLECFIT:

- Pixels with possible defects that could affect the fit quality can be excluded from the fit in two ways. First, the critical pixels can be listed as a parameter or in a FITS file specified in the parameter `prange_exclude` (see Section 6.3). Second, pixels can be skipped by adding a mask column to an input ASCII or FITS table, or adding an mask extension to a FITS image. In both cases, the name of the column/extension has to be given by the fourth COLUMNS parameter.

- The resulting best-fit parameters of MOLECFIT are written in the `BEST_FIT_PARAMETERS.fits` file. In the case of a complex fit, the more reliable fit parameters could be taken from this file and used as (fixed) input for another iteration of the fitting procedure.

- Changing the fit parameters FTOL and XTOL (see Section 6.3) can significantly affect the code run time and the quality of the fit. In the case of unsatisfying fit results, it may be an option to change the default values. However, the effect is often unpredictable, since more relaxed convergence criteria can lead to worse as well as better fit quality.

- To achieve an optimal performance of the code, one should fit only those molecules that significantly contribute to the wavelength range of the fit. We suggest to base the selection of relevant molecules on the information given in Section 8.3.

- In principle, PWV values can be measured by means of all kinds of spectra of bright standard stars which show telluric lines in absorption and atmospheric emission spectra in the thermal IR. However, a good fit requires significant H_2O features. This criterion cannot be fulfilled if the water lines are very weak as in the optical. Moreover, too low resolution can smooth out the crucial lines, which can make the fit very unstable or even impossible.

- For MOLECFIT applications aiming at the derivation of the atmospheric water vapour content or the telluric absorption correction of astronomical spectra, it is often sufficient to set the abundances of other molecules to a fixed value. For the more frequent molecules in the atmosphere (see Section 8.3), the column density from the input standard profile, i.e. RELCOL = 1, is usually relatively close to the true value. It has to be taken into account that the MIPAS standard profiles were created in 2001 (see Section 8.1.1), which causes deviations for greenhouse gases that indicate a significant long-term increase in atmospheric abundance. For example, the global CO_2 concentration increased by about 6% in one decade, which suggests RELCOL = 1.06.

- The user can explicitly set the initial value of the constant term of the polynomials for the continuum correction and the wavelength solution. The higher-order coefficients are automatically set to reasonable start values if required (see Section 6.3). Setting WLC_CONST is only recommended if a wavelength shift towards a certain direction is expected. The CONT_CONST is more critical. If the continuum level of the input spectrum strongly deviates from 1, even after setting the scaling factor FLUX_UNIT (see Section 6.3), it is prudent to adapt this term.

- For a correct wavelength fit especially at high resolution, it is important to have a correct setting of the VAC_AIR parameter. The wavelength system depends on the instrument and the wavelength calibration approach by the data reduction pipeline. IR data (e.g. CRIRES, VISIR) tend to be provided in vacuum wavelengths, whereas data at shorter wavelengths tend to be provided in air wavelengths (e.g. UVES, X-shooter). If the user does not know this necessary input, it can be easily derived by running MOLECFIT with both VAC_AIR options.

- For the instrumental profile created by the convolution of boxcar, Gaussian, and Lorentzian kernels (see Section 8.5), a derivation from the fit can be difficult. If the χ^2 degeneration by the kernel parameters causes a bad fit, we recommend to modify the initial values of these parameters or to fix the width of a kernel element. Some knowledge on the true functional form of the instrumental profile can be quite helpful. For a first test, the width of the Lorentzian might be fixed. For CRIRES, a value of 0.75 plus a large kernel size could be reasonable (A. Smette 2012, priv. comm.). For our tests, we used 0.5 pixels for all instruments, although X-shooter spectra do not indicate a significant contribution of a Lorentzian. The boxcar kernel can probably be neglected if the slit width does not have a significant impact on the line FWHM. If the user has access to a suitable line profile kernel, it can be imported via the parameter KERNEL_FILE (see Section 8.5.3). The three-component profile fitting is switched off in this case.

- The run time of the radiative transfer code LBLRTM depends on the width of the fitted wavelength ranges. For a better performance, it is, therefore, recommended to use fit ranges as narrow as possible. It also makes the polynomial continuum fit more reliable. For e.g. X-shooter, several representative ranges over the entire wavelength range covering lines of all critical molecules could be defined. A suitable bin size is 10 nm. CRIRES spectra are sufficiently narrow that the full wavelength range can be used for the fitting procedure.

- For spectra covering a wide wavelength range and narrow fit ranges, the degree of the polynomial for the wavelength solution WLC_N should be set to 0 to avoid unpredictable wavelength corrections outside the fit ranges if a telluric absorption correction is required. For X-shooter, it is also strongly recommended to set VARKERN = 1 in order to consider the change of the instrumental profile with wavelength (see Section 6.3).

- For telluric absorption correction in the thermal IR, it could be advantageous to use the strong sky emission extracted as 1D spectrum as input for `molecfit_model`, `molecfit_calctrans` will then calculate the transmission function belonging to the best-fit sky radiance spectrum. Finally, the object spectrum can be corrected for telluric absorption via `molecfit_correct`.

A Expert Fitting

The spectral domain of the input data may be split into several ranges either by user declaration or by physical specifications of multi chip detectors. The expert mode offers greater access to the molecfit fitting parameters with regards these ranges and allows for more flexible iterative fitting. On the other hand, it increases the complexity of the fitting procedure and there is a chance of erroneous results if the masked files are changed and the range and chip related parameters are not carefully adapted. For this reason, it has not been implemented as standard mode for molecfit.

A.1 Mapping Multiple Extension Data

It is possible to represent several science spectra to be processed in mutiple extensions of a single fits file. When the parameter "USE_ONLY_INPUT_PRIMARY_DATA" is set to false, molecfit will assume that all extensions in the input science data fit contain spectra data to be processed. When the "CHIP_EXTENSIONS" flag is set to false it is assumed that they are to be processed independently of each other. In this format, the output files will have multiple extensions each corresponding to the extension index of the input science frame. For example consider the following science frame, CRIRES_HighSNR_Telluric_input_AM1p472.fits, that contains science spectra in extensions 1 and 2, but not in the primary:

Table A.1: Table showing the summary for CRIRES_HighSNR_Telluric_input_AM1p472.fits

Index	Extension	Type	Dimension	Comment
0	Primary	Image	0	
1	SCIENCE1	Binary table	9x1024	Contains the first science spectra
2	SCIENCE2	Binary table	9x1024	Contains the second science spectra

The molecfit_model recipe would then generate derivative files with two extensions: SCIENCE1 and SCIENCE2. For example, the BEST_FIT_PARAMETER.fits file would consist of the following:

Table A.2: Table showing the summary for BEST_FIT_PARAMETERS.fits

Index	Extension	Type	Dimension	Comment
0	Primary	Image	0	
1	SCIENCE1	Binary	3x30	Contains the fitted parameters for SCIENCE1
2	SCIENCE2	Binary	3x30	Contains the fitted parameters for SCIENCE2

When dealing with multiple extension input and multiple extension parameters there may arise a need to specify which extension parameters are to be applied to which science extension input. In particular when applying a specific telluric correction that has already been generated for a set of science spectra. For this purpose the following mapping parameters are available:

Table A.3: Table showing the mapping parameters

Parameter Name	Maps from Extension of:	Maps to Extension of:
MODEL_MAPPING_KERNEL	MOLECFIT_SCIENCE	MOLECFIT_MODEL_KERNEL_LIBRARY
CALCTRANS_MAPPING_KERNEL	MOLECFIT_SCIENCE	MOLECFIT_CALCTRANS_KERNEL_LIBRARY
MAPPING_ATMOSPHERIC	MOLECFIT_SCIENCE	MOLECFIT_ATM_PARAMETERS
MAPPING_CONVOLVE	MOLECFIT_SCIENCE	BEST_FIT_PARAMETERS and ATM_PARAMETERS (process will create an extension in MOLECFIT_TELLURIC_CORR in line with the MOLECFIT_SCIENCE extension)
MAPPING_CORRECT	MOLECFIT_SCIENCE	MOLECFIT_TELLURIC_CORR

The mapping parameters are defined as a string of comma separated extension indices that lists, in order of the primary (index=0) and extensions (index>0) of the input science fits file, the extension number of the associated file they are to be mapped too. For example assume that the science frame contains data in extensions 1,2 and 3, then the identity mapping would be "0,1,2,3", that is Primary is mapped to Primary, extension 1 is mapped to extension 1, extension 2 is mapped to extension 2 and extension 3 is mapped to extension 4. Similarly, a mapping of "2,2,2,2" implies use extension 2 of the associated file for all the input science extensions.

It is further possible to define these extension mappings in separate fits files, by setting the above parameters to NULL and declaring the fits files in the SOF file with the associated tagnames which is the same as the parameter name. Those fits files must then contain the index list in a single column binary table. The name of the column must be the name of the parameter with the suffix "_EXT". For example, if parameter MODEL_MAPPING_KERNEL is set to "NULL" then the fits file specified in the SOF file with tag MODEL_MAPPING_KERNEL must have a single column binary table with the column name as MODEL_MAPPING_KERNEL_EXT.

A.2 Mapping Ranges to Chips

With the definition of multiple ranges and multiple chips some with overlapping wavelength ranges, ambiguity may arise as to which chip a range is to be associated with. To prevent such ambiguity there is a mapping parameter MAP_REGIONS_TO_CHIP which is a string of comma separated chip indices that lists, in order of the range indices. If this string is set as "NULL" it is then assumed that there is a fits table assigned to specify the wavelength inclusion ranges and that there is a column "MAPPED_TO_CHIP" from which the mappings are declared.

A.3 Expert Range Specific Continuum Modelling

The default when fitting for continuum is to model the continuum for each region as an independent polynomial representation of an order which is common for all regions. The fitting options can be extended to specify a sub-selection of regions to fit for only and for each region specify a unique polynomial order for model representation. The following parameters define these range specifc values:

CONTINUUM_N:

A list of integers specifying the polynomial order for each range. For example CONTIUUM_N="0,3,0,1" implies that the polynomial order for ranges 1,2,3,4 are 0, 3, 0 and 1 respectively. If the string is set to "NULL" then it is assumed that there is a fits table assigned to specify the wavelength inclusion ranges and that there is a column "CONT_POLY_ORDER" from which the orders are declared.

FIT_CONTINUUM:

A list of integers (0 or 1) specifying whether a range is to be fitted for or not. For example FIT_CONTIUUM="0,1,0,1" implies that the ranges 2 and 4 are to be fitted for but ranges 1 and 3 are not. If the string is set to "NULL" then it is assumed that there is a fits table assigned to specify the wavelength inclusion ranges and that there is a column "CONT_FIT_FLAG" from which the flags are declared.

A.4 Expert Chip Specific Wavelength Correcting

The default when fitting for wavelength corrections is to model the pixel specific corrections for each chip as an independent polynomial of an order which is common for all chips. As with expert continuum modelling the fitting options can be extended to specify a sub-selection of the ranges associated with each chip to include in the fittings. Note, unlike expert continuum modelling there is no means to uniquely specify a polynomial order for each chip. Further, a chip is not explicitly declared as to be fitted for or not. Rather this is ascertained by whether at least one range associated with that chip has been flagged for fitting. The ranges are flagged with the following parameter.

FIT_WLC:

A list of integers (0 or 1) specifying whether a range is to be included in a wavelength correction of the chip that that range is assigned to. For example FIT_WLC="0,1,0,1" implies that the ranges 2 and 4 are to be included in such fittings but ranges 1 and 3 are not. If the string is set to "NULL" then it is assumed that there is a fits table assigned to specify the wavelength inclusion ranges and that there is a column "WLC_FIT_FLAG" from which the flags are declared.

A.5 Expert Mode Initial Fitting Values

As with any non-linear fitting algorithm, molecfit requires initial values (guess values) as a starting point for the fitting prcedure. These values can have a strong impact on both the run-time and the successful conclusion of the fitting. The closer the initial values are to the correct values the greater the chance of success and the shorter the run-time and vice versa. With regards the polynomial coefficients for continuum modelling and wavelength correction, the default parameters: CONTINUUM_CONST and WLC_CONST only allow the initial definition for these polynomial as a simple shift.The molecfit Expert Mode allows initial values for all the coefficients of all the polynomials being fitted for to be read from a binary table of fits a file. The accepted format of such a table includes the format as rendered in the BEST_FITS_PARAMETER.fits file allowing the user to select the results from a previous fitting as the starting point of a new fitting.

To use this expert mode the .rc file must include the optional parameter EXPERT_MODE set to TRUE and the .sof file must include the pathname for an initial values fits file with tagname "INIT_FIT_PARAMETERS". The expected format of the fits file is to contain a binary table with at least two columns: "parameter" and "value". The expert mode will override default initial values from the values in this table associated with parameter names that match expected format (see table below).

Table A.4: Table showing the parameter name matching formats

Parameter Name String Format:	Case Sensitive:	Examples:
`"boxfwhm"`	YES	boxfwhm
`"gaussfwhm"`	YES	gaussfwhm
`"lorentzfwhm"`	YES	lorentzfwhm
`rel_mol_col_[MOLECULE NAME]`	YES	rel_mol_col_H2O rel_mol_col_O3
`"*RANGE [range idx]*COEF [coef idx]*"`	NO	Range 1, Coef 0 Range 1, chip 1, coef 0 range 1 coef 0
`"*CHIP [chip idx]*COEF [coef idx]*"`	NO	Chip 2, Coef 1 Chip 2, Coef 1

The expert mode will only utilise coefficient values found that are associated with the list of coefficients that have been assigned to any of the previously declared chips or ranges. Any coefficient that is not included in the fits table will be given the value 0.0.

A.6 Expert Analysis of the BEST_FIT_PARAMETERS.fits FILE

Whilst the BEST_FIT_PARAMETER.fits file is generated by molecfit_model for use as input for the molecfit_calctrans recipe, the expert user may look at the contents of this file to determine if the parameter fitting process of molecfit_model has progressed as expected. For such individuals we highlight the structure of this file.

The file consists of a single binary table with three columns labelled: parameter; value; and uncertainty. The parameter column and the value column contains the name and value/final fitted value of each parameter involved within the process. This includes parameters that were chosen to be fitted, parameters that were chosen to be fixed and those parameters for which fitting is not applicable, eg the number of data points. The uncertainty column shows: a value of -1 for those parameters for which fitting is not applicable; 0.0 for those parameters set as a fixed value; and the estimated error of fit for those actually fitted.

For the most part the parameter names listed will make sense, but a few need further clarification.

A.6.1 The value of the `Status` variable

Status is the variable that represents the integer output 'info' from the mpfit function. If the user has terminated execution, info is set to the (negative) value of iflag. see description of fcn. otherwise, info is set as follows.

info = 0 improper input parameters.

info = 1 both actual and predicted relative reductions in the sum of squares are at most ftol.

info = 2 relative error between two consecutive iterates is at most xtol.

info = 3 conditions for info = 1 and info = 2 both hold.

info = 4 the cosine of the angle between fvec and any column of the jacobian is at most gtol in absolute value.

info = 5 number of calls to fcn has reached or exceeded maxfev.

info = 6 ftol is too small. no further reduction in the sum of squares is possible.

info = 7 xtol is too small. no further improvement in the approximate solution x is possible.

info = 8 gtol is too small. fvec is orthogonal to the columns of the jacobian to machine precision.

A.6.2 Wavelength correction

Coefficients for the polynomial fittings for range specific continuum models and the chip specific wavelength corrections:

The wavelength correction is defined as a Chebyshev polynomial correction to the given wavelength calibrations of the chip pixels. For each coefficient of each chip, the parameter name is given in the format: `Chip` *[chip idx]*`,` `coeff` *[coef idx]*

where *coef idx* is the coefficient associated with the Chebyshev function of that order. For example, if the wavelength correction function for chip 2 is a quadratic, then the three associated coefficient parameters would be labeled:

```
Chip 2, coef 0
Chip 2, coef 1
Chip 2, coef 2
```

and *x* represents the wavelength normalized to the whole spectral range of the chip.

If we symbolise these coefficient parameters as c_0, c_1 and c_2 respectively, then the correction function is defined as

$$c_0 T_0(x) + c_1 T_1(x) + c_2 T_2(x) \tag{8}$$

where T_0, T_1 and T_2 are the first three Chebyshev functions

$$\begin{aligned} T_o(x) &= 1 \\ T_1(x) &= x \\ T_2(x) &= 2x^2 - 1 \end{aligned} \tag{9}$$

It is important to note that the identity correction (i.e. no correction) is defined as $c_0 = 0$, $c_1 = 1$ and $c_i = 0$ for all $i \neq 1$.

Thus to model the correction function as a shift to be fitted would be to fit a value for c_0 and set c_1 as fixed with the value 1. This maybe confusing as such a fit is declared by specifying that the polynomial order be zero when in fact it will be of order one by only one coefficient is actually fitted.

A.6.3 Continuum

The situation with the continuum ranges is similar except that standard polynomials are used instead of Chebyshev, and that the parameter name is given by `Range` *[range idx]*`, chip` *[chip idx]* `coef` *[coef idx]*

and that the polynomial order of fit can be unique per range. The storage of the range coeffiecients is that of a rectangular table of size the number of ranges by the maximum polynomial order specified. Coefficients that are not used are fixed at 0.0 and will be displayed in the parameter column with value 0.0 and uncertainty 0.0. Note that the spectral range of the chip is normalized to [-1,1]. For a constant shift (i.e., c_1 = 1, c_i = 0 for $i > 1$), the shift of wavelength is given by [c_0 (λ_{max} - λ_{min}) + (λ_{max} + λ_{min})]/2, where λ_{min} and λ_{max} are the minimum and maximum wavelengths of the spectral range, respectively.

A.7 Scaling the transmission function to the PWV of the science spectrum

In some cases, it is advantageous to execute `molecfit_model` on a telluric star spectrum and `molecfit_calctrans` on a science spectrum. It may happen that the amount of precipitable water vapour (PWV) has changed significantly between the observing time of the two spectra.

Therefore, since version 4.3, the `molecfit_calctrans` recipe includes the capability to scale the transmission function to the PWV of the science spectrum as provided as input to `molecfit_calctrans`. The PWV of the science spectrum may be specified via a numerical value or via values located in header keywords of the science product. This is controlled via the `SCALE_PWV` parameter which must be set to enable the scaling (see Tab. 6.2). The `H2O` column of the `ATM_PARAMETERS` table is then scaled by the ratio of the PWV values. The FITS headers of the `molecfit_calctrans` output products will then contain the additional keywords 'ESO DRS PWV SCI' (Value for the science spectrum), 'ESO DRS PWV TELL' (Best fit from telluric spectrum) and their ratio 'ESO DRS PWV RATIO' (the value of 'ESO DRS PWV SCI' divided by 'ESO DRS PWV TELL').

In addition to these header keywords, the `molecfit_calctrans` output products will also contain the header keywords 'ESO DRS PWV DELTA MJD' and 'ESO DRS PWV DELTA AIRM', providing the difference in Modified Julian Date and Airmass between the science spectrum and telluric spectrum. These are helpful in order to keep track of the time and airmass that has elapsed between the spectra.

In order for these keywords to be meaningful, the appropriate header keywords must be specified via the `molecfit_calctrans` parameters `HDR_MJD`, `HDR_EXP`, `HDR_AIR1` and `HDR_AIR2` (see Tab. 6.2). These header keywords are read directly from the science spectrum, but for the telluric spectrum they are read from a copy of the telluric header that is copied to `BEST_FIT_PARAMETERS` by `molecfit_model`. For these values to be successfully read, the latest version of `molecfit_model` must have produced the `BEST_FIT_PARAMETERS` table. If `SCALE_PWV` is not the default ('none'), then an error will be raised if any of these `HDR_*` keywords are missing. An error does not occur in the default case, but a warning may instead be given.

The success of the PWV scaling depends on the fact that the exposure time of the telluric used by molecfit_model is short, so that the airmass does not change much during the exposure. Hence the airmass mean value is negligibly different from its median value.

Caveat: Currently the PWV scaling works in the simplest case for single extension datasets. Users trying to process more complex multiple extension datasets (e.g. KMOS) may encounter difficulties, in which case it is recommended to leave SCALE_PWV as the default 'none'. The next molecfit release will include a fix for this issue.

A.8 Quality of the correction parameter QC MEAN_ABS_DEV MAX

In previous versions of molecfit_calctrans it was not possible to determine how well the airmass-scaled transmission matches the science spectrum. Since version 4.3, an algorithm has now been incorporated into molecfit_calctrans that determines the quality of the correction. The corrected spectrum is smoothed by running a Savitzky-Golay filter of width SGWL pixels (see Tab. 6.2). The corrected spectrum is divided by the smoothed spectrum in an attempt to remove slopes and other intrinsic features in the spectrum. The normalized result is then subtracted from 1.0 and a mean absolute deviation of the resulting spectrum is calculated (excluding NaN values over the included regions). The value of this quantity is reported in the keyword 'ESO QC MEAN_ABS_DEV MAX' of the products of molecfit_calctrans. The value of this keyword may be compared between different output products of molecfit_calctrans in order to gauge the quality of the fit as determined by various other parameters.

The optional parameter SGWL_ASMAX (see Tab. 6.2) instructs the above algorithm to be run multiple times for a variety of window lengths, namely from SGWL down to a minimum of 5 at decrements of 2 pixels. In this case the value of 'ESO QC MEAN_ABS_DEV MAX' will represent the maximum over this interval. Additional header keywords include 'ESO QC MEAN_ABS_DEV MAX WIDTH', representing the width of the Savitzky-Golay filter at the maximum which the 'ESO QC MEAN_ABS_DEV MAX' value was determined, and 'ESO QC MEAN_ABS_DEV INPUT WIDTH' which stores the input value of SGWL.

B Parameters

B.1 Molecfit Parameter Name Changes: MOLECFIT_MODEL

Table B.1: Table showing how the parameter names for MOLECFIT_MODEL have changed between the original MOLECFIT (4.2.x) and the current version (4.3)

MOLECFIT 4.3+	Type	MOLECFIT 4.2.x
USE_ONLY_INPUT_PRIMARY_DATA	BOOL	–
USE_DATA_EXTENSION_AS_DFLUX	INT (0 implies no)	–
USE_DATA_EXTENSION_AS_MASK	INT (0 implies no)	–
USE_INPUT_KERNEL	BOOL	–
MODEL_MAPPING_KERNEL	STRING	KERNEL_FILE
LIST_MOLEC	STRING	LIST_MOLEC
FIT_MOLEC	STRING	FIT_MOLEC
REL_COL	STRING	RELCOL
WAVE_INCLUDE	STRING	WRANGE_INCLUDE
WAVE_EXCLUDE	STRING	WRANGE_EXCLUDE
PIXEL_EXCLUDE	STRING	PRANGE_EXCLUDE
TELLURICCORR_PATH	STRING	–
TELLURICCORR_DATA_PATH	STRING	–
TMP_PATH	STRING	–
OPENMP_THREADS	INT	–
SILENT_EXTERNAL_BINS	BOOL	–
TRANSMISSION	BOOL	TRANS
COLUMN_LAMBDA	STRING	COL_LAM
COLUMN_FLUX	STRING	COL_FLUX
COLUMN_DFLUX	STRING	COL_DFLUX
COLUMN_MASK	STRING	COL_MASK
MASK_BINARY	BOOL	–
DEFAULT_ERROR	DOUBLE	DEFAULT_ERROR
WLG_TO_MICRON	DOUBLE	WLGTOMICRON
WAVELENGTH_FRAME (has value VAC or AIR)	STRING	VAC_AIR
CLEAN_MODEL_FLUX	BOOL	CLEAN_MFLUX
FTOL	DOUBLE	FTOL
XTOL	DOUBLE	XTOL
FLUX_UNIT	DOUBLE	FLUX_UNIT
FIT_TELESCOPE_BACKGROUND	BOOL	TELBACK
TELESCOPE_BACKGROUND_CONST	DOUBLE	TELBACK
FIT_CONTINUUM	BOOL	FIT_CONT
CONTINUUM_N	INT	CONT_N
CONTINUUM_CONST	DOUBLE	CONT_CONST
FIT_WLC	BOOL	FIT_WLC

continuation of Table B.1

MOLECFIT 4.3+	Type	MOLECFIT 4.2.x
WLC_N	INT	FIT_WLC
WLC_CONST	DOUBLE	WLC_CONST
FIT_RES_BOX	BOOL	FIT_RES_BOX
RES_BOX	DOUBLE	RELRES_BOX
FIT_RES_GAUSS	BOOL	FIT_RES_GAUSS
RES_GAUSS	DOUBLE	RES_GAUSS
FIT_RES_LORENTZ	BOOL	FIT_RES_LORENTZ
RES_LORENTZ	DOUBLE	RES_LORENTZ
KERNMODE	BOOL	KERNMODE
KERNFAC	DOUBLE	KERNFAC
VARKERN	BOOL	VARKERN
OBSERVING_DATE_KEYWORD	STRING	OBSDATE_KEY
OBSERVING_DATE_VALUE	INT	OBSDATE
UTC_KEYWORD	STRING	UTC_KEY
UTC_VALUE	DOUBLE	UTC
TELESCOPE_ANGLE_KEYWORD	STRING	TELALT_KEY
TELESCOPE_ANGLE_VALUE	DOUBLE	TELALT
RELATIVE_HUMIDITY_KEYWORD	STRING	RHUM_KEY
RELATIVE_HUMIDITY_VALUE	DOUBLE	RHUM
PRESSURE_KEYWORD	STRING	PRES_KEY
PRESSURE_VALUE	DOUBLE	PRES
TEMPERATURE_KEYWORD	STRING	TEMP_KEY
TEMPERATURE_VALUE	DOUBLE	TEMP
MIRROR_TEMPERATURE_KEYWORD	STRING	M1TEMP_KEY
MIRROR_TEMPERATURE_VALUE	DOUBLE	M1TEMP
ELEVATION_KEYWORD	STRING	GEOELEV_KEY
ELEVATION_VALUE	DOUBLE	GEOELEV
LONGITUDE_KEYWORD	STRING	LONGITUDE_KEY
LONGITUDE_VALUE	DOUBLE	LONGITUDE_
LATITUDE_KEYWORD	STRING	LATITUDE_KEY
LATITUDE_VALUE	DOUBLE	LATITUDE
SLIT_WIDTH_KEYWORD	STRING	SLITW_KEY
SLIT_WIDTH_VALUE	DOUBLE	SLITW
PIX_SCALE_KEYWORD	STRING	PIXSC_KEY
PIX_SCALE_VALUE	DOUBLE	PIXSC
REFERENCE_ATMOSPHERIC	STRING	REF_ATM
GDAS_PROFILE	STRING	GDAS_PROF
LAYERS	BOOL	LAYERS
EMIX	DOUBLE	EMIX
PWV	DOUBLE	PWV
LNFL_LINE_DB		–
LNFL_LINE_DB_FORMAT		–

continuation of Table B.1

MOLECFIT 4.3+	Type	MOLECFIT 4.2.x
LBLRTM_ICNTNM		–
LBLRTM_IAERSL		–
LBLRTM_MPTS		–
LBLRTM_NPTS		–
LBLRTM_V1		–
LBLRTM_V2		–
LBLRTM_SAMPLE		–
LBLRTM_ALFAL0		–
LBLRTM_AVMASS		–
LBLRTM_DPTMIN		–
LBLRTM_DPTFAC		–
LBLRTM_TBOUND		–
LBLRTM_SREMIS1		–
LBLRTM_SREMIS2		–
LBLRTM_SREMIS3		–
LBLRTM_SRREFL1		–
LBLRTM_SRREFL2		–
LBLRTM_SRREFL3		–
LBLRTM_MODEL		–
LBLRTM_ITYPE		–
LBLRTM_NOZERO		–
LBLRTM_NOPRNT		–
LBLRTM_IPUNCH		–
LBLRTM_RE		–
LBLRTM_HSPACE		–
LBLRTM_REF_LAT		–
LBLRTM_H1		–
LBLRTM_H2		–
LBLRTM_RANGE		–
LBLRTM_BETA		–
LBLRTM_LEN		–
LBLRTM_HOBS		–
LBLRTM_AVTRAT		–
LBLRTM_TDIFF1		–
LBLRTM_TDIFF2		–
LBLRTM_ALTD1		–
LBLRTM_ALTD2		–
LBLRTM_DELV		–
CHIP_EXTENSIONS	BOOL	CHIP

B.2 Molecfit Parameter Name Changes: MOLECFIT_CALCTRANS

Table B.2: Table showing how the parameter names for MOLECFIT_CALCTRANS have changed between the original MOLECFIT (4.2.x) and the current version (4.3)

MOLECFIT 4.3+	Type	MOLECFIT 4.2.x
USE_ONLY_INPUT_PRIMARY_DATA	BOOL	–
USE_DATA_EXTENSION_AS_DFLUX	INT (0 implies no)	–
USE_DATA_EXTENSION_AS_MASK	INT (0 implies no)	–
USE_INPUT_KERNEL	BOOL	–
CALCTRANS_MAPPING_KERNEL	STRING	KERNEL_FILE
MAPPING_ATMOSPHERIC	STRING	–
MAPPING_CONVOLVE	STRING	–
CHIP_EXTENSIONS	BOOL	CHIP

B.3 Molecfit Parameter Name Changes: MOLECFIT_CORRECT

Table B.3: Table showing how the parameter names for MOLECFIT_CORRECT have changed between the original MOLECFIT (4.2.x) and the current version (4.3)

MOLECFIT 4.3+	Type	MOLECFIT 4.2.x
USE_ONLY_INPUT_PRIMARY_DATA	BOOL	–
USE_DATA_EXTENSION_AS_DFLUX	INT (0 implies no)	–
USE_DATA_EXTENSION_AS_MASK	INT (0 implies no)	–
SUPPRESS_EXTENSION	BOOL	–
MAPPING_CORRECT	STRING	–
CHIP_EXTENSIONS=FALSE	BOOL	CHIP

B.4 molecfit_model SOF Tag names

Table B.4: Table showing the SOF Tagnames and their purpose as used in the MOLECFIT_MODEL recipe.

Tag Name	Tag Purpose
SCIENCE	INPUT (mandatory) Contains the spectral data affected by telluric contamination.
STD_MODEL	INPUT Alternative tag name for SCIENCE
MOLECULES	INPUT Only used if recipe parameter, LIST_MOLEC, is set to NULL, in which case this file must be provided. Declares the LIST_MOLEC parameters in a binary table extension with three columns: LIST_MOLEC, FIT_MOLEC, and REL_COL.
WAVE_INCLUDE	INPUT Only used if recipe parameter, WAVE_INCLUDE, is set to NULL, but does not have to be provided. Declares the WAVE_INCLUDE parameters in a binary table extension with two columns: LOWER_LIMIT, and UPPER_LIMIT
WAVE_EXCLUDE	INPUT Only used if recipe parameter, WAVE_EXCLUDE, is set to NULL, but does not have to be provided. Declares the WAVE_EXCLUDE parameters in a binary table extension with two columns: LOWER_LIMIT, and UPPER_LIMIT.
PIXEL_EXCLUDE	INPUT Only used if recipe parameter, PIXEL_EXCLUDE, is set to NULL, but does not have to be provided. Declares the PIXEL_EXCLUDE parameters in a binary table extension with two columns: LOWER_LIMIT, and UPPER_LIMIT
GDAS	INPUT & OUTPUT (optional) User-supplied GDAS profile FITS file
ATM_PROFILE_STANDARD	INPUT (optional) User defined atmospheric standard profile.
ATM_PROFILE_COMBINED	INPUT (optional). User specified file that contains comprehensive grid data of meteorological and molecular abundance information necessary to construct an atmospheric parameter set to use in the LBLRTM tools.
KERNEL_LIBRARY	Library of user provided kernels.
MAPPING_KERNEL	FITS binary table mapping the kernel in the library KERNEL_LIBRARY with the SCIENCE spectrum.
MODEL_KERNEL_LIBRARY	Alternative tag name for KERNEL_LIBRARY
MODEL_MAPPING_KERNEL	Alternative tag name for MAPPING_KERNEL

B.5 molecfit_calctrans SOF Tag Names

Table B.5: Table showing the SOF Tagnames and their puprose as used in the MOLECFIT_CALCTRANS recipe

Tag Name	Tag Purpose
SCIENCE	INPUT (mandatory) Contains the spectral data of interest with telluric contamination.
SCIENCE_CALCTRANS	INPUT Alternative tag name for SCIENCE
MODEL_MOLECULES	INPUT (mandatory) Local copy of a FITS file that contains the LIST_MOLEC parameters as used by the model recipe execution.
KERNEL_LIBRARY	Library of user provided kernels.
MAPPING_KERNEL	FITS binary table mapping the kernel in the library KERNEL_LIBRARY with the SCIENCE spectrum.
CALCTRANS_KERNEL_LIBRARY	Alternative tag name for KERNEL_LIBRARY
CALCTRANS_MAPPING_KERNEL	Alternative tag name for MAPPING_KERNEL
MAPPING_ATMOSPHERIC	FITS binary table mapping the parameters in MOLECFIT_ATM_PARAMETERS with the SCIENCE spectrum.
MAPPING_CONVOLVE	FITS binary table mapping the parameters in both BEST_FIT_PARAMETERS and ATM_PARAMETERS with the SCIENCE spectrum.
ATM_PARAMETERS	INPUT (mandatory) Atmospheric model file containing the necessary parameters to be used in the LBLRTM tools.
BEST_FIT_PARAMETERS	INPUT (mandatory) Parameter file containing the values (assumed derived by the model recipe) fitted for.

B.6 molecfit_correct SOF Tag Names

Table B.6: Table showing the SOF Tagnames and their puproses as used in the MOLECFIT_CORRECT recipe

Tag Name	Tag Purpose
SCIENCE	INPUT (mandatory) Contains the spectral data of interest with telluric contamination.
MAPPING_CORRECT	FITS binary table mapping the kernel in MOLECFIT_TELLURIC_CORR with the SCIENCE spectrum.
TELLURIC_CORR	Telluric correction data.
TELLURIC_DATA	Telluric correction data, mandatory if WLC_REF=MODEL.

B.7 Molecfit Output Files

B.7.1 molecfit_model Output Files

Table B.7: Table showing output files of molecfit_model and their purpose.

File name	File purpose
MODEL_MOLECULES.fits	The list of molecules used in the model execution. This is either constructed from the LIST_MOLEC parameter supplied in the .rc file or is a local copy of the file specified by the MOLECULES tag in the SOF file. To be used as input for caltrans execution when specified in the MODEL_MOLECULES tag of the calctrans related SOF file.
ATM_PARAMETERS.fits	The list of atmospheric parameters used in the model execution. To be used as input for calctrans execution when specified by the ATM_PARAMETERS tag of the calctrans related SOF file.
BEST_FIT_PARAMETERS.fits	The list of fitted parameter used derived in the model execution. To be used as input for calctrans execution when specified by the BEST_FIT_PARAMETERS tag of the calctrans related SOF file.
BEST_FIT_MODEL.fits	Spectra of the fitted model as derived in the model execution.
GDAS.fits	The GDAS meteorological profile data used with MIPAS and EMM data to construct an atmospheric model. Contains grid data for pressure height temperature and relative humidity at a physical range around the required location at the time of observation. Note this is either an interpolation from GDAS_BEFORE.fits and GDAS_AFTER.fits or is a local copy of the profile file specified in the relevant SOF file with the GDAS tagname.
GDAS_BEFORE.fits	GDAS meteorological profile data from a time prior to the science frame capture time. Used with GDAS_AFTER.fits to interpolate GDAS.fits Note this is either a copy extracted from the NOAA packaged data[14] or is a local copy of the profile file specified in the relevant SOF file with the GDAS_BEFORE tagname.
GDAS_AFTER.fits	GDAS meteorological profile data from a time after the science frame capture time. Used with GDAS_BEFORE.fits to interpolate GDAS.fits Note this is either a copy extracted from the NOAA packaged data[14] or is a local copy of the profile file specified in the relevant SOF file with the GDAS_AFTER tagname.

continuation of Table B.7

File name	File purpose
ATM_PROFILE_STANDARD.fits	MIPAS atmospheric composition data used with GDAS and EMM data to construct an atmospheric model. Note this is either a copy extracted from the ESA packaged data[15] or is a local copy of the profile file specified in the relevant SOF file with the ATM_PROFILE_STANDARD tagname.
ATM_PROFILE_COMBINED.fits	The result of combining the GDAS, MIPAS and EMM data and represents the gridded atmospheric profile data to be used in the LNFL and LBLRTM third-party tools. Required as input for the calctrans recipe.

B.7.2 molecfit_calctrans Output Files

Table B.8: Table showing output files of molecfit_calctrans and their purpose.

File name	File purpose
LBLRTM_RESULTS.fits	
TELLURIC_DATA.fits	Input spectrum in BINTABLE format
TELLURIC_CORR.fits	Contains the telluric correction transformation data to be used for convolution with the uncorrected spectrum.

B.7.3 molecfit_correct Output Files

Table B.9: Table showing output files of molecfit_correct and their purpose

File name	File purpose
SCIENCE_TELLURIC_CORR_[RFNAME]	The input science frame corrected for telluric contamination.

Where [RFNAME] is the name of the raw input science file – e.g. if the input science file was

CRIRES_HighSNR_Telluric_input_AM1p472.fits

Then the above output files would be

SCIENCE_TELLURIC_CORR_CRIRES_HighSNR_Telluric_input_AM1p472.fits

[14] Located in the installed file /share/molecfit/data/profiles/gdas/gdas_profiles_C-70.4-24.6.tar.gz

[15] Located within the installed subdirectory /share/molecfit/data/profiles/mipas

C Molecular Spectra

The following figures provide the transmission spectra for specific molecules. Digital version of these atlases are available as FITS binary tables under the share/esopipes/datastatic/molecfit-<version>/

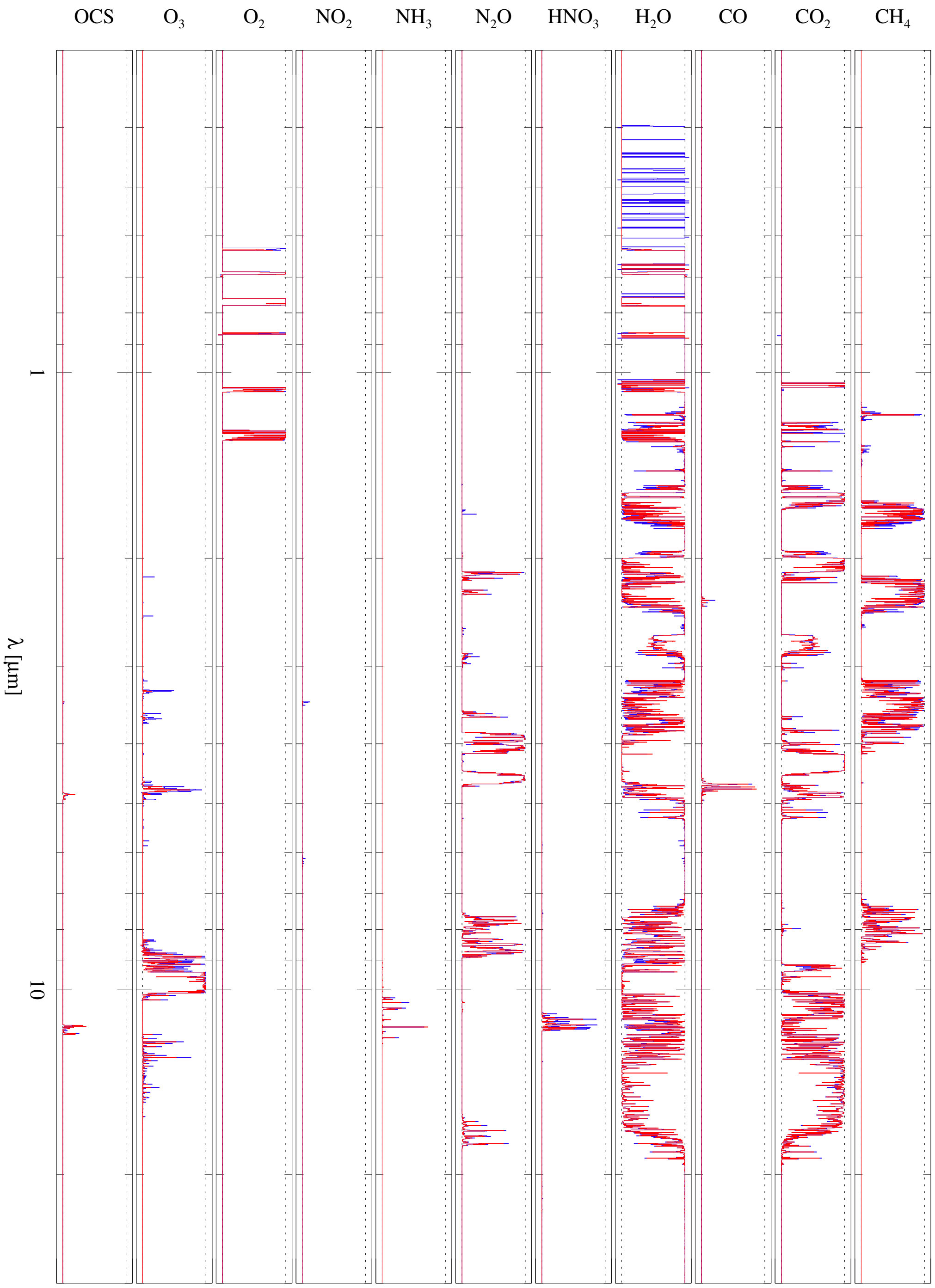

Figure C.1: *Influence of individual molecules as a function of wavelength:* this figure shows the relative importance of all molecules at a given wavelength, which exceed more than 5% of the total radiance (in red) or transmission (in blue).

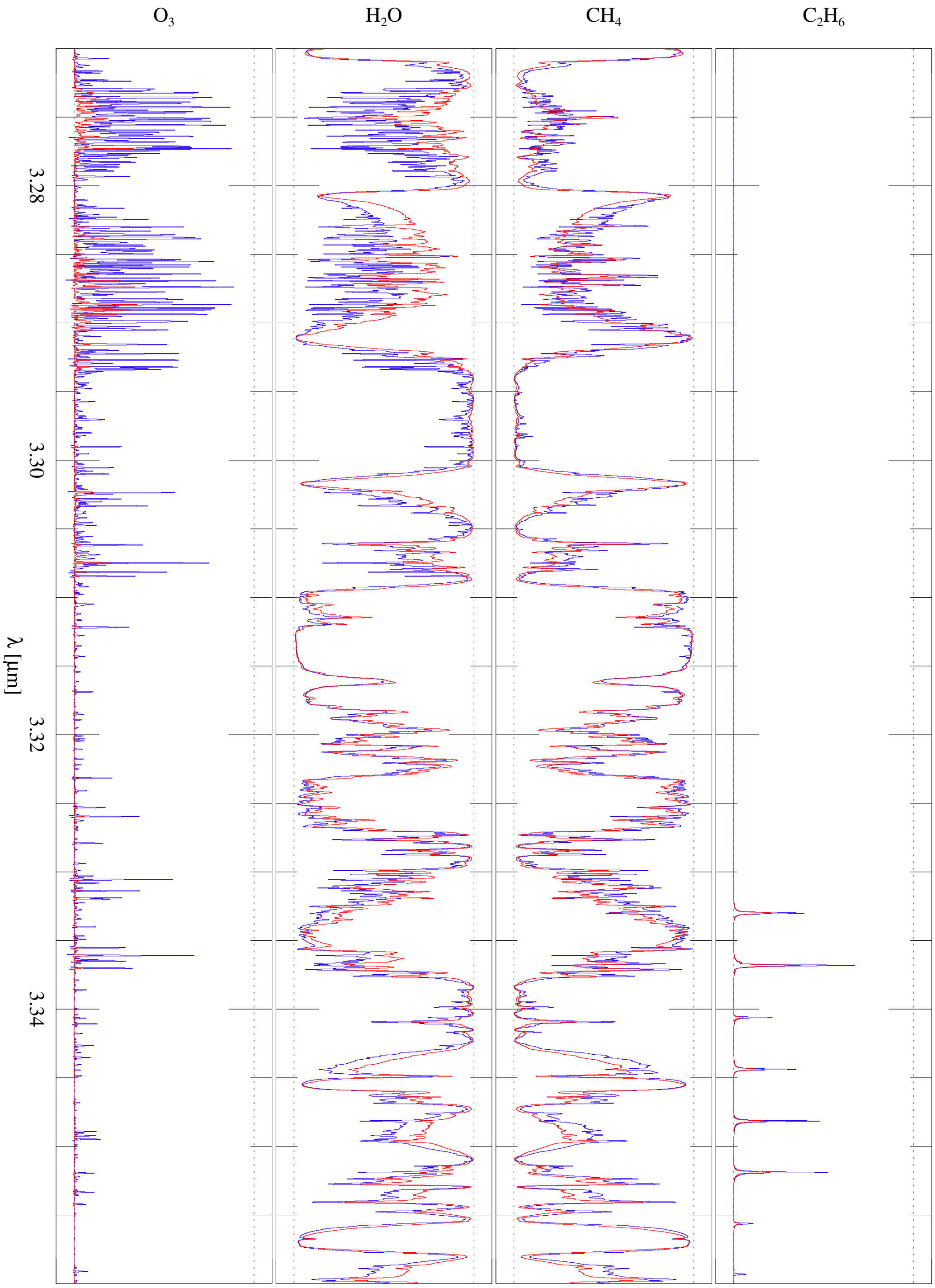

Figure C.2: *Influence of individual molecules as a function of wavelength:* same as Figure C.1, but for the CRIRES test data wavelength regime.

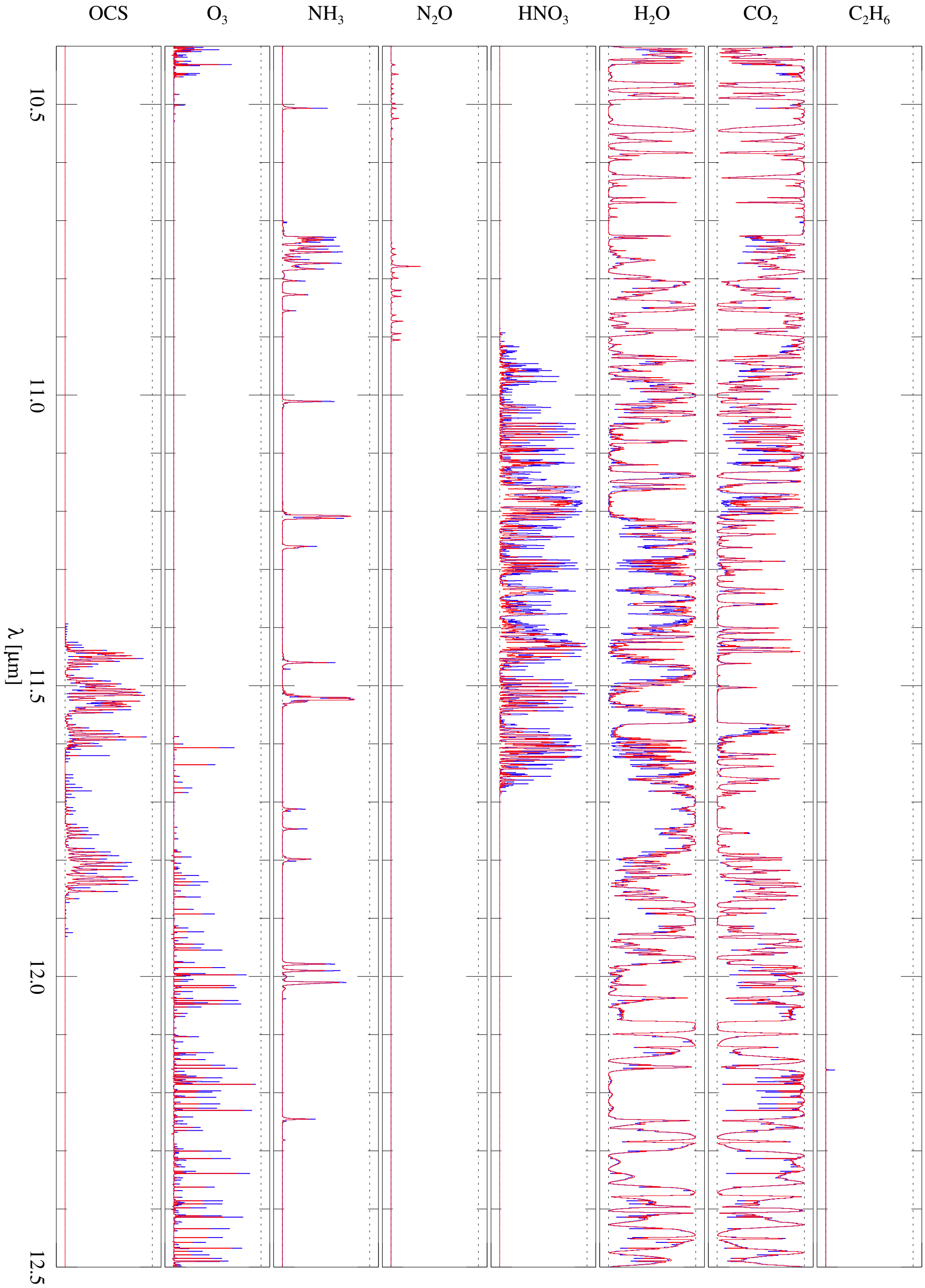

Figure C.3: *Influence of individual molecules as a function of wavelength:* same as Figure C.1, but for the VISIR test data wavelength regime.

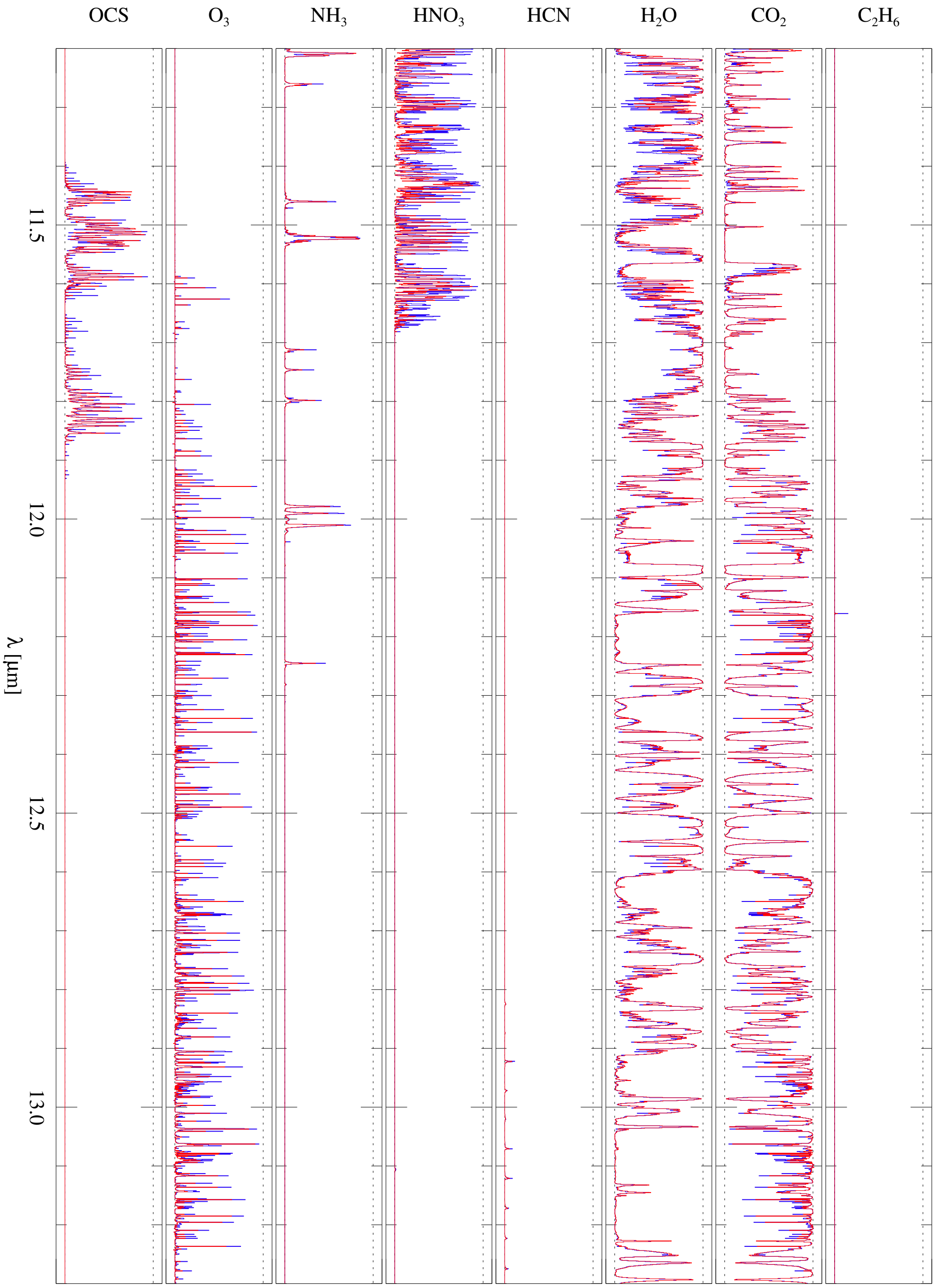

Figure C.4: *Influence of individual molecules as a function of wavelength:* same as Figure C.1, but for the VISIR test data wavelength regime.

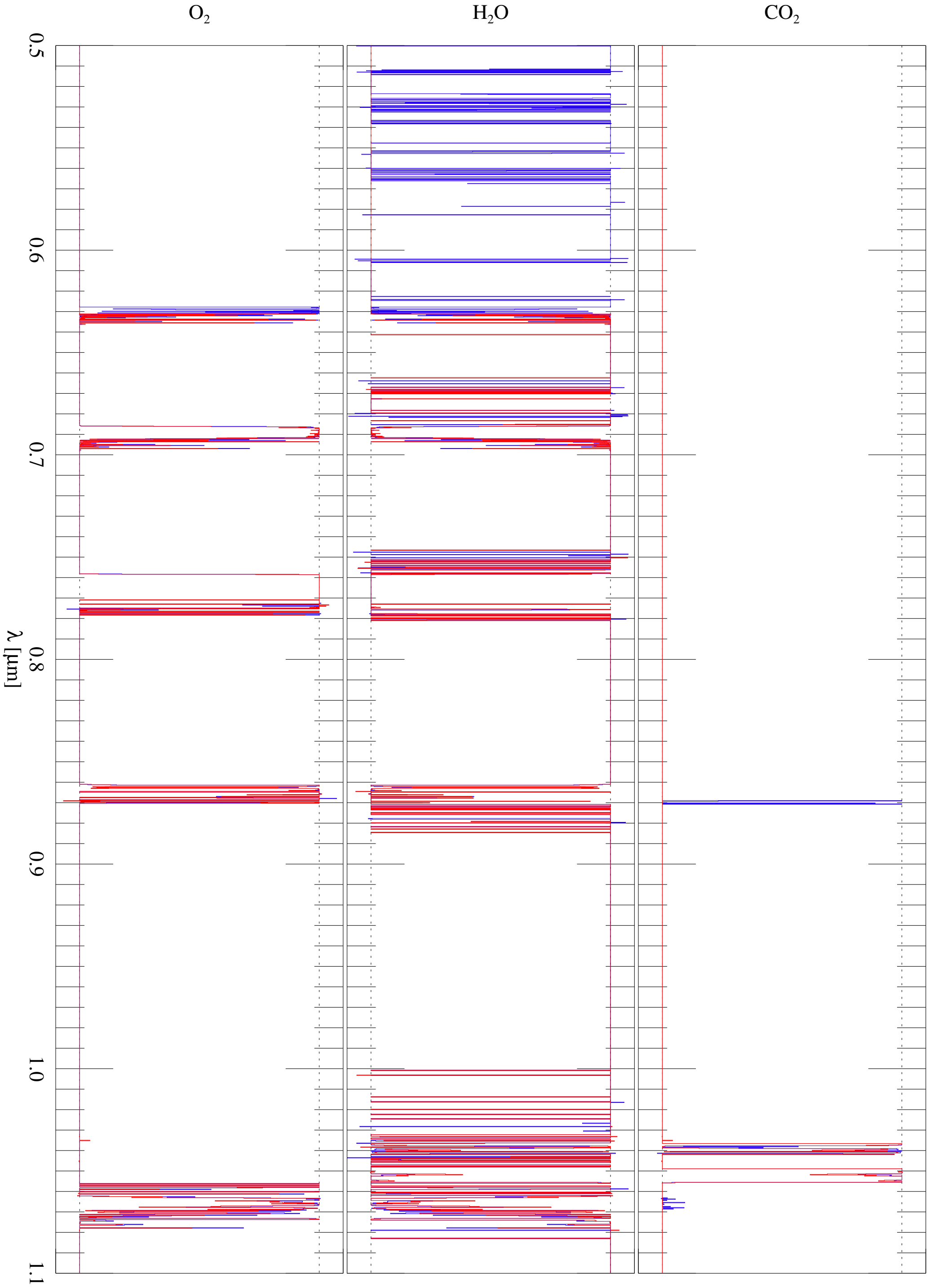

Figure C.5: *Influence of individual molecules as a function of wavelength:* same as Figure C.1, but for the X-Shooter VIS-arm test data wavelength regime.

D Maintenance and License Issues

D.1 Maintenance

D.1.1 Introduction

The package MOLECFIT relies on some external code and data, which might undergo a change due to development after the current release. Although it is not possible to foresee all possible modifications, some hints can be given to use MOLECFIT even with updates from the external sources at a later stage. This particularly applies to the radiative transfer code LNFL/LBLRTM, and the GDAS data. In this appendix, we summarise the issues to be taken into account for a successful usage of MOLECFIT with later versions of LNFL/LBLRTM, the line database, and the GDAS data.

D.1.2 Radiative transfer code LNFL/LBLRTM

The radiative transfer code package LNFL/LBLRTM is developed by AER [RD25] and can directly be obtained from there. MOLECFIT is delivered with LNFL v3.2, LBLRTM v12.11, and the modified AER line database v3.8.1.2.

In principle, MOLECFIT should also work with later versions as its principle usage has been unchanged for several years. This means that MOLECFIT is expected to provide all functionality as long as it is retained. In particular, the principle usage of the TAPE<xx> files [RD28], and the format of the LBLRTM input file `TAPE5` MUST remain unchanged. However, as LBLRTM is widely used in atmospheric research, we do not expect major changes in the near future. In the course of the former DR-, and later SM-projects, we used several versions without problems.

What might change is the installation procedure of the LNFL/LBLRTM codes. If a future release of MOLECFIT is provided with a newer version of the radiative transfer code, the install script might need to be modified accordingly. For that purpose, the developers from AER provide a README file in the tree of the sources. In order to maximise flexibility, MOLECFIT only searches for executables labelled `lnfl` and `lblrtm` in the `<INST_DIR>/bin/` directory. Therefore, a soft link to the actual binaries of LNFL and LBLRTM has to be provided by the installation script. To facilitate the modification in the script, the corresponding section is marked there.

We recommend the following procedure for the update of MOLECFIT to later versions of LNFL/LBLRTM:

- Updates of the LNFL/LBLRTM are usually once or twice per year (see Section "What's new" at [RD25]). Therefore, a quarterly check for new versions is sufficient.
- Download the latest version of LNFL/LBLRTM from [RD25]
- Make sure that the principle usage has not changed. Check the documents in the `docs` directory in the tree of the LBLRTM sources. There, you should find a FAQ document, the release notes, and a detailed HTML description of the `TAPE5` input file for LBLRTM.
- Read the `README` files whether the installation procedure has changed. These files should be located in the `src/build` directory of LNFL and LBLRTM.

- Check whether there is also a line list update and follow the steps described in Section D.1.3.
- Adapt the installation script of MOLECFIT.
- Evaluate by means of the delivered examples.

D.1.3 AER Line parameter list

Usually the line parameter list delivered with the LNFL/LBLRTM package is updated at the same time as the code. We recommend to use the version which is delivered together with the corresponding code version.

To integrate the new line list, simply unpack the tarball and copy the new line parameter file (usually labelled `aer_v_<version>` to the directory `<INST_DIR>/data/hitran/` of your MOLECFIT installation. This can be done also be done by modifying the install script, as the corresponding section is marked. The parameter `LNFL_LINE_DB` should be updated accordingly.

Note that it has turned out that at long wavelengths relevant for VISIR, MOLECFIT can be significantly slower if the recommended `aer` line list is used instead of the basic HITRAN 2008 database. Unfortunately, this line list is no more available at the HITRAN website [RD20], which now provides the 2016 version. The new version has not been tested in terms of performance in MOLECFIT. Hence, it cannot be guaranteed that it works at all. If you want to test an original HITRAN line list, you have to put it in the `data/hitran/` folder and modify _LINE_DB in the `lblrtm_setup` file. In addition, the parameter _LINE_DB_FMT has to be changed from 100 to 160 because of a different file format.

The version of the database provided with version 4.3 of molecfit was built from version aer_v_3.8.1 (`https://github.com/AER-RC/AER_Line_File/wiki/What's-New#aer-line-file`) but modified by removing duplicated lines. In addition, as suggested by Romain Allart, the H2O lines at wavenumber = 16984.8471 cm^{-1} (588.7608 nm) and wavenumber = 15437.8563 cm^{-1} (647.7583 nm) have also been removed.

D.1.4 GDAS data

In contrast to the radiative transfer code, which is updated once/twice per year, the GDAS profiles are updated permanently. MOLECFIT initially searches in a local tarball for the corresponding profiles (script `<INST_DIR>/bin/get_gdas_profiles.sh`), and tries to download it from the web in the case they are not found locally. The latter is done by the script `<INST_DIR>/bin/extract_grib.sh` with the help of the GRIB software.

However, the incorporated download server[16] does not seem to be very stable and organised in a predictable way. It might happen, that the profiles are not found. Therefore data tarballs updated daily are available from `https://ftp.eso.org/pub/dfs/pipelines/skytools/molecfit/gdas`. They can be downloaded and placed into the `<INST_DIR>/data/profiles/gdas` folder in case the automatic update mechanism fails. Note that this directory needs to be write enabled to have the new version permanently available.

[16] `http://nomad3.ncep.noaa.gov/pub/gdas/rotating/`

Additionally MOLECFIT contains the script used to produce the updated files for the website. It can also be used to update the local GDAS database. The script downloads the GDAS data archive from a different server[17]. The data there is organised on weekly basis, each stored in a single file labelled `gdas1.<mmm><yy>.w#`, being <mmm> the month (e.g. jan, feb, mar,...) and <yy> the year. The extension "w#" describes the number of the week of this month:

#=1 - days 1-7 of the month
#=2 - days 8-14
#=3 - days 15-21
#=4 - days 22-28
#=5 - days 29 - rest of the month

For example, the file `gdas1.apr07.w3` contains data of the third week of April 2007 (15th to 21st of April 2007). More information on the file structure is given here[18].

These files provide the GDAS profiles of the world wide grid. Therefore, it is necessary to extract the profiles in the same way as described in Section 8.1. This can be achieved with the shell script `update_gdas_db.sh`. It downloads the data on monthly basis, and extracts the profiles as required by MOLECFIT, and adds them to the local database. However, this may take a while as the amount of data to be downloaded is large (up to 580 MB per week) and the server is fairly slow.

The shell script invokes a C-programme `extract_gdas_profiles`. Although, both tools are part of the molecfit package, they can be used independently of MOLECFIT to provide the possibility to update the GDAS archive on different machines. This might be useful as some disc space is required temporarily. If this is intended, both tools have to be installed in the following way on the target machine:

- create subfolders `bin/` and `data/downloads/paranal/` in the target directory on the target machine.
- Copy the shell script `update_gdas_db.sh` and the C source file `extract_gdas_profiles.c` to the `bin/` folder in target directory on the target machine.
- There compile the C programme by invoking

```
gcc extract_gdas_profiles.c -lm -o extract_gdas_profiles
```

- Copy the tarball file `gdas_profiles_C-70.4-24.6.tar.gz` delivered with the MOLECFIT package to the directory `<targetdir@targetmachine>/data/downloads/paranal/`. This tarball is being updated.

To update the local GDAS database please follow these steps:

- The final release of MOLECFIT contains GDAS profiles up to March 2014. So, if this is your first update, invoke the update script with the parameter oct13:

[17] ftp://ftp.arl.noaa.gov/archives/gdas1/
[18] http://www.ready.noaa.gov/gdas1.php

```
cd <INST_DIR>/
```

OR

```
cd <targetdir@targetmachine>
```

(depending on your installation (see above)

- invoke the update script

```
bin/update_gdas_database.sh oct13 P
```

and proceed until the end of the last fully missing month, if necessary. Note: You only can download full months. The file `data/paranal/last_month.txt` shows a history of past updates.

- Copy the resulting tarball `data/downloads/paranal/gdas_profiles_C-70.4-24.6.tar.gz` to the folder `<INST_DIR>/data/profiles/gdas/` in your MOLECFIT installation path.
- Optionally: delete the downloaded files in `data/downloads/gdas/`.

Some dates are missing. Please check the website[19] for more details. There are no GDAS data available before 1st of December 2004.

D.2 License issues

D.2.1 LNFL/LBLRTM

URL: http://rtweb.aer.com/lblrtm_frame.html

LNFL Terms and Conditions The LNFL software you are downloading has been copyrighted by Atmospheric and Environmental Research, Inc. (AER) and is proprietary to AER. AER grants USER the right to download, install, use and copy this software for scientific and research purposes only. By downloading this software, you are agreeing that:

- If you redistribute any portion of the software, whether in original or modified form, each source code file redistributed will include the AER copyright notice currently present in that file.
- The software or any modified version of the software may not be incorporated into proprietary software or commercial software offered for sale.
- Should you utilize the results of LNFL in your research, we request that you appropriately reference this usage in any associated publications.
- This software is provided as is without any express or implied warranties.

[19] http://ready.arl.noaa.gov/archives.php

LBLRTM Terms and Conditions The LBLRTM software you are downloading has been copyrighted by Atmospheric and Environmental Research, Inc. (AER) and is proprietary to AER. AER grants USER the right to download, install, use and copy this software for scientific and research purposes only. By downloading this software, you are agreeing that:

- If you redistribute any portion of the software, whether in original or modified form, each source code file redistributed will include the AER copyright notice currently present in that file.
- The software or any modified version of the software may not be incorporated into proprietary software or commercial software offered for sale.
- Should you utilize the results of LBLRTM in your research, we request that you appropriately reference this usage in any associated publications.
- This software is provided as is without any express or implied warranties.

Principal References Clough, S. A., M. W. Shephard, E. J. Mlawer, J. S. Delamere, M. J. Iacono, K. Cady-Pereira, S. Boukabara, and P. D. Brown, Atmospheric radiative transfer modeling: a summary of the AER codes, Short Communication, J. Quant. Spectrosc. Radiat. Transfer, 91, 233-244, 2005.

Clough, S.A., M.J. Iacono, and J.-L. Moncet, Line-by-line calculation of atmospheric fluxes and cooling rates: Application to water vapor. J. Geophys. Res., 97, 15761-15785, 1992.

D.2.2 cmpfit library

URL : http://www.physics.wisc.edu/ craigm/idl/cmpfit.html **URL :** http://cow.physics.wisc.edu/ craigm/idl/idl.html

DISCLAIMER MPFIT: A MINPACK-1 Least Squares Fitting Library in C

Original public domain version by B. Garbow, K. Hillstrom, J. More' (Argonne National Laboratory, MINPACK project, March 1980) Copyright (1999) University of Chicago (see below)

Tranlation to C Language by S. Moshier (moshier.net) (no restrictions placed on distribution)

Enhancements and packaging by C. Markwardt (comparable to IDL fitting routine MPFIT

see http://cow.physics.wisc.edu/ craigm/idl/idl.html) Copyright (C) 2003, 2004, 2006, 2007 Craig B. Markwardt

This software is provided as is without any warranty whatsoever. Permission to use, copy, modify, and distribute modified or unmodified copies is granted, provided this copyright and disclaimer are included unchanged.

Source code derived from MINPACK must have the following disclaimer text provided.

==
Minpack Copyright Notice (1999) University of Chicago. All rights reserved

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. WARRANTY DISCLAIMER. THE SOFTWARE IS SUPPLIED "AS IS" WITHOUT WARRANTY OF ANY KIND. THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, AND THEIR EMPLOYEES: (1) DISCLAIM ANY WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT, (2) DO NOT ASSUME ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF THE SOFTWARE, (3) DO NOT REPRESENT THAT USE OF THE SOFTWARE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS, (4) DO NOT WARRANT THAT THE SOFTWARE WILL FUNCTION UNINTERRUPTED, THAT IT IS ERROR-FREE OR THAT ANY ERRORS WILL BE CORRECTED.

5. LIMITATION OF LIABILITY. IN NO EVENT WILL THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, OR THEIR EMPLOYEES: BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL OR PUNITIVE DAMAGES OF ANY KIND OR NATURE, INCLUDING BUT NOT LIMITED TO LOSS OF PROFITS OR LOSS OF DATA, FOR ANY REASON WHATSOEVER, WHETHER SUCH LIABILITY IS ASSERTED ON THE BASIS OF CONTRACT, TORT (INCLUDING NEGLIGENCE OR STRICT LIABILITY), OR OTHERWISE, EVEN IF ANY OF SAID PARTIES HAS BEEN WARNED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGES.

D.2.3 HITRAN

URL: http://www.cfa.harvard.edu/hitran/
URL: http://www.cfa.harvard.edu/hitran/Updated/ref-table.pdf

Principal references Rothman et al., "The HITRAN 2008 molecular spectroscopic database", Journal of Quantitative Spectroscopy and Radiative Transfer, vol. 110, pp. 533-572 (2009)

D.2.4 GDAS

URL: http://ready.arl.noaa.gov/gdas1.php

Air Resources Laboratory - Disclaimer The information on government servers is in the public domain, unless specifically annotated otherwise, and may be used freely by the public. Before using information obtained from this server special attention should be given to the date and time of the data and products being displayed. This information shall not be modified in content and then presented as official government material. The user shall credit NOAA Air Resources Laboratory in any publications, presentations, or other derivative works that result from the use of this material. NOAA Air Resources Laboratory is not obligated to provide any support, consulting, training, or assistance of any kind with regard to the use of this material.

This material is provided "AS IS" and any express or implied warranties including, but not limited to, the warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall NOAA Air Resources Laboratory be liable to you or to any third party for any direct, indirect, incidental, consequential, special or exemplary damages or lost profit resulting from any use or misuse of this data.

This server is available 24 hours a day, seven days a week, however this server is not operational and is not maintained 24 hours a day, seven days a week. Timely delivery of data and products from this server through the Internet is not guaranteed.

Any references or links from the ARL web server to any non-Federal Government Web sites do not imply endorsement of any particular product, service, organization, company, information provider, or content. We are not responsible for the contents of any "off-site" web pages referenced from ARL servers.