

COCCON Software Description



Contact: F. Hase, KIT, frank.hase@kit.edu

DISCLAIMER: THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This documentation explains in short the usage of software tools provided here for the COCCON (Collaborative Carbon Column Observing Network) data analysis. The development of the preprocessing tool has been supported by ESA. Peripheral items are still under development (e.g. batch processing for larger number of days, generation of netCDF files from ASCII-table output, ...), but the main functionalities are all available in this distribution, so the code is ready to use. The data processing has two steps, reflected in 1 + 2 independent pieces of code:

- (1) The preprocessing: generates spectra in a binary format which are then used for the subsequent quantitative trace gas analysis. This code also checks the quality of the raw data, it generates a *.bin output spectrum only if no problems were detected during the processing chain (error flag status zero). It performs a DC correction, FFT, phase correction, and a

resampling of the spectrum to a minimally sampled grid. This code has been created by KIT in the framework of ESA's COCCON-PROCEEDS project.

- (2) The quantitative trace gas analysis. This is a 2-step-sequence: (a) The "pcxs" program (pcxs: pre-calculate x-sections) is used for tabulating the cross-sections of all relevant gases in the $\sim 4000 \dots 8600 \text{ cm}^{-1}$ spectral region. This program needs to be started once per measurement day, the tabulated values are reused for all retrievals afterwards. The execution takes several minutes, a big "...abscos.bin" results from this action (file size $\sim 270\text{MB}$). (b) afterwards, the "invers" program (performs the inversion) is used for performing the trace gas retrievals for all species on all available calibrated spectra. The program fits the column amount of each target gas (and further interfering species and auxiliary quantities), and finally performs the required post-processing steps for creating the final estimates of column-averaged abundances (called XCO_2 , XCH_4 , etc.) by applying air-mass-independent and air-mass-dependent corrections.

On the longer run, we assume that performing the preprocessing will be the only required action by the site operator. After having performed this operation, the operator will submit the calibrated spectra (together with some auxiliary data) to a central facility for the standardized trace gas analysis. After the processing on the central facility, the operator is informed that the column-averaged abundances retrieved from the submitted spectra are available for download. The operator will decide whether the spectra should be kept at the central facility for further use (e.g. reanalysis if new line lists or other improvements become available), and whether the trace gas results should be added to the public COCCON data pool. Until this final goal is achieved, the complete processing as described above needs to be done by the instrument operator.

The COCCON data processing software suite is source open, so you are not required to sign a license: all source codes are included in the distribution, you are free to use the included executables (currently for windows), to study or modify the source codes, etc., but KIT will not take any liability (note the above disclaimer!). All three codes are written in Fortran and are largely portable. The preprocessing tool is fully compatible with Fortran 2003 standard and therefore is fully portable to other platforms. The "pcxs" and "invers" codes follow Fortran 95 standard using a few Lahey compiler specific non-standard commands, so minor adjustments might be required when using other compilers or operating systems. We plan to upgrade these sources for reaching full compatibility with Fortran 2003 in the near future.

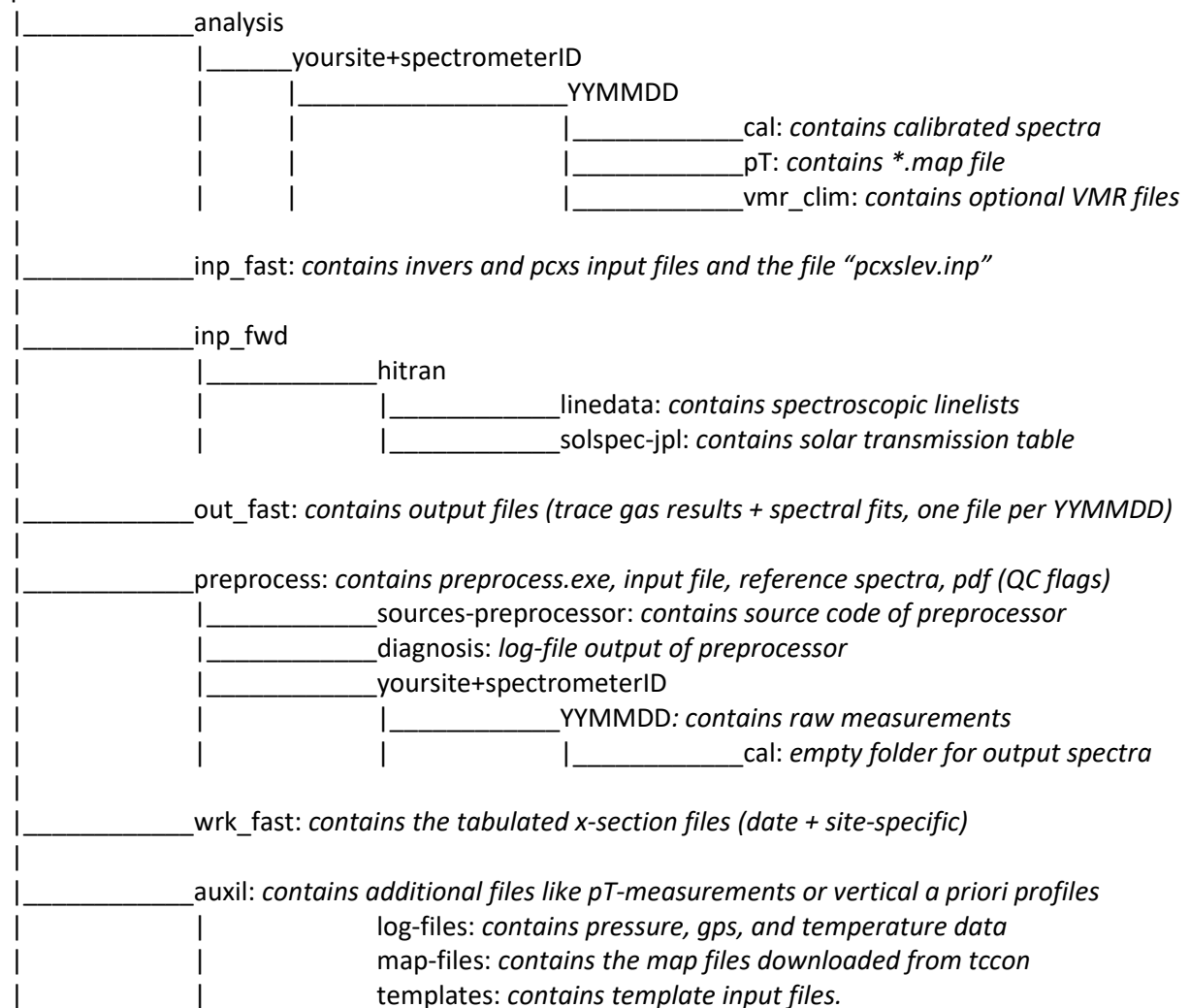
The codes have been tested on a large dataset of raw measurements and for different EM27/SUN spectrometers. We will add more tests from additional sites in the future (and we are happy about your helpful feedbacks if you detect problems with the code!). The current version does not yet support mobile applications, but assumes the standard situation of COCCON that a spectrometer is setup at a chosen place and operated during one or several measurement days.

We strongly suggest to maintain the chosen folder structure and naming conventions. Especially, the "invers" and "pcxs" executables should reside below the "prf96-em27-fast" folder. As in the demo file, the folders "analysis", "inp_fast", "inp_fwd", "out_fast", and "preprocess", and "wrk_fast" should also reside there.

NOTE: For a large number of files, the procedure described in this file is quite cumbersome. Hence, we provide several python scripts, which automatized the procedure in this file. The purpose of this file is mainly to get deeper insides of the software.

Recommended folder tree structure

prf96-em27-fast

**Further remarks concerning folder structure:**

- The “analysis” folder might contain several different “yoursite+spectrometerID” subfolder structures
- The “preprocess” folder might contain several different “yoursite+spectrometerID” subfolder structures – but we recommend storing raw OPUS files and output files in this part of the folder tree only temporarily. The backup of OPUS files should reside elsewhere, and we suggest to remove the raw OPUS files after successful calibration and to move the “preprocess\yoursite+spectrometerID\YYMMDD\cal” folder with its contents to “analysis\ yoursite+spectrometerID\YYMMDD\cal” for the further analysis. Store all calibrated spectra permanently in the “analysis” for further use and possible reprocessing.
- We assume that users far from the Greenwich meridian will use local time on their computer (see comments on time zone below) and sort the raw OPUS files and resulting calibrated spectra according to their “local date”, thereby collecting the complete dataset of a measurement day in one single YYMMDD folder (despite a date change has occurred in Greenwich during this “local date”).

General remarks to the input files

The key to understand the input files are the “\$” signs: The file contains comments and data, where the data is stored in blocks. The algorithm reads the file line by line until it finds a “\$”-sign. This tells the read-in algorithm to be at the next data block and hence the next few lines contain relevant data. What kind of data, and how many lines the block comprises, is hardcoded in the code itself. To sum it up, this means for users dealing with these files:

- It is possible to add as much comments as you like *before* each dollar sign.
- One must *NOT* change the order or the formatting of the entries of the lines containing the actual data.
- In the convention of these files, the data-line of each block are described in the same order as the data itself.

Perform Preprocessing

Manual starting the preprocessing:

1. Put your OPUS files recorded during measurement day YYMMDD in “*prf96-em-27-fast\preprocess\mysite\YYMMDD*”,. Within this newly created folder, make an empty subfolder “*prf96-em-27-fast\preprocess\mysite\YYMMDD\cal*”.
2. Edit “*prf96-em-27-fast\preprocess\ preprocess4.inp*”, with your site-specific inputs:
 - a. ILS parameters of your spectrometer from the table provided in the paper by Frey et al., AMT, 2018. If your spectrometer has not yet been calibrated, use the average guess value of “0.983 0.000” for both channels.
 - b. latitude and longitude (in deg), altitude (in km)
 - c. list of raw OPUS files of the measurement day to be processed (add *** as end marker below list)
3. Now you can start the preprocessing either by typing “*preprocess4.exe*” or via the “*start-preprocess.bat*” batch. The latter will start a selectable number of preprocessor runs in parallel (this is realized by a call with additional parameters, each call will process different subsets of the whole list).
The output spectra are generated in the “*cal*” folder, the folder “*prf96-em-27-fast\preprocess\diagnosis*” folder will contain the log files.
4. Finally, copy the “*cal*” folder to “*prf96-em-27-fast\analysis\mysite\YYMMDD*”. The spectra are named HHMMSSSN.bin (main channel) or HHMMSSSM.bin (CO channel). The time used for the naming is the start time of the measurement as given in the OPUS file. Hence it adopts your choice of time zone when the OPUS file was generated. For European investigators, we recommend use of UT throughout. If you are far from the Greenwich longitude, we suggest using your local time (but perhaps ignore summer time?), specify the time offset wrt UT in the input file. We assume that an observer far away from Greenwich will collect all spectra recorded during a local solar day in one single YYMMDD folder. Using local time will be useful to ensure increasing values of HHMMSS during the whole measurement day.

Since for a large number of files, it does not make sense to do those steps for each file manually, it is recommended to use the python script mentioned in the introduction of this document.

Performing the pre-calculation of x-sections:

This step requires location and date-specific meteorological information: the atmospheric pressure and temperature profiles and the a-priori mixing ratio of each trace gas (H₂O, HDO, CO₂, N₂O, CO, CH₄, O₂, and HF). This information is required for each day extra.

In order to be compatible with the TCCON analysis, we directly process *.map files, which contain the profiles and mixing-ratios. Note that file formats differ between Linux and Windows, so a unix2dos or dos2unix tool has to be used if the *.map file has been generated on a different platform!. *It is important to correctly specify the average ground pressure at the observer altitude for the day under consideration.* (If no in-situ data are available, the pressure can be interpolated from the *.map file by choosing a pressure value of 9999.9. If in-situ values are available, please use the average pressure for the measurement period.)

Edit the input file “prf96-em-27-fast\inp_fast\pcxs10_sod2017_em27sn039_170608.inp”. Save this file under a specific name e.g. “pcx10-SiteID-YYMMDD.inp”. You can give it a new name for every retrieval run, if you want to. Note that this file MUST be located in the “inp_fast” folder, since its location path is hardcoded. Large parts of the file contents are invariable, just specify your location, a best guess for the average ground pressure at the observer altitude for the day under consideration, and which *.map file to use for p,T and gas profile information. The map file should reside in “prf96-em-27-fast\auxil\map-files\mySite”. Note that the map file is referred to several times in the input file (firstly where the p,T information is requested and further down where the a-priori gas profiles are requested. In this lower section, you can decide for each species whether you want to adopt it from the *map file or whether you want to read it from another file, e.g. a *.prf files located in “prf96-em-27-fast\auxil\mysite\YYMMDD\vmr_clim”. Note that the O₂ profile can be taken from the *.map file although this VMR entry is actually missing – this choice simply triggers the code to generate an O₂ profile.)

The example is started by typing in a command prompt “pcxs10 pcxs10.inp”(BH: This file does not exist anymore. I guess that it should be “pcxs10.exe pcxs10_sod2017_em27sn039_170608.inp” and “pcxs10.exe pcxs10_sod2017_em27sn039_170609.inp”). The input file is handled as command line parameter, so several calls for generating x-sections for different days can be processed in parallel, by starting the code with different input file names. Note that you must not give the folder name for the input file, since it is hardcoded.

The output of this program is a file “*abscos.bin” which is saved in “prf96-em-27-fast\wrk_fast\” and called later by the next program.

Starting the retrieval:

After the x-section table has been generated, the inversion of all spectra recorded during a single measurement day can be performed.

Open “invers10.inp” (again, you are free to choose a specific file name) and insert specific inputs. Only the first and last input blocks contain variable user-specific inputs. Important: *Note that SM spectra are not to be listed in the list of spectra, the code will check for the presence of a corresponding SM spectrum for each listed SN spectrum.* For calling the invers code, type “invers.exe invers10.inp”. The input file is handled as command line parameter, so several calls for analyzing spectra of different days

can be processed in parallel (assuming that an x-section table has been generated before for each of these days), by starting the code with different input file names.

While the pre-calculation of x-sections needs to assume a single set of daily meteorological parameters, it also provides derivatives of x-sections with respect to ground pressure and boundary layer temperature. Therefore, intraday ground pressure changes and intraday changes of PBL (Planetary Boundary Level) temperature due to radiative heating of the ground can be taken into account. The assumed PBL thickness is already set in "pcxs.inp": the `n_Tdisturb` variable sets how many levels are affected by a change of temperature (the associated altitude levels can be looked up in `pcxslev.inp`). Typically, the PBL will heat up *and* thicken during the day, the latter effect is neglected, but the temperature change could be scaled to approximate this effect. Note that the relevant temperature change strongly differs the temperature change measured at the observer altitude, the amplitude will be much lower than the temperature change at ground (a temperature measured at several hundred meter altitude would be representative). If no tall tower or simulated data are available, we suggest to omit the correction for variable temperature, the correction due to changing ground pressure is more important anyway.

The shape of the required file reporting the intraday pressure change can be looked up in the Sodankyla example. The file needs to be included in the "*prf96-EM27-fast\analysis\mySite\YYMMDD\pt*" folder and the invariable file name is "*pT_intraday.inp*". The HHMMSS time values used in the file needs to match with the time zone used for naming the calibrated HHMMSS...BIN spectra.

Result files:

The results are found in "*prf96-EM27-fast\out_fast*". The site and day specific output name allows storing a larger number of output files in this folder. The "...invparms.dat" is the ASCII table reporting the trace gas results. All spectral fits (measurement + calculation) are stored in the *.spc files, in `auxil\demo-viewer` you can find a simple demo program explaining the file format and demonstrating visualization of fits.

The post processing includes airmass independent and airmass dependent corrections. These corrections are set according to the values provided in the invariable part of the "invers.inp" input file. These values have been adjusted to bring the COCCON results in close agreement with TCCON. Exceptions are XCO, where we assume that the TCCON calibration has a low bias of about 5%, and the deviating definition of XAIR: here we use the ratio of the spectroscopically derived ground pressure over the measured ground pressure, and we apply a normalization factor to ensure the value is close to unity for an ideal measurement. XAIR is a valuable indicator for instrumental problems, time offsets, incorrect observer coordinates, and incorrect values for pressure at observer altitude, etc. If XAIR is sloping during the day or shows an offset by more than ~1 %, investigating possible causes is advisable.

References:

Frey M., M. K. Sha, F. Hase, M. Kiel, T. Blumenstock, R. Harig, G. Surawicz, N. M. Deutscher, Kei Shiomi, J. Franklin, H. Bösch, J. Chen, M. Grutter, H. Ohyama, Y. Sun, A. Butz, G. Mengistu Tsidu, D. Ene, D. Wunch, C. Z. Song, O. Garcia, M. Ramonet, F. Vogel, and J. Orphal: Building the COllaborative Carbon Column Observing Network (COCCON): Long term stability and ensemble performance of the EM27/SUN Fourier transform spectrometer, submitted to AMT, 2018

Code changes:

180727 – (in pcxs) new choice of air mass values for set of selected line-of-sight angles

180731 – (in invers) corrected bug in raytracing formula (bending angle was too small by factor of ~60), this bug created obvious decrease of XAIR at large SZA

180806 – (in invers) changed meaning of ADCF parameters: prior parameters ADCF1 and ADCF2 were interpreted as prefactors for x^2 and x^4 polynomials, now for x^4 and x^8