**Minutes of IRWG/NORS Workshop on uncertainties**

**NCAR, Jan. 28-Feb. 1, 2013**

**Participants:**

James Hannigan, Martine De Mazière, co-chairs of IRWG

Matthias Palm (UBremen),

Bavo Langerock (BIRA)

Hideaki Nakajima, Yoshihiro Nagahama (NIES)

Nicholas Jones (UWollongong)

Camille Viatte , Stephanie Conway , Pierre Fogal (UToronto)

 Andreas Ostler (KIT, Garmisch-P.)

Philippe Demoulin  (ULiège)

**Goal of the meeting:**

* Discuss common tools
1. for calculation of uncertainties budget
2. for processing of FTIR data, from data acquisition to data submission to NDACC DHF

Presentations made by participants are collected on

ftp://acd.ucar.edu//user/jamesw/IRWG\_NORS\_EW/:

* Presentation by J. Hannigan with outline of meeting and some basic ideas
* Presentation by B. Langerock about fundamentals of error calculations
* Presentation by M. Palm about basic ideas behind optimal estimation and methods for iterative solution, and convergence criterion in SFIT2 & SFIT4
* Presentation by J. Hannigan about structure of SFIT4
* Presentation by J. Hannigan about his batching procedure

**Discussions**

* tools will be built around SFIT4, because this is the S/W that will be maintained in the future
* main factor in error calculations is K- matrix

=> suggestion that SFIT4 calculates all required K-elements (in the retrieved state! ) and the full averaging kernel, and that uncertainties calculation are a separate toolbox

=> suggestion that every uncertainty is a separate output, even if the random and systematic uncertainties are lumped together in the HDF file. This allows inspection of the relative importance of the separate components.

* Next important element is the covariance matrix for each error component (Sb) :
	+ these will be different for each site but we should provide recipes about how to create them.
	+ Should we archive them in a meta-datafile per site? This file will be reasonably stable.
		- Cf. issue of traceability, NORS Data User Guide and NORS Guide about Uncertainties
		- This description can be pointed to in the HDF file.
* Discussion about required error components and distinction between systematic and random errors
	+ E.g., should we include errors due to solar lines? - ask FH (1%-strength, 1e-6-shift) ~3% limb darkening, limb Doppler effect, observer Doppler effect (fwdmdl error – not accounted for)
	+ Do we have enough information for all error components to give reasonable estimates for the systematic and random components?
	+ Circulate a doc to accumulate Sb & physical sources of uncertainties including the distinction between systematic and random (statistical) components.
* Discussion about the computation time for the uncertainties calculations
	+ need to optimize the calculations to avoid redundant I/O, redundant calls to SFIT4, etc. – will be avoided with proposed sfit4 error computations
	+ need for parallelization
* Discussion about construction/optimization of regularization matrix (Sa) and the choice between Tikhonov regularization or full covariance matrix:
	+ Tikhonov is well behaved and useful for low DOFS (tropospheric) species
	+ Tikhonov L1 has the disadvantage that sensitivity is equal 1 for all altitudes, inducing artifacts in the retrieved profile
	+ Importance of good-looking Averaging kernels (Ak): is a criterion to evaluate retrieval method and choice of Sa
		- Warning: units of Ak : relative (to xa) units or VMR/VMR units (the unit used in the HDF archive)
	+ Se (⬄ SNR ) in retrieval can (should ?) be optimized per spectrum, in agreement with the spectrum characteristics. Scaling of Se or Sa are equivalent.
* Distribution, implementation and tests of SFIT4

* + SFIT4 is a research tool and enables many features that are not necessarily used for operational processing.
	+ Required:
		- good documentation about various features, inputs and outputs
		- some guidelines as to which features (not) to use in ‘normal’ solar absorption conditions, e.g., Levenberg-Marquardt fit, SNR fit, etc.
* Discussion about batching tools:
	+ It seems difficult to provide a common tool (everybody has different implementations of his processing chain) especially on front end.
	+ Useful to have:
		- A repository of codes for basic manipulations/ calculations,
			* E.g., SZA calculation
			* Conversion between OPUS, bnr and T15asc formats
			* Conversion from SFIT2 to SFIT4 input
			* Etc
		- A workflow
		- An example batch procedure for newcomers
			* Hideaki offers to translate the existing NCAR batch procedure to IDL and Python

IRWG Error Analysis Workshop Day 4

Data quality control:

Should we provide a spectrum "checker" tool before the retrievals? - can use OPUS or ckop.pro (idl) for quick visual checking

Maybe not :

* because noisy data is still data. you don' t want to throw away useful window. i.e. Don't want to throw away the baby with the water, save but remove from normal process flow.
* it will be site specific. i.e. in Antarctica, even if with noisy spectra you can get information.
* we are experimental physicists, so we SHOULD to actually look at the spectra

But the filtering by "eye" is also inconsistent with global database (ex in Bremen, they have too much spectra per day to do that, operational way).

Quality filter should be based on:

* S/N
* baseline control (zero levels and fluctuation)
* cloud detector
* quadrant sensor detector
* DC signal
* Post retrieval
	+ Fit Rms (& calculated RMS & ratio)
	+ dofs
	+ convergence criteria
	+ vmr artifacts (negative, oscillations), negative columns
	+ look at complete run / outliers
* Post error calculation
	+ Spar
	+ Are errors large if negative values? Is zero within the error bar? Statement of 1sigma error needs to clear on user doc or/and hdf.

Local database:

should it be in HDF? May have problems with file sizes > 2 Gb, then common among all groups

have to think about it more...

Batching:

controlling / steering function, responsibility of each site? i.e. conversion to reference.prf, t15asc...

use or extend runlog/sunrun (spectra list) (get existing format)

basic batch file made available, but added complexity up to each group?

Multiple fits (this is part of the added complexity mentioned above) – current thinking is multiple runs are done one by one … for now

retrieve water for each spectrum? i.e. Each filter?

 Find µw’s from MUSICA, Reunion for each filter.

Should be able to do this simultaneously, i.e. A separate window for water which is then used as the *background* for water in the window that we are retrieving the gas of interest. But it doesn't always seem to work.

Have a separate refmaker.input similar to hbin.input? Either way we would indicate in this file where the vmr, and zpt information for reference.prf and then they would be interpolated by the refmaker program.

-> make output for pre-fit gases easier to input into refmaker?

-> See Bavo's presentation for BIRA's master control file in MATLAB example of more sophisticated multi fit recipe.

\

Other notes:

In an attempt to try to minimize the amount of variables that need changing in sfit4.ctl the following variables are going to move to the t15asc file: SNR, OPD FOV

Need an Sb for each gas
-> comes from WACCM, use avg(mean) as true and use historical data to determine covariance
-> there's a covariance function in python

Additional step @ BIRA: calibration for pressure, temperature and time

-> taken from TCCON people

Have to run hbin every time you change sftit4.ctl microwindows. Should there be an error message?

Notes regarding Mathias' python scripts:

In example you have to change sys.path.append('*directory\_name*') to the folder that contains the scripts

python environment is very rudimentary, ipython is better

to run files in ipython:

execfile('*filename.py*')

programs: show\_all\_spectra.py, show\_spectra\_by\_gas.py, create\_hdf5\_result.py

eg use import functionality:

import tables as h5

h5.file('file name') opens h5 file

Python has an extendable array, Earray: to store something on the fly

*N.B. iteration nr-1 is the last iteration*

copy sfit4 output and setup files to a folder, either names after the daytime of measurement of spectrum file name. Have the setup files and a sub folder

Running IDL from script or commandline:

To pipe commands into idl:
idl <<EOF <enter>

*command <enter>*

EOF <enter>

or idl -e “*command*”

Running IDL from matlab example for idlhdfcr8.sav:

sprintf('idl -e "%s,''%s.meta'',''%s.data'',''%s'',''%s''"',idlfile,[precisefolder mol],[precisefolder mol],config.hdf\_table, precisefolder)

Routines

Refmaker – read different profile formats & make a reference.prf file

* these different input readers are user supplied
* used in / out of batch

runlogmaker – makes an input list for new batch ggg file + extensions

* input routines are user supplied
* offline operation

Spectrum parser – reads OPUS (or ??) & creates t15 spectrum

* calcs SNR,
* puts SZA in t15