MT_CKD_H2O User Guide

1. Introduction

The MT_CKD_H2O package consists of a netCDF file containing the MT_CKD_H2O foreign and self water vapor continuum coefficients and the self continuum temperature coefficients, along with code to obtain water vapor absorption coefficients at the desired pressure, temperature and wavenumber range. The code is a FORTRAN subroutine called mt_ckd_h2o_absco. A simple driver (drive_mt_ckd_h2o) is provided to demonstrate how to call the subroutine. Figure 1 illustrates the principal components of the MT_CKD_H2O package (which can be obtained from either https://github.com/AER-RC/MT_CKD_H2O or https://hitran.org/mtckd/) and how it fits within the AER RT model framework. A more detailed diagram of the MT_CKD_H2O components can be found in the Appendix (Figure A1).

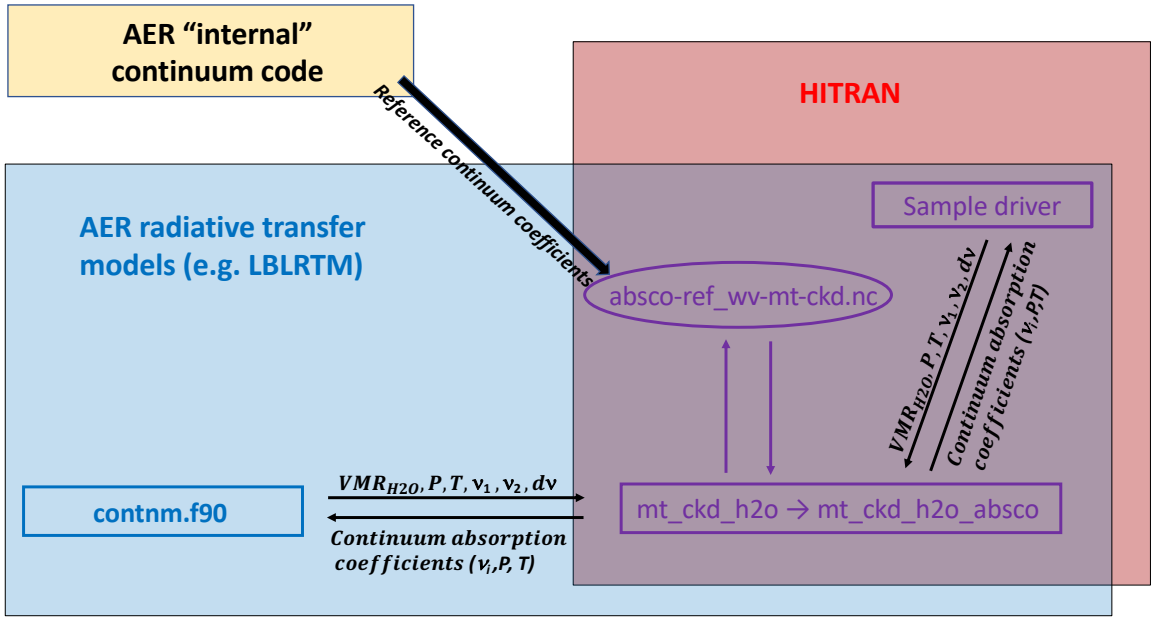

Figure 1: The purple section of this schematic diagram shows the MT_CKD water vapor continuum model (MT_CKD_H2O), which is common to both HITRAN (red) and what is distributed with AER radiative transfer models like LBLRTM (blue). For both HITRAN and the AER models, MT_CKD_H2O takes as input the water vapor abundance, pressure, temperature, and wavenumber specifications, and returns the water vapor continuum absorption coefficients (either with or without the radiation term). In its calculation, the mt_ckd_h2o_absco code accesses the continuum data stored in absco-ref_wv-mt-ckd.nc, which is created by the AER internal water vapor continuum code (yellow box).

2. Compiling and running

To compile the code:

```
cd build
make -f make_mt_ckd_h2o linuxGNUdbl
```

This command calls the gfortran compiler. This is the only compiler that has been tested with this code so far. The directory structure is shown in the appendix.

The example case can be found in the run_example directory. Run the example by typing:

```
../ mt_ckd_h2o_vx.x_linux_gnu_dbl
```

where x.x indicates the version number of code and is defined in make_mt_ckd_h2o.

3. Components

a. mt_ckd_h2o_module: FORTRAN90

This module contains the mt_ckd_h2o_absco subroutine, which reads in the absco-ref_wv-mt-ckd netCDF file and calculates the water vapor continuum absorption coefficients for the requested pressure, temperature, and spectral range. If requested, the coefficients will be multiplied by the radiation term. A simple driver (see section d) illustrates how to call mt_ckd_h2o_absco. The output is provided in the sh2o and fh2o arrays, which must be declared in the calling program. The module also contains three short private routines adapted from LBLRTM: myradfn, myxint and pre_xint. Also included in the package is the read_data_module, which contains the get_data routine used to read the netCDF file.

Definition: subroutine mt_ckd_h2o (p_atm,t_atm ,h2o_vmr,wv1,wv2,dwv,self_absco, for_absco, radflag)

Inputs:

p_atm - atmospheric pressure for the calculation of continuum coefficients (mbar)
t_atm - atmospheric temperature for the calculation of continuum coefficients (K)
h2o_vmr - fraction of the total number of molecules that are water vapor
wv1 - initial wavenumber for which continuum coefficients will be computed (cm-1)
wv2 - final wavenumber for which continuum coefficients will be computed (cm-1)
dwv - wavenumber spacing for the continuum coefficient calculation (cm-1)
radflag - (optional) if true, multiply by radiation term (default); if false, do not

Output:

self_absco –

- for radflag = 0, computed self continuum coefficients (cm2/(molecule cm-1))
- for radflag = 1, computed self continuum absorption coefficients (cm2/molecule)*

for_absco -

- for radflag = 0, computed foreign continuum coefficients (cm2/(molecule cm-1))
- for radflag = 1, computed foreign continuum absorption coefficients (cm2/molecule)*

If radflag is set to 1, the calculated continuum coefficient ($\tilde{C}_x$) is multiplied by the radiation term (R) (Equations 1 and 2); it then becomes an absorption coefficient (C), which when multiplied by the molecular amount provides the optical depth.

$$C(\nu, T, \rho_x) = \tilde{C}_x(\nu, T, \rho_x)\, R(\nu, T) \qquad (1)$$

The radiation term *R* is given by

$$R(\nu, T) = \nu \tanh\left(\frac{hc\nu}{2kT}\right) \qquad (2)$$

where *h* is Planck's constant, *c* is the speed of light, and *k* is Boltzmann's constant.

b. absco-ref_wv-mt-ckd.nc

The absco-ref_wv-mt-ckd.nc netCDF files is generated by the internal (not publicly available) continuum code, which outputs the water continuum coefficients in a modern and portable format for use by the mt_ck_h2o routine.

Below is a listing of the dimensions and variables in the absco-ref_wv-mt-ckd.nc file:

```
dimensions:
        wavenumbers = 2003 ;
variables:
        double wavenumbers(wavenumbers) ;
                wavenumbers:units = "cm-1" ;
                wavenumbers:description = "Spectral domain of water vapor continuum" ;
        double self_absco_ref(wavenumbers) ;
                self_absco_ref:units = "cm**2/molecule cm-1" ;
                self_absco_ref:description = "Water vapor self continuum coefficients" ;
        double for_absco_ref(wavenumbers) ;
                for_absco_ref:units = "cm**2/molecule cm-1" ;
                for_absco_ref:description = "Water vapor foreign continuum coefficients" ;
        double self_texp(wavenumbers) ;
                self_texp:units = "dimensionless" ;
```

```
        self_texp:description = "Temperature exponent for water vapor self continuum" ;
    double ref_press ;
        ref_press:units = "mbar" ;
        ref_press:description = "Reference pressure" ;
    double ref_temp ;
        ref_temp:units = "K" ;
        ref_temp:description = "Reference temperature" ;
```

c. drive_mt_ckd_h20.f90: FORTRAN

drive_mt_ckd_h2o is a simple FORTRAN driver for mt_ckd_h2o. It reads in the required inputs for absco-ref_mt-ckd-h2o from a NAMELIST file and outputs the spectral grid and the self and foreign coefficients in a netCDF file.

Below is a sample namelist file:

```
&mt_ckd_input
    p_atm=1013.
    t_atm=300.
    h2o_frac= 0.00990098
    wv1=497.
    wv2=603.
    dwv=1.
/
```

The output is provided in mt_ckd_h2o_output.nc, whose contents are listed below:

```
netcdf mt_ckd_h2o_output {
dimensions:
    nwvn = 107 ;
variables:
    double wavenumbers(nwvn) ;
        wavenumbers:units = "cm-1" ;
        wavenumbers:description = "Wavenumbers\f" ;
    double self_absorption(nwvn) ;
        self_absorption:units = "cm**2/molecule)" ;
        self_absorption:description = "Water vapor continuum self absorption coefficients"
;
    double frgn_absorption(nwvn) ;
        frgn_absorption:units = "cm**2/molecule)" ;
        frgn_absorption:description = "Water vapor foreign continuum absorption
coefficients";
```

Appendix

This appendix contains more detailed information on the components of the MT_CKD_H2O package, as shown in Figure A1.

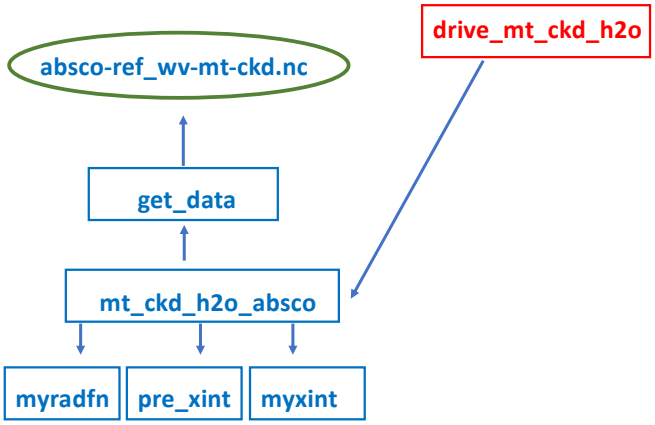

Figure A1.: Flow chart of the MT_CKD_H2O model

Directory structure

The listing below shows the directory structure and files after compilation:

LICENSE.md mt_ckd_h2o_v4.1_linux_gnu_dbl* README.md

build:
addlibs.inc make_mt_ckd_h2o* mt_ckd_h2o_v4.1_linux_gnu_dbl.obj/ read_file.mod
makefile.common* mt_ckd_h2o.mod phys_consts.mod

data:
absco-ref_wv-mt-ckd.nc

docs:
mt_ckd_h2o_user_guide.pdf

run_example:
absco-ref_wv-mt-ckd.nc@ mt_ckd.config mt_ckd_h2o_output.nc

src:
drive_mt_ckd_h2o.f90 mt_ckd_h2o_module.f90 phys_consts.f90 read_module.f90

Secondary model components

a. **myradfn:** calculates the radiation term for a specific temperature and an array of frequencies. Note that the radfn function in LBLRTM calculates this term for a single frequency

Definition: integer function myradfn (vi,xkt,nvi,rad)

Inputs:

v1: array of frequencies (cm-1)
xkt: ratio of layer temperature to radcn2 (cm-1)
nvi: number of frequencies in v1

Outputs:

rad: array of radiation terms (cm-1)

b. **pre_xint:** finds the starting and ending indices to use in the interpolation to the requested grid

Definition: subroutine pre_xint(v1ss,v2ss,v1abs,dvabs,nptabs,ist,lst)

Inputs:

v1ss: first frequency of coefficient grid (cm-1)
v2ss: last frequency of coefficient grid (cm-1)
v1abs: first frequency of output grid (cm-1)
v2abs: last frequency of output grid (cm-1)
dvabs: resolution of output grid (cm-1)
nptabs: number of points in output grid

Outputs:

ist: index of first point to interpolate to in output grid
lst: index of last point to interpolate to in output grid

c. **myxint:** interpolates to output grid and adds to existing absorption/optical depth

Definition: subroutine myxint (v1a,v2a,dva,a,afact,vft,dvr3,r3,n1r3,n2r3)

Inputs:

v1a: first frequency of coefficient grid (cm-1)
v2a: last frequency of coefficient grid (cm-1)

dva: resolution of coefficient grid (cm-1)
a: array of calculated absorption (cm2/molec) or optical depth
afact: 1.0
vft: starting frequency of output grid (cm-1)
dvr3: resolution of output grid (cm-1)
n1r3: index of first point to interpolate to in output grid
n2r3: index of last point to interpolate to in output grid

Input/Output

r3: array of interpolated and summed absorption (molec/cm2) or optical depth

d. read_module: FORTRAN90

This module defines a data type (data2read) to store the contents of absco-ref_wv-mt-ckd.nc, reads in the contents of this file with the get_data routine and returns them to the calling routine in the defined datatype.

Input: absco-ref_wv-mt-ckd.nc

Output: dat: user defined datatype data2read containing all the data in absco-ref_wv-mt-ckd.nc:

```
type data2read
   real(kind=8), allocatable, dimension(:) :: wavenumber
   real(kind=8), allocatable, dimension(:) :: for_absco_ref
   real(kind=8), allocatable, dimension(:) :: self_absco_ref
   real(kind=8), allocatable, dimension(:) :: self_texp
```

.