RADIS Documentation

Release 0.13.1

Erwan Pannier, Dirk van den Bekerom, Nicolas Minesi, et al. (https

Aug 28, 2022

CONTENTS

RADIS is a fast *line-by-line code* for high resolution infrared molecular spectra (emission / absorption, equilibrium / nonequilibrium).

It also includes *post-processing tools* to compare experimental spectra and spectra calculated with RADIS, or with other spectral codes.

CHAPTER

ONE

GETTING STARTED

1.1 Install

Assuming you have Python installed with the Anaconda distribution just use:

```
pip install radis
```

That's it! You can now run your first example below. If you encounter any problem or if you need to upgrade, please refer to the *detailed installation procedure*. If you don't have a Python environment, try *RADIS Online* first !

1.2 Quick Start

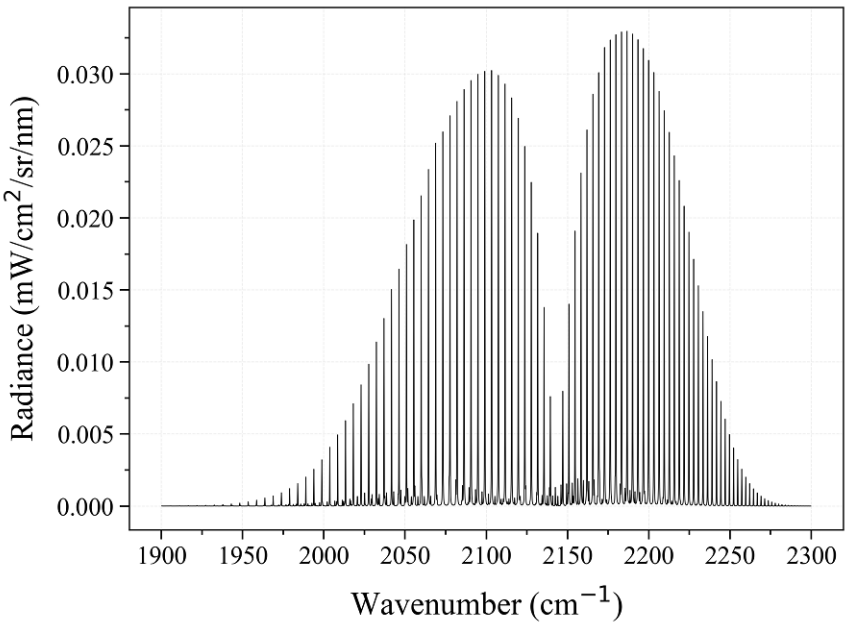
Calculate a CO equilibrium spectrum from the [HITRAN-2020] database, using the `calc_spectrum()` function. Lines are downloaded automatically using Astroquery (based on [HAPI]). Output is a *Spectrum object*:

```
from radis import calc_spectrum
s = calc_spectrum(1900, 2300,         # cm-1
                  molecule='CO',
                  isotope='1,2,3',
                  pressure=1.01325,   # bar
                  Tgas=700,           # K
                  mole_fraction=0.1,
                  path_length=1,      # cm
                  databank='hitran',  # or 'hitemp', 'geisa', 'exomol'
                  )
s.apply_slit(0.5, 'nm')       # simulate an experimental slit
s.plot('radiance')
```

Calculate a CO *nonequilibrium* spectrum from the HITRAN database, with arbitrary `units` (on your first call, this will compute and store the CO(X) rovibrational energies):

```
from astropy import units as u
s2 = calc_spectrum(1900 / u.cm, 2300 / u.cm,
                  molecule='CO',
                  isotope='1,2,3',
                  pressure=1.01325 * u.bar,
                  Tvib=700 * u.K,
                  Trot=300 * u.K,
                  mole_fraction=0.1,
```

(continues on next page)



(continued from previous page)

```
                  path_length=1 * u.cm,
                  databank='hitran',  # or 'hitemp', 'geisa', 'exomol'
                  )
s2.apply_slit(0.5, 'nm')
s2.plot('radiance', nfig='same')    # compare with previous
```

Experimental spectra can be loaded using the `experimental_spectrum()` function and compared with the `plot_diff()` function. For instance:

```
from numpy import loadtxt
from radis import experimental_spectrum, plot_diff
w, I = loadtxt('my_file.txt').T    # assuming 2 columns
sexp = experimental_spectrum(w, I, Iunit='mW/cm2/sr/nm')
plot_diff(sexp, s)    # comparing with a spectrum 's' calculated previously
```

Refer to the *Spectrum object guide* for more post-processing functions (*rescale* , *crop*, *remove baselines*, *store*, *combine along the line-of-sight*, *identify each line*, *manipulate multiple spectra at once*, etc.)

1.3 More examples

- *Load an experimental spectrum*
- *Line Survey*
- *Blackbody radiation*
- *Remove a baseline*
- *GPU Accelerated Spectra*
- *Calculate non-LTE spectra of carbon-monoxide*
- *Use different plot themes*

- *Calculate a large spectrum by part*
- *Compare CO spectrum from the GEISA and HITRAN database*
- *Calculate a spectrum from HITEMP*
- *Calculate a spectrum from ExoMol*
- *Real-time GPU Accelerated Spectra (Interactive)*
- *See populations of computed levels*
- *Spectrum Database*
- *Compare CO xsections from the ExoMol and HITEMP database*
- *Post-process using Specutils*
- *Multi-temperature Fit*
- *1 temperature fit*

The Quick Start examples automatically downloaded the line databases from [HITRAN-2020], which is valid for temperatures below 700 K. For *high temperature* cases, you may need to use *other line databases* such as [HITEMP-2010] (typically T < 2000 K) or [CDSD-4000] (T < 5000 K). These databases must be described in a `~/radis.json` *Configuration file*.

Note: starting from radis==0.9.30 you can also download HITEMP and ExoMol directly. Just use `databank='hitemp'` or `databank='exomol'` in the initial example. This will automatically download, unzip and setup the database files in a ~/.radisdb folder.

More complex *examples* will require to use the `SpectrumFactory` class, which is the core of RADIS line-by-line calculations.

Refer to the *Examples* and Example Gallery sections for more examples, and to the *User Documentation* for more details on the code. You can also ask questions on the Q&A Forum or on the community chats on Gitter or Slack

1.4 In the browser (no installation needed!)

Alternatively, you can also run RADIS directly in the browser with the RADIS Interactive Examples project. For instance, run the Quick Start example on the link below:

Or use *RADIS-lab* to start a full online environment for advanced spectrum processing and comparison with experimental data :

1.5 Cite

RADIS is built on the shoulders of many state-of-the-art packages and databases. If using RADIS to compute spectra, make sure you cite all of them, for proper reproducibility and acknowledgement of the work ! See *How to cite?*

CHAPTER

TWO

CONTENT

- *The Line-by-line (LBL) module*

  This is the core of RADIS: it calculates the spectral densities for a homogeneous slab of gas, and returns a `Spectrum` object. Calculations are performed within the `SpectrumFactory` class. `calc_spectrum()` is a high-level wrapper to `SpectrumFactory` for most simple cases.

- *Line-of-sight (LOS) module*

  This module takes several `Spectrum` objects as input and combines then along the line-of-sight (`SerialSlabs()`) or at the same spatial position (`MergeSlabs()`), to reproduce line-of-sight experiments. The module allows combination of Spectra such as:

```
s_line_of_sight = (s_plasma_CO2 // s_plasma_CO) > (s_room_absorption)
```

- *The Spectrum object guide*

  This module contains the `Spectrum` object itself, with several methods that can be applied after the Spectrum was calculated: rescale, apply instrumental slit function, store or retrieve from a Spectrum database, plot or compare with another Spectrum object.

- modindex

2.1 Features

2.1.1 Description

Written as a general purpose radiative solver, the code is built around the [HITRAN-2020], [HITEMP-2010] and [CDSD-4000] databases for molecules in their electronic ground state. Energy levels are read from tabulated databases or calculated from Dunham developments. Boltzmann, Treanor, and state specific vibrational distributions can be generated. Thus far, CO_2, CO are featured for non-equilibrium calculations (`MOLECULES_LIST_NONEQUILIBRIUM`), and all species present in the HITRAN database are featured for equilibrium calculations (`MOLECULES_LIST_EQUILIBRIUM`).

To fit experimental spectra, RADIS includes a `LineSurvey` tool, an interface with a look-up `SpecDatabase` to improve fitting convergence times, and a *multi-slab module* with a radiative transfer equation solver to reproduce line-of-sight experiments. *Validation cases* against existing spectral codes and experimental results from various plasma sources are included [RADIS-2018].

2.1.2 Features

RADIS is both an infrared *line-by-line code* and a *post-processing library*. It includes:

- Absorption and emission spectra of all [HITRAN-2020] and [ExoMol-2020] species under equilibrium calculations (`MOLECULES_LIST_EQUILIBRIUM`)
- Absorption and emission spectra of CO2 and CO for non-LTE calculations (see `MOLECULES_LIST_NONEQUILIBRIUM`)
- Different Line Databases: support of [HITRAN-2020], [HITEMP-2010], [CDSD-4000], [ExoMol-2020], [GEISA-2020] line databases (see `KNOWN_DBFORMAT`)
- Calculation of *Rovibrational Energies of molecules*.
- Calculation of equilibrium and nonequilibrium *Partition Functions*.
- Spatially heterogeneous spectra (see *see line-of-sight*)
- Post-processing tools to load and *compare with experimental spectra* (see *the Spectrum object*)
- A *Line Survey* tool to identify which lines correspond to a spectral feature.

RADIS does *not* include, so far:

- Line-mixing effects and speed-dependant lineshapes. [HAPI] is a Python alternative that does it.
- Collisional-induced absorption (CIA) or emission.
- Electronic states other than electronic ground states
- Hamiltonian calculations (a private module for CO2 is available on request)
- Raman spectra (contribute in #43)

RADIS also features:

- *High Performances*: spectra are calculated up to several orders of magnitude faster than equivalent line-by-line codes.
- In-the-browser calculations (no install needed) : see *RADIS Online*.
- Automatic download of the latest HITRAN and HITEMP databases with `calc_spectrum()`
- Automatic testing and continuous integration tools for a reliable *Open-source Development*.

Remarks and request for features can be done on GitHub , on the Q&A forum or on the Gitter community chat:

Use Cases

Use RADIS to:

- Quickly compare different line databases: Various line database formats are supported by RADIS, and can be easily switched to be compared. See the list of supported line databases formats: `KNOWN_DBFORMAT` and refer to the *Configuration file* on how to use them.

  See the comparison of two CO2 spectra calculated with [HITEMP-2010] and [CDSD-4000] below:

- Use the RADIS post-processing methods with the calculation results of another spectral code. For instance, pySpecair, the Python interface to SPECAIR, uses the RADIS `Spectrum` object for post-processing (see *How to generate a Spectrum?*)

Refer to the *Examples* section for more examples, or to the RADIS Interactive Examples project.

See the *Architecture* section for an overview of the RADIS calculation steps.

2.1.3 Line Databases

List of supported line databases formats: `KNOWN_DBFORMAT` :

- [HITRAN-2016]
- [HITRAN-2020]
- [HITEMP-2010]
- [CDSD-4000]
- [ExoMol-2020]

For download and configuration of line databases, see the *Line Databases section*

2.1.4 Interfaces

RADIS includes parsers and interfaces to read and return data in different formats:

Thermodynamic codes

Cantera

RADIS can compute gas mixture compositions under chemical equilibrium using CANTERA (in particular the [CANTERA] `equilibrate()` function). Refer to `get_eq_mole_fraction()` for more information.

2.1.5 New features

RADIS is open-source, so everyone can contribute to the code development. Read the *Developer Guide* to get started.

You can also suggest or vote for new features below:

2.2 Line-by-line module

This is the core of RADIS: it calculates the spectral densities for a homogeneous slab of gas, and returns a `Spectrum` object.

Calculations are performed within the `SpectrumFactory` class. `calc_spectrum()` is a high-level wrapper to `SpectrumFactory` for most simple cases.

- *Cite all references used*
- *Calculate a full range spectrum*

For any other question you can use the Q&A forum, the GitHub issues or the community chats on Gitter or Slack .

2.2.1 Line databases

List of supported line databases formats: `KNOWN_DBFORMAT`

HITRAN

RADIS can automatically fetch HITRAN lines using the Astroquery module. This is done by specifying `databank=='hitran'` in `calc_spectrum()` or by using the `fetch_databank()` method in `SpectrumFactory`. Refer to `fetch_astroquery()` for more information.

You can also download the HITRAN databases files locally:

- HITRAN can be downloaded from https://hitran.org/lbl/. Expect ~80 Mb for CO2, or 50 Mb for H2O. Cite with [HITRAN-2020].

Note: RADIS has parsers to read line databases in Pandas dataframes. This can be useful if you want to edit the database. see `hit2df()`

There are also functions to get HITRAN molecule ids, and vice-versa: `get_molecule()`, `get_molecule_identifier()`

HITEMP

RADIS can read files from the HITEMP database.

- HITEMP-2010 files can be downloaded from https://hitran.org/hitemp/. Expect ~3 Gb for CO2 or ~10 Gb for H2O. Cite with [HITEMP-2010]

The `~/radis.json` is then used to properly handle the line databases on the User environment. See the *Configuration file* section, as well as the `radis.misc.config` module and the `getDatabankList()` function for more information.

starting from radis==0.9.28 you can also download HITEMP directly. Example

```
from radis import calc_spectrum
calc_spectrum(
    wavenum_min=2500 / u.cm,
    wavenum_max=4500 / u.cm,
    molecule="OH",
    Tgas=600,
    databank="hitemp",  # test by fetching directly
    verbose=False,
)
```

- Some HITEMP line databases are pre-configured in the *RADIS-lab* online environment. No install needed !
- If you just want to parse the HITEMP files, use `fetch_hitemp()`

```
from radis.io.hitemp import fetch_hitemp
fetch_hitemp("NO")
```

CDSD-4000

RADIS can read files from the CDSD-4000 database, however files have to be downloaded manually.

- CDSD-4000 files can be downloaded from ftp://ftp.iao.ru/pub/. Expect ~50 Gb for all CO2. Cite with [CDSD-4000].
- Tabulated partition functions are availabe in the `partition_functions.txt` file on the [CDSD-4000] FTP : ftp://ftp.iao.ru/pub/CDSD-4000/ . They can be loaded and interpolated with `PartFuncCO2_CDSDtab`. This can be done automatically providing `parfuncfmt: cdsd` and `parfunc = PATH/TO/cdsd_partition_functions.txt` is given in the `~/radis.json` configuration file (see the *Configuration file*).

The `~/radis.json` is used to properly handle the line databases on the User environment. See the *Configuration file* section, as well as the `radis.misc.config` module and the `getDatabankList()` function for more information.

Note: See `cdsd2df()` for the conversion to a Pandas DataFrame.

2.2.2 Calculating spectra

Calculate one molecule spectrum

In the following example, we calculate a CO spectrum at equilibrium from the latest HITRAN database, and plot the transmittance:

```
s = calc_spectrum(
                wavenum_min=1900,
                wavenum_max=2300,
                Tgas=700,
                path_length=0.1,
                molecule='CO',
                mole_fraction=0.5,
                isotope=1,
    wstep=0.01,
    databank='hitran'   # or 'hitemp'
                )
s.plot('transmittance_noslit')
```

Calculate multiple molecules spectrum

RADIS can also calculate the spectra of multiple molecules. In the following example, we add the contribution of CO2 and plot the transmittance:

```
s = calc_spectrum(
        wavenum_min=1900,
        wavenum_max=2300,
        Tgas=700,
        path_length=0.1,
        mole_fraction={'CO2':0.5, 'CO':0.5},
        wstep=0.01,
  isotope=1,
```

(continues on next page)

(continued from previous page)

```
        )
s.plot('transmittance_noslit')
```

Note that you can indicate the considered molecules either as a list in the `molecule` parameter, or in `isotope` or `mole_fraction`. The following commands give the same result:

```
# Give molecule:
s = calc_spectrum(
        wavelength_min=4165,
        wavelength_max=5000,
        Tgas=1000,
        path_length=0.1,
        molecule=["CO2", "CO"],
        mole_fraction=1,
        isotope={"CO2": "1,2", "CO": "1,2,3"},
        wstep=0.01,
        verbose=verbose,
  )


# Give isotope only
s = calc_spectrum(
    wavelength_min=4165,
    wavelength_max=5000,
    Tgas=1000,
    path_length=0.1,
    isotope={"CO2": "1,2", "CO": "1,2,3"},
    verbose=verbose,
)

# Give mole fractions only
s = calc_spectrum(
    wavelength_min=4165,
    wavelength_max=5000,
    Tgas=1000,
    path_length=0.1,
    mole_fraction={"CO2": 0.2, "CO": 0.8},
    isotope="1,2",
    verbose=verbose,
)
```

Be careful to be consistent and not to give partial or contradictory inputs.

```
# Contradictory input:
s = calc_spectrum(
         wavelength_min=4165,
         wavelength_max=5000,
         Tgas=1000,
         path_length=0.1,
         molecule=["CO2"],  # contradictory
         mole_fraction=1,
         isotope={"CO2": "1,2", "CO": "1,2,3"},
```

(continues on next page)

(continued from previous page)

```
        verbose=verbose,
    )

# Partial input:
s = calc_spectrum(
        wavelength_min=4165,
        wavelength_max=5000,
        Tgas=1000,
        path_length=0.1,
        molecule=["CO2", "CO"],  # contradictory
        mole_fraction=1,
        isotope={"CO2": "1,2"},  # unclear for CO
        verbose=verbose,
    )
```

Equilibrium Conditions

By default RADIS calculates spectra at thermal equilibrium (one temperature).

The `calc_spectrum()` function requires a given mole fraction, which may be different from chemical equilibrium.

You can also compute the chemical equilibrium composition in other codes like [CANTERA], and feed the output to RADIS `calc_spectrum()`. The `get_eq_mole_fraction()` function provides an interace to [CANTERA] directly from RADIS

```
from radis import calc_spectrum, get_eq_mole_fraction

# calculate gas composition of a 50% CO2, 50% H2O mixture at 1600 K:
gas = get_eq_mole_fraction('CO2:0.5, H2O:0.5', 1600, # K
                                       1e5  # Pa
                           )
# calculate the contribution of H2O to the spectrum:
calc_spectrum(...,
              mole_fraction=gas['H2O']
              )
```

Nonequilibrium Calculations

Non-LTE calculations (multiple temperatures) require to know the vibrational and rotational energies of each level in order to calculate the nonequilibrium populations.

You can either let RADIS calculate rovibrational energies with its built-in *spectroscopic constants*, or supply an energy level database. In the latter case, you need to edit the *Configuration file* .

Fit a Spectrum

Calculating spectrum using GPU

RADIS also supports CUDA-native parallel computation, specifically for lineshape calculation and broadening. To use these GPU-accelerated methods to compute the spectra, use either `calc_spectrum()` function with parameter `mode` set to `gpu`, or `eq_spectrum_gpu()`. In order to use these methods, ensure that your system has an Nvidia GPU with compute capability of atleast 3.0 and CUDA Toolkit 8.0 or above. Refer to GPU Spectrum Calculation on RADIS to see how to setup your system to run GPU accelerated spectrum calculation methods, examples and performance tests.

Currently, GPU-powered spectra calculations are supported only at thermal equilibrium and therefore, the method to calculate the spectra has been named `eq_spectrum_gpu()`. In order to use this method to calculate the spectra, follow the same steps as in the case of a normal equilibrium spectra, and if using `calc_spectrum()` function set the parameter `mode` to `gpu`, or use `eq_spectrum_gpu()`

One could compute the spectra with the assistance of GPU using the following code as well

```
s = calc_spectrum(
            wavenum_min=1900,
            wavenum_max=2300,
            Tgas=700,
            path_length=0.1,
            mole_fraction=0.01,
            isotope=1,
            mode='gpu'
            )
```

Refer to GPU Spectrum Calculation on RADIS for more details.

2.2.3 Under the hood

Flow Chart

RADIS can calculate populations of emitting/absorbing levels by scaling tabulated data (equilibrium) or from the rovibrational energies (nonequilibrium), get the emission and absorption coefficients from *Line Databases*, calculate the line broadening using various strategies to improve *Performances*, and produce a *Spectrum object*. These steps can be summarized in the flow chart below:

The detail of the functions that perform each step of the RADIS calculation flow chart is given in *Architecture*.

The Spectrum Factory

Most RADIS calculations can be done using the `calc_spectrum()` function. Advanced examples require to use the `SpectrumFactory` class, which is the core of RADIS line-by-line calculations. `calc_spectrum()` is a wrapper to `SpectrumFactory` for the simple cases.

The `SpectrumFactory` allows you to :

- calculate multiple spectra (batch processing) with a same line database
- edit the line database manually
- have access to intermediary calculation variables
- connect to a database of precomputed spectra on your computer

To use the `SpectrumFactory`, first load your own line database with `load_databank()`, and then calculate several spectra in batch using `eq_spectrum()` and `non_eq_spectrum()`, and `units`

```
import astropy.units as u
from radis import SpectrumFactory
sf = SpectrumFactory(wavelength_min=4165 * u.nm,
                     wavelength_max=4200 * u.nm,
                     path_length=0.1 * u.m,
                     pressure=20 * u.mbar,
                     molecule='CO2',
                     wstep = 0.01,
                     isotope='1,2',
                     cutoff=1e-25,               # cm/molecule
                     broadening_max_width=10,    # cm-1
                     )
sf.load_databank('HITRAN-CO2-TEST', load_columns='noneq')        # this database must be␣
↪defined in ~/radis.json
s1 = sf.eq_spectrum(Tgas=300 * u.K)
s2 = sf.eq_spectrum(Tgas=2000 * u.K)
s3 = sf.non_eq_spectrum(Tvib=2000 * u.K, Trot=300 * u.K)
```

Note that for non-LTE calculations, specific columns must be loaded. This is done by using the `load_columns='noneq'` parameter. See `load_databank()` for more information.

Configuration file

The `~/radis.json` configuration file is used to initialize your `radis.config`. It will :

- store the list and attributes of the Line databases available on your computer.
- change global user preferences, such as plotting styles and libraries, or warnings thresholds, or default algorithms.

The list of all available parameters is given in the default_radis.json file. Any key added to your `~/radis.json` will override the value of `default_radis.json`.

Note: You can also update config parameters at runtime by setting:

```
import radis
radis.config["SOME_KEY"] = "SOME_VALUE"
```

Although it is recommended to simply edit your `~/radis.json` file.

Databases downloaded from 'hitran', 'hitemp' and 'exomol' with `calc_spectrum()` or `fetch_databank()` are automatically registered in the `~/radis.json` configuration file. The default download path is `~/.radisdb`. You can change this at runtime by setting the `radis.config["DEFAULT_DOWNLOAD_PATH"]` key, or (recommended) by adding a `DEFAULT_DOWNLOAD_PATH` key in your `~/radis.json` configuration file.

The configuration file will help to:

- handle local line databases that contains multiple files
- use custom tabulated partition functions for equilibrium calculations
- use custom, precomputed energy levels for nonequilibrium calculations

---

Note: it is also possible to work with local line databases without a configuration file, either by giving a file to the `databank=...` parameter of `calc_spectrum()` , or by giving to `load_databank()` the line database path, format, and partition function format directly.

However, this is not recommended and should only be used if for some reason you cannot create a configuration file.

A `~/radis.json` is user-dependant, and machine-dependant. It contains a list of database, each of which is specific to a given molecule. It typically looks like:

str: Typical expected format of a ~/radis.json entry:

```
{
  "database": {                                    # database key: all databanks␣
↪information are stored in this key
      "MY-HITEMP-CO2": {                           # your databank name: use this in␣
↪calc_spectrum()
                                                   # or SpectrumFactory.load_databank()
        "path": [                                  # no "", multipath allowed
            "D:\\Databases\\HITEMP-CO2\\hitemp_07",
            "D:\\Databases\\HITEMP-CO2\\hitemp_08",
            "D:\\Databases\\HITEMP-CO2\\hitemp_09"
        ],
        "format": "hitran",                        # 'hitran' (HITRAN/HITEMP), 'cdsd-
↪hitemp', 'cdsd-4000'
                                                   # databank text file format. More␣
↪info in
                                                   # SpectrumFactory.load_databank␣
↪function.
        "parfuncfmt": "hapi"                       # calculate partition functions
      }
  }
}
```

Following is an example where the path variable uses a wildcard * to find all the files that have `hitemp_*` in their names:

```
{
  "database": {                                    # database key: all databanks␣
↪information are stored in this key
      "MY-HITEMP-CO2": {                           # your databank name: use this in␣
↪calc_spectrum()
                                                   # or SpectrumFactory.load_databank()
        "path": "D:\\Databases\\HITEMP-CO2\\hitemp_*",   # To load all hitemp files␣
```

(continues on next page)

(continued from previous page)

```
↪directly
        "format": "hitran",                          # 'hitran' (HITRAN/HITEMP), 'cdsd-
↪hitemp', 'cdsd-4000'
                                                     # databank text file format. More␣
↪info in
                                                     # SpectrumFactory.load_databank␣
↪function.
        "parfuncfmt": "hapi"                         # calculate partition functions
      }
  }
}
```

In the former example, for equilibrium calculations, RADIS uses [HAPI] to retrieve partition functions tabulated with TIPS-2017. It is also possible to use your own partition functions, for instance:

```
{
  "database": {                                      # database key: all databanks␣
↪information are stored in this key
      "MY-HITEMP-CO2": {                             # your databank name: use this in␣
↪calc_spectrum()
                                                     # or SpectrumFactory.load_
↪databank()
        "path": [                                    # no "", multipath allowed
            "D:\\Databases\\HITEMP-CO2\\hitemp_07",
            "D:\\Databases\\HITEMP-CO2\\hitemp_08",
            "D:\\Databases\\HITEMP-CO2\\hitemp_09"
        ],
        "format": "hitran",                          # 'hitran' (HITRAN/HITEMP), 'cdsd-
↪hitemp', 'cdsd-4000'
                                                     # databank text file format. More␣
↪info in
                                                     # SpectrumFactory.load_databank␣
↪function.
        "parfuncfmt": "cdsd",                        # 'cdsd', 'hapi', etc.
                                                     # format to read tabulated␣
↪partition function
                                                     # file. If `hapi`, then HAPI␣
↪(HITRAN Python
                                                     # interface) is used to retrieve␣
↪them (valid if
                                                     # your databank is HITRAN data).␣
↪HAPI is embedded
                                                     # into RADIS. Check the version.␣
↪If not specified then 'hapi'
                                                     # is used as default
        "parfunc": "PATH/TO/cdsd_partition_functions.txt"
                                                     # path to tabulated partition␣
↪function to use.
                                                     # If `parfuncfmt` is `hapi` then␣
↪`parfunc`
                                                     # should be the link to the hapi.
↪py file. If
```

(continues on next page)

(continued from previous page)

```
                                                    # not given, then the hapi.py␣
↪embedded in RADIS
                                                    # is used (check version)
      }
  }
}
```

By default, for nonequilibrium calculations, RADIS built-in *spectroscopic constants* are used to calculate the energy levels for CO2. It is also possible to use your own Energy level database. For instance:

```
{
  "database": {                                     # database key: all databanks␣
↪information are stored in this key
      "MY-HITEMP-CO2": {                            # your databank name: use this in␣
↪calc_spectrum()
                                                    # or SpectrumFactory.load_
↪databank()
        "path": [                                   # no "", multipath allowed
            "D:\\Databases\\HITEMP-CO2\\hitemp_07",
            "D:\\Databases\\HITEMP-CO2\\hitemp_08",
            "D:\\Databases\\HITEMP-CO2\\hitemp_09"
        ],
        "format": "hitran",                          # 'hitran' (HITRAN/HITEMP), 'cdsd-
↪hitemp', 'cdsd-4000'
                                                    # databank text file format.␣
↪More info in
                                                    # SpectrumFactory.load_databank␣
↪function.
                                                    # is used (check version)
        "levels_iso1": "D:\\PATH_TO\\energies_of_626_isotope.levels",
        "levels_iso2": "D:\\PATH_TO\\energies_of_636_isotope.levels",
        "levelsfmt": "cdsd",                        # 'cdsd', etc.
                                                    # how to read the previous file.␣
↪Default None.
        "levelszpe": "2531.828"                     # zero-point-energy (cm-1):␣
↪offset for all level
                                                    # energies. Default 0 (if not␣
↪given)
      }
  }
}
```

The full description of a `json` entry is given in `DBFORMAT`:

- `path` corresponds to Line databases (here: downloaded from [HITEMP-2010]) and the `levels_iso` are user generated Energy databases (here: calculated from the [CDSD-4000] Hamiltonian on non-distributed code, which takes into account non diagonal coupling terms).
- `format` is the databank text file format. It can be one of `'hitran'` (for HITRAN / HITEMP 2010), `'cdsd-hitemp'` and `'cdsd-4000'` for the different CDSD versions (for CO2 only). See full list in `KNOWN_DBFORMAT`.
- `parfuncfmt`: `cdsd`, `hapi` is the format of the tabulated partition functions used. If `'hapi'`, then [HAPI] is used to retrieve them (valid if your databank is HITRAN data). See full list in `KNOWN_PARFUNCFORMAT`

- `parfunc` is the path to the tabulated partition function to use in in equilibrium calculations (`eq_spectrum()`). If `parfuncfmt` is `'hapi'` then `parfunc` should be the link to the hapi.py file. If not given, then the `hapi` embedded in RADIS is used (check version)
- `levels_iso#` are the path to the energy levels to use for each isotope, which are needed for nonequilibrium calculations (`non_eq_spectrum()`).
- `levelsfmt` is the energy levels database format. Typically, `'radis'`, and various implementation of [CDSD-4000] nonequilibrium partitioning of vibrational and rotational energy: `'cdsd-pc'`, `'cdsd-pcN'`, `'cdsd-hamil'`. See full list in `KNOWN_LVLFORMAT`

How to create the configuration file?

A default `~/radis.json` configuration file can be generated with `setup_test_line_databases()`, which creates two test databases from fragments of [HITRAN-2020] line databases:

```
from radis.test.utils import setup_test_line_databases
setup_test_line_databases()
```

which will create a `~/radis.json` file with the following content

```
{
  "database": {
      "HITRAN-CO2-TEST": {
        "info": "HITRAN 2016 database, CO2, 1 main isotope (CO2-626), bandhead: 2380-
↪2398 cm-1 (4165-4200 nm)",
        "path": "PATH_TO\\radis\\radis\\test\\files\\hitran_co2_626_bandhead_4165_4200nm.
↪par",
        "format": "hitran",
        "parfuncfmt": "hapi",
        "levelsfmt": "radis"
      },
      "HITRAN-CO-TEST": {
        "info": "HITRAN 2016 database, CO, 3 main isotopes (CO-26, 36, 28), 2000-2300 cm-
↪1",
        "path": "PATH_TO\\radis\\radis\\test\\files\\hitran_co_3iso_2000_2300cm.par",
        "format": "hitran",
        "parfuncfmt": "hapi",
        "levelsfmt": "radis"
      },
      "HITEMP-CO2-TEST": {
        "info": "HITEMP-2010, CO2, 3 main isotope (CO2-626, 636, 628), 2283.7-2285.1 cm-1
↪",
        "path": "PATH_TO\\radis\\radis\\test\\files\\cdsd_hitemp_09_fragment.txt",
        "format": "cdsd-hitemp",
        "parfuncfmt": "hapi",
        "levelsfmt": "radis"
      }
  }
}
```

If you configuration file exists already, the test databases will simply be appended.

2.2.4 Advanced

Calculation Flow Chart

Refer to *Architecture* for an overview of how equilibrium and nonequilibrium calculations are conducted.

Use Custom Spectroscopic constants

Spectroscopic constants are a property of the RADIS `ElectronicState` class. All molecules are stored in the `Molecules` dictionary. You need to update this dictionary before running your calculation in order to use your own spectroscopic constants.

An example of how to use your own spectroscopic constants:

```
from radis import calc_spectrum
from radis.db.molecules import Molecules, ElectronicState

Molecules['CO2'][1]['X'] = ElectronicState('CO2', isotope=1, state='X', term_symbol='1u+
↪',
                           spectroscopic_constants='my_constants.json',  # <<< YOUR␣
↪FILE HERE
                           spectroscopic_constants_type='dunham',
                           Ediss=44600,
                           )
s = calc_spectrum(...)
```

- *Calculate Rovibrational Energies*

Vibrational bands

To calculate vibrational bands of a given spectrum separately (vibrational-state-specific calculations), use the `eq_bands()` and `non_eq_bands()` methods. See the `test_plot_all_CO2_bandheads()` example in `radis/test/lbl/test_bands.py` for more information.

Connect to a Spectrum Database

In RADIS, the same code can be used to retrieve precomputed spectra if they exist, or calculate them and store them if they don't. See *Precompute Spectra*

- *Spectrum Database*

2.2.5 Performance

RADIS is very optimized, making use of C-compiled libraries (NumPy, Numba) for computationally intensive steps, and data analysis libraries (Pandas) to handle lines databases efficiently. Additionaly, different strategies and parameters are used to improve performances further:

Line Database Reduction Strategies

By default:

- *linestrength cutoff* : lines with low linestrength are discarded after the new populations are calculated. Parameter: `cutoff` (see the default value in the arguments of `eq_spectrum()`)

Additional strategies (deactivated by default):

- *weak lines* (pseudo-continuum): lines which are close to a much stronger line are called weak lines. They are added to a pseudo-continuum and their lineshape is calculated with a simple rectangular approximation. See the default value in the arguments of `pseudo_continuum_threshold` (see arguments of `eq_spectrum()`)

Lineshape optimizations

Lineshape convolution is usually the performance bottleneck in any line-by-line code.

Two approaches can be used:

- improve the convolution efficiency. This involves using an efficient convolution algorithm, using a reduced convolution kernel, analytical approximations, or multiple spectral grid.
- reduce the number of convolutions (for a given number of lines): this is done using the LDM strategy.

RADIS implements the two approaches as well as various strategies and parameters to calculate the lineshapes efficiently.

- *broadening width* : lineshapes are calculated on a reduced spectral range. Voigt computation calculation times scale linearly with that parameter. Gaussian x Lorentzian calculation times scale as a square with that parameter. parameters: broadening_max_width
- *Voigt approximation* : Voigt is calculated with an analytical approximation. Parameter : `broadening_max_width` and default values in the arguments of `eq_spectrum()`. See `voigt_lineshape()`.
- *Fortran precompiled* : previous Voigt analytical approximation is precompiled in Fortran to improve performance times. This is always the case and cannot be changed on the user side at the moment. See the source code of `voigt_lineshape()`.
- *Multiple spectral grids* : many LBL codes use different spectral grids to calculate the lineshape wings with a lower resolution. This strategy is not implemented in RADIS.
- *LDM* : lines are projected on a Lineshape database to reduce the number of calculated lineshapes from millions to a few dozens. With this optimization strategy, the lineshape convolution becomes almost instantaneous and all the other strategies are rendered useless. Projection of all lines on the lineshape database becomes the performance bottleneck. parameters: `ldm_res_L`, `ldm_res_G`. (this is the default strategy implemented in RADIS). Learn more in [Spectral-Synthesis-Algorithm]

More details on the parameters below:

Computation parameters

If performance is an issue (for instance when calculating polyatomic spectra on large spectral ranges), you may want to tweak the computation parameters in `calc_spectrum()` and `SpectrumFactory`. In particular, the parameters that have the highest impact on the calculation performances are:

- The `broadening_max_width`, which defines the spectral range over which the broadening is calculated.
- The linestrength `cutoff`, which defines which low intensity lines should be discarded. See `plot_linestrength_hist()` to choose a correct cutoff.

Check the [RADIS-2018] article for a quantitative assessment of the influence of the different parameters.

Other strategies are possible, such as calculating the weak lines in a pseudo-continuum. This can result in orders of magnitude improvements in computation performances.:

- The `pseudo_continuum_threshold` defines which treshold should be used.

See the `test_abscoeff_continuum()` case in `radis/test/lbl/test_broadening.py` for an example, which can be run with (you will need the CDSD-HITEMP database installed)

```
pytest radis/test/lbl/test_broadening.py -m "test_abscoeff_continuum"
```

Choose the right wavenumber grid

`wstep` determines the wavenumber grid's resolution. Smaller the value, higher the resolution and vice-versa. By default **radis** uses `wstep=0.01`. You can manually set the `wstep` value in `calc_spectrum()` and `SpectrumFactory`. To get more accurate result you can further reduce the value, and to increase the performance you can increase the value.

Based on `wstep`, it will determine the number of gridpoints per linewidth. To make sure that there are enough gridpoints, Radis will raise an Accuracy Warning `_check_accuracy()` if number of gridpoints are less than `GRIDPOINTS_PER_LINEWIDTH_WARN_THRESHOLD` and raises an Accuracy Error if number of gridpoints are less than `GRIDPOINTS_PER_LINEWIDTH_ERROR_THRESHOLD`.

From `0.9.30` a new mode `wstep='auto'` has been added which directly computes the optimum value of `wstep` ensuring both performance and accuracy. It is ensured that there are slightly more or less than `GRIDPOINTS_PER_LINEWIDTH_WARN_THRESHOLD` points for each linewidth.

Note: wstep = 'auto' is optimized for performances while ensuring accuracy, but is still experimental in 0.9.30. Feedback welcome!

Sparse wavenumber grid

To compute large band spectra with a small number of lines, RADIS includes a sparse wavenumber implementation of the DIT algorithm, which is activated based on a scarcity criterion (`Nlines/Ngrid_points > 1`).

The sparse version can be forced to be activated or deactivated. This behavior is done by setting the `SPARSE_WAVENUMBER` key of the `radis.config` dictionary, or of the ~/radis.json user file.

See the *HITRAN full-range example* for an example.

Database loading

Line database can be a performance bottleneck, especially for large polyatomic molecules in the [HITEMP-2010] or [CDSD-4000] databases. Line database files are automatically cached by RADIS under a `.h5` format after they are loaded the first time. If you want to deactivate this behaviour, use `use_cached=False` in `calc_spectrum()`, or `db_use_cached=False, lvl_use_cached=False` in `SpectrumFactory`.

You can also use `init_databank()` instead of the default `load_databank()`. The former will save the line database parameter, and only load them if needed. This is useful if used in conjonction with `init_database()`, which will retrieve precomputed spectra from a database if they exist.

Manipulate the database

If for any reason, you want to manipulate the line database manually (for instance, keeping only lines emitting by a particular level), you need to access the `df0` attribute of `SpectrumFactory`.

> **Warning:** never overwrite the `df0` attribute, else some metadata may be lost in the process. Only use inplace operations.

For instance:

```
sf = SpectrumFactory(
    wavenum_min= 2150.4,
    wavenum_max=2151.4,
    pressure=1,
    isotope=1)
sf.load_databank('HITRAN-CO-TEST')
sf.df0.drop(sf.df0[sf.df0.vu!=1].index, inplace=True)   # keep lines emitted by v'=1 only
sf.eq_spectrum(Tgas=3000, name='vu=1').plot()
```

`df0` contains the lines as they are loaded from the database. `df1` is generated during the spectrum calculation, after the line database reduction steps, population calculation, and scaling of intensity and broadening parameters with the calculated conditions.

Tabulated Partition Functions

At nonequilibrium, calculating partition functions by full summation of all rovibrational levels can become costly. Radis offers to tabulate them just-in-time, using the `parsum_mode='tabulation'` of `calc_spectrum()` or `SpectrumFactory`. See `parsum_mode`.

Profiler

You may want to track where the calculation is taking some time. You can set `verbose=1` or higher to print the time spent on the different calculation steps at runtime. Example with `verbose=3`:

```
s = calc_spectrum(1900, 2300,         # cm-1
                  molecule='CO',
                  isotope='1,2,3',
                  pressure=1.01325,   # bar
                  Tvib=1000,          # K
                  Trot=300,           # K
                  mole_fraction=0.1,
                  verbose=3,
                  )
```

Performance profiles are kept in the output spectrum `conditions['profiler']` dictionary. You can also use the `print_perf_profile()` method in the SpectrumFactory object or the `print_perf_profile()` method in the Spectrum object to print them in the console :

For the above example:

```
s.print_perf_profile()
```

::

 # Output:

spectrum_calculation 0.189s

check_line_databank 0.000s check_non_eq_param 0.042s fetch_energy_5 0.015s calc_weight_trans 0.008s reinitialize 0.002s

copy_database 0.000s memory_usage_warning 0.002s reset_population 0.000s

calc_noneq_population 0.041s

part_function 0.035s population 0.006s

scaled_non_eq_linestrength 0.005s

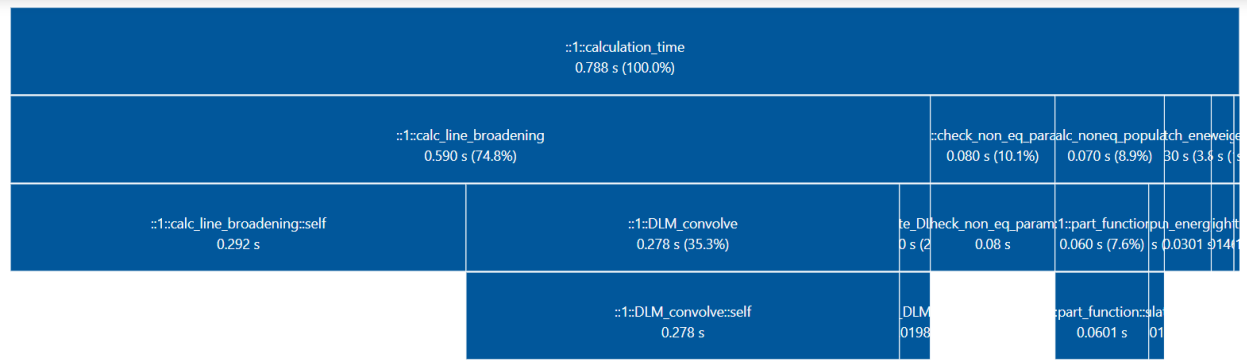
map_part_func 0.001s corrected_population_se 0.003s

calc_emission_integral 0.006s applied_linestrength_cutoff 0.002s calc_lineshift 0.001s calc_hwhm 0.007s generate_wavenumber_arrays 0.001s calc_line_broadening 0.074s

precompute_LDM_lineshapes 0.012s LDM_Initialized_vectors 0.000s LDM_closest_matching_line 0.001s LDM_Distribute_lines 0.001s LDM_convolve 0.060s others 0.001s

calc_other_spectral_quan 0.003s generate_spectrum_obj 0.000s others -0.016s

Finally, you can also use the SpectrumFactory `generate_perf_profile()` Spectrum `generate_perf_profile()` methods to generate an interactive profiler in the browser.



Predict Time

`predict_time()` function uses the input parameters like `Spectral Range`, `Number of lines`, `wstep`, `truncation` to predict the estimated calculation time for the Spectrum broadening step(bottleneck step) for the current optimization and broadening_method. The formula for predicting time is based on benchmarks performed on various parameters for different optimization, broadening_method and deriving its time complexity.

The following Benchmarks were used to derive the time complexity:

https://anandxkumar.github.io/Benchmark_Visualization_GSoC_2021/

Complexity vs Calculation Time Visualizations for different optimizations and broadening_method:

LBL>Voigt: LINK
DIT>Voigt: LINK
DIT>FFT: LINK

Precompute Spectra

See `init_database()`, which is the direct integration of `SpecDatabase` in a `SpectrumFactory`

2.3 The Spectrum object

RADIS has powerful tools to post-process spectra created by *the line-by-line module* or by other spectral codes.

At the core of the post-processing is the `Spectrum` class, which features methods to:

- *generate Spectrum objects* from text files or python arrays.
- *rescale a spectrum* without redoing the line-by-line calculation.
- apply *instrumental slit functions*.
- *plot* with one line and in whatever unit.
- *crop, offset* or *interpolate*.
- *remove baselines*.
- *multiply or add constants* as simply as with `s=10*s` or `s=s+0.2` in Python.
- *store* an experimental or a calculated spectrum while retaining the metadata.
- *compare different spectra*.
- combine multiple spectra along the *line-of-sight*.
- manipulate a folder of spectra easily with *spectrum Databases*.
- compute transmittance from absorbance, or whatever *missing spectral arrays*.
- use the *line survey* tool to identify each line.
- *Load an experimental spectrum*
- *Line Survey*
- *Blackbody radiation*
- *Remove a baseline*
- *GPU Accelerated Spectra*
- *Calculate non-LTE spectra of carbon-monoxide*
- *Use different plot themes*
- *Calculate a large spectrum by part*
- *Compare CO spectrum from the GEISA and HITRAN database*
- *Calculate a spectrum from HITEMP*
- *Calculate a spectrum from ExoMol*
- *Real-time GPU Accelerated Spectra (Interactive)*
- *See populations of computed levels*
- *Spectrum Database*
- *Compare CO xsections from the ExoMol and HITEMP database*
- *Post-process using Specutils*

- *Multi-temperature Fit*
- *1 temperature fit*

Refer to the guide below for an exhaustive list of all features:

For any other question you can use the Q&A forum, the GitHub issues or the community chats on Gitter or Slack .

2.3.1 How to generate a Spectrum?

Calculate a Spectrum

Usually a Spectrum object is the output from a line-by-line (LBL) radiation code. Refer to the LBL documentation for that. Example using the RADIS LBL module with `calc_spectrum()`:

```
from radis import calc_spectrum
s = calc_spectrum(...)        # s is a Spectrum object
```

Or with the `SpectrumFactory` class, which can be used to batch-generate multiple spectra using the same line database:

```
from radis import SpectrumFactory
sf = SpectrumFactory(...)
sf.fetch_databank("hitemp", load_columns='noneq')  # or 'hitran', 'exomol', etc.
s = sf.eq_spectrum(...)
s2 = sf.non_eq_spectrum(...)
```

Note that the `SpectrumFactory` class has the `init_database()` method that automatically retrieves a spectrum from a `SpecDatabase` if you calculated it already, or calculates it and stores it if you didn't. Very useful for spectra that require long computational times!

Initialize from Python arrays

The standard way to build a Radis Spectrum is from a dictionary of numpy arrays:

```
# w, k, I are numpy arrays for wavenumbers, absorption coefficient, and radiance.
from radis import Spectrum
s = Spectrum({"wavenumber":w, "abscoeff":k, "radiance_noslit":I},
             units={"radiance_noslit":"mW/cm2/sr/nm", "abscoeff":"cm-1"})
```

Or:

```
s = Spectrum({"abscoeff":(w,k), "radiance_noslit":(w,I)},
             wunit="cm-1"
             units={"radiance_noslit":"mW/cm2/sr/nm", "abscoeff":"cm-1"})
```

You can also use the `from_array()` convenience function:

```
# w, T are two numpy arrays
from radis import Spectrum
s = Spectrum.from_array(w, T, 'transmittance_noslit',
                        wunit='nm', unit='') # adimensioned
```

Dimensionned arrays can also be used directly

```
import astropy.units as u
w = np.linspace(200, 300) * u.nm
I = np.random.rand(len(w)) * u.mW/u.cm**2/u.sr/u.nm
s = Spectrum.from_array(w, I, 'radiance_noslit')
```

Other convenience functions have been added to handle the usual cases: `calculated_spectrum()`, `transmittance_spectrum()` and `experimental_spectrum()`:

```
# w, T, I are numpy arrays for wavelength, transmittance and radiance
from radis import calculated_spectrum, transmittance_spectrum, experimental_spectrum
s1 = calculated_spectrum(w, I, wunit='nm', Iunit='W/cm2/sr/nm')     # creates 'radiance_
↪noslit'
s2 = transmittance_spectrum(w, T, wunit='nm')                       # creates
↪'transmittance_noslit'
s3 = experimental_spectrum(w, I, wunit='nm', Iunit='W/cm2/sr/nm')   # creates 'radiance'
```

Initialize from Specutils

Use `from_specutils()` to convert from a `specutils` specutils.spectra.spectrum1d.Spectrum1D object

```
from radis import Spectrum
Spectrum.from_specutils(spectrum)
```

Initialize from a text file

Spectrum objects can also be generated directly from a text file.

From a file, use `from_txt()`

```
# 'exp_spectrum.txt' contains a spectrum
from radis import Spectrum
s = Spectrum.from_txt('exp_spectrum.txt', 'radiance',
                          wunit='nm', unit='mW/cm2/sr/nm')
```

It is, however, recommended to use the RADIS `.spec` json format to store and load arrays :

Load from a .spec file

A `.spec` file contains all the Spectrum spectral arrays as well as the input conditions used to generate it. To retrieve it use the `load_spec()` function:

```
s = load_spec('my_spectrum.spec')
```

Sometimes the `.spec` file may have been generated under a compressed format where redundant spectral arrays have been removed (for instance, transmittance if you already know absorbance). Use the `update()` method to regenerate missing spectral arrays:

```
s = load_spec('my_spectrum.spec', binary=True)
s.update()
```

If many spectra are stored in a folder, it may be time to set up a `SpecDatabase` structure to easily see all Spectrum conditions and get Spectrum that suits specific parameters

- *Load an experimental spectrum*
- *Remove a baseline*
- *1 temperature fit*

Load from a HDF5 file

This is the fastest way to read a Spectrum object from disk. It keeps metadata and units, and you can also load only a part of a very large spectrum. Use `from_hdf5()`

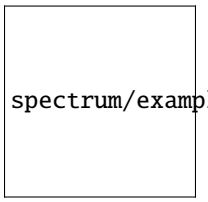
```
Spectrum.from_hdf5("rad_hdf.h5", wmin=2100, wmax=2200, columns=['abscoeff', 'emisscoeff
↪'])
```

Calculate a test spectrum

You need a spectrum immediatly, to run some tests ? Use `test_spectrum()`

```
s = radis.test_spectrum()
s.apply_slit(0.5, 'nm')
s.plot('radiance')
```

This returns the CO spectrum from the *first documentation example*



scale
60 %

Generate a Blackbody (Planck) function object

In RADIS you can either use the `planck()` and `planck_wn()` functions that generate Planck radiation arrays for wavelength and wavenumber, respectively.

Or, you can use the `sPlanck()` function that returns a `Spectrum` object, with all the associated methods (add in a line-of-sight, compare, etc.)

Example:

```
s = sPlanck(wavelength_min=3000, wavelength_max=50000,
            T=288, eps=1)
s.plot()
```

- *Blackbody radiation*

2.3.2 Spectral Arrays

A `Spectrum` object can contain one spectral arrays, such as `'radiance'` for emission spectra, or `'transmittance'` for absorption spectra. It can also contain both emission and absorption quantities to be later combined with other spectra by solving the *radiative transfer equation*.

Some variables represent quantities that have been convolved with an instrumental slit function, as measured in experimental spectra:

- `'radiance'`: the spectral radiance, convolved by the instrument function (typically in :math:`'mW/cm^2/sr/nm'`). This is sometimes confusingly called *spectral intensity*.
- `'transmittance'`: the directional spectral transmittance (:math:`0` to :math:`1`), convolved by the instrument function.
- `'emissivity'`: the directional spectral emissivity (:math:`0` to :math:`1`), convolved by the instrument function. The spectral emissivity is the radiance emitted by a surface divided by that emitted by a black body at the same temperature as that surface. This value is only defined under thermal equilibrium.

Other variables represent quantities that have not been convolved (theoretical spectra):

- `'radiance_noslit'`: the spectral radiance (typically in $mW/cm^2/sr/nm$). This

is sometimes confusingly called *spectral intensity*. - `'transmittance_noslit'`: the directional spectral transmittance (:math:`0` to :math:`1`) - `'emissivity_noslit'`: spectral emissivity (`0` to `1`) *i.e.* the radiance emitted by a surface

  divided by that emitted by a black body at the same temperature as that surface. This value is only defined under thermal equilibrium.

- `'emisscoeff'`: the directional spectral emission density (typically in :math:`'mW/cm^3/sr/nm'`).
- `'absorbance'`: the directional spectral absorbance (no dimension), also called *optical depth*.
- `'abscoeff'`: spectral absorption coefficient (typically in :math:`'cm^{-1}'`), also called *opacity*. This is sometimes found as the *extinction coefficient* in the literature (strictly speaking, *extinction* is *absorption* + *diffusion*, the latter being negligible in the infrared).
- `'xsection'`: absorption cross-section, typically in cm2

Additionally, RADIS may calculate extra quantities such as:

- `'emisscoeff_continuum'`: the pseudo-continuum part of the spectral emission density `'emisscoeff'`, that can be generated by `SpectrumFactory`
- `'abscoeff_continuum'` the pseudo-continuum part of the spectral absorption coefficient `'abscoeff'`, that can be generated by `SpectrumFactory`

See the latest list in the `CONVOLUTED_QUANTITIES` and `NON_CONVOLUTED_QUANTITIES`.

Custom spectral arrays

A `Spectrum` object is built on top of a dictionary structure, and can handle spectral arrays with any name.

Custom spectral arrays with arbitrary units can be defined when creating a Spectrum object, for instance:

```
# w, I are two numpy arrays
s = Spectrum.from_array(w, I, 'irradiance',
                        wunit='nm', unit='w/cm2/nm')
```

Although not recommended, it is also possible to directly edit the dictionary containing the objects. For instance, this is done in CO2 radiative forcing example to calculate irradiance from radiance (by multiplying by `'pi'` and changing the unit):

```
s._q['irradiance'] = s.get('radiance_noslit')[1]*pi
s.units['irradiance'] = s.units['radiance_noslit'].replace('/sr', '')
```

The unit conversion methods will properly work with custom units.

Warning: Rescaling or combining spectra with custom quantities may result in errors.

Relations between quantities

Most of the quantities above can be recomputed from one another. In a homogeneous slab, one requires an emission spectral density, and an absorption spectral density, to be able to recompute the other quantities (provided that conditions such as path length are given). Under equilibrium, only one quantity is needed. Missing quantities can be recomputed automatically with the `update()` method.

Units

Default units are stored in the `units` dictionary

It is strongly advised not to modify the dictionary above. However, spectral arrays can be retrieved in arbitrary units with the `get()` method.

When a spectral unit is convolved with `apply_slit()`, a new convolved spectral array is created. The unit of the convolved spectral array may be different, depending on how the slit function was normalized. Several options are available in RADIS. Please refer to the documentation of the `apply_slit()` method.

2.3.3 How to access Spectrum properties?

Get spectral arrays

Spectral Arrays of a Spectrum object can be stored in arbitrary wavespace (wavenumbers, wavelengths in air, or wavelengths in vacuum) and arbitrary units.

Therefore, it is recommendeded to use the `get()` method to retrieve the quantity un the units you want:

```
w, I = s.get('transmittance_noslit', wunit='cm-1')
_, R = s.get('radiance_noslit', wunit='nm', Iunit='W/cm2/sr/nm',
             medium='air')
```

Use with `return_units` to get dimensioned Astropy Quantities

```
w, R  = s.get('radiance_noslit', return_units=True)
# w, R are astropy quantities
```

See spectral arrays for the list of spectral arrays.

Get wavelength/wavenumber

Use the `get_wavelength()` and `get_wavenumber()` methods:

```
w_nm = s.get_wavelength()
w_cm = s.get_wavenumber()
```

Print Spectrum conditions

Want to know under which calculation conditions was your Spectrum object generated, or under which experimental conditions it was measured? Just print it:

```
print(s)
```

(that shows all spectral arrays stored in the object, all keys and values in the `conditions` dictionary, and all atoms/molecules stored in the `populations` dictionary)

You can also show the conditions only with `print_conditions()`:

```
s.print_conditions()
```

Plot spectral arrays

Use `plot()`:

```
s.plot('radiance_noslit')
```

You can plot on the same figure as before using the convenient `nfig` parameter:

```
s_exp.plot('radiance_noslit', nfig='same')
```

But for comparing different spectra you may want to use `plot_diff()` directly.

Plot populations

Get or plot populations computed in calculations. Use `get_populations()` or `plot_populations()`:

```
s.plot_populations('vib', nunit='cm-3')
```

- *See populations of computed levels*

Plot line survey

Use the `line_survey()` method. Example of output:

```
from radis import SpectrumFactory
sf = SpectrumFactory(
                     wavenum_min=2380,
                     wavenum_max=2400,
                     mole_fraction=400e-6,
                     path_length=100,  # cm
```

(continues on next page)

(continued from previous page)

```
                    isotope=[1],
                    )
sf.load_databank('HITRAN-CO2-TEST')
s = sf.eq_spectrum(Tgas=1500)
s.apply_slit(0.5)
s.line_survey(overlay='radiance_noslit', barwidth=0.01)
```

- *Line Survey*
- *See populations of computed levels*

Know if a spectrum has nan

`has_nan()` looks over all spectral quantities. `print(s)` will also show the number of nans per quantity

```
s = radis.test_spectrum()
s.has_nan()
```

2.3.4 How to export ?

Save a Spectrum object

To store use the `store()` method:

```
# s is a Spectrum object
s.store('temp_file.spec')
from radis import load_spec
s2 = load_spec('temp_file.spec')
assert s == s2  # compare both
```

The generated `.spec` file can be read (and edited) with any text editor. However, it may take a lot of space. If memory is important, you may use the `compress=True` argument which will remove redundant spectral arrays (for instance, transmittance if you already know absorbance), and store the .spec file under binary format. Use the `update()` method to regenerate missing quantities:

```
s.store('temp_file.spec', compress=True, if_exists_then='replace')
s2 = load_spec('temp_file.spec')
s2.update()    # regenerate missing quantities
```

If calculating many spectra, you may want to handle all of them together in a `SpecDatabase`. You can add them to the existing database with the `add()` method:

```
db = SpecDatabase(r"path/to/database")     # create or loads database
db.add(s)
```

Note that if using the RADIS LBL code to generate your spectra, the `SpectrumFactory` class has the `init_database()` method that automatically retrieves a spectrum from a database if you calculated it already, or calculates it and stores it if you didn't. Very useful for spectra that requires long computational times!

Export to hdf5

This is the fastest way to dump a Spectrum object on disk (and also, it keeps metadata and therefore units !). Use `to_hdf5()`

```
s.to_hdf5('spec01.h5')
```

Export to txt

Saving to `.txt` in general isn't recommended as you will loose some information (for instance, the conditions). You better use `store()` and export to `.spec` [a hidden `.json`] format.

If you really need to export a given spectral arrays to txt file (for use in another software, for instance), you can use the `savetxt()` method that will export a given spectral arrays:

```
s.savetxt('radiance_W_cm2_sr_um.csv', 'radiance_noslit', wunit='nm', Iunit='W/cm2/sr/µm')
```

Export to Pandas

Use `to_pandas()`

```
s.to_pandas()
```

This will return a DataFrame with all spectral arrays as columns.

Export to Specutils

Use `to_specutils()` to convert to a to `specutils` `specutils.spectra.spectrum1d.Spectrum1D` object

```
s.to_specutils()
```

- *Post-process using Specutils*

2.3.5 How to modify a Spectrum object?

Calculate missing quantities

Some spectral arrays can be infered from quantities stored in the Spectrum if enough conditions are given. For instance, transmittance can be recomputed from the spectral absorption coefficient if the path length is stored in the conditions.

The `update()` method can be used to do that. In the example below, we recompute transmittance from the absorption coefficient (opacity)

```
# w, A are numpy arrays for wavenumber and absorption coefficient
s = Spectrum.from_array(w, A, 'abscoeff', wunit='cm-1')
s.update('transmittance_noslit')
```

All derivable quantities can be computed using `.update('all')` or simply `.update()`:

```
s.update()
```

Update Spectrum conditions

Spectrum conditions are stored in a `conditions` dictionary

Conditions can be updated *a posteriori* by modifying the dictionary:

```
s.conditions['path_length'] = 10    # cm
```

Rescale Spectrum with new path length

Path length can be changed after the spectra were calculated with the `rescale_path_length()` method. If the spectrum is not optically thin, this requires solving the radiative transfer equation again, so the `emisscoeff` and `abscoeff` (opacity) quantities will have to be stored in the Spectrum, or any equivalent combination (radiance_noslit and absorbance, for instance).

Example:

```
from radis import load_spec
s = load_spec('co_calculation.spec')
s.rescale_path_length(0.5)      # calculate for new path_length
```

Rescale Spectrum with new mole fraction

Warning: Rescaling mole fractions neglects the changes in collisional broadening

mole fraction can also be changed in post-processing, using the `rescale_mole_fraction()` method that works similarly to the `rescale_path_length()` method. However, the broadening coefficients are left unchanged, which is valid for small mole fraction changes. However, for large mole fraction changes you will have to recalculate the spectrum from scratch.

```
>>> s.rescale_mole_fraction(0.02)   # calculate for new mole fraction
```

Apply instrumental slit function

Use `apply_slit()`:

```
s.apply_slit(1.5)    # nm
```

By default, convoluted spectra are thinner than non-convoluted spectra, to remove side effects. Use the `mode=` argument to change this behavior.

- *Line Survey*
- *GPU Accelerated Spectra*
- *Calculate non-LTE spectra of carbon-monoxide*
- *Calculate a spectrum from ExoMol*

Plot the slit function that was applied

Use `plot_slit()`. You can also change the unit:

```
s.apply_slit(0.5, 'cm-1')    # for instance
s.plot_slit('nm')
```

Multiply, subtract

Sometimes you need to manipulate an experimental spectrum, to account for calibration or remove a baseline. Spectrum operations are done just for that:

- `add_constant()`
- `add_array()`
- `add_spectra()`
- `substract_spectra()`

Most of these functions are implemented with the standard operators. Ex:

```
((s_exp - 0.1)*10).plot()   # works for a Spectrum s_exp
```

Note that these operators are purely algebraic and should not be used in place of the line-of-sight functions, i.e, `SerialSlabs()` (>) and `MergeSlabs()` (//)

Most of these functions will only work if there is only one spectral arrays defined in the Spectrum. If there is any ambiguity, use the `take()` method. For instance, the following line is a valid RADIS command to plot the spectral radiance of a spectrum with a low resolution:

```
(10*(s.apply_slit(10, 'nm')).take('radiance')).plot()
```

Algebraic operations also work with dimensioned `Quantity`. For instance, remove a constant baseline in a given unit:

```
s -= 0.1 * u.Unit('W/cm2/sr/nm')
```

The `max()` function returns a dimensionned value, therefore it can be used to normalize a spectrum directly :

```
s /= s.max()
```

Or below, we calibrate a Spectrum, assuming the spectrum units is in "count", and that our calibration show we have 94 $mW/cm2/sr/nm$ per count.

```
s *= 94 * u.Unit("mW/cm2/sr/nm") / u.Unit("count")
```

Offset, crop

Use the associated functions: `crop()`, `offset()`.

They are also defined as methods of the Spectrum objects (see `crop()`, `offset()`), so they can be used directly with:

```
s.offset(3, 'nm')
s.crop(370, 380, 'nm')
```

By default, using methods that will modify the object in place, using the functions will generate a new Spectrum.

- *Load an experimental spectrum*

Normalize

Use `normalize()` directly, if your spectrum only has one spectral arrays

```
s.normalize()
s.normalize(normalize_how="max")
s.normalize(normalize_how="area")
```

You can also normalize only on limited range. Useful for noisy spectra

```
s.normalize(wrange=(2250, 2500), wunit="cm-1", normalize_how="mean")
```

This returns a new spectrum and does not modify the Spectrum itself. To do so use:

```
s.normalize(inplace=True)
```

Chaining

You can chain the various methods of Spectrum. For instance:

```
s.normalize().plot()
```

Or:

```
s.crop(4120, 4220, 'nm').apply_slit(3, 'nm').take('radiance')
```

If you want to create a new spectrum, don't forget to set `inplace=False` for the first command that allows it. i.e

```
s2 = s.crop(4120, 4220, 'nm', inplace=False).apply_slit(3, 'nm').offset(1.5, 'nm')
```

Remove a baseline

Either use the `add_constant()` mentionned above, which is implemented with the `-` operator:

```
s2 = s - 0.1
```

Or remove a linear baseline with:

- `get_baseline()`
- `sub_baseline()`

You could also use the functions available in `pyspecutils`, see `to_specutils()`.

Calculate transmittance from radiance with Kirchoff's law

RADIS can be used to infer spectral arrays from others if they can be derived. If on top that, equilibrium is assumed, then Kirchoff's law is used. See `How to ... calculate missing quantities?` and the `update()` method with argument `assume_equilibrium=True`. Example:

```
s = calculated_spectrum(...)     # defines 'radiance_noslit')
s.update('transmittance_noslit')
s.plot('transmittance_noslit')
```

You can infer if a Spectrum is at (thermal) equilibrium with the `is_at_equilibrium()` method, that looks up the declared spectrum conditions and ensures `Tgas==Tvib==Trot`. It does not imply chemical equilibrium (mole fractions are still arbitrary)

2.3.6 How to handle multiple Spectra?

Build a line-of-sight profile

RADIS allows the combination of Spectra such as:

```
s_line_of_sight = (s_plasma_CO2 // s_plasma_CO) > (s_room_absorption)
```

Refer to the line-of-sight module

Compare two Spectra

You can compare two Spectrum objects using the `compare_with()` method, or simply the == statement (which is essentially the same thing):

```
s1 == s2
>>> True/False
s1.compare_with(s2)
>>> True/False
```

However, `compare_with()` allows more freedom regarding what quantities to compare. == will compare everything of two spectra, including input conditions, units under which spectral quantities are stored, populations of species if they were saved, etc. In many situations, we may want to simply compare the spectra themselves, or even a particular quantity like *transmittance_noslit*. Use:

```
s1.compare_with(s2, spectra_only=True)                    # compares all spectral arrays
s1.compare_with(s2, spectra_only='transmittance_noslit')  # compares transmittance only
```

The aforementionned methods will return a boolean array (True/False). If you need the difference, or ratio, or distance, between your two spectra, or simply want to plot the difference, you can use one of the predefined functions `get_diff()`, `get_ratio()`, `get_distance()`, `get_residual()` or the plot function `plot_diff()`:

```
from radis.spectrum import plot_diff
s1 = load_spec(temp_file_name)
s2 = load_spec(temp_file_name2)
plot_diff(s1, s2, 'radiance')
```

These functions usually require that the spectra are calculated on the same spectral range. When comparing, let's say, a calculated spectrum with experimental data, you may want to interpolate: you can have a look at the `resample()` method. See *Interpolate a Spectrum on another* for details.

In `plot_diff()`, you can choose to plot the absolute difference (`method='diff'`), or the ratio (`method='ratio'`), or both:

```
# Below we compare 2 CO2 spectra s_cdsd and s_hitemp previously calculated with two␣
↪different line databases.
from radis import plot_diff
plot_diff(s_cdsd, s_hitemp, method=['diff', 'ratio'])
```

Plot in log scale

If you wish to plot in a logscale, it can be done in the following way:':

```
fig, [ax0, ax1] = plot_diff(s_expe, s_test, normalize=False, verbose=False)
ylim0 = ax0.get_ybound()
ax0.set_yscale("log")
ax0.set_ybound(ylim0)
```

Fit an experimental spectrum

RADIS does not include fitting algorithms. To fit an experimental spectrum, one should use one of the widely available optimization algorithms from the Python ecosystem, for instance `scipy.optimize.minimize()`.

The `get_residual()` and `get_residual_integral()` functions can be used to return a scalar to feed to the `minimize()` function.

A simple fitting procedure could be:

```
from scipy.optimize import minimize
from radis import calc_spectrum, experimental_spectrum

s_exp = experimental_spectrum(...)

def cost_function(T):
    calc_spectrum(Tgas=T,
                  ... # other parameters )
    return get_residual(s_exp, s)

best = minimize(cost_function,
                800, # initial value
                bounds=[500, 2000],
                )
T_best = best.x
```

Note however that the performances of a fitting procedure can be vastly improved by not reloading the line database every time. In that case, it becomes interesting to use the `SpectrumFactory` class.

An example of a script that uses the `SpectrumFactory`, multiple fitting parameters, and plots the residual and the calculated spectrum in real-time, can be found *in the Examples page*

Interpolate a Spectrum on another

Let's interpolate a calculated spectrum on an experimental spectrum, using the `resample()` and, for instance, the `get_wavelength()` method:

```
# let's say we have two objects:
s_exp = load_spec('...')
s_calc = calc_spectrum(...)
# resample:
s_calc.resample(s_exp.get_wavelength(), 'nm')
```

Energy conservation is ensured and an error is raised if your interpolation is too bad. If you need to adjust the error threshold, see the parameters in `resample()`.

Create a database of Spectrum objects

Use the `SpecDatabase` class. It takes a folder as an argument, and converts it in a list of `Spectrum` objects. Then, you can:

- see the properties of all spectra within this folder with `see()`
- select the Spectrum that match a given set of conditions with `get()`, `get_unique()` and `get_closest()`
- fit an experimental spectrum against all precomputed spectra in the folder with `fit_spectrum()`

See more information about databases below.

2.3.7 Spectrum Database

`Spectrum` objects can be stored/loaded to/from .spec JSON files using the `store()` method and the `load_spec()` function.

- *Spectrum Database*

It is also possible to set up a `SpecDatabase` which reads all .spec files in a folder. The `SpecDatabase` can then be connected to a `SpectrumFactory` so that spectra already in the database are not recomputed, and that new calculated spectra are stored in the folder

Example:

```
db = SpecDatabase(r"path/to/database")     # create or loads database

db.update()  # in case something changed
db.see(['Tvib', 'Trot'])   # nice print in console

s = db.get('Tvib==3000')[0]  # get a Spectrum back
db.add(s)  # update database (and raise error because duplicate!)
```

A `SpecDatabase` can also be used to compare the physical and computation parameters of all spectra in a folder. Indeed, whenever the database is loaded, a summary `.csv` file is generated that contains all conditions and can be read, for instance, with Excel.

Example:

```
from radis import SpecDatabase
SpecDatabase(r".")     # this generates a .csv file in the current folder
```

The examples below show some actions that can be performed on a database:

Iterate over all Spectra in a database

Both methods below are equivalent. Directly iterating over the database:

```
db = SpecDatabase('.')
for s in db:
    print(s.name)
```

Or using the `get()` method with no filtering condition:

```
db = SpecDatabase('.')
for s in db.get():
    print(s.name)
```

You can also use dictionary-like methods: `keys()`, `values()` and `items()` where Spectrum are returned under a `{path:Spectrum}` dictionary.

Filter spectra that match certain conditions

If you want to get Spectra in your database that match certain conditions (e.g: a particular temperature), you may want to have a look at the `get()`, `get_unique()` and `get_closest()` methods

Fit an experimental spectrum against precomputed spectra

The `fit_spectrum()` method of `SpecDatabase` can be used to return the spectrum of the database that matches the best an experimental spectrum:

```
s_exp = experimental_spectrum(...)
db = SpecDatabase('...')
db.fit_spectrum(s_exp)
```

By default `fit_spectrum()` uses the `get_residual()` function. You can use an customized function too (below: to get the transmittance):

```
from radis import get_residual
db.fit_spectrum(s_exp, get_residual=lambda s_exp, s: get_residual(s_exp, s, var=
↪'transmittance'))
```

You don't necessarily need to precompute spectra to fit an experimental spectrum. You can find an example of *multi temperature fitting script* in the Example pages, which shows the evolution of the spectra in real-time. You can get inspiration from there!

Updating a database

Update all spectra in current folder with a new condition ('author'), making use of the `items()` method:

```
from radis import SpecDatabase
db = SpecDatabase('.')
for path, s in db.items():
    s.conditions['author'] = 'me'
    s.store(path, if_exists_then='replace')
```

You may also be interested in the `map()` method.

When not to use a Database

If you simply want to store and reload one `Spectrum` object, no need to use a database: you better use the `store()` method and `load_spec()` function.

Databases prove useful only when you want to filter precomputed Spectra based on certain conditions.

2.4 Line-of-sight module

This module takes several `Spectrum` objects as an input and combines then along the line-of-sight (`SerialSlabs()`) or at the same spatial position (`MergeSlabs()`), to reproduce line-of-sight experiments

2.4.1 How to combine slabs?

Along the line-of-sight

Use the `SerialSlabs()` function:

```
s1 = calc_spectrum(...)
s2 = calc_spectrum(...)
s3 = SerialSlabs(s1, s2)
```

You can also use the > operator. The previous line is equivalent to:

```
s3 = s1 > s2
```

At the same spatial position

Use the `MergeSlabs()` function:

Merge two spectra calculated with different species (true only if broadening coefficient dont change much):

```
from radis import calc_spectrum, MergeSlabs
s1 = calc_spectrum(...)
s2 = calc_spectrum(...)
s3 = MergeSlabs(s1, s2)
```

You can also use the // operator. The previous line is equivalent to:

```
s3 = s1 // s2
```

- *Calculate a large spectrum by part*

2.4.2 Practical Examples

Below are some practical examples of the use of the Line-of-sight module:

Build a large spectrum

If you want to calculate a spectrum on a very large spectral range which cannot be handled in memory at once, you can calculate partial, non-overlapping spectral ranges and use `MergeSlabs()` to combine them. In that case, we tell `MergeSlabs()` to use the full spectral range and that the partial spectra are transparent outside of their definition range:

```
from radis import load_spec, MergeSlabs
spectra = []
for f in ['spec1.spec', 'spec2.spec', ...]:  # precomputed spectra
    spectra.append(load_spec(f))
s = MergeSlabs(*spectra, resample='full', out='transparent')
s.plot()
```

Get the contribution of each slab along the LOS

Let's say you have a total line of sight:

```
s_los = s1 > s2 > s3
```

If you want to get the contribution of `s2` to the line-of-sight emission, you need to discard the emission of `s3` but take into account its absorption. This is done using the `PerfectAbsorber()` function, which returns a new Spectrum with all the emission features set to 0:

```
from radis import PerfectAbsorber
(s2 > PerfectAbsorber(s3)).plot('radiance_noslit')
```

And the contribution of `s1` would be:

```
(s1 > PerfectAbsorber(s2>s3)).plot('radiance_noslit')
```

2.5 Examples

Many other examples scripts are available on the radis-examples project.

2.5.1 Line Survey

Example of output produced by the `LineSurvey` tool:

```
from radis import SpectrumFactory
sf = SpectrumFactory(
                     wavenum_min=2380,
                     wavenum_max=2400,
                     mole_fraction=400e-6,
                     path_length=100,  # cm
                     isotope=[1],
                     )
sf.load_databank('HITRAN-CO2-TEST')
s = sf.eq_spectrum(Tgas=1500)
s.apply_slit(0.5)
s.line_survey(overlay='radiance_noslit', barwidth=0.01)
```

The graph is a html file that can be shared easily even to non-Python users.

2.5.2 RADIS in-the-browser

RADIS in-the-browser sessions can be run from the RADIS examples project. No installation needed, you don't even need Python on your computer.

For example, run the Quick Start *first example* by clicking on the link below:

Or start a bare RADIS online session:

The full list can be found on the RADIS Interactive Examples project.

2.5.3 Get rovibrational energies

RADIS can simply be used to calculate the rovibrational energies of molecules, using the built-in *spectroscopic constants*. See the `getMolecule()` function, and the `Molecules` list containing all `ElectronicState` objects.

Here we get the energy of the asymmetric mode of CO2:

```
from radis import getMolecule
CO2 = getMolecule('CO2', 1, 'X')
print(CO2.Erovib(0, 0, 0, 1, 0))
>>> 2324.2199999
```

Here we get the energy of the v=6, J=3 level of the 2nd isotope of CO:

```
CO = getMolecule('CO', 2, 'X')
print(CO.Erovib(6, 3))
>>> 12218.8130906978
```

2.5.4 Calculate Partition Functions

By default and for equilibrium calculations, RADIS calculates Partition Functions using the TIPS program ([TIPS-2020]) through [HAPI]. These partition functions can be retrieved with the `PartFunc_Dunham` class:

```
from radis.levels.partfunc import PartFuncTIPS
from radis.db.classes import get_molecule_identifier

M = get_molecule_identifier('N2O')
iso=1

Q = PartFuncTIPS(M, iso)
print(Q.at(T=1500))
```

RADIS can also be used to compute Partition Functions from rovibrational energies calculated with the built-in *spectroscopic constants*.

The calculation uses the `at()` method of the `PartFunc_Dunham` class, which reads `Molecules`. A minimal working example is:

```
from radis.levels.partfunc import PartFunc_Dunham
from radis.db.molecules import Molecules
iso=1
electronic_state = 'X'
S = Molecules['CO2'][iso][electronic_state]
Qf = PartFunc_Dunham(S)
print(Qf.at(T=3000))      # K
```

Nonequilibrium partition functions can also be computed with `at_noneq()`

```
print(Qf.at_noneq(Tvib=2000, Trot=1000))    # K
```

`at_noneq()` can also return the vibrational partition function and the table of rotational partition functions for each vibrational state.

2.5.5 Multi Temperature Fit

A 3 temperature fitting example . reproducing the validation case of Klarenaar 2017[1], who calculated a transmittance spectrum from the initial data of Dang 1973[2], with a 1 rotational temperature + 3 vibrational temperature (Treanor distributions) model.

CO_2 Energies are calculated from Dunham developments in an uncoupled harmonic oscillator - rigid rotor model. The example is based on one of RADIS validation cases. It makes use of the RADIS Spectrum class and the associated compare and load functions

The fitting script can be found in the radis-examples project .

[1] Klarenaar et al 2017, "Time evolution of vibrational temperatures in a CO2 glow discharge measured with infrared absorption spectroscopy" doi/10.1088/1361-6595/aa902e

[2] Dang et al 1982, "Detailed vibrational population distributions in a CO2 laser discharge as measured with a tunable diode laser" doi/10.1007/BF00694640

2.5.6 CH4 Full Spectrum Benchmark

Here we reproduce the full spectrum (0.001 - 11500 cm-1) of Methane for a `broadening_max_width` corresponding to about 50 HWHMs, as in the Benchmark case of [HAPI], Table 7, Methane_III, also featured in the [RADIS-2018] article

```
from radis import SpectrumFactory

benchmark_line_brd_ratio = 50    # "WavenumberWingHW"/HWHMs
dnu = 0.01          # step in HAPI Benchmark article
molecule = 'CH4'
wavenum_min = 0.001
wavenum_max = 11505
pressure_bar = 1.01315
T = 296
isotopes = [1, 2, 3, 4]

sf = SpectrumFactory(wavenum_min=wavenum_min,
                     wavenum_max=wavenum_max,
                     isotope=isotopes,  #'all',
                     verbose=2,
                     wstep=dnu,      # depends on HAPI benchmark.
                     cutoff=1e-23,
                     broadening_max_width=5.73,  # Corresponds to WavenumberWingHW/
↪HWHM=50 in HAPI
                     molecule=molecule,
                     optimization=None,
                     )
sf.fetch_databank('astroquery')

s = sf.eq_spectrum(Tgas=T, pressure=pressure_bar)
s.plot()
```

The comparison in terms of performance with HAPI can be found in the `radis/test/benchmark/radis_vs_hapi_CH4_full_spectrum.py` case:

```
cd radis
python radis/test/benchmark/radis_vs_hapi_CH4_full_spectrum.py
```

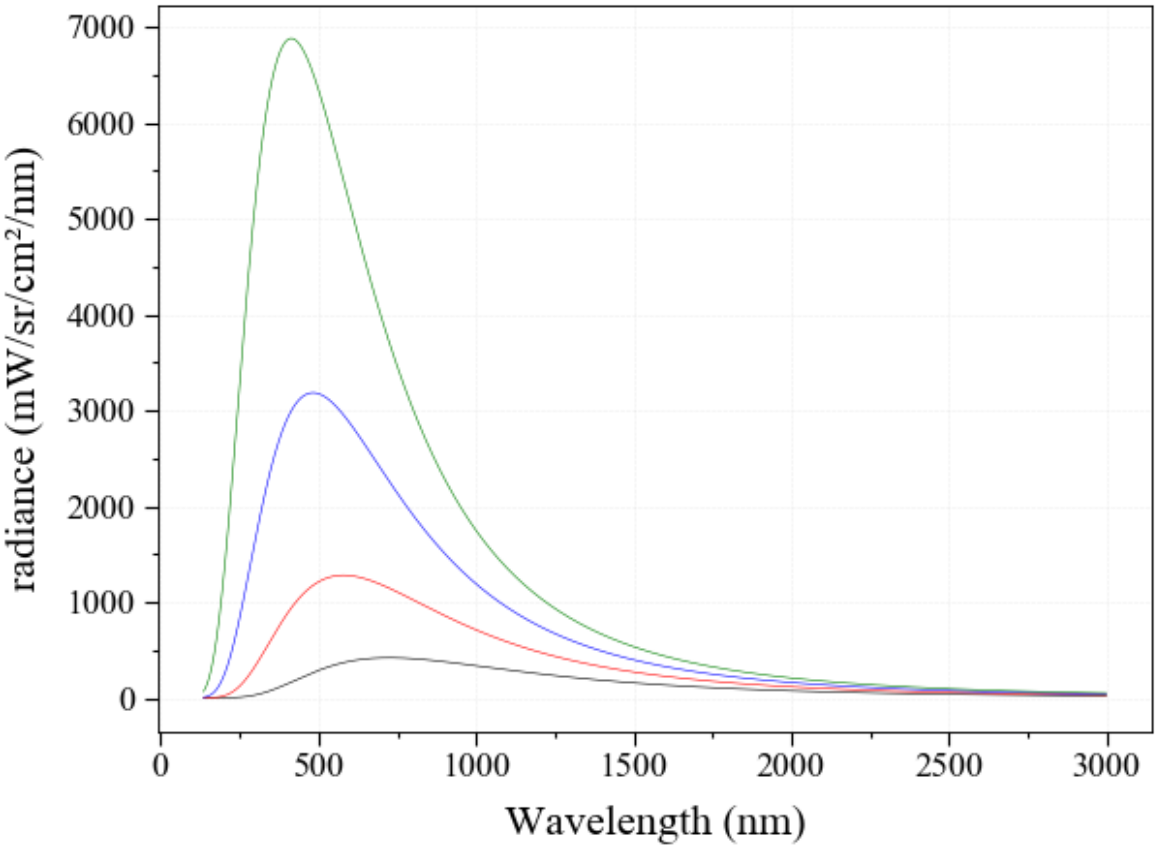
Using the different *Performance* optimisations available in RADIS, the calculation is typically 100 times faster in RADIS:

```
>>> Calculated with HAPI in 157.41s
>>> Calculated with RADIS in 1.65s
```

2.5.7 Compute Blackbody Radiation Spectrum

Compute a Planck Blackbody radiation spectrum in Python using the RADIS `sPlanck()` function

```
from radis import sPlanck

sPlanck(wavelength_min=135, wavelength_max=3000, T=4000).plot()
sPlanck(wavelength_min=135, wavelength_max=3000, T=5000).plot(nfig='same')
sPlanck(wavelength_min=135, wavelength_max=3000, T=6000).plot(nfig='same')
sPlanck(wavelength_min=135, wavelength_max=3000, T=7000).plot(nfig='same')
```



The same function can compute the Planck blackbody radiation against wavenumbers rather than wavelengths.

2.6 Example gallery

These examples show simple calculations and tasks that can be done with the RADIS package.

Other examples can be found in the dedicated radis-examples repository.

All examples can be run online, without any install required:

- Download the Jupyter notebook `ipynb` in the example page.
- start Radis-Lab, upload the Jupyter notebook, and run it from there.

Note: Radis-Lab allows you to share your session URL with colleagues. They will access the latest saved version of your notebook as long as your session is running.

2.6.1 Calculate Rovibrational Energies

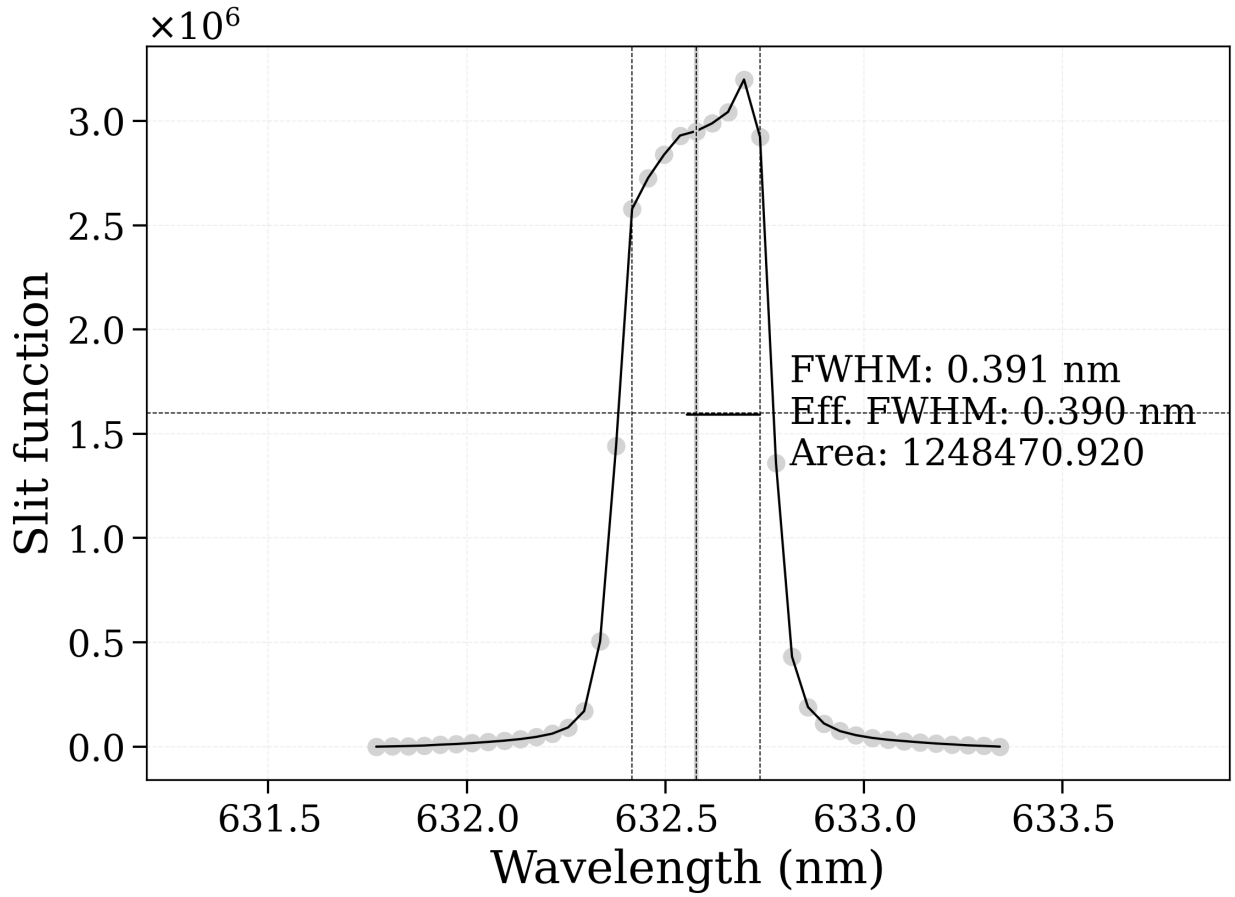
RADIS can simply be used to calculate the rovibrational energies of molecules, using the built-in *spectroscopic constants* (or your own!). See the `getMolecule()` function, and the `Molecules` list containing all `ElectronicState` objects.

```
from radis import getMolecule

# Here we get the energy of the v=6, J=3 level of the 2nd isotope of CO::

CO = getMolecule("CO", 2, "X")
print(CO.Erovib(6, 3))
```

Total running time of the script: (0 minutes 0.000 seconds)

2.6.2 Slit Function

Quickly load and plot a slit function with `plot_slit()`



```
(<Figure size 640x480 with 1 Axes>, <AxesSubplot:xlabel='Wavelength (nm)', ylabel='Slit␣
↪function'>)
```

```
from radis import plot_slit
from radis.test.utils import getTestFile

my_slit = getTestFile("slitfunction.txt")  # for the example here
plot_slit(my_slit, wunit="nm")
```

Total running time of the script: (0 minutes 1.144 seconds)

2.6.3 Partition Functions from TIPS

By default and for equilibrium calculations, RADIS calculates Partition Functions using the TIPS program ([TIPS-2020]) through [HAPI]. These partition functions can be retrieved with the `PartFuncTIPS` class:

See Also

`PartFunc_Dunham`

```
from radis.db.classes import get_molecule_identifier
from radis.levels.partfunc import PartFuncHAPI

M = get_molecule_identifier("N2O")
iso = 1

Q = PartFuncHAPI(M, iso)
print(Q.at(T=1500))
```

Total running time of the script: (0 minutes 0.000 seconds)

2.6.4 Download the HITEMP database

Database will be downloaded automatically and can be edited locally.

To compute a spectrum with the HITEMP database, see the *Calculate a HITEMP spectrum example*

By default the database is returned as a Pandas DataFrame. To explore huge databases (like CO2, CH4 or H2O) that do not fit in RAM, RADIS allows you to use a Vaex DataFrame instead (out-of-RAM). See the *Explore Database with Vaex example*

```
from radis.io.hitemp import fetch_hitemp

df = fetch_hitemp("OH")
print(df.columns)
```

```
Downloading 13_HITEMP2020.par.bz2 for OH (1/1).
Download complete. Parsing OH database to /home/docs/.radisdb/hitemp/OH-13_HITEMP2020.
↪hdf5
<frozen importlib._bootstrap>:219: RuntimeWarning: numpy.ndarray size changed, may␣
↪indicate binary incompatibility. Expected 80 from C header, got 96 from PyObject
Added HITEMP-OH database in /home/docs/radis.json
Index(['id', 'iso', 'wav', 'int', 'A', 'airbrd', 'selbrd', 'El', 'Tdpair',
       'Pshft', 'globu', 'globl', 'locu', 'locl', 'ierr', 'iref', 'lmix', 'gp',
       'gpp'],
      dtype='object')
```

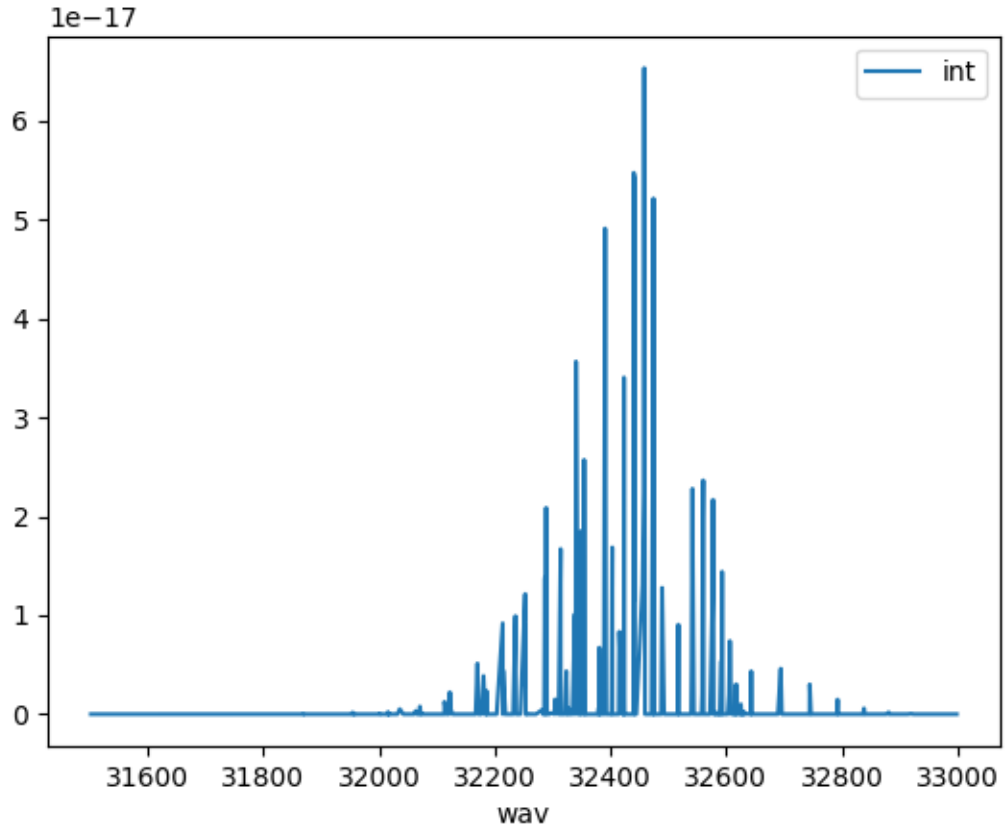
Returns:

```
Index(['id', 'iso', 'wav', 'int', 'A', 'airbrd', 'selbrd', 'El', 'Tdpair',
       'Pshft', 'globu', 'globl', 'locu', 'locl', 'ierr', 'iref', 'lmix', 'gp',
       'gpp'],
      dtype='object')
```

Columns are described in `columns_2004`

A specific can be retrieved with the same `fetch_hitemp()` function. The already downloaded database will be used:

```
df = fetch_hitemp("OH", load_wavenum_min=31500, load_wavenum_max=33000, isotope="1")
df.plot("wav", "int")
```



```
<AxesSubplot:xlabel='wav'>
```

Total running time of the script: (0 minutes 2.983 seconds)

2.6.5 Cite all references used

RADIS is built on the shoulders of many state-of-the-art packages and databases. If using RADIS for your work, **cite all of them that made it possible**.

Starting from 0.9.30, you can retrieve the bibtex entries of all papers and references that contribute to the calculation of a Spectrum, with the `cite()` method.

Example

Below, we compute a non-equilibrium spectrum. The `cite()` method returns the computational algorithm used, the line database, the spectroscopic constants used to compute rovibrational energies, and the data retrieval tools.

See Also

`RefTracker`

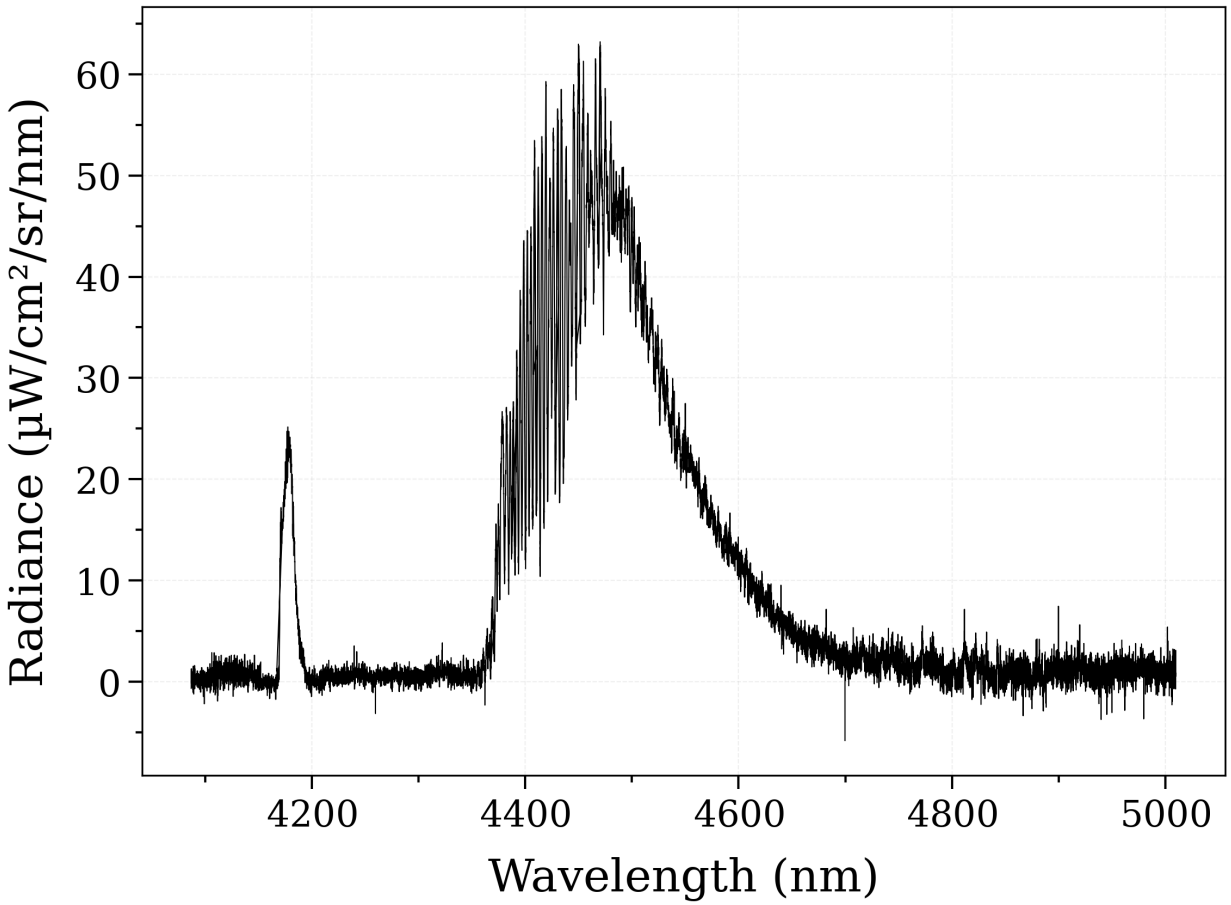
```
from radis import calc_spectrum

s = calc_spectrum(
    1900,
    2300,  # cm-1
    molecule="CO",
    isotope="1,2,3",
    pressure=1.01325,  # bar
    Tvib=2000,  #
    Trot=300,
    mole_fraction=0.1,
    path_length=1,  # cm
    databank="hitran",
)

s.cite()
```

Total running time of the script: (0 minutes 0.000 seconds)

2.6.6 Load an experimental spectrum

Load an experimental spectrum stored as `.spec` (with units and metadata) using `load_spec()` (we could also have used `plot_spec()` directly !)

See more loading and post-processing functions on the *Spectrum page*.




```
Spectrum Name:  CO2_measured_spectrum_4-5um.spec
Spectral Quantities
-----------------------------------------
   radiance     [µW/cm2/sr/nm]  (18,400 points)
Physical Conditions
-----------------------------------------
Computation Parameters
-----------------------------------------
   filename             12_StepAndGlue_30us_Cathode_30us_avg_spectra_stacked.txt
   filename_created     04/08/2017
   waveunit             nm
Config parameters
-----------------------------------------
-----------------------------------------

25.161498577151097 uW / (cm2 nm sr)
```

```
from radis.test.utils import getTestFile
```

(continues on next page)

(continued from previous page)

```
from radis.tools.database import load_spec

my_file = getTestFile("CO2_measured_spectrum_4-5um.spec")  # for the example here

s = load_spec(my_file)
s.plot()

# Print all metadata:
print(s)

# Or retrieve an information :
print(s.crop(4160, 4200, "nm").max())
```

Total running time of the script: (0 minutes 0.725 seconds)

2.6.7 Line Survey

Plot details of every single line in a spectrum.

Uses the `line_survey()` function.

```
HAPI version: 1.2.2.0
To get the most up-to-date version please check http://hitran.org/hapi
ATTENTION: Python versions of partition sums from TIPS-2021 are now available in HAPI␣
↪code

           MIT license: Copyright 2021 HITRAN team, see more at http://hitran.org.

           If you use HAPI in your research or software development,
           please cite it using the following reference:
           R.V. Kochanov, I.E. Gordon, L.S. Rothman, P. Wcislo, C. Hill, J.S. Wilzewski,
           HITRAN Application Programming Interface (HAPI): A comprehensive approach
           to working with spectroscopic data, J. Quant. Spectrosc. Radiat. Transfer 177,
↪ 15-30 (2016)
           DOI: 10.1016/j.jqsrt.2016.03.005

           ATTENTION: This is the core version of the HITRAN Application Programming␣
↪Interface.
                      For more efficient implementation of the absorption coefficient␣
↪routine,
                      as well as for new profiles, parameters and other functional,
                      please consider using HAPI2 extension library.
                      HAPI2 package is available at http://github.com/hitranonline/hapi2

Using /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2


Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO2_1
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_1.header
END DOWNLOAD
                     Lines parsed: 174446
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO2_2
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_2.header
END DOWNLOAD
                     Lines parsed: 69870
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO2_3
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_3.header
END DOWNLOAD
```

(continues on next page)

(continued from previous page)

```
                     Lines parsed: 122140
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO2_4
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
```

(continues on next page)

(continued from previous page)

```
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
```

(continues on next page)

(continued from previous page)

```
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
```

(continues on next page)

(continued from previous page)

```
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_4.header
END DOWNLOAD
                     Lines parsed: 73939
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO2_5
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
```

(continues on next page)

(continued from previous page)

```
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
 65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_5.header
END DOWNLOAD
                     Lines parsed: 41058
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO2_6
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_6.header
END DOWNLOAD
                     Lines parsed: 23607
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO2_7
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_7.header
END DOWNLOAD
                     Lines parsed: 10498
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO2_8
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
```

(continues on next page)

(continued from previous page)

```
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_8.header
END DOWNLOAD
                     Lines parsed: 14623
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO2_9
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.
↪data
```

(continues on next page)

(continued from previous page)

```
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO2/CO2_9.header
END DOWNLOAD
                    Lines parsed: 6493
PROCESSED
Added HITRAN-CO2 database in /home/docs/radis.json
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: HighTemperatureWarning: HITRAN is valid for low␣
↪temperatures (typically < 700 K). For higher temperatures you may need HITEMP or CDSD.␣
↪See the 'databank=' parameter
  warnings.warn(WarningType(message))
Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 1500 K
   Trot                 1500 K
   Tvib                 1500 K
   isotope              1
   mole_fraction        0.0004
   molecule             CO2
   overpopulation       None
   path_length          100 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2400.0000 cm-1
   wavenum_min          2380.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             hitran
   dbpath               /home/docs/.radisdb/hitran/CO2.hdf5
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           50 cm-1
   waveunit             cm-1
   wstep                0.01 cm-1
   zero_padding         -1
----------------------------------------
0.04s - Spectrum calculated
288 lines
```

(continues on next page)

(continued from previous page)

```
Figure({
    'data': [{'name': 'linestrength',
              'text': array(['CO2 [Q32](11$^{1}$0 r=2)->(03$^{3}$1 r=1)<br>int (intensity at␣
↪296K): 2.12e-29  [cm-1/(molecule/cm-2)]<br>A (Einstein A coefficient): 3.62e-05  [s-1]
↪<br>El (lower-state energy): 2.35e+03  [cm-1]<br>s.lines index: 0',
                             'CO2 [P52](10$^{0}$0 r=2)->(10$^{0}$1 r=1)<br>int (intensity at␣
↪296K): 7.49e-26  [cm-1/(molecule/cm-2)]<br>A (Einstein A coefficient): 0.0867  [s-1]
↪<br>El (lower-state energy): 2.36e+03  [cm-1]<br>s.lines index: 1',
                             'CO2 [Q41](11$^{1}$0 r=2)->(03$^{3}$1 r=1)<br>int (intensity at␣
↪296K): 1.13e-29  [cm-1/(molecule/cm-2)]<br>A (Einstein A coefficient): 5.33e-05  [s-1]
↪<br>El (lower-state energy): 2.6e+03  [cm-1]<br>s.lines index: 2',
                             ...,
                             'CO2 [P34](20$^{0}$0 r=2)->(20$^{0}$1 r=1)<br>int (intensity at␣
↪296K): 8.86e-28  [cm-1/(molecule/cm-2)]<br>A (Einstein A coefficient): 0.0691  [s-1]
↪<br>El (lower-state energy): 3.13e+03  [cm-1]<br>s.lines index: 329',
                             'CO2 [P5](10$^{0}$1 r=2)->(10$^{0}$2 r=1)<br>int (intensity at 296K):␣
↪3.12e-29  [cm-1/(molecule/cm-2)]<br>A (Einstein A coefficient): 0.196  [s-1]<br>El␣
↪(lower-state energy): 3.62e+03  [cm-1]<br>s.lines index: 330',
                             'CO2 [P32](20$^{0}$0 r=3)->(20$^{0}$1 r=2)<br>int (intensity at␣
↪296K): 2.21e-27  [cm-1/(molecule/cm-2)]<br>A (Einstein A coefficient): 0.0789  [s-1]
↪<br>El (lower-state energy): 2.96e+03  [cm-1]<br>s.lines index: 331'],
                            dtype=object),
              'type': 'bar',
              'width': 0.20010000000436776,
              'x': array([2380.016336, 2380.081298, 2380.114148, ..., 2399.778997, 2399.
↪900272,
                          2399.962486]),
              'y': array([5.52334543e-27, 2.06836810e-23, 8.10163669e-27, ..., 5.
↪02840796e-24,
                          1.19565901e-24, 6.37839163e-24])},
             {'name': 'radiance',
              'type': 'scatter',
              'x': array([2380.  , 2380.01, 2380.02, ..., 2399.99, 2400.  , 2400.01]),
              'y': array([nan, nan, nan, ..., nan, nan, nan]),
              'yaxis': 'y2'}],
    'layout': {'hovermode': 'closest',
               'showlegend': False,
               'template': '...',
               'title': {'text': 'Line Survey'},
               'xaxis': {'range': [2380.0, 2400.010000000437], 'title': {'text':
↪'Wavenumber (cm-1)'}},
               'yaxis': {'range': [-27, -17],
                         'tickfont': {'color': '#1f77b4'},
                         'title': {'font': {'color': '#1f77b4'}, 'text': 'Linestrength␣
↪(cm-1/(molecule/cm-2))'},
                         'type': 'log'},
               'yaxis2': {'overlaying': 'y',
                          'side': 'right',
                          'tickfont': {'color': '#ff7f0e'},
                          'title': {'font': {'color': '#ff7f0e'}, 'text': 'Radiance (mW/
↪cm2/sr/cm-1)'},
```

(continues on next page)

(continued from previous page)

```
                            'type': 'log'}}
})
```

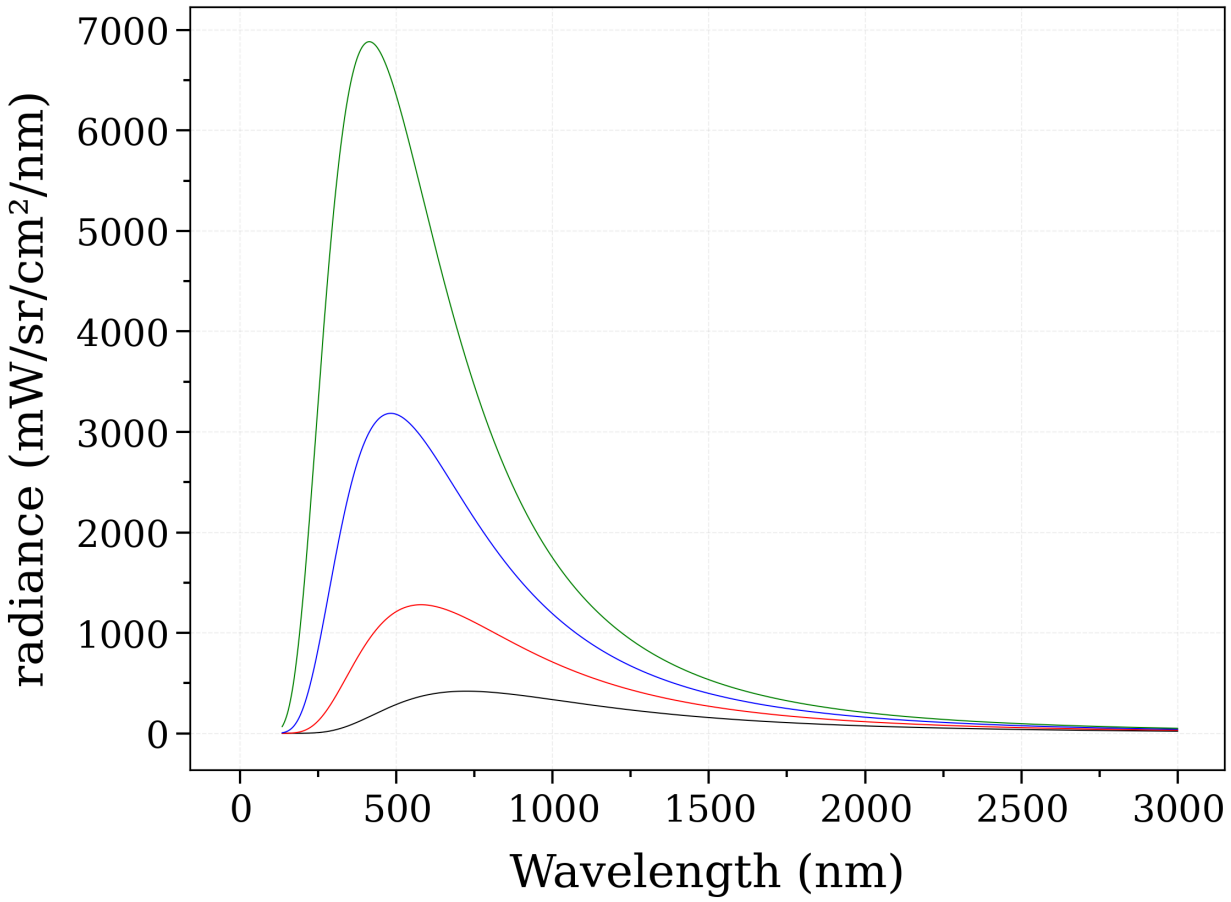
```
from radis import calc_spectrum

s = calc_spectrum(
    wavenum_min=2380,
    wavenum_max=2400,
    mole_fraction=400e-6,
    path_length=100,  # cm
    Tgas=1500,
    molecule="CO2",
    isotope=[1],
    databank="hitran",
    export_lines=True,
)
s.apply_slit(2, "nm")
s.line_survey(overlay="radiance", barwidth=0.01)
```

Total running time of the script: (0 minutes 46.882 seconds)

2.6.8 Blackbody radiation

Compute Planck's blackbody emission and return a RADIS `Spectrum` object for easier post-processing.

Uses `sPlanck`.




```
findfont: Font family ['sans-serif'] not found. Falling back to DejaVu Sans.
findfont: Generic family 'sans-serif' not found because none of the following families␣
↪were found: Helvetica, Bitstream Vera Sans, Lucida Grande, Verdana, Geneva, Lucid,␣
↪Arial, Avant Garde, sans-serif

<matplotlib.lines.Line2D object at 0x7f848c136190>
```

```
from radis.phys.blackbody import sPlanck

sPlanck(wavelength_min=135, wavelength_max=3000, T=4000).plot()
sPlanck(wavelength_min=135, wavelength_max=3000, T=5000).plot(nfig="same")
sPlanck(wavelength_min=135, wavelength_max=3000, T=6000).plot(nfig="same")
sPlanck(wavelength_min=135, wavelength_max=3000, T=7000).plot(nfig="same")
```

Total running time of the script: (0 minutes 1.204 seconds)

2.6.9 Partition Functions from spectroscopic constants

RADIS can calculate equilibrium and non-LTE Partition Functions from a given set of spectroscopic constants using a Dunham expansion.

Calculations use the the `PartFunc_Dunham` class

Default spectroscopic constants and spectroscopic models used are given in *default spectroscopic constants*. You can also use your own set of spectroscopic constants.

See Also

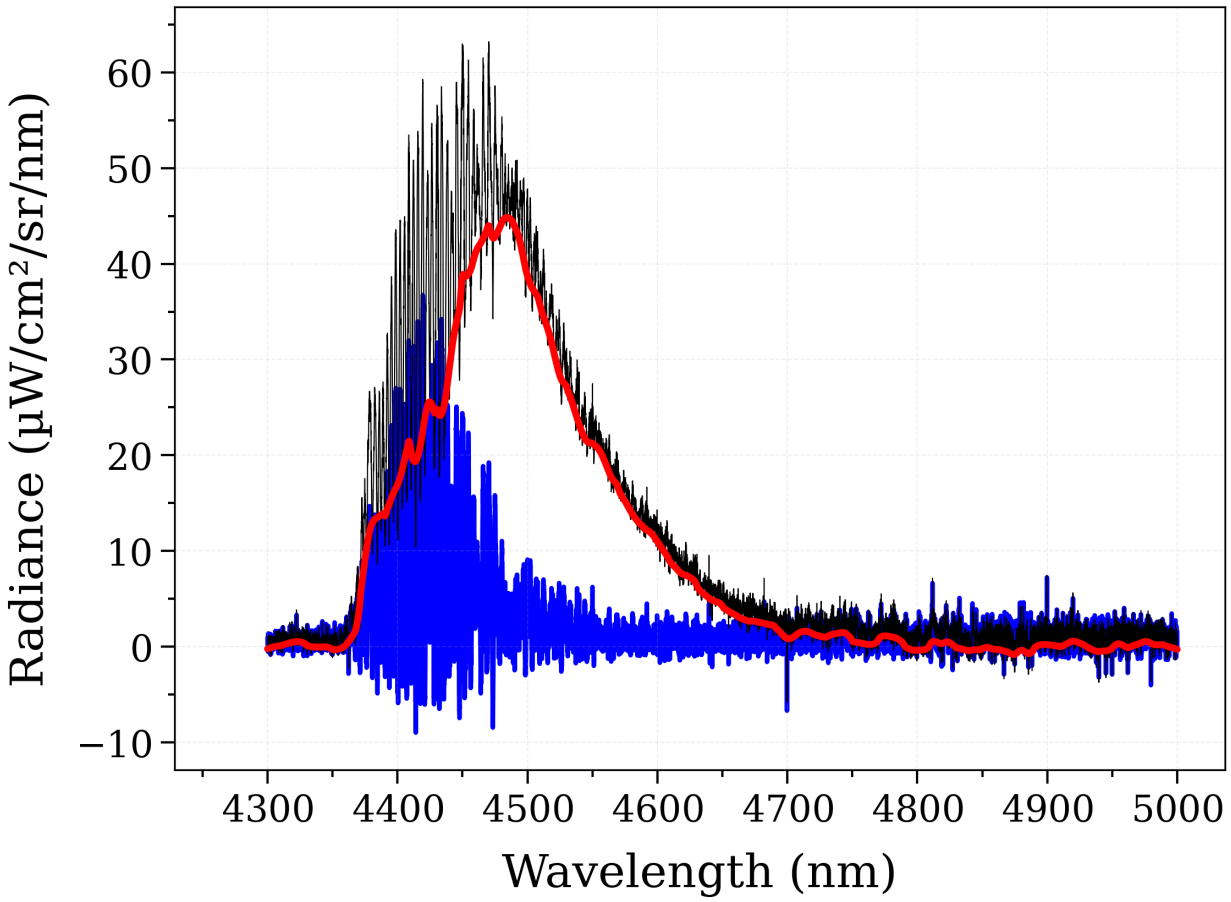
`PartFuncHAPI`

```
from radis.db.molecules import Molecules
from radis.levels.partfunc import PartFunc_Dunham

isotope = 1
electronic_state = "X"
S = Molecules["CO"][isotope][electronic_state]

# Equilibrium partition functions :
Qf = PartFunc_Dunham(S)
print(Qf.at(T=3000))  # K

# Nonequilibrium partition functions :
print(Qf.at_noneq(Tvib=2000, Trot=1000))  # K
```

Total running time of the script: (0 minutes 0.000 seconds)

2.6.10 Remove a baseline

Remove a baseline from an experimental spectrum :




```
<matplotlib.lines.Line2D object at 0x7f84aa8bd280>
```

```
# Get a spectrum:
from radis import load_spec
from radis.test.utils import getTestFile

s_exp = (
    load_spec(getTestFile(r"CO2_measured_spectrum_4-5um.spec"), binary=True)
    .crop(4300, 5000)
    .sort()
)


sb = s_exp.get_baseline(algorithm="als")

s_exp.plot()
sb.plot(nfig="same", lw=3)

# Plot the spectrum minus the baseline :
```

(continues on next page)

(continued from previous page)

```
s = s_exp - sb
s.plot(nfig="same", lw=2, zorder=0)
```

Total running time of the script: (0 minutes 1.275 seconds)

2.6.11 Explore Line Database Parameters

Database will be downloaded automatically and can be edited locally.

The *Download HITEMP Database example* showed how to download the HITEMP database under Pandas format.

RADIS can also be used to explore and visualize larger databases, using the underlying `vaex` library.

```
from radis.io.hitemp import fetch_hitemp

df = fetch_hitemp("CO2", output="vaex", load_wavenum_min=2150, load_wavenum_max=2450)
print(f"{len(df)} lines in HITEMP CO2; 2150 - 2450 cm-1")
```

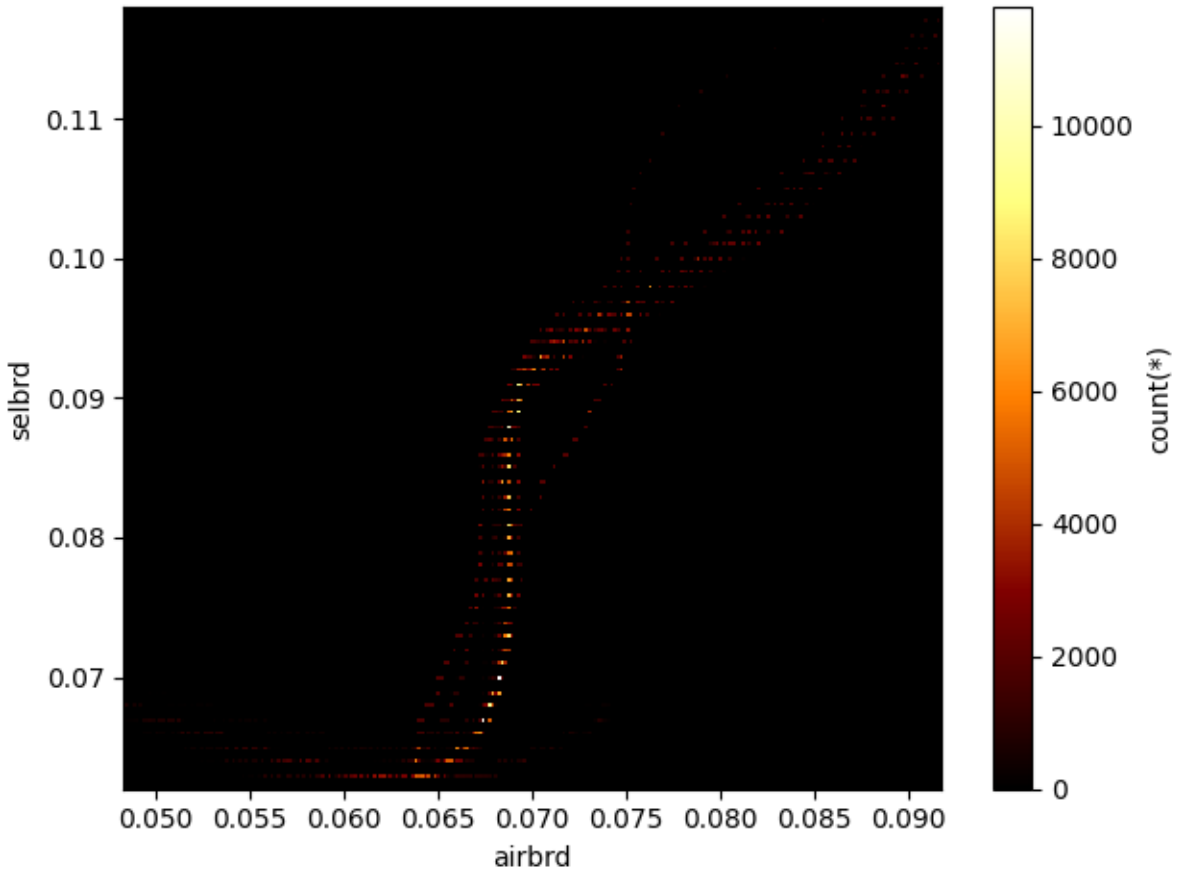
```
HITEMP keep only relevant input files: ['/home/docs/.radisdb/hitemp/CO2-02_02125-02250_
↪HITEMP2010.hdf5', '/home/docs/.radisdb/hitemp/CO2-02_02250-02500_HITEMP2010.hdf5']
HITEMP keep only relevant input files: ['/home/docs/.radisdb/hitemp/CO2-02_02125-02250_
↪HITEMP2010.hdf5', '/home/docs/.radisdb/hitemp/CO2-02_02250-02500_HITEMP2010.hdf5']
Downloading 02_02125-02250_HITEMP2010.zip for CO2 (1/2).
Download complete. Parsing CO2 database to /home/docs/.radisdb/hitemp/CO2-02_02125-02250_
↪HITEMP2010.hdf5

Downloading 02_02250-02500_HITEMP2010.zip for CO2 (2/2).
Download complete. Parsing CO2 database to /home/docs/.radisdb/hitemp/CO2-02_02250-02500_
↪HITEMP2010.hdf5
Added HITEMP-CO2 database in /home/docs/radis.json
HITEMP keep only relevant input files: ['/home/docs/.radisdb/hitemp/CO2-02_02125-02250_
↪HITEMP2010.hdf5', '/home/docs/.radisdb/hitemp/CO2-02_02250-02500_HITEMP2010.hdf5']
1355761 lines in HITEMP CO2; 2150 - 2450 cm-1
```

Note the use of `output='vaex'` in `fetch_hitemp()` above. The returned DataFrame is a Vaex DataFrame. Loading times takes only few tens of milliseconds even for the largest HITEMP or ExoMol databases

We can also use Vaex graph functions. See Vaex vizualisations : https://vaex.readthedocs.io/en/latest/guides/advanced_plotting.html#

For instance, we plot a `heatmap()` showing Self and Air-broadening coefficients

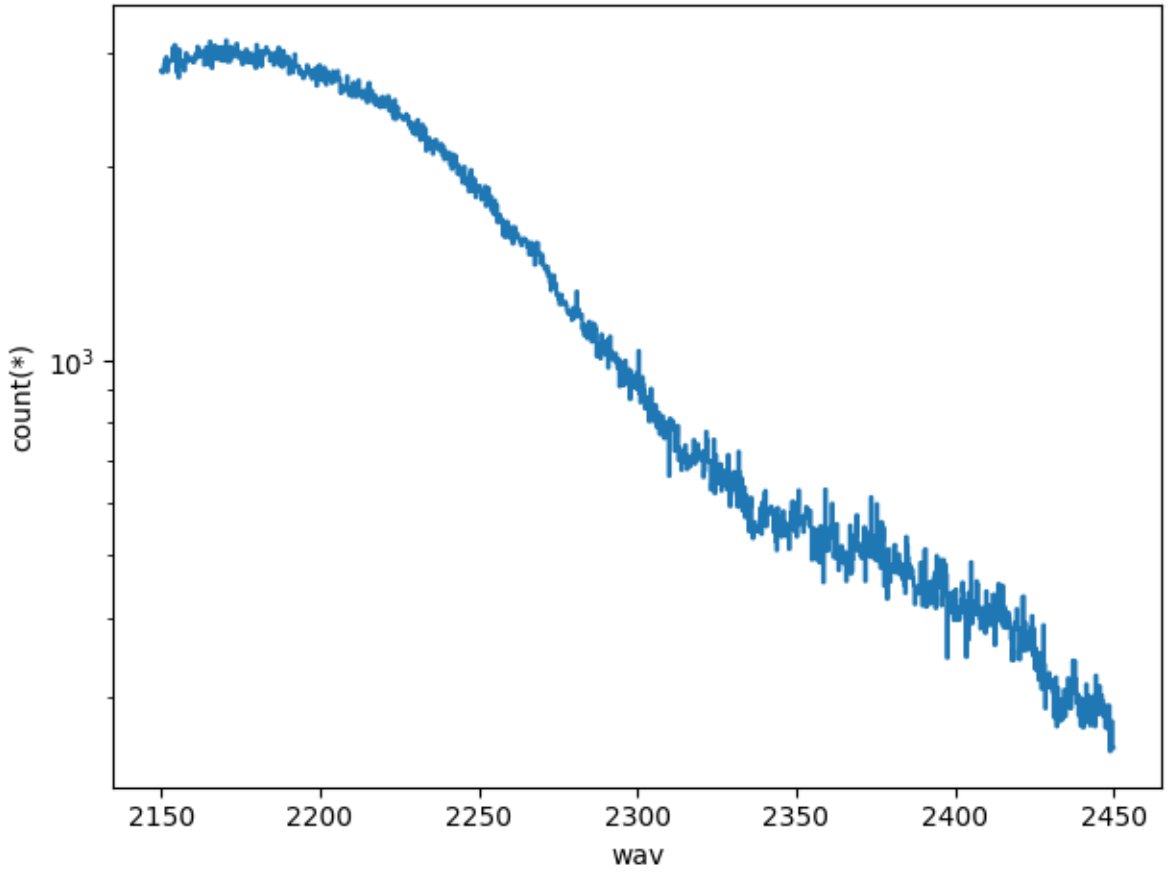
```
import matplotlib.pyplot as plt

plt.figure()
df.viz.heatmap("airbrd", "selbrd", limits="99%")

# TODO / idea : compare CO2 broadening with Air broadening for HITRAN database ?
```




```
<matplotlib.image.AxesImage object at 0x7f846f24c220>
```

Or below we plot the number of lines using Vaex's `histogram()`

```
plt.figure()
df.viz.histogram("wav", shape=1000)
plt.yscale("log")
```



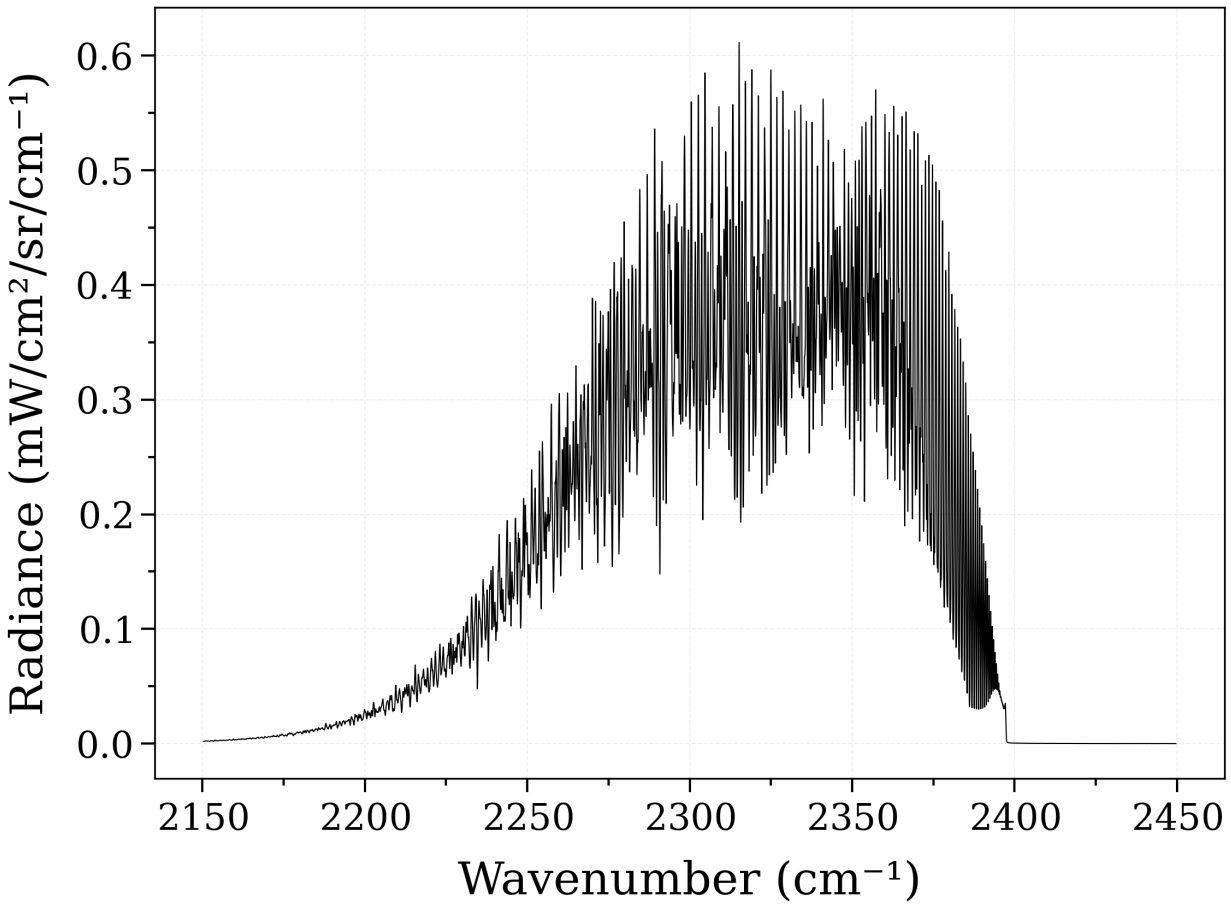
Total running time of the script: (1 minutes 12.068 seconds)

2.6.12 GPU Accelerated Spectra

Example using GPU calculation with `eq_spectrum_gpu()`

This method requires CUDA compatible hardware to execute. For more information on how to setup your system to run GPU-accelerated methods using CUDA and Cython, check GPU Spectrum Calculation on RADIS

Note: in the example below, the GPU code runs on CPU, using the parameter `emulate=True`. In your environment, to run the GPU code with the full power of the GPU, remove this line or set `emulate=False` (default)




```
HITEMP keep only relevant input files: ['/home/docs/.radisdb/hitemp/CO2-02_02125-02250_
↪HITEMP2010.hdf5', '/home/docs/.radisdb/hitemp/CO2-02_02250-02500_HITEMP2010.hdf5']
HITEMP keep only relevant input files: []
HITEMP keep only relevant input files: ['/home/docs/.radisdb/hitemp/CO2-02_02125-02250_
↪HITEMP2010.hdf5', '/home/docs/.radisdb/hitemp/CO2-02_02250-02500_HITEMP2010.hdf5']
Number of lines loaded: 1128265

Finished calculating spectrum!
1.24s - Spectrum calculated
Spectrum Name:  CO2-hitemp-radisdb-1100.0K-#5456
Spectral Quantities
----------------------------------------
   abscoeff     [cm-1]  (150,001 points)
   absorbance   (150,001 points)
   emissivity   (150,001 points)
   emissivity_noslit    (150,001 points)
   transmittance_noslit         (150,001 points)
   radiance_noslit      [mW/cm2/sr/cm-1]        (150,001 points)
   transmittance        (150,001 points)
Physical Conditions
----------------------------------------
   Tgas                 1100.0 K
   Trot                 1100.0 K
```

(continues on next page)

(continued from previous page)

```
   Tvib                 1100.0 K
   isotope              1,2,3
   mole_fraction        0.8
   molecule             CO2
   overpopulation       None
   path_length          0.2 cm
   pressure_mbar        1000.0 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   thermal_equilibrium  True
   vib_distribution     boltzmann
   wavenum_max          2450.0000 cm-1
   wavenum_min          2150.0000 cm-1
Computation Parameters
----------------------------------------
   NwG                  3
   NwL                  6
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               0 cm-1/(#.cm-2)
   dbformat             hitemp-radisdb
   dbpath               /home/docs/.radisdb/hitemp/CO2-02_02125-02250_HITEMP2010.hdf5,/
↪home/docs/.radisdb/hitemp/CO2-02_0225...
   default_output_unit  cm-1
   emulate_gpu          True
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   profiler             {'spectrum_calculation': {'scaled_S0': 0.0367154770065099, 'calc_
↪other_spectral_quan': 0.01340713699...
   pseudo_continuum_threshold  0
   radis_version        0.13.1
   sparse_ldm           auto
   spectral_points      150000.0
   truncation           50 cm-1
   waveunit             cm-1
   wstep                0.002 cm-1
   zero_padding         -1
Config parameters
----------------------------------------
   DEFAULT_DOWNLOAD_PATH  ~/.radisdb
   GRIDPOINTS_PER_LINEWIDTH_ERROR_THRESHOLD  1
   GRIDPOINTS_PER_LINEWIDTH_WARN_THRESHOLD  3
   SPARSE_WAVERANGE     auto
Information
----------------------------------------
```

(continues on next page)

(continued from previous page)

```
   calculation_time     1.2341494829888688 s
   chunksize            None
   db_use_cached        True
   dxG                  0.13753507880165727
   dxL                  0.20180288881201608
   export_lines         False
   export_populations   None
   export_rovib_fraction  False
   levelsfmt            radis
   lines_calculated     1,128,265
   load_energies        False
   lvl_use_cached       True
   parfuncpath          None
   total_lines          1128265
   warning_broadening_threshold  0.01
   warning_linestrength_cutoff  0.01
   wavenum_max_calc     2450.0000 cm-1
   wavenum_min_calc     2150.0000 cm-1
----------------------------------------


<matplotlib.lines.Line2D object at 0x7f8462569970>
```

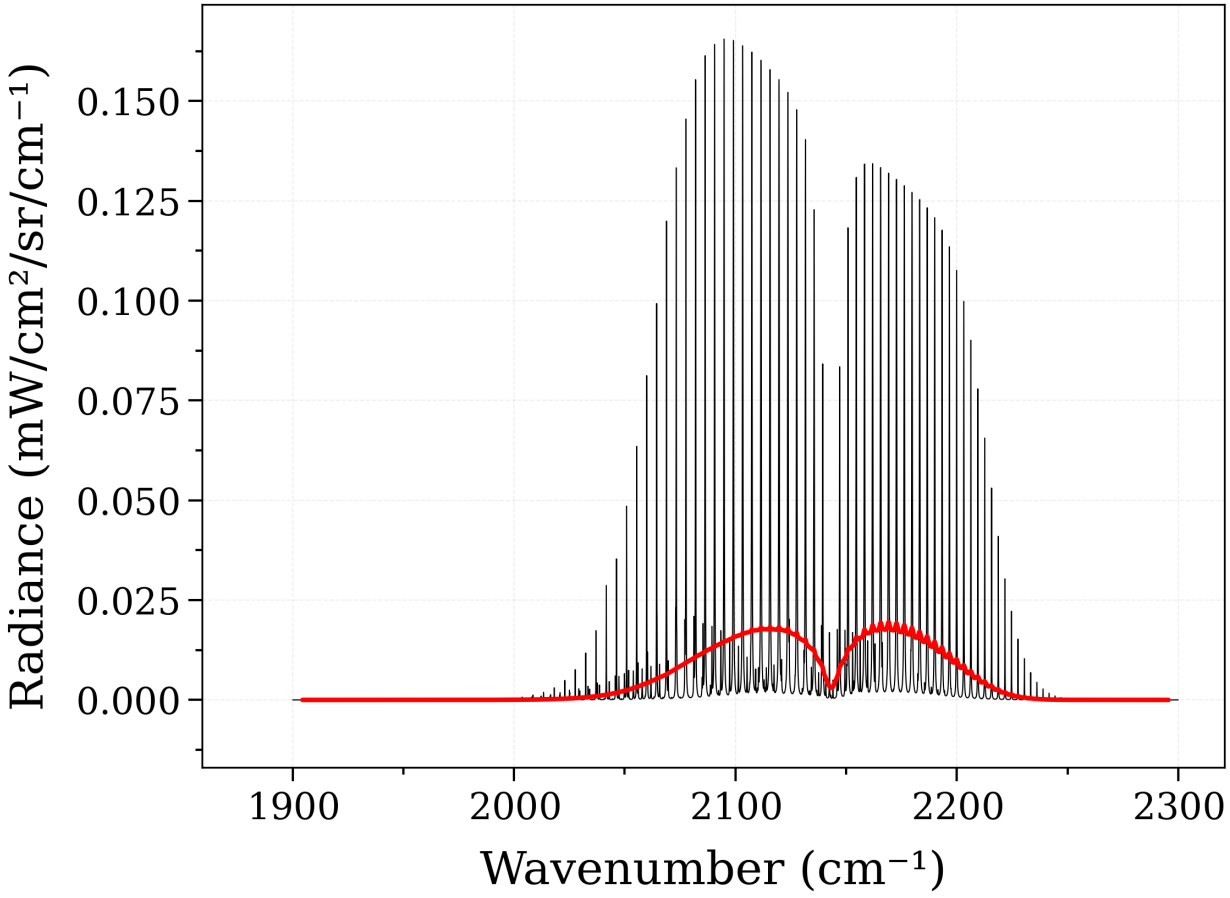
```
from radis import SpectrumFactory

sf = SpectrumFactory(
    2150,
    2450,  # cm-1
    molecule="CO2",
    isotope="1,2,3",
    wstep=0.002,
)

sf.fetch_databank("hitemp")

s = sf.eq_spectrum_gpu(
    Tgas=1100.0,  # K
    pressure=1,  # bar
    mole_fraction=0.8,
    path_length=0.2,  # cm
    emulate=True,  # if True, runs CPU code on GPU. Set to False or remove to run on the␣
↪GPU
)
print(s)

s.apply_slit(0.5)  # cm-1
s.plot("radiance", show=True)
```

Total running time of the script: (0 minutes 2.408 seconds)

2.6.13 Calculate non-LTE spectra of carbon-monoxide

Compute a CO spectrum with the temperature of the vibrational mode different from the temperature of the rotational mode.

This example uses the `calc_spectrum()` function, the [HITRAN-2016] line database to derive the line positions and intensities, and the default RADIS spectroscopic constants to compute nonequilibrium energies and populations, but it can be extended to other line databases and other sets of spectroscopic constants.




```
Using /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO


Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO_1
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_1.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_1.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_1.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_1.header
END DOWNLOAD
                    Lines parsed: 1344
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO_2
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_2.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_2.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_2.header
END DOWNLOAD
                    Lines parsed: 1042
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO_3
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_3.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_3.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_3.header
END DOWNLOAD
                    Lines parsed: 920
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO_4
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_4.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_4.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_4.header
END DOWNLOAD
                    Lines parsed: 800
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO_5
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_5.
↪data
```

(continues on next page)

(continued from previous page)

```
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_5.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_5.header
END DOWNLOAD
                     Lines parsed: 674
PROCESSED

Data is fetched from http://hitran.org

BEGIN DOWNLOAD: CO_6
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_6.
↪data
  65536 bytes written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_6.
↪data
Header written to /home/docs/.radisdb/hitran/downloads__can_be_deleted/CO/CO_6.header
END DOWNLOAD
                     Lines parsed: 601
PROCESSED
Added HITRAN-CO database in /home/docs/radis.json
Calculating energy levels with Dunham expansion for CO(X1+)(iso1)
Database generated up to v=48, J=238
Calculating energy levels with Dunham expansion for CO(X1+)(iso2)
Database generated up to v=48, J=243
Calculating energy levels with Dunham expansion for CO(X1+)(iso3)
Database generated up to v=48, J=243
Calculating Non-Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 300.0 K
   Trot                 300.0 K
   Tvib                 700.0 K
   isotope              1,2,3
   mole_fraction        0.1
   molecule             CO
   path_length          1.0 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2300.0000 cm-1
   wavenum_min          1900.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             hitran
   dbpath               /home/docs/.radisdb/hitran/CO.hdf5
   folding_thresh       1e-06
   include_neighbouring_lines  True
```

(continues on next page)

(continued from previous page)

```
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           50 cm-1
   waveunit             cm-1
   wstep                0.01 cm-1
   zero_padding         -1
----------------------------------------
Fetching Evib & Erot.
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: NegativeEnergiesWarning:

There are negative rotational energies in the database

/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: PerformanceWarning:

'gu' was recomputed although 'gp' already in DataFrame. All values are equal

... sorting lines by vibrational bands
... lines sorted in 0.0s
0.19s - Spectrum calculated

<matplotlib.lines.Line2D object at 0x7f848b3bdd30>
```

```
from astropy import units as u

from radis import calc_spectrum

s2 = calc_spectrum(
    1900 / u.cm,
    2300 / u.cm,
    molecule="CO",
    isotope="1,2,3",
    pressure=1.01325 * u.bar,
    Tvib=700 * u.K,
    Trot=300 * u.K,
    mole_fraction=0.1,
    path_length=1 * u.cm,
    databank="hitran",  # or use 'hitemp'
)
s2.plot("radiance_noslit")
```

(continues on next page)

(continued from previous page)

```
# Apply a (large) instrumental slit function :
s2.apply_slit(10, "nm")
s2.plot("radiance", nfig="same", lw=2)  # compare with previous
```

Total running time of the script: (0 minutes 4.418 seconds)

2.6.14 Use different plot themes

Shows how to customize plot styles, using `seaborn` , `matplotlib` or publib

Examples

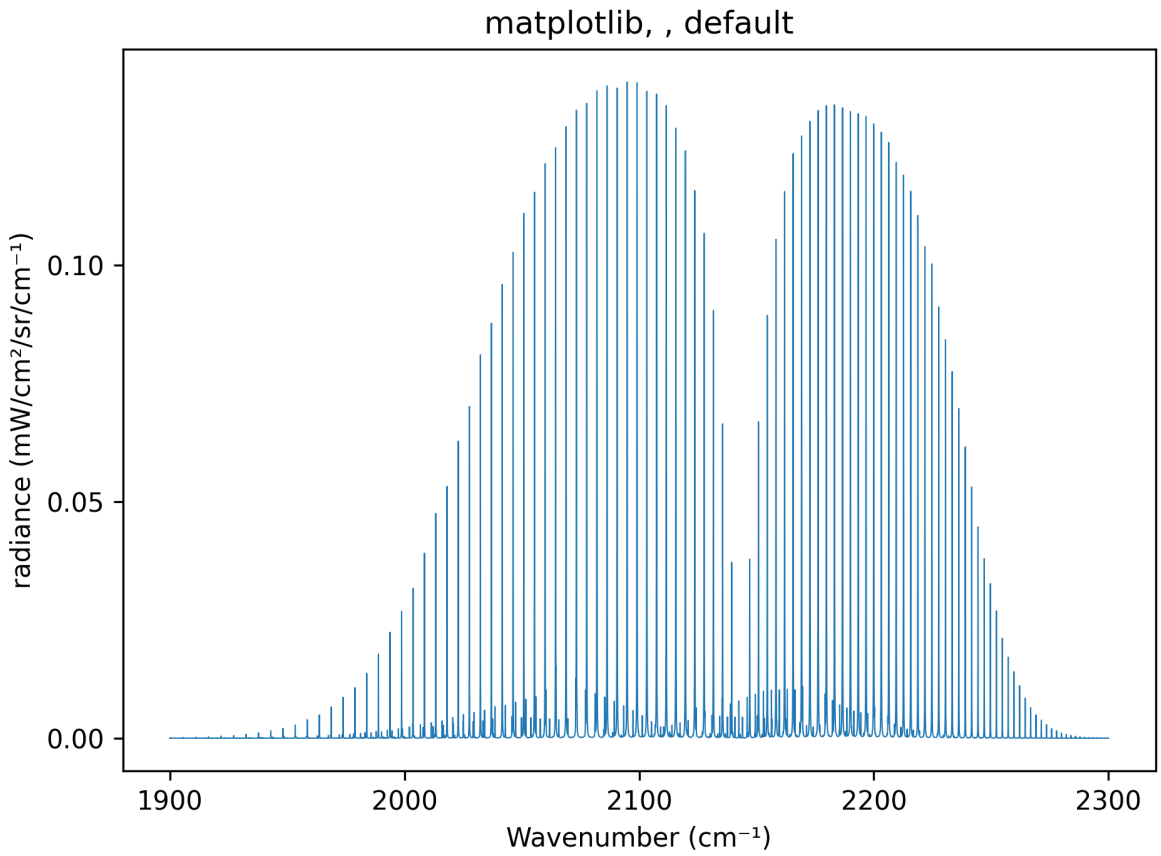
To change it in your user script, set the keys of the `"plot"` bloc in `radis.config` :

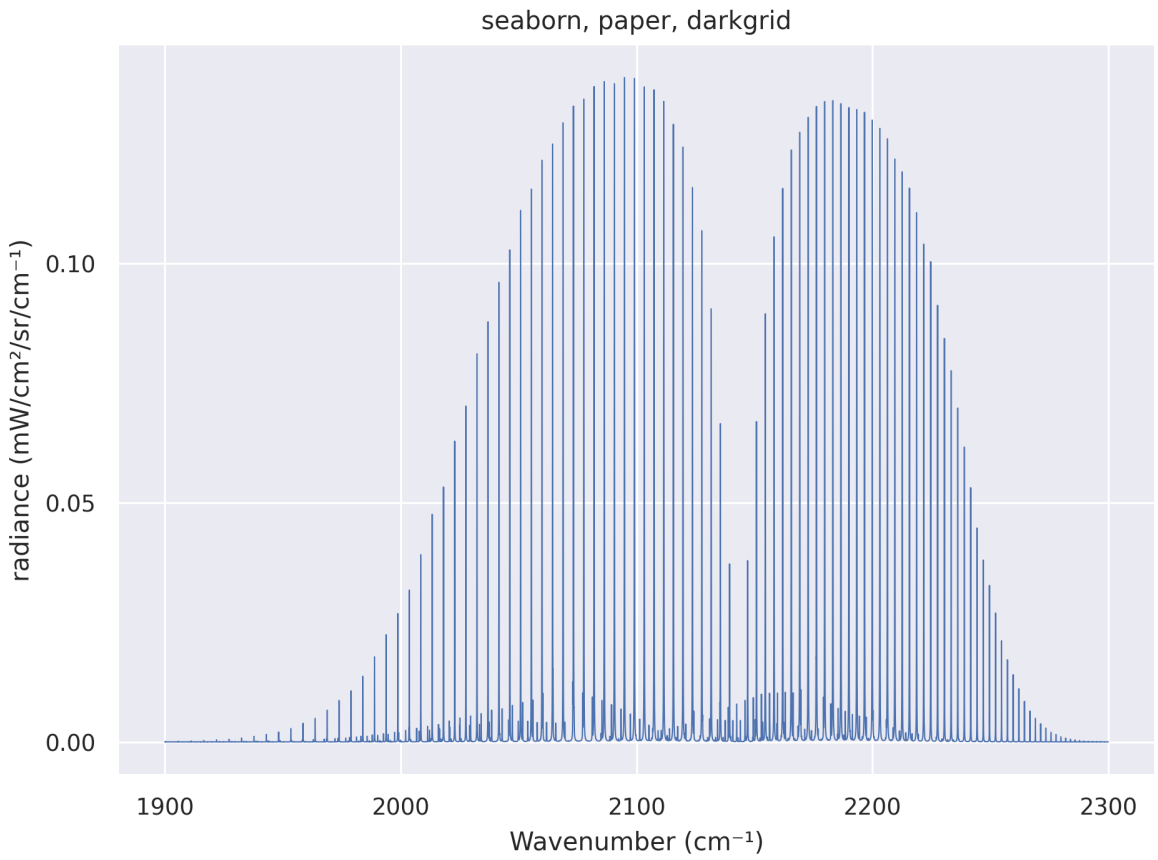
```
import radis
radis.config["plot"]["plotlib"] = "seaborn"
radis.config["plot"]["context"] = "paper"
radis.config["plot"]["style"] = "darkgrid"
```

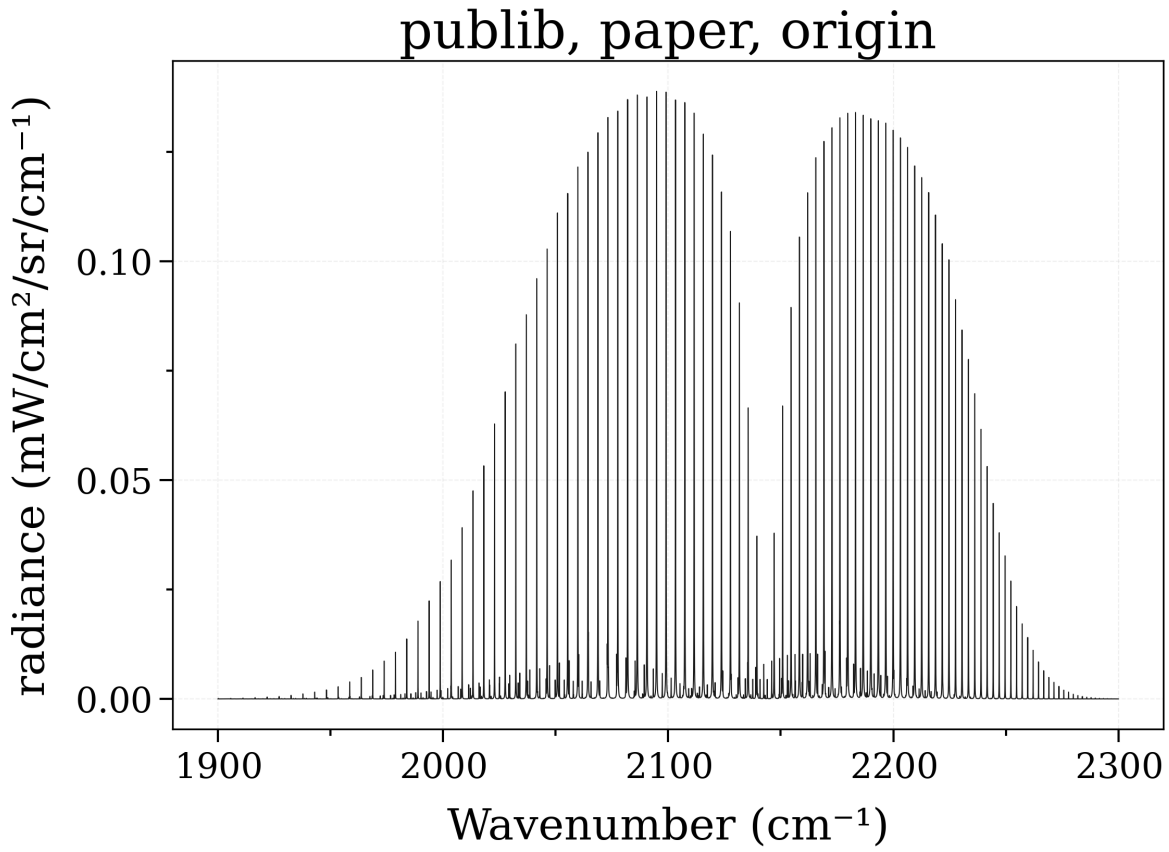
To change your default settings, edit the `~/radis.json` *Configuration file*

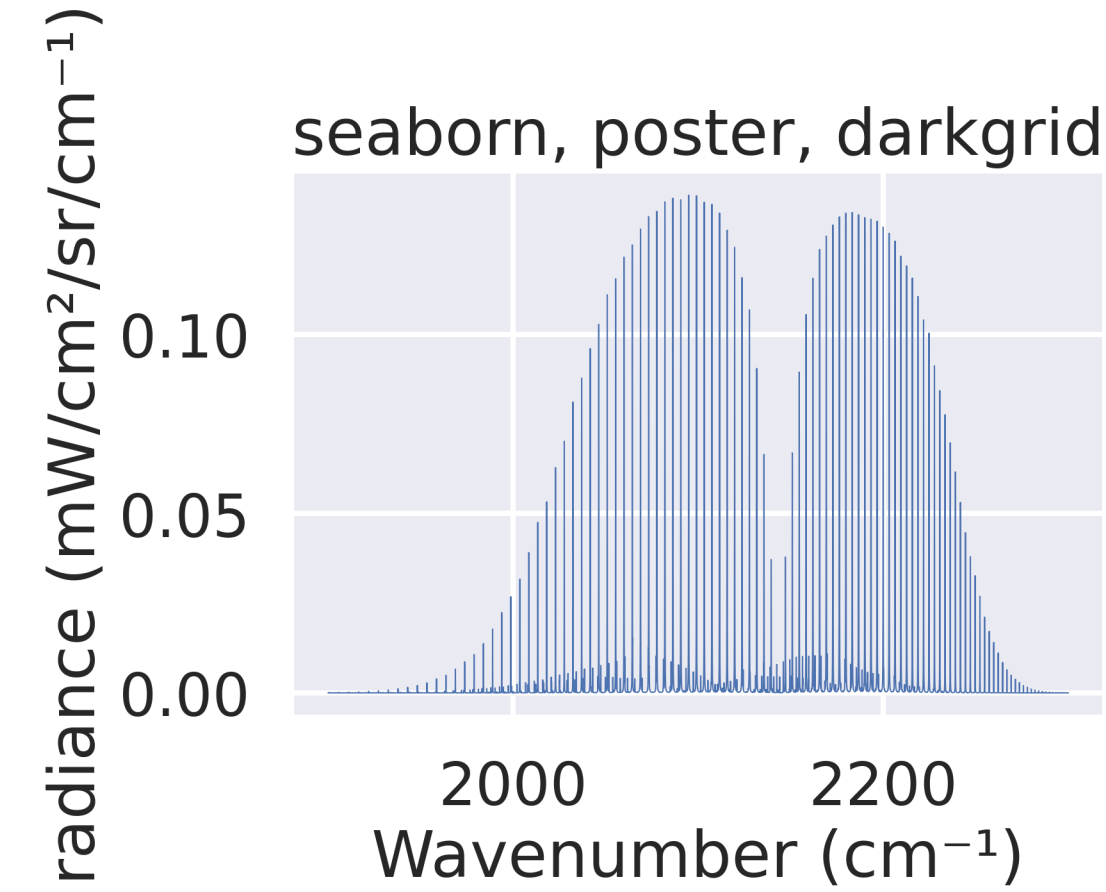
See Also

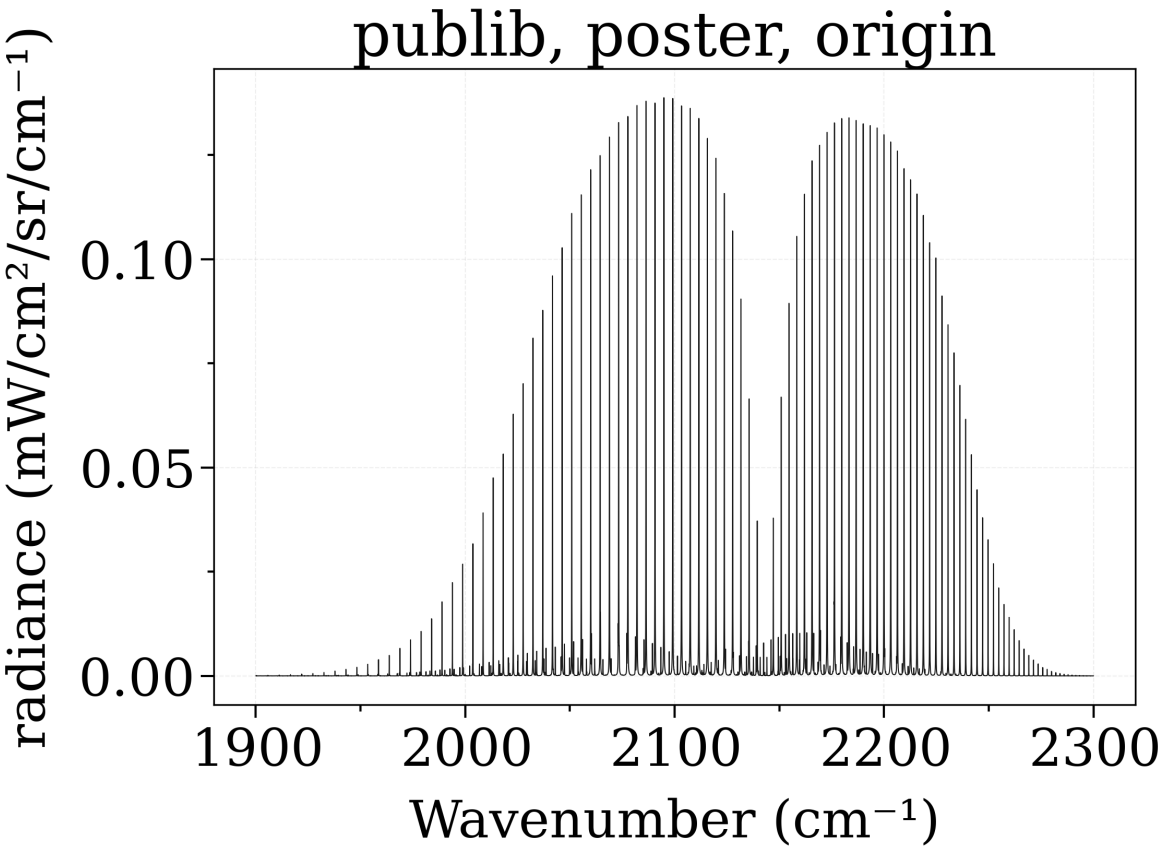
`set_style()`, `fix_style()`,


matplotlib, , default
radiance (mW/cm²/sr/cm⁻¹)
0.10
0.05
0.00
1900
2000
2100
2200
2300
Wavenumber (cm⁻¹)


seaborn, paper, darkgrid
radiance (mW/cm²/sr/cm⁻¹)
0.10
0.05
0.00
1900
2000
2100
2200
2300
Wavenumber (cm⁻¹)


publib, paper, origin
radiance (mW/cm²/sr/cm⁻¹)
0.10
0.05
0.00
1900
2000
2100
2200
2300
Wavenumber (cm⁻¹)


seaborn, poster, darkgrid
radiance (mW/cm²/sr/cm⁻¹)
0.10
0.05
0.00
2000
2200
Wavenumber (cm⁻¹)




```
Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 700 K
   Trot                 700 K
   Tvib                 700 K
   isotope              1,2,3
   mole_fraction        0.1
   molecule             CO
   overpopulation       None
   path_length          1 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2300.0000 cm-1
   wavenum_min          1900.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             hitran
```

(continues on next page)

(continued from previous page)

```
   dbpath                /home/docs/.radisdb/hitran/CO.hdf5
   folding_thresh        1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines       0 cm-1
   optimization          simple
   parfuncfmt            hapi
   parsum_mode           full summation
   pseudo_continuum_threshold  0
   sparse_ldm            auto
   truncation            50 cm-1
   waveunit              cm-1
   wstep                 0.01 cm-1
   zero_padding          -1
----------------------------------------
0.08s - Spectrum calculated
```

```
import matplotlib.pyplot as plt

import radis

plt.close("all")
s = radis.test_spectrum()

for plotlib, context, style in [
    ("matplotlib", "", "default"),
    ("seaborn", "paper", "darkgrid"),
    ("publib", "paper", "origin"),
    ("seaborn", "poster", "darkgrid"),
    ("publib", "poster", "origin"),
]:
    radis.config["plot"]["plotlib"] = plotlib
    radis.config["plot"]["context"] = context
    radis.config["plot"]["style"] = style
    s.plot()
    plt.title(", ".join(f"{v}" for v in radis.config["plot"].values()))
    plt.tight_layout()
```

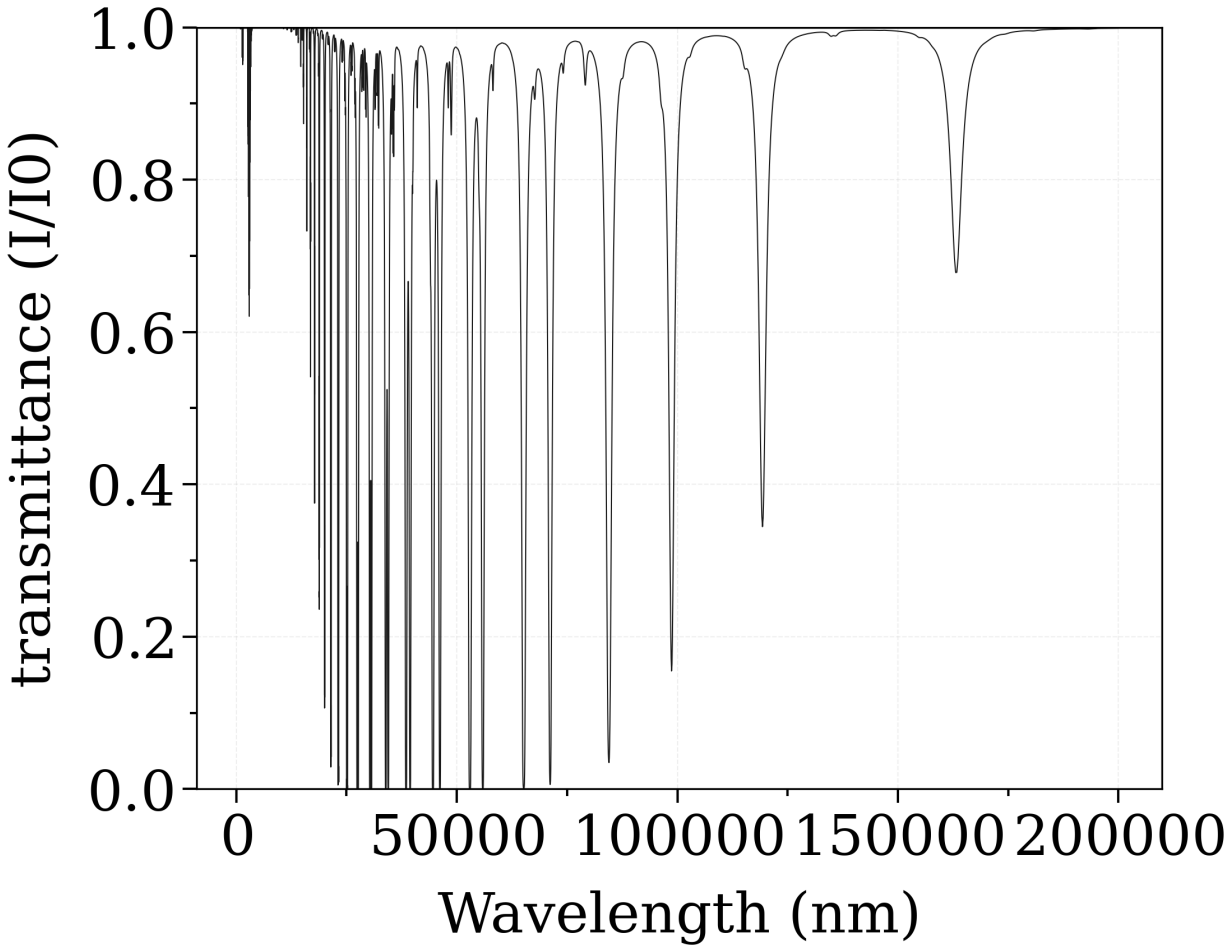
Total running time of the script: (0 minutes 3.282 seconds)

2.6.15 Calculate a large spectrum by part

In may be faster to calculate a full-range spectrum on several partial spectral ranges, and combine them.

Uses the `MergeSlabs()` function for that.

Starting from 0.11, RADIS introduces a sparse waverange implementation that should make it possible to directly compute a full range spectrum. See the *HITRAN full-range example*




```
OH-hitemp-radisdb-1000K-#5120 new quantities added: ['emisscoeff']
OH-hitemp-radisdb-1000K-#5376 new quantities added: ['emisscoeff']
OH-hitemp-radisdb-1000K-#5632 new quantities added: ['emisscoeff']
Spectrum Name:  OH-hitemp-radisdb-1000K-#5120//OH-hitemp-radisdb-1000K-#5376//OH-hitemp-
↪radisdb-1000K-#5632
Spectral Quantities
-----------------------------------------
   abscoeff     [cm-1]  (99,502 points)
   emisscoeff   [mW/cm3/sr/cm-1]        (99,502 points)
   radiance_noslit      [mW/cm2/sr/cm-1]        (99,502 points)
   emissivity_noslit    (99,502 points)
   absorbance   (99,502 points)
   transmittance_noslit         (99,502 points)
Physical Conditions
```

(continues on next page)

(continued from previous page)

```
----------------------------------------
   Tgas                 1000 K
   Trot                 1000 K
   Tvib                 1000 K
   isotope              N/A
   mole_fraction        0.1
   molecule             {'OH'}
   overpopulation       None
   path_length          1 cm
   pressure_mbar        10000.0 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   thermal_equilibrium  True
   vib_distribution     boltzmann
   wavenum_max          N/A cm-1
   wavenum_min          N/A cm-1
Computation Parameters
----------------------------------------
   NwG                  N/A
   NwL                  4
   Tref                 296 K
   add_at_used          cython
   broadening_method    voigt
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             hitemp-radisdb
   dbpath               /home/docs/.radisdb/hitemp/OH-13_HITEMP2020.hdf5
   default_output_unit  cm-1
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   profiler             N/A
   pseudo_continuum_threshold  0
   radis_version        0.13.1
   sparse_ldm           True
   spectral_points      N/A
   truncation           50 cm-1
   waveunit             cm-1
   wstep                0.1 cm-1
   zero_padding         N/A
Config parameters
----------------------------------------
   DEFAULT_DOWNLOAD_PATH  ~/.radisdb
   GRIDPOINTS_PER_LINEWIDTH_ERROR_THRESHOLD  1
   GRIDPOINTS_PER_LINEWIDTH_WARN_THRESHOLD  3
   SPARSE_WAVERANGE     auto
Information
----------------------------------------
```

(continues on next page)

(continued from previous page)

```
   calculation_time     0.22960855498968158 s
   chunksize            None
   db_use_cached        True
   dxG                  0.13753507880165727
   dxL                  0.20180288881201608
   export_lines         False
   export_populations   None
   export_rovib_fraction  True
   levelsfmt            None
   lines_calculated     5,672
   lines_cutoff         21,775
   lines_in_continuum   0
   load_energies        False
   lvl_use_cached       True
   parfuncpath          None
   total_lines          27447
   warning_broadening_threshold  0.01
   warning_linestrength_cutoff  0.01
   wavenum_max_calc     N/A cm-1
   wavenum_min_calc     N/A cm-1
----------------------------------------


(0.0, 1.0)
```

```
from radis import MergeSlabs, calc_spectrum

spectra = []
for (wmin, wmax) in [(50, 3000), (3000, 7000), (7000, 10000)]:

    spectra.append(
        calc_spectrum(
            wmin,
            wmax,
            Tgas=1000,
            pressure=10,  # bar
            molecule="OH",
            path_length=1,
            mole_fraction=0.1,
            wstep=0.1,
            databank="hitemp",
            verbose=False,
        )
    )

s = MergeSlabs(*spectra, resample="full", out="transparent")
print(s)
```

(continues on next page)

(continued from previous page)

```
s.plot("transmittance_noslit", wunit="nm")

import matplotlib.pyplot as plt

plt.ylim(0, 1)
```

Total running time of the script: (0 minutes 1.325 seconds)

2.6.16 Compare CO spectrum from the GEISA and HITRAN database

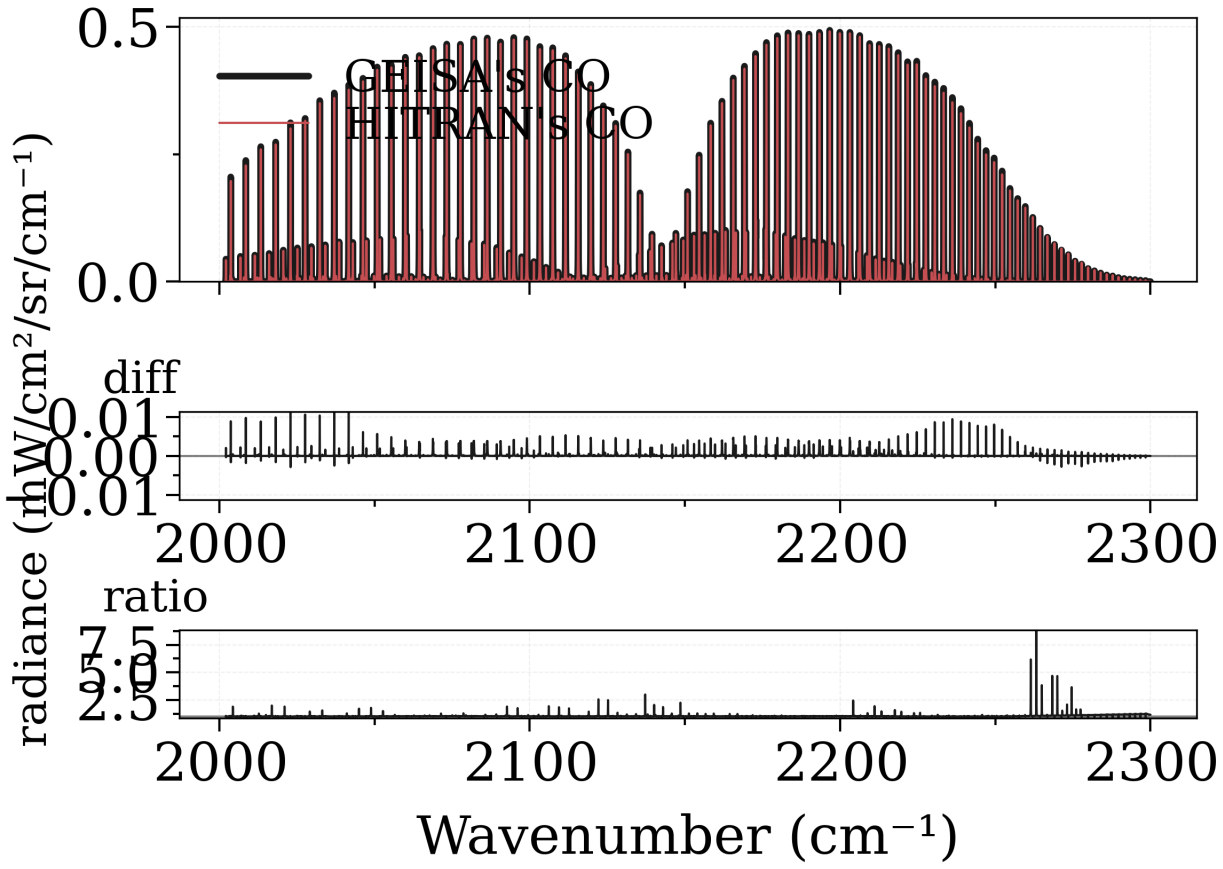
GEISA Database has been newly implemented in RADIS 0.13 release on May 15, 2022. This is among the very first attemps to compare the spectra generated from the two databases.

Auto-download and calculate CO spectrum from the GEISA database, and the HITRAN database.

Output should be similar, but not exactly! By default these two databases provide different broadening coefficients. However, the Einstein coefficients & linestrengths should be approximately the same, therefore the integrals under the lines should be similar.

You can see it by running the code below.

For your interest, GEISA and HITRAN lines can be downloaded and accessed separately using `fetch_geisa()` and `fetch_hitran()`



```
Molecule: CO
Downloading line_GEISA2020_asc_gs08_v1.0_co for CO (1/1).
Added GEISA-CO database in /home/docs/radis.json
Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 1000 K
   Trot                 1000 K
   Tvib                 1000 K
   isotope              1,2,3,4,5,6
   mole_fraction        0.1
   molecule             CO
   overpopulation       None
   path_length          1 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2300.0000 cm-1
   wavenum_min          2002.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             geisa
   dbpath               /home/docs/.radisdb/geisa/CO-line_GEISA2020_asc_gs08_v1.hdf5
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           50 cm-1
   waveunit             cm-1
   wstep                0.01 cm-1
   zero_padding         -1
----------------------------------------
0.07s - Spectrum calculated
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: HighTemperatureWarning:

HITRAN is valid for low temperatures (typically < 700 K). For higher temperatures you␣
↪may need HITEMP or CDSD. See the 'databank=' parameter

Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
```

(continues on next page)

(continued from previous page)

```
   Tgas                 1000 K
   Trot                 1000 K
   Tvib                 1000 K
   isotope              1,2,3,4,5,6
   mole_fraction        0.1
   molecule             CO
   overpopulation       None
   path_length          1 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2300.0000 cm-1
   wavenum_min          2002.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             hitran
   dbpath               /home/docs/.radisdb/hitran/CO.hdf5
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           50 cm-1
   waveunit             cm-1
   wstep                0.01 cm-1
   zero_padding         -1
----------------------------------------
0.07s - Spectrum calculated

(<Figure size 640x480 with 3 Axes>, [<AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:xlabel=
↪'Wavenumber (cm$^{-1}$)'>])
```

```
import astropy.units as u

from radis import calc_spectrum, plot_diff

conditions = {
```

(continues on next page)

(continued from previous page)

```
    "wmin": 2002 / u.cm,
    "wmax": 2300 / u.cm,
    "molecule": "CO",
    "pressure": 1.01325,  # bar
    "Tgas": 1000,  # K
    "mole_fraction": 0.1,
    "path_length": 1,  # cm
    "verbose": True,
}

s_geisa = calc_spectrum(**conditions, databank="geisa", name="GEISA's CO")

s_hitran = calc_spectrum(
    **conditions,
    databank="hitran",
    name="HITRAN's CO",
)

"""

In :py:func:`~radis.io.geisa.fetch_geisa`, you can choose to additionally plot the
absolute difference (method='diff') by default, or the ratio (method='ratio'), or both.

"""

plot_diff(s_geisa, s_hitran, method=["diff", "ratio"])
```

Total running time of the script: (0 minutes 3.587 seconds)

2.6.17 Get Molecular Parameters

Retrieve isotopologue abundances or molecular mass, based on HITRAN data, using `MolParams`

See also how to use custom (non-terrestrial) abundances in *the Custom Abundances example*

```
from radis.db.molparam import MolParams

molpar = MolParams()

print("CO2 abundances for the first 2 isotopes :")
print(molpar.get("CO2", 1, "abundance"))  # 1 for isotopologue number
print(molpar.get("CO2", 2, "abundance"))  # 1 for isotopologue number


print("H2O molar mass for the first 2 isotopes")
print(molpar.get("H2O", 1, "mol_mass"))  # 1 for isotopologue number
print(molpar.get("CO2", 2, "mol_mass"))  # 1 for isotopologue number


print("Tabulated CO partition function at 296 K for the first 2 isotopes")
print(molpar.get("CO", 1, "Q_296K"))  # 1 for isotopologue number
print(molpar.get("CO", 2, "Q_296K"))  # 1 for isotopologue number
```

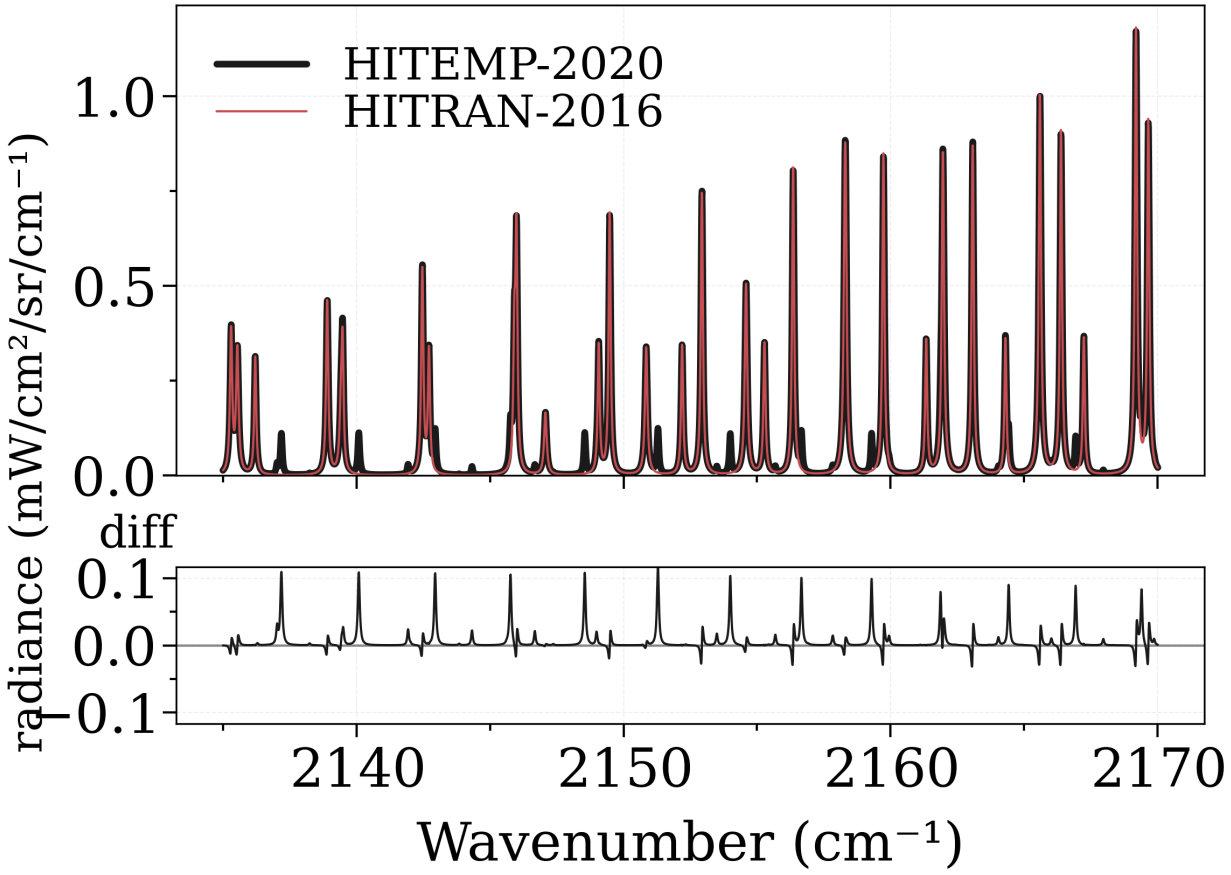
```
print("All parameters: ", list(molpar.df.columns))
print(molpar.df)
```

Total running time of the script: (0 minutes 0.000 seconds)

2.6.18 Calculate a spectrum from HITEMP

Auto-download and compute a CO spectrum from HITEMP ([HITEMP-2010]).

Compare with HITRAN-2016 ([HITRAN-2016]).




```
Downloading 05_HITEMP2019.par.bz2 for CO (1/1).
Download complete. Parsing CO database to /home/docs/.radisdb/hitemp/CO-05_HITEMP2019.
↪hdf5
Added HITEMP-CO database in /home/docs/radis.json
Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 2000 K
   Trot                 2000 K
   Tvib                 2000 K
   isotope              1
   mole_fraction        0.1
```

(continues on next page)

(continued from previous page)

```
   molecule             CO
   overpopulation       None
   path_length          1 cm
   pressure_mbar        3000.0 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2170.0000 cm-1
   wavenum_min          2135.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             hitemp-radisdb
   dbpath               /home/docs/.radisdb/hitemp/CO-05_HITEMP2019.hdf5
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           50 cm-1
   waveunit             cm-1
   wstep                0.01 cm-1
   zero_padding         -1
----------------------------------------
0.03s - Spectrum calculated
Added HITRAN-CO2-TEST database in /home/docs/radis.json
Added HITRAN-CO-TEST database in /home/docs/radis.json
Added HITEMP-CO2-TEST database in /home/docs/radis.json
Added HITEMP-CO2-HAMIL-TEST database in /home/docs/radis.json
Using database: HITRAN-CO-TEST
'HITRAN-CO-TEST':
{'info': 'HITRAN 2016 database, CO, 3 main isotopes (CO-26, 36, 28), 2000-2300 cm-1',
 ↪'path': ['/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/
 ↪python3.8/site-packages/radis/test/files/hitran_co_3iso_2000_2300cm.par'], 'format':
 ↪'hitran', 'parfuncfmt': 'hapi', 'levelsfmt': 'radis'}



Generating cache file /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
 ↪lib/python3.8/site-packages/radis/test/files/hitran_co_3iso_2000_2300cm.h5 with␣
 ↪metadata :
{'last_modification': 'Sun Aug 28 21:55:18 2022', 'wavenum_min': 2000.052539, 'wavenum_
 ↪max': 2298.445736}
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
```

(continues on next page)

(continued from previous page)

```
↪packages/radis/misc/warning.py:354: HighTemperatureWarning:

HITRAN is valid for low temperatures (typically < 700 K). For higher temperatures you␣
↪may need HITEMP or CDSD. See the 'databank=' parameter

Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 2000 K
   Trot                 2000 K
   Tvib                 2000 K
   isotope              1
   mole_fraction        0.1
   molecule             CO
   overpopulation       None
   path_length          1 cm
   pressure_mbar        3000.0 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2170.0000 cm-1
   wavenum_min          2135.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             hitran
   dbpath               /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/
↪master/lib/python3.8/site-packages/radis...
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           50 cm-1
   waveunit             cm-1
   wstep                0.01 cm-1
   zero_padding         -1
----------------------------------------
0.03s - Spectrum calculated

(<Figure size 640x480 with 2 Axes>, [<AxesSubplot:>, <AxesSubplot:xlabel='Wavenumber␣
↪(cm⁻¹)'>])
```

```
from radis import calc_spectrum

s = calc_spectrum(
    2135,
    2170,  # cm-1
    molecule="CO",
    isotope="1",
    pressure=3,  # bar
    Tgas=2000,  # K
    mole_fraction=0.1,
    path_length=1,  # cm
    databank="hitemp",  # latest version is fetched
    name="HITEMP-2020",
)

from radis.test.utils import setup_test_line_databases

setup_test_line_databases()

s2 = calc_spectrum(
    2135,
    2170,  # cm-1
    molecule="CO",
    isotope="1",
    pressure=3,  # bar
    Tgas=2000,  # K
    mole_fraction=0.1,
    path_length=1,  # cm
    databank="HITRAN-CO-TEST",  # we could also have fetched the latest HITRAN with
↪"databank='hitran'"
    name="HITRAN-2016",
)

# Compare:
from radis import plot_diff

plot_diff(s, s2)
```

Total running time of the script: (0 minutes 25.616 seconds)

2.6.19 Use Custom Abundances

Custom isotopologue abundances can be defined, to model non-terrestrial atmospheres.

Abundances are read and set directly in the `SpectrumFactory` using the `get_abundance()` and `set_abundance()` methods.

Below, we compute a CO2 spectrum with custom abundances for the two main terrestrial isotopes (12C-16O2 ; 13C-16O2)

See Also

MolParams

```
from radis.test.utils import setup_test_line_databases

setup_test_line_databases()  # creates "HITEMP-CO2-TEST" for this example

from radis import SpectrumFactory

sf = SpectrumFactory(
    2284.2,
    2284.6,
    wstep=0.001,  # cm-1
    pressure=20 * 1e-3,  # bar
    mole_fraction=400e-6,
    molecule="CO2",
    isotope="1,2",
    verbose=False,
)
sf.load_databank("HITEMP-CO2-TEST")
```

```
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: MissingReferenceWarning:

Missing doi for CDSD-HITEMP. Use HITEMP-2010?
```

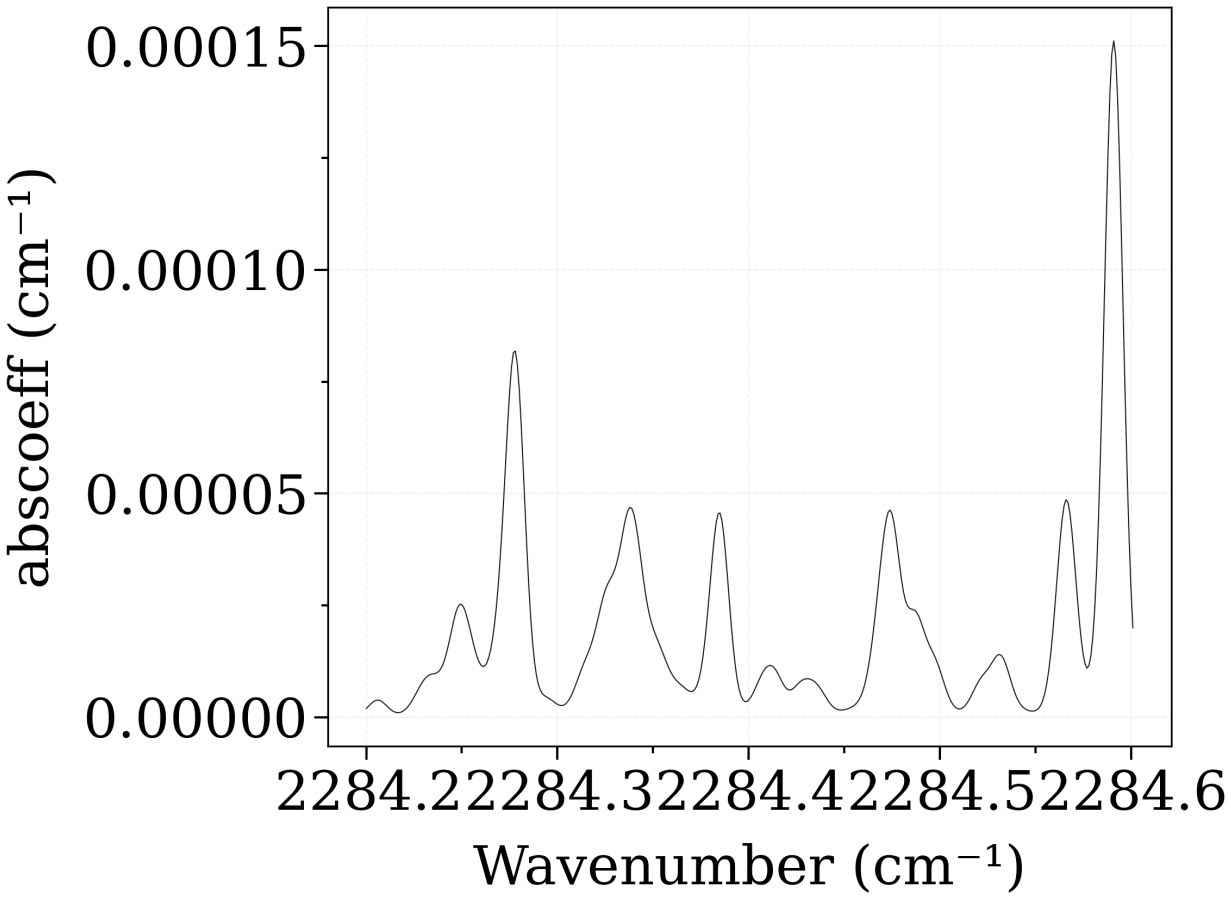
To explicitely identify the isotopes we can use the molparam attribute the Factory

```
print(sf.molparam.get("CO2", 1, "isotope_name"))  # >> (12C)(16O)2
print(sf.molparam.get("CO2", 2, "isotope_name"))  # >> (13C)(16O)2
```

```
(12C)(16O)2
(13C)(16O)2
```

Print the default abundance of the CO2 isotopes, compute a spectrum

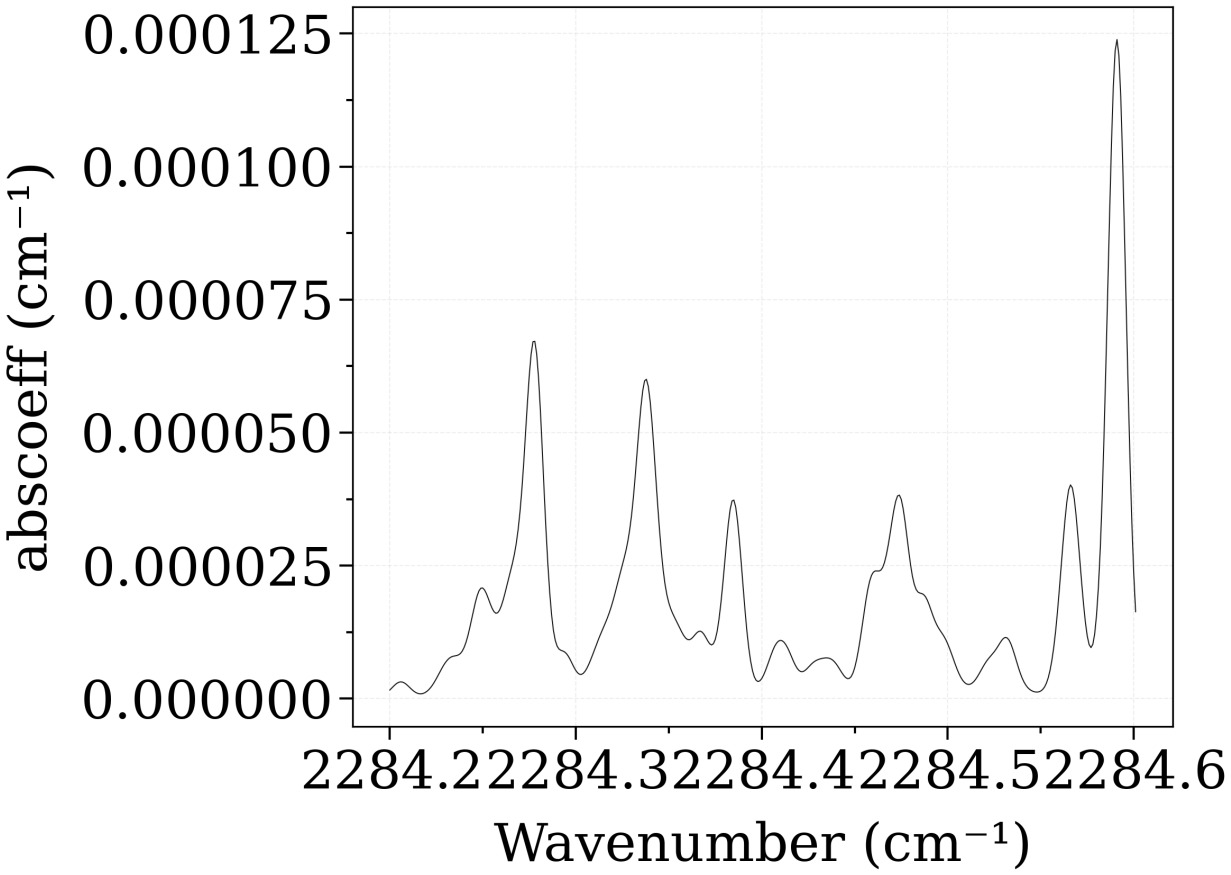
```
print("Abundance of CO2[1,2]", sf.get_abundance("CO2", [1, 2]))
sf.eq_spectrum(2000).plot("abscoeff")
```




```
Abundance of CO2[1,2] [0.984204  0.0110574]

<matplotlib.lines.Line2D object at 0x7f848b38efa0>
```

Set the abundance of CO2(626) to 0.8; and the abundance of CO2(636) to 0.2 (arbitrary):

```
sf.set_abundance("CO2", [1, 2], [0.8, 0.2])
print("New abundance of CO2[1,2]", sf.get_abundance("CO2", [1, 2]))

sf.eq_spectrum(2000).plot("abscoeff", nfig="same")
```




```
New abundance of CO2[1,2] [0.8 0.2]

<matplotlib.lines.Line2D object at 0x7f846d346ee0>
```

Total running time of the script: (0 minutes 1.649 seconds)

2.6.20 Calculate a spectrum from ExoMol

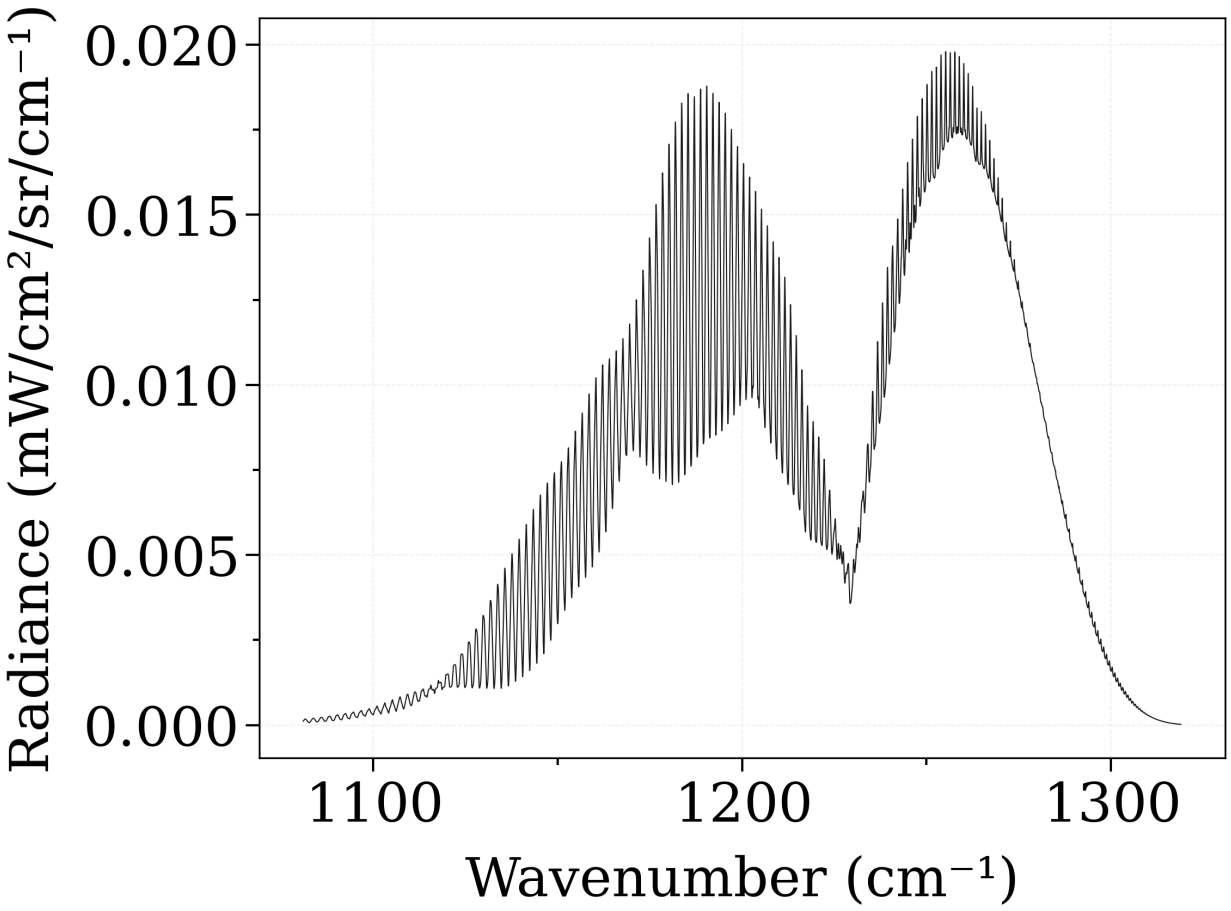
Auto-download and compute a SiO spectrum from the ExoMol database ([ExoMol-2020])

```
from radis import calc_spectrum

s = calc_spectrum(
    1080,
    1320,  # cm-1
    molecule="SiO",
    isotope="1",
    pressure=1.01325,  # bar
    Tgas=1000,  # K
    mole_fraction=0.1,
    path_length=1,  # cm
    broadening_method="fft",  # @ dev: Doesn't work with 'voigt'
    databank=("exomol", "EBJT"),  # Simply use 'exomol' for the recommended database
```

(continues on next page)

(continued from previous page)

```
)
s.apply_slit(1, "cm-1")  # simulate an experimental slit
s.plot("radiance")
```




```
Background atmosphere:  Air
Downloading http://www.exomol.com/db/SiO/28Si-16O/EBJT/28Si-16O__EBJT.def
Downloading http://www.exomol.com/db/SiO/28Si-16O/EBJT/28Si-16O__EBJT.pf
Downloading http://www.exomol.com/db/SiO/28Si-16O/EBJT/28Si-16O__EBJT.states.bz2
Downloading http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__H2.broad
Error: Couldn't download .broad file at http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__
↪H2.broad and save.
Downloading http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__He.broad
Error: Couldn't download .broad file at http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__
↪He.broad and save.
Downloading http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__air.broad
Error: Couldn't download .broad file at http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__
↪air.broad and save.
Note: Caching states data to the vaex format. After the second time, it will become much␣
↪faster.
Reading transition file
Downloading http://www.exomol.com/db/SiO/28Si-16O/EBJT/28Si-16O__EBJT.trans.bz2
Note: Caching line transition data to the vaex format. After the second time, it will␣
```

(continues on next page)

(continued from previous page)

```
↪become much faster.
.broad is used.
Warning: Cannot load .broad. The default broadening parameters are used.
Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 1000 K
   Trot                 1000 K
   Tvib                 1000 K
   isotope              1
   mole_fraction        0.1
   molecule             SiO
   overpopulation       None
   path_length          1 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          1320.0000 cm-1
   wavenum_min          1080.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    fft
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             exomol-radisdb
   dbpath               /home/docs/.radisdb/exomol/SiO/28Si-16O/EBJT
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           exomol
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           None cm-1
   waveunit             cm-1
   wstep                0.01 cm-1
   zero_padding         -1
----------------------------------------
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: MissingPressureShiftWarning:

Pressure-shift coefficient not given in database: assumed 0 pressure shift

0.07s - Spectrum calculated

<matplotlib.lines.Line2D object at 0x7f848ce01e50>
```

```
"""ExoMol lines can be downloaded and accessed separately using
:py:func:`~radis.io.exomol.fetch_exomol`
"""

# See line data:
from radis.io.exomol import fetch_exomol

df = fetch_exomol("SiO", database="EBJT", isotope="1", load_wavenum_max=5000)
print(df)
```

```
Background atmosphere:  Air
Downloading http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__H2.broad
Error: Couldn't download .broad file at http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__
↪H2.broad and save.
Downloading http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__He.broad
Error: Couldn't download .broad file at http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__
↪He.broad and save.
Downloading http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__air.broad
Error: Couldn't download .broad file at http://www.exomol.com/db/SiO/28Si-16O/28Si-16O__
↪air.broad and save.
Reading transition file
.broad is used.
Warning: Cannot load .broad. The default broadening parameters are used.
        i_upper  i_lower             A  ...            int  airbrd  Tdpair
0         24300    24299  1.926600e-12  ...  3.923990e-167    0.07     0.5
1          4698     5073  3.431600e-10  ...  2.626438e-202    0.07     0.5
2         24273    24272  1.652000e-10  ...  2.386983e-165    0.07     0.5
3         24301    24300  1.725200e-11  ...  2.947629e-166    0.07     0.5
4         24232    24231  7.675700e-10  ...  1.291448e-164    0.07     0.5
...         ...      ...           ...  ...            ...     ...     ...
249599     3769     1818  2.429700e-04  ...   3.947150e-91    0.07     0.5
249600     8052     6269  1.481800e-02  ...   2.484685e-72    0.07     0.5
249601     8693     6938  1.601100e-02  ...   1.337757e-67    0.07     0.5
249602     8659     6904  1.166600e-02  ...   3.267363e-66    0.07     0.5
249603     7665     5871  3.160000e-03  ...   4.399687e-64    0.07     0.5

[249604 rows x 11 columns]
```

See the list of recommended databases for the 1st isotope of SiO :

```
from radis.io.exomol import get_exomol_database_list, get_exomol_full_isotope_name

databases, recommended = get_exomol_database_list(
    "SiO", get_exomol_full_isotope_name("SiO", 1)
)
print("Databases for SiO: ", databases)
print("Database recommended by ExoMol: ", recommended)
```

```
Databases for SiO:  ['Kurucz-SiO', 'EBJT', 'xsec-SiOUVenIR', 'SiOUVenIR']
Database recommended by ExoMol:  SiOUVenIR
```

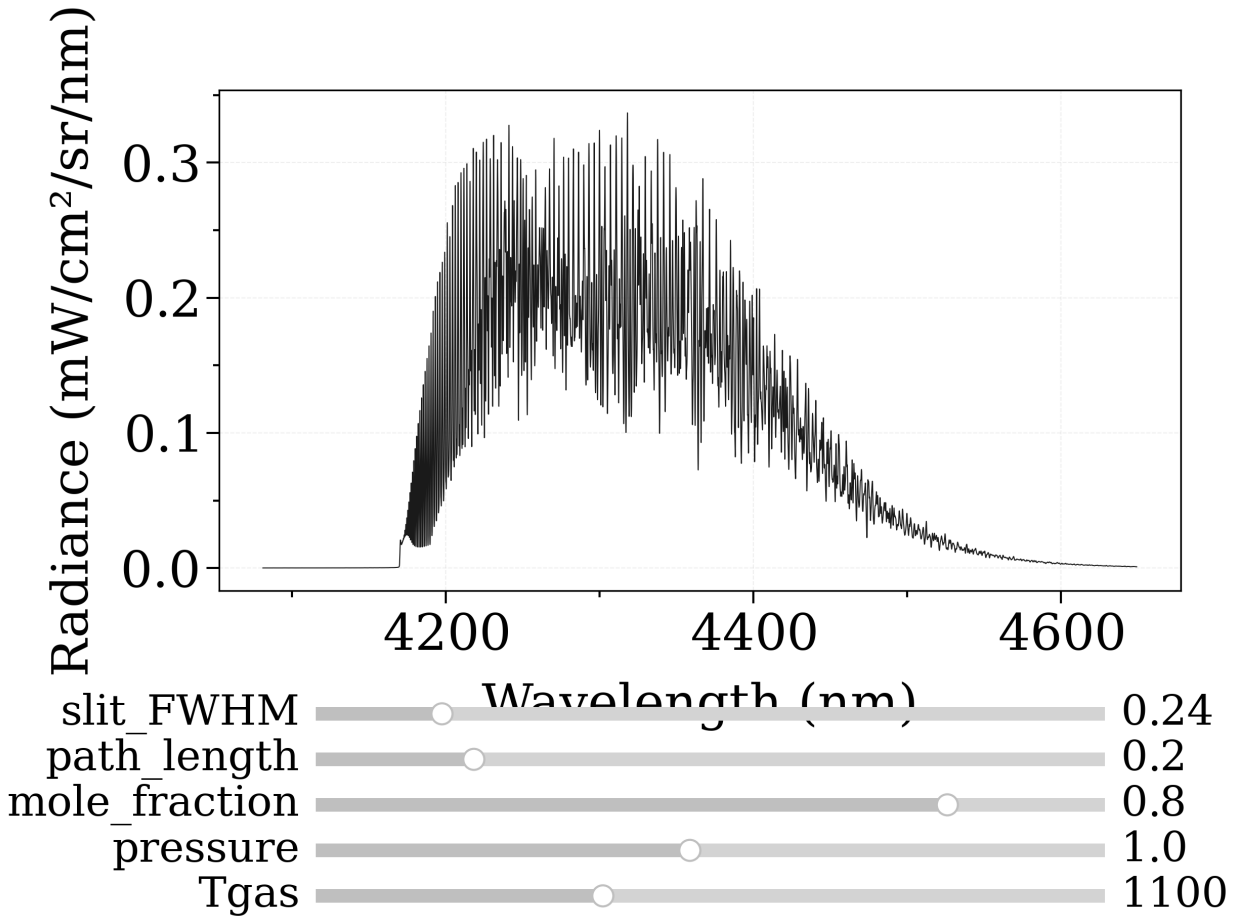
Total running time of the script: (0 minutes 10.747 seconds)

2.6.21 Real-time GPU Accelerated Spectra (Interactive)

Example using GPU sliders and GPU calculation with `eq_spectrum_gpu()`

This method requires CUDA compatible hardware to execute. For more information on how to setup your system to run GPU-accelerated methods using CUDA and Cython, check GPU Spectrum Calculation on RADIS

Note: in the example below, the GPU code runs on CPU, using the parameter `emulate=True`. In your environment, to run the GPU code with the full power of the GPU, remove this line or set `emulate=False` (default)




```
HITEMP keep only relevant input files: ['/home/docs/.radisdb/hitemp/CO2-02_02125-02250_
↪HITEMP2010.hdf5', '/home/docs/.radisdb/hitemp/CO2-02_02250-02500_HITEMP2010.hdf5']
HITEMP keep only relevant input files: []
HITEMP keep only relevant input files: ['/home/docs/.radisdb/hitemp/CO2-02_02125-02250_
↪HITEMP2010.hdf5', '/home/docs/.radisdb/hitemp/CO2-02_02250-02500_HITEMP2010.hdf5']
Number of lines loaded: 1128265

Finished calculating spectrum!
1.19s - Spectrum calculated
Spectrum Name:  CO2-hitemp-radisdb-1100.0K-#1168
Spectral Quantities
----------------------------------------
```

(continues on next page)

(continued from previous page)

```
   abscoeff     [cm-1]  (150,001 points)
   absorbance   (150,001 points)
   emissivity   (150,001 points, 242 nans)
   emissivity_noslit    (150,001 points)
   transmittance_noslit         (150,001 points)
   radiance_noslit      [mW/cm2/sr/cm-1]        (150,001 points)
   transmittance        (150,001 points, 242 nans)
   radiance     [mW/cm2/sr/cm-1]        (150,001 points, 242 nans)
Physical Conditions
----------------------------------------
   Tgas                 1100.0 K
   Trot                 1100.0 K
   Tvib                 1100.0 K
   isotope              1,2,3
   mole_fraction        0.8
   molecule             CO2
   overpopulation       None
   path_length          0.2 cm
   pressure_mbar        1000.0 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   thermal_equilibrium  True
   vib_distribution     boltzmann
   wavenum_max          2450.0000 cm-1
   wavenum_min          2150.0000 cm-1
Computation Parameters
----------------------------------------
   NwG                  3
   NwL                  6
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               0 cm-1/(#.cm-2)
   dbformat             hitemp-radisdb
   dbpath               /home/docs/.radisdb/hitemp/CO2-02_02125-02250_HITEMP2010.hdf5,/
↪home/docs/.radisdb/hitemp/CO2-02_0225...
   default_output_unit  cm-1
   emulate_gpu          True
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   norm_by              area
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pressure             1.0
   profiler             {'spectrum_calculation': {'scaled_S0': 0.03147624399571214,
↪'calc_other_spectral_quan': 0.0132299820...
   pseudo_continuum_threshold  0
   radis_version        0.13.1
```

(continues on next page)

(continued from previous page)

```
   slit_FWHM            0.24
   slit_dispersion      None
   slit_dispersion_threshold  0.01
   slit_function        0.24
   slit_shape           triangular
   slit_unit            cm-1
   sparse_ldm           auto
   spectral_points      150000.0
   truncation           50 cm-1
   waveunit             cm-1
   wstep                0.002 cm-1
   zero_padding         -1
Config parameters
----------------------------------------
   DEFAULT_DOWNLOAD_PATH  ~/.radisdb
   GRIDPOINTS_PER_LINEWIDTH_ERROR_THRESHOLD  1
   GRIDPOINTS_PER_LINEWIDTH_WARN_THRESHOLD  3
   SPARSE_WAVERANGE     auto
Information
----------------------------------------
   calculation_time     1.192465706000803 s
   chunksize            None
   db_use_cached        True
   dxG                  0.13753507880165727
   dxL                  0.20180288881201608
   export_lines         False
   export_populations   None
   export_rovib_fraction  False
   levelsfmt            radis
   lines_calculated     1,128,265
   load_energies        False
   lvl_use_cached       True
   parfuncpath          None
   total_lines          1128265
   warning_broadening_threshold  0.01
   warning_linestrength_cutoff  0.01
   wavenum_max_calc     2450.0000 cm-1
   wavenum_min_calc     2150.0000 cm-1
----------------------------------------
```

```
from radis import SpectrumFactory

# from radis.test.utils import getTestFile
from radis.tools.plot_tools import ParamRange

## Add an experimental file :
# my_file = getTestFile("CO2_measured_spectrum_4-5um.spec")  # for the example here
```

(continues on next page)

(continued from previous page)

```
# s_exp = load_spec(my_file)
# s_exp.crop(4120, 4790).plot(Iunit="mW/cm2/sr/nm")
#
## This spectrum is significantly absorbed by atmospheric CO2
## so it will never match the synthetic spectrum.
## TODO: find different spectrum for this example.


sf = SpectrumFactory(
    2150,
    2450,  # cm-1
    molecule="CO2",
    isotope="1,2,3",
    wstep=0.002,
)

sf.fetch_databank("hitemp")

s = sf.eq_spectrum_gpu_interactive(
    var="radiance",
    Tgas=ParamRange(300.0, 2500.0, 1100.0),  # K
    pressure=ParamRange(0.1, 2, 1),  # bar
    mole_fraction=ParamRange(0, 1, 0.8),
    path_length=ParamRange(0, 1, 0.2),  # cm
    slit_FWHM=ParamRange(0, 1.5, 0.24),  # cm-1
    emulate=True,  # if True, runs CPU code on GPU. Set to False or remove to run on the␣
↪GPU
    plotkwargs={"wunit": "nm"},  # "nfig": "same",
)

print(s)
```

Total running time of the script: (0 minutes 2.881 seconds)

2.6.22 See populations of computed levels

Spectrum methods gives you access to the populations of levels used in a nonequilibrium spectrum :

- `get_populations()` provides the populations

of all levels of the molecule, used to compute the nonequilibrium partition function

- `lines` returns all informations

of the visible lines, in particular the populations of the upper and lower levels of absorbing/emitting lines in the spectrum.

```
from astropy import units as u
```

We first compute a noneq CO spectrum. Notice the keys `export_lines=True` and `export_populations=True`

```
from radis import calc_spectrum

```

(continues on next page)

(continued from previous page)

```
s = calc_spectrum(
    1900 / u.cm,
    2300 / u.cm,
    molecule="CO",
    isotope="1",
    pressure=1.01325 * u.bar,
    Tvib=3000 * u.K,
    Trot=1000 * u.K,
    mole_fraction=0.1,
    path_length=1 * u.cm,
    databank="hitran",  # or 'hitemp', 'geisa', 'exomol'
    export_lines=True,
    export_populations="rovib",
)
```

```
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: HighTemperatureWarning:

HITRAN is valid for low temperatures (typically < 700 K). For higher temperatures you␣
↪may need HITEMP or CDSD. See the 'databank=' parameter

Calculating Non-Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 1000.0 K
   Trot                 1000.0 K
   Tvib                 3000.0 K
   isotope              1
   mole_fraction        0.1
   molecule             CO
   path_length          1.0 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2300.0000 cm-1
   wavenum_min          1900.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             hitran
   dbpath               /home/docs/.radisdb/hitran/CO.hdf5
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
```

(continues on next page)

(continued from previous page)

```
   parsum_mode           full summation
   pseudo_continuum_threshold  0
   sparse_ldm            auto
   truncation            50 cm-1
   waveunit              cm-1
   wstep                 0.01 cm-1
   zero_padding          -1
----------------------------------------
Fetching Evib & Erot.
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: NegativeEnergiesWarning:

There are negative rotational energies in the database

/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: PerformanceWarning:

'gu' was recomputed although 'gp' already in DataFrame. All values are equal

... sorting lines by vibrational bands
... lines sorted in 0.0s
0.14s - Spectrum calculated
```

Below we plot the populations We could also have used `plot_populations()` directly

```
pops = s.get_populations("CO")["rovib"]
```

```
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/spectrum/spectrum.py:2249: UserWarning:

Populations valid for partition function calculation but sometimes NOT for spectra␣
↪calculations, e.g. CO2.
                          See help on how to use 's.lines.query' instead, for instance in␣
↪https://github.com/radis/radis/issues/508.
                          Turn off warning with 'show_warning=False'
```

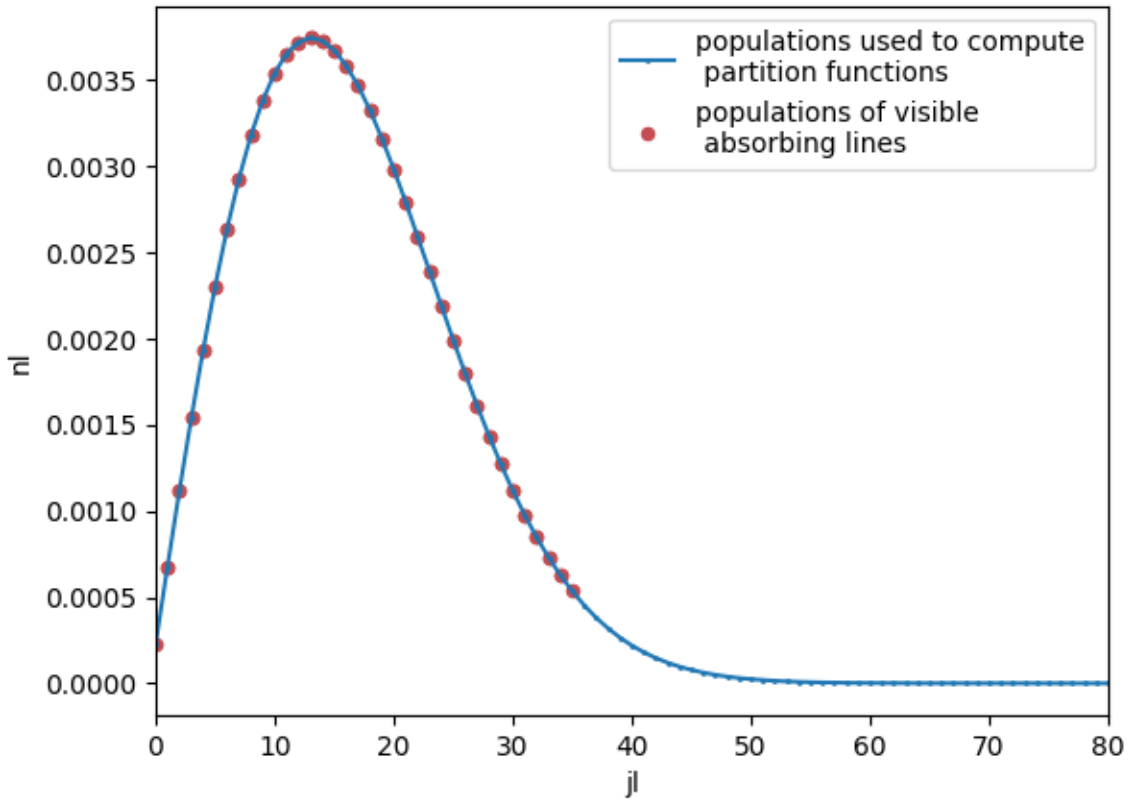
Plot populations : Here we plot population fraction vs rotational number of the 2nd vibrational level v==2, for all levels as well as for levels of lines visible in the spectrum

```
import matplotlib.pyplot as plt

pops.query("v==2").plot(
    "j",
    "n",
    label="populations used to compute\n partition functions",
    style=".-",
    markersize=2,
)
s.lines.query("vl==2").plot(
    "jl",
    "nl",
    ax=plt.gca(),
```

(continues on next page)

(continued from previous page)

```
    label="populations of visible\n absorbing lines",
    kind="scatter",
    color="r",
)
plt.xlim((0, 80))
plt.legend()
```




```
<matplotlib.legend.Legend object at 0x7f848b426730>
```

Print all levels information

```
print(f"{len(pops)} levels ")
print(pops)
print(pops.columns)
```

```
7766 levels
       v   j             E  ...             n        Qrot      nrot
0      0   0      0.000000  ...  1.754208e-03  362.691791  0.002757
1      0   1      3.845033  ...  5.233590e-03  362.691791  0.008226
2      0   2     11.534953  ...  8.626674e-03  362.691791  0.013559
3      0   3     23.069466  ...  1.187857e-02  362.691791  0.018670
4      0   4     38.448131  ...  1.493823e-02  362.691791  0.023479
```

(continues on next page)

(continued from previous page)

```
...   ..  ..           ...  ...           ...         ...       ...
7761  48  38  89299.135499  ...  2.237048e-21  360.424333  0.003612
7762  48  39  89447.643402  ...  1.853601e-21  360.424333  0.002993
7763  48  40  89599.882150  ...  1.526677e-21  360.424333  0.002465
7764  48  41  89755.845909  ...  1.249931e-21  360.424333  0.002018
7765  48  42  89915.528699  ...  1.017299e-21  360.424333  0.001643

[7766 rows x 13 columns]
Index(['v', 'j', 'E', 'Evib', 'viblvl', 'gj', 'gvib', 'grot', 'Erot', 'nvib',
       'n', 'Qrot', 'nrot'],
      dtype='object')
```

Print visible lines information :

```
print(f"{len(s.lines)} visible lines in the spectrum")
print(s.lines)
print(s.lines.columns)
```

```
274 visible lines in the spectrum
             wav           int      A  ...  hwhm_voigt  hwhm_lorentz  hwhm_gauss
0    1900.341956  3.539000e-29  12.45  ...    0.016713      0.015650    0.004067
1    1902.414114  2.295000e-31  24.91  ...    0.017474      0.016455    0.004072
2    1905.778633  9.041000e-29  12.54  ...    0.016872      0.015814    0.004079
3    1907.676341  5.311000e-31  25.11  ...    0.017672      0.016659    0.004083
4    1911.187699  2.268000e-28  12.64  ...    0.017028      0.015973    0.004091
..           ...           ...    ...  ...         ...           ...         ...
269  2291.548195  1.124000e-28  21.70  ...    0.017153      0.015650    0.004905
270  2293.336784  4.416000e-29  21.77  ...    0.017008      0.015491    0.004909
271  2295.082667  1.703000e-29  21.83  ...    0.016901      0.015372    0.004912
272  2296.785698  6.450000e-30  21.90  ...    0.016752      0.015208    0.004916
273  2298.445736  2.400000e-30  21.97  ...    0.016609      0.015049    0.004919

[274 rows x 49 columns]
Index(['wav', 'int', 'A', 'airbrd', 'selbrd', 'El', 'Tdpair', 'Pshft', 'gp',
       'gpp', 'branch', 'jl', 'vu', 'vl', 'ju', 'Eu', 'Evibl', 'Evibu',
       'Erotu', 'Erotl', 'gju', 'gjl', 'grotu', 'grotl', 'gvibu', 'gvibl',
       'gu', 'gl', 'Rs2', 'Blu', 'Bul', 'Aul', 'viblvl_l', 'viblvl_u', 'band',
       'Qrotl', 'Qrotu', 'nu_vib', 'nl_vib', 'nu_rot', 'nl_rot', 'nu', 'nl',
       'S', 'Ei', 'shiftwav', 'hwhm_voigt', 'hwhm_lorentz', 'hwhm_gauss'],
      dtype='object')
```

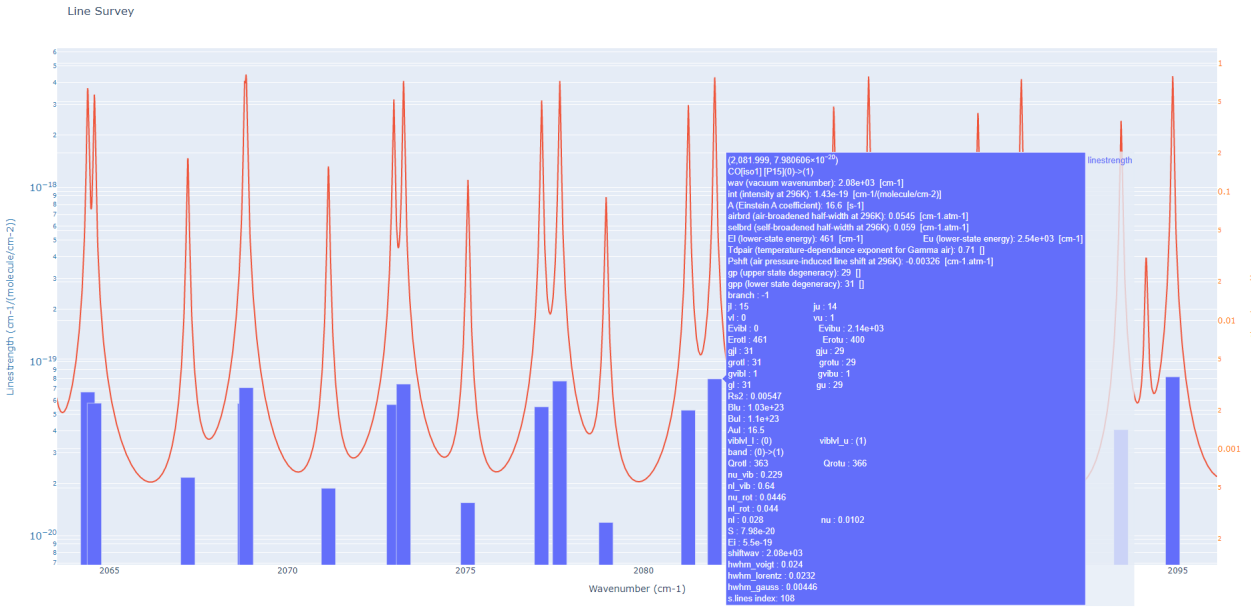
We can also look at unique levels; for instance sorting by quantum numbers `v, J` of the lower level : `vl, jl`

```
print(
    len(s.lines.groupby(["vl", "jl"])),
    " visible absorbing levels (i.e. unique 'vl', 'jl' ",
)
```

```
141  visible absorbing levels (i.e. unique 'vl', 'jl'
```

`line_survey()` is also a convenient way to explore populatinos and other line parameters, for instance

```
s.line_survey(overlay="absorbance", barwidth=0.001, lineinfo="all")
```



Total running time of the script: (0 minutes 0.438 seconds)

2.6.23 Calculate a full range spectrum

We compute the transmittance of an homogeneous, 1-km atmosphere layer at 300 K and 1 atm, with 420 ppm CO2 and 2% H2O.

This example makes use of a lot of the RADIS optimization and convenience functions :

- The database is retrieved from the latest HITRAN data using [HAPI], parsed with RADIS and stored to memory-mapping formats. After the first calls, retrieving the database will be very fast
- The RADIS algorithm includes a sparse wavenumber implementation, which is activated by default in the following example where the waverange is very large. This behavior can be changed by setting the `SPARSE_WAVENUMBER` key of the `radis.config` dictionary, or of the ~/radis.json user file.
- The wavenumber grid is automatically computed using the `wstep='auto'` parameter.
- A linestrength cutoff reduces the initial number of lines.

Eventually, the full-range spectrum (about 4.8M points) is computed within a minute, with about 140k lines resolved, i.e. a performance of about 1e10 gridpoints*lines/s.

Extra optimization could be achieved by cutting the spectrum in small intervals, but requires an a-priori knowledge of the absorption & emission ranges of the spectrum. See the *Calculate a large spectrum by part*

```
import astropy.units as u

from radis import calc_spectrum

s = calc_spectrum(
    wmin=0.5 * u.um,
    wmax=15 * u.um,  # cm-1
    mole_fraction={"CO2": 420e-6, "H2O": 0.02},
```

(continues on next page)

(continued from previous page)

```
    isotope="1,2,3",
    pressure=1.01325,  # bar
    Tgas=300,  # K
    path_length=1e5,  # 1 km in cm
    verbose=2,
    databank="hitran",
    wstep="auto",
)
```

Plot low and high resolution spectra on the same graph :

```
s.apply_slit(10, "nm")
import matplotlib.pyplot as plt

plt.figure(figsize=(16, 6))
s.plot("transmittance_noslit", wunit="nm", color="k", alpha=0.1, nfig="same")
s.plot("transmittance", wunit="nm", color="k", nfig="same")
```

Print some details about the computed spectrum : We could also simply use `print(s)`

```
print("Number of grid points: ", len(s))
print(
    "Number of lines: ", s.c["total_lines"]
)  # some are discarded by linestrength cutoff
print("Number of lines: ", s.c["lines_calculated"])
print("Whether the sparse wavenumber algorithm was activated: ", s.c["sparse_ldm"])
print(f"Lineshape truncation used: {s.c['truncation']:.1f}cm-1")
print(f"Total calculation time: {s.c['calculation_time']:.1f}s")
print(
    f"Gridpoints * lines/s : {len(s)*s.c['lines_calculated']/s.c['calculation_time']:.1e}
↪"
)
```

Total running time of the script: (0 minutes 0.000 seconds)

2.6.24 Spectrum Database

RADIS has `SpecDatabase` feature used to store and retrieve calculated Spectrums. A path can be specified for SpecDatabase all Spectrums are stored as .spec files which can be loaded from the SpecDatabase object itself. A csv file is generated which contains all input and conditional parameters of Spectrum.

RADIS also has `init_database()` feature which initializes the SpecDatabase for the SpectrumFactory and every Spectrum generated from it will be stored in the SpecDatabase automatically.

You can use `plot_cond()` to make a 2D plot using the conditions of the Spectrums in the SpecDatabase and use a z_label to plot a heat map based on it.

`SpecDatabase` also has useful `fit_spectrum()` and `interpolate()` methods.

```
from pathlib import Path

import numpy as np
```

(continues on next page)

(continued from previous page)

```
from radis import SpectrumFactory
from radis.tools import SpecDatabase

sf = SpectrumFactory(
    wavenum_min=2900,
    wavenum_max=3200,
    molecule="OH",
    truncation=5,  # cm-1
    medium="vacuum",
    verbose=0,  # more for more details
    pressure=10,
    wstep=0.1,
)
sf.warnings = {"AccuracyError": "ignore"}
sf.fetch_databank("hitemp")

# Generating a Spectrum
s1 = sf.eq_spectrum(Tgas=300, path_length=1)
```

Creating SpecDatabase This is simply a local folder on your computer.

```
from radis.test.utils import getTestFile

my_folder = Path(getTestFile(".")) / "SpecDatabase_Test"
db = SpecDatabase(my_folder)
```

```
Database SpecDatabase_Test initialised in /home/docs/checkouts/readthedocs.org/user_
↪builds/radis/envs/master/lib/python3.8/site-packages/radis/test/files
*** Loading the database with 1 processor (0 files)***
```

There are many ways to add spectra to the database : Method 1: Creating .spec file and adding manually to SpecDatabase

```
db.add(s1)
```

```
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828.spec (0.1Mb)
loaded 20220828.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.


'/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/test/files/SpecDatabase_Test/20220828.spec'
```

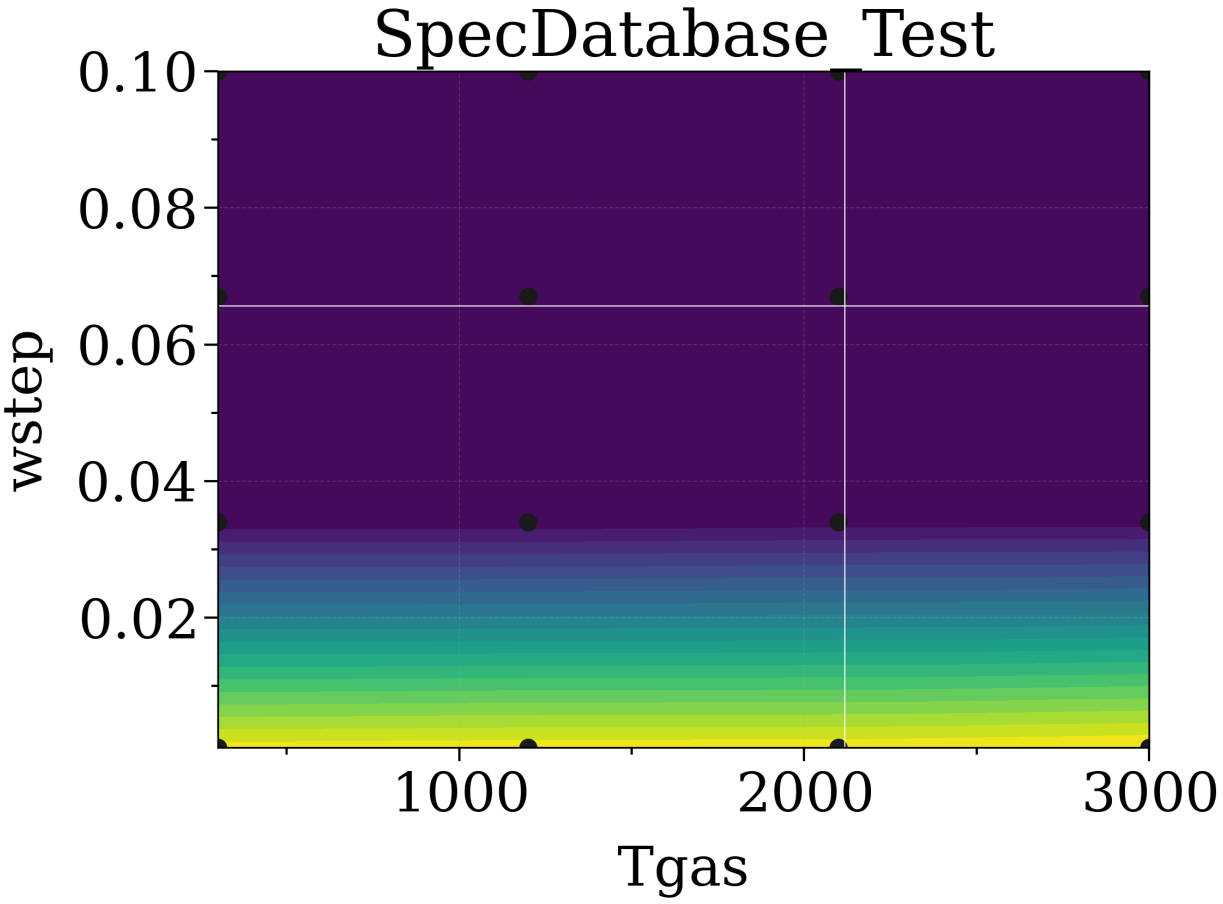
Method 2: Using init_database() SpecDatabase is connected to the SpectrumFactory `sf` above Generated Spectrum are added to SpecDatabase automatically :

```
sf.init_database(my_folder)
```

(continues on next page)

(continued from previous page)

```
wstep = np.linspace(0.1, 0.001, 4)
Tgas = np.linspace(300, 3000, 4)


# Multiple Spectrum calculation based on different values of Tgas and wstep
for i in wstep:
    sf._wstep = i
    sf.params.wstep = i
    for j in Tgas:
        sf.eq_spectrum(Tgas=j, path_length=1)


# Loading SpecDatabase
db_new = SpecDatabase(my_folder)

# Plot Tgas vs wstep for all Spectrums, heatmap based on calculation_time
db_new.plot_cond("Tgas", "wstep", "calculation_time")
```




```
Loading database SpecDatabase_Test
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib1200K_
↪Trot1200K.spec (0.1Mb)
```

(continues on next page)

(continued from previous page)

```
loaded 20220828_Tvib1200K_Trot1200K.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib2100K_
↪Trot2100K.spec (0.1Mb)
loaded 20220828_Tvib2100K_Trot2100K.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib3000K_
↪Trot3000K.spec (0.1Mb)
loaded 20220828_Tvib3000K_Trot3000K.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib300K_
↪Trot300K.spec (0.2Mb)
loaded 20220828_Tvib300K_Trot300K.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib1200K_
↪Trot1200K_1.spec (0.2Mb)
loaded 20220828_Tvib1200K_Trot1200K_1.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib2100K_
↪Trot2100K_1.spec (0.2Mb)
```

(continues on next page)

(continued from previous page)

```
loaded 20220828_Tvib2100K_Trot2100K_1.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib3000K_
↪Trot3000K_1.spec (0.2Mb)
loaded 20220828_Tvib3000K_Trot3000K_1.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib300K_
↪Trot300K_1.spec (0.3Mb)
loaded 20220828_Tvib300K_Trot300K_1.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib1200K_
↪Trot1200K_2.spec (0.4Mb)
loaded 20220828_Tvib1200K_Trot1200K_2.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib2100K_
↪Trot2100K_2.spec (0.4Mb)
loaded 20220828_Tvib2100K_Trot2100K_2.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
```

(continues on next page)

(continued from previous page)

```
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib3000K_
↪Trot3000K_2.spec (0.4Mb)
loaded 20220828_Tvib3000K_Trot3000K_2.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib300K_
↪Trot300K_2.spec (10.7Mb)
loaded 20220828_Tvib300K_Trot300K_2.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib1200K_
↪Trot1200K_3.spec (12.2Mb)
loaded 20220828_Tvib1200K_Trot1200K_3.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib2100K_
↪Trot2100K_3.spec (12.4Mb)
loaded 20220828_Tvib2100K_Trot2100K_3.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
↪version. Use pandas.concat instead.

Warning. File already exists. Filename is incremented
Spectrum stored in /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/SpecDatabase_Test/20220828_Tvib3000K_
↪Trot3000K_3.spec (12.4Mb)
loaded 20220828_Tvib3000K_Trot3000K_3.spec
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/database.py:2350: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a future␣
```

(continues on next page)

(continued from previous page)

```
↪version. Use pandas.concat instead.

Loading database SpecDatabase_Test
```

Total running time of the script: (0 minutes 7.933 seconds)

2.6.25 Compare CO xsections from the ExoMol and HITEMP database

Auto-download and calculate CO spectrum from the HITEMP database, and the ExoMol database. ExoMol references multiple databases for CO. Here we do not use the ExoMol recommended database (see `get_exomol_database_list()`) but we use the HITEMP database hosted on ExoMol.

Output should be similar, but no ! By default these two databases provide different broadening coefficients. However, the Einstein coefficients & linestrengths should be the same, therefore the integrals under the lines should be similar.

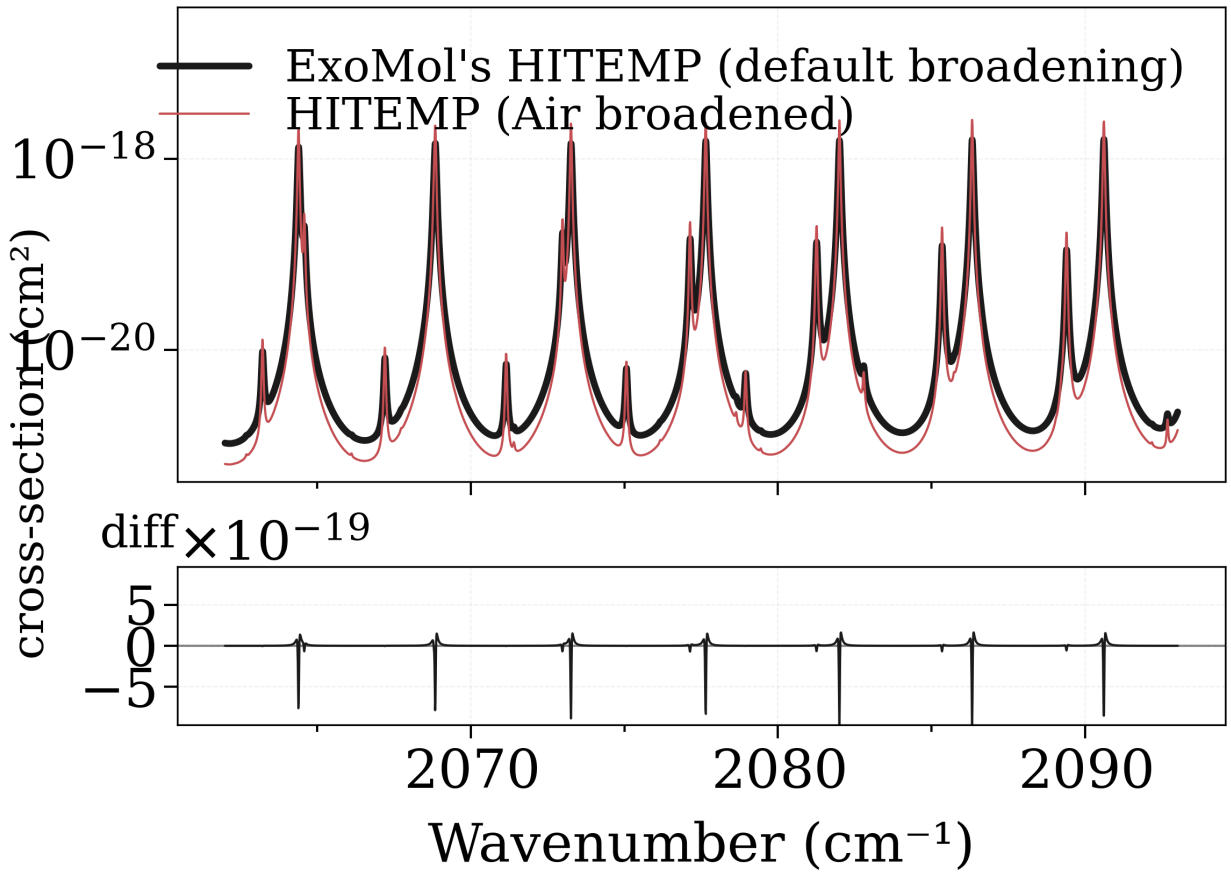
We verifiy this below :

For further exploration, ExoMol and HITEMP lines can be downloaded and accessed separately using `fetch_exomol()` and `fetch_hitemp()`

```
import astropy.units as u

from radis import calc_spectrum, plot_diff

conditions = {
    "wmin": 2062 / u.cm,
    "wmax": 2093 / u.cm,
    "molecule": "CO",
    "isotope": "1",
    "pressure": 1.01325,  # bar
    "Tgas": 1000,  # K
    "mole_fraction": 0.1,
    "path_length": 1,  # cm
    "broadening_method": "fft",  # @ dev: Doesn't work with 'voigt'
    "verbose": True,
    "neighbour_lines": 20,
}
```

Note that we account for the effect on neighbour_lines by computing `20cm-1` on the side (`neighbour_lines` condition above)

```
s_exomol = calc_spectrum(
    **conditions, databank="exomol", name="ExoMol's HITEMP (default broadening)"
)
s_hitemp = calc_spectrum(
    **conditions,
    databank="hitemp",
    name="HITEMP (Air broadened)",
)
fig, [ax0, ax1] = plot_diff(s_exomol, s_hitemp, "xsection", yscale="log")
# Adjust diff plot to be in linear scale
ax1.set_yscale("linear")
ax0.set_ylim(ymax=ax0.get_ylim()[1] * 10)  # more space for legend
```




```
Using ExoMol database Li2015 for 12C-16O (recommended by the ExoMol team). All available␣
↪databases are ['HITEMP', 'Li', 'xsec-Li2015', 'xsec-VUV-DTU', 'Li2015']. Select one of␣
↪them with `radis.fetch_exomol(DATABASE_NAME)`, `SpectrumFactory.fetch_databank('exomol
↪', exomol_database=DATABASE_NAME')`, or `calc_spectrum(..., databank=('exomol',␣
↪DATABASE_NAME))`

Background atmosphere:  Air
Downloading http://www.exomol.com/db/CO/12C-16O/Li2015/12C-16O__Li2015.def
Downloading http://www.exomol.com/db/CO/12C-16O/Li2015/12C-16O__Li2015.pf
Downloading http://www.exomol.com/db/CO/12C-16O/Li2015/12C-16O__Li2015.states.bz2
Downloading http://www.exomol.com/db/CO/12C-16O/12C-16O__H2.broad
Downloading http://www.exomol.com/db/CO/12C-16O/12C-16O__He.broad
Downloading http://www.exomol.com/db/CO/12C-16O/12C-16O__air.broad
Error: Couldn't download .broad file at http://www.exomol.com/db/CO/12C-16O/12C-16O__air.
↪broad and save.
Note: Caching states data to the vaex format. After the second time, it will become much␣
↪faster.
Reading transition file
Downloading http://www.exomol.com/db/CO/12C-16O/Li2015/12C-16O__Li2015.trans.bz2
Note: Caching line transition data to the vaex format. After the second time, it will␣
↪become much faster.
.broad is used.
Warning: Cannot load .broad. The default broadening parameters are used.
Calculating Equilibrium Spectrum
```

(continues on next page)

(continued from previous page)

```
Physical Conditions
----------------------------------------
   Tgas                 1000 K
   Trot                 1000 K
   Tvib                 1000 K
   isotope              1
   mole_fraction        0.1
   molecule             CO
   overpopulation       None
   path_length          1 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2093.0000 cm-1
   wavenum_min          2062.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    fft
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             exomol-radisdb
   dbpath               /home/docs/.radisdb/exomol/CO/12C-16O/Li2015
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      20 cm-1
   optimization         simple
   parfuncfmt           exomol
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           None cm-1
   waveunit             cm-1
   wstep                0.01 cm-1
   zero_padding         -1
----------------------------------------
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: MissingPressureShiftWarning:

Pressure-shift coefficient not given in database: assumed 0 pressure shift

0.02s - Spectrum calculated
Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 1000 K
   Trot                 1000 K
   Tvib                 1000 K
   isotope              1
```

(continues on next page)

(continued from previous page)

```
   mole_fraction        0.1
   molecule             CO
   overpopulation       None
   path_length          1 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2093.0000 cm-1
   wavenum_min          2062.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    fft
   cutoff               1e-27 cm-1/(#.cm-2)
   dbformat             hitemp-radisdb
   dbpath               /home/docs/.radisdb/hitemp/CO-05_HITEMP2019.hdf5
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      20 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           None cm-1
   waveunit             cm-1
   wstep                0.01 cm-1
   zero_padding         -1
----------------------------------------
0.02s - Spectrum calculated

(4.1789321827835834e-22, 3.8605771326502384e-17)
```

Broadening coefficients are different in these databases, so lineshapes end up being very different; however the areas under the lines should be the same. We verify this :

```
import numpy as np

try:
    assert np.isclose(
        s_exomol.get_integral("xsection"), s_hitemp.get_integral("xsection"), rtol=0.001
    )

except:
    # @dev: someone there is un expected error with this example on ReadThedocs.
    # Escaping for the moment. See https://github.com/radis/radis/issues/501
    pass
```

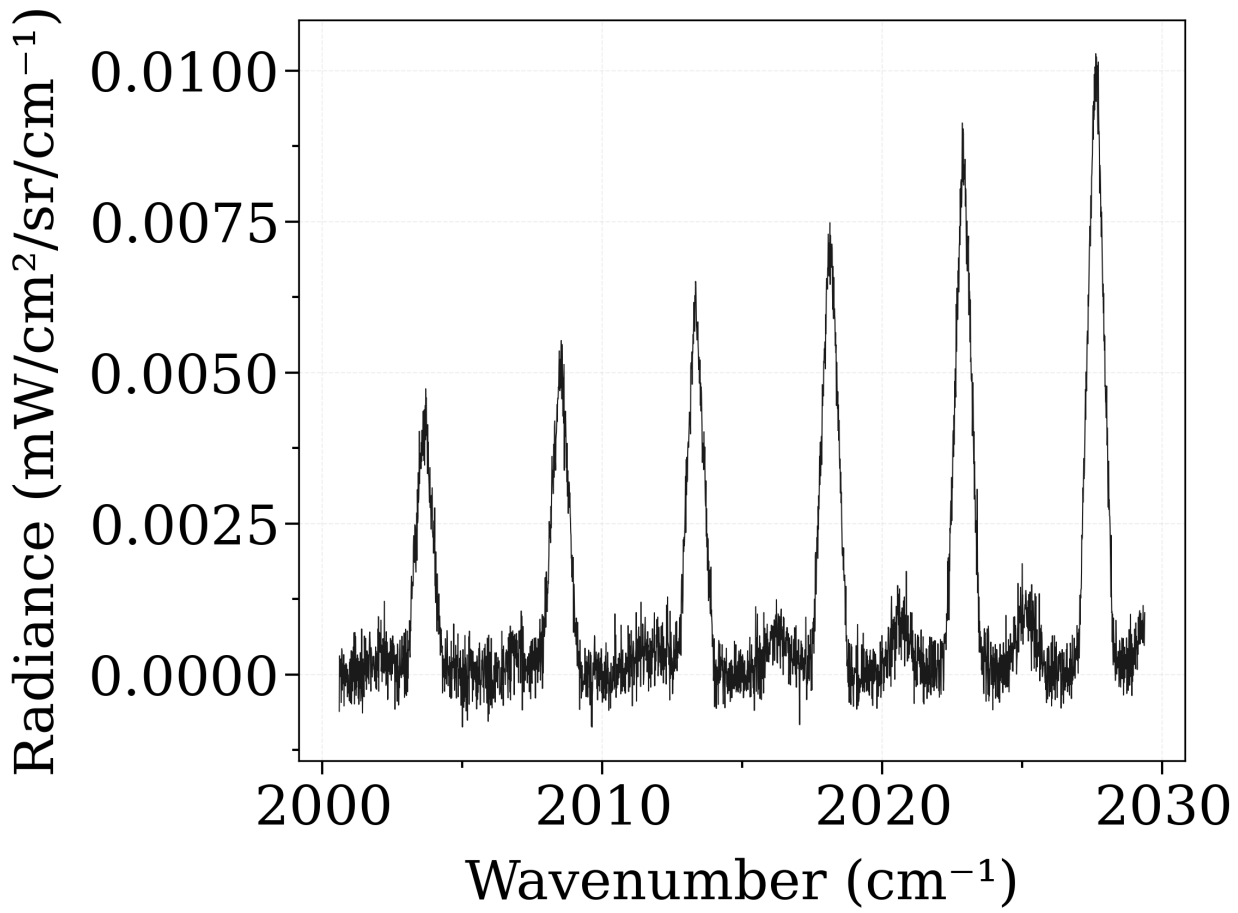
Total running time of the script: (0 minutes 7.079 seconds)

2.6.26 Post-process using Specutils

Find peaks or uncertainties using the `specutils` library. A Radis Spectrum object can easily be converted to a `specutils` `specutils.spectra.spectrum1d.Spectrum1D` using `to_specutils()`.

Below, we create a noisy spectrum based on a synthetic CO spectrum, we convert it to `specutils`, add uncertainties by targeting a noisy region, then determine the lines using `find_lines_threshold()` :

```
import astropy.units as u
import numpy as np

from radis import test_spectrum

""" We create a synthetic CO spectrum"""

s = (
    test_spectrum(molecule="CO", wavenum_min=2000, wavenum_max=2030)
    .apply_slit(1.5, "nm")
    .take("radiance")
)
s.trim()  # removes nans created by the slit convolution boundary effects
noise = np.random.normal(0.0, s.max().value * 0.03, len(s))
s_exp = s + noise

s_exp.plot()
```




```
Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 700 K
   Trot                 700 K
   Tvib                 700 K
   isotope              1,2,3
   mole_fraction        0.1
   molecule             CO
   overpopulation       None
   path_length          1 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2030.0000 cm-1
   wavenum_min          2000.0000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
```

(continues on next page)

(continued from previous page)

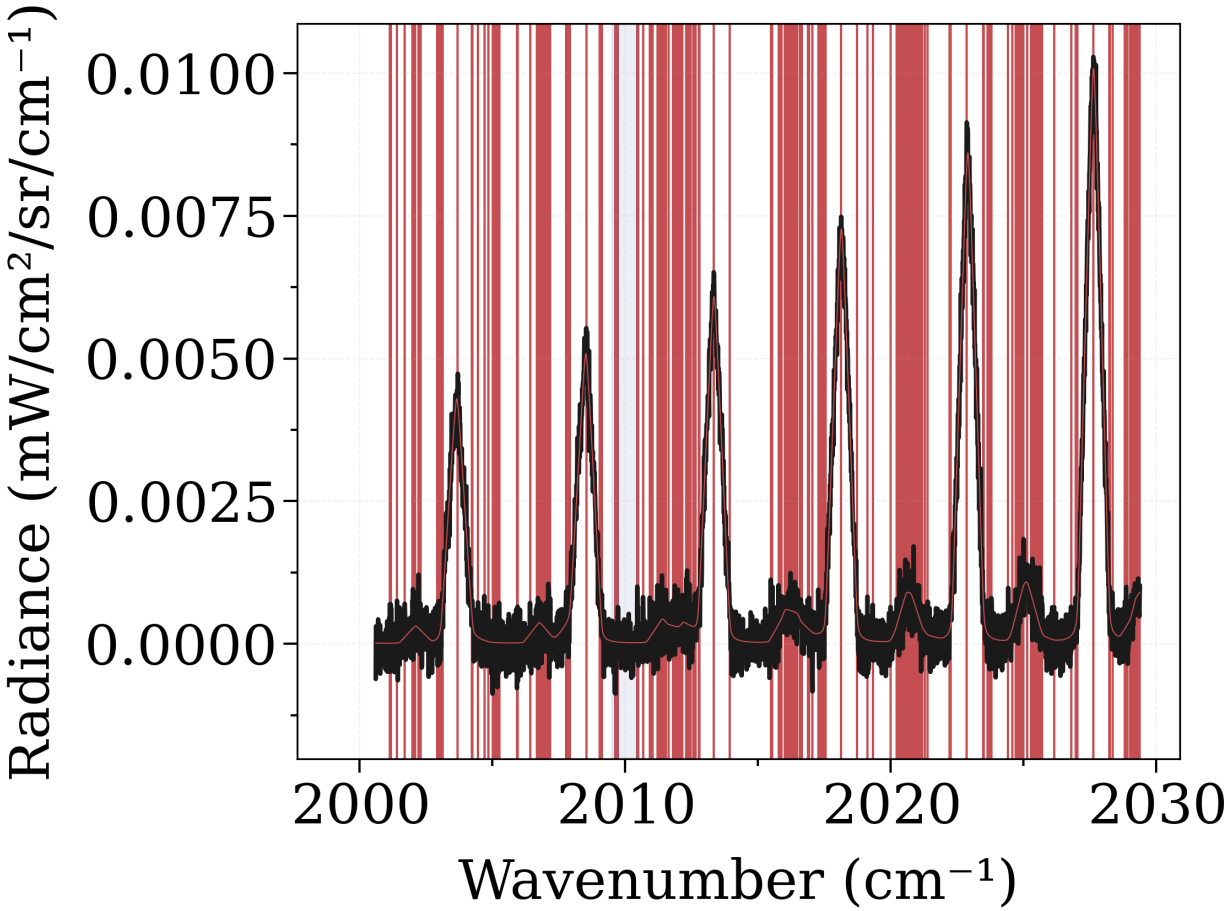
```
   cutoff                1e-27 cm-1/(#.cm-2)
   dbformat              hitran
   dbpath                /home/docs/.radisdb/hitran/CO.hdf5
   folding_thresh        1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines       0 cm-1
   optimization          simple
   parfuncfmt            hapi
   parsum_mode           full summation
   pseudo_continuum_threshold  0
   sparse_ldm            auto
   truncation            50 cm-1
   waveunit              cm-1
   wstep                 0.01 cm-1
   zero_padding          -1
-----------------------------------------
0.03s - Spectrum calculated

<matplotlib.lines.Line2D object at 0x7f848b9cbd30>
```

Determine the noise level by selecting a noisy region from the graph above :

```
spectrum = s_exp.to_specutils()

from specutils import SpectralRegion
from specutils.manipulation import noise_region_uncertainty

noise_region = SpectralRegion(2010.5 / u.cm, 2009.5 / u.cm)
spectrum = noise_region_uncertainty(spectrum, noise_region)
```

Find lines :

```
from specutils.fitting import find_lines_threshold

lines = find_lines_threshold(spectrum, noise_factor=2)

print(lines)


s_exp.plot(lw=2, show_ruler=True)
import matplotlib.pyplot as plt

for line in lines.to_pandas().line_center.values:
    plt.axvline(line, color="r", zorder=-1)
s.plot(nfig="same")
plt.axvspan(noise_region.lower.value, noise_region.upper.value, color="b", alpha=0.1)
```




```
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/specutils/analysis/flux.py:290: AstropyUserWarning:

Spectrum is not below the threshold signal-to-noise 0.01. This may indicate you have not␣
↪continuum subtracted this spectrum (or that you have but it has high SNR features).

If you want to suppress this warning either type 'specutils.conf.do_continuum_function_
↪check = False' or see http://docs.astropy.org/en/stable/config/#adding-new-
↪configuration-items for other ways to configure the warning.

   line_center     line_type  line_center_index
      1 / cm
----------------- ---------- -----------------
 2001.139999999999   emission                53
 2001.169999999999   emission                56
2001.4099999999987   emission                80
2001.6999999999985   emission               109
2001.9899999999982   emission               138
2002.0099999999982   emission               140
2002.0399999999981   emission               143
 2002.079999999998   emission               147
 2002.099999999998   emission               149
 2002.219999999998   emission               161
```

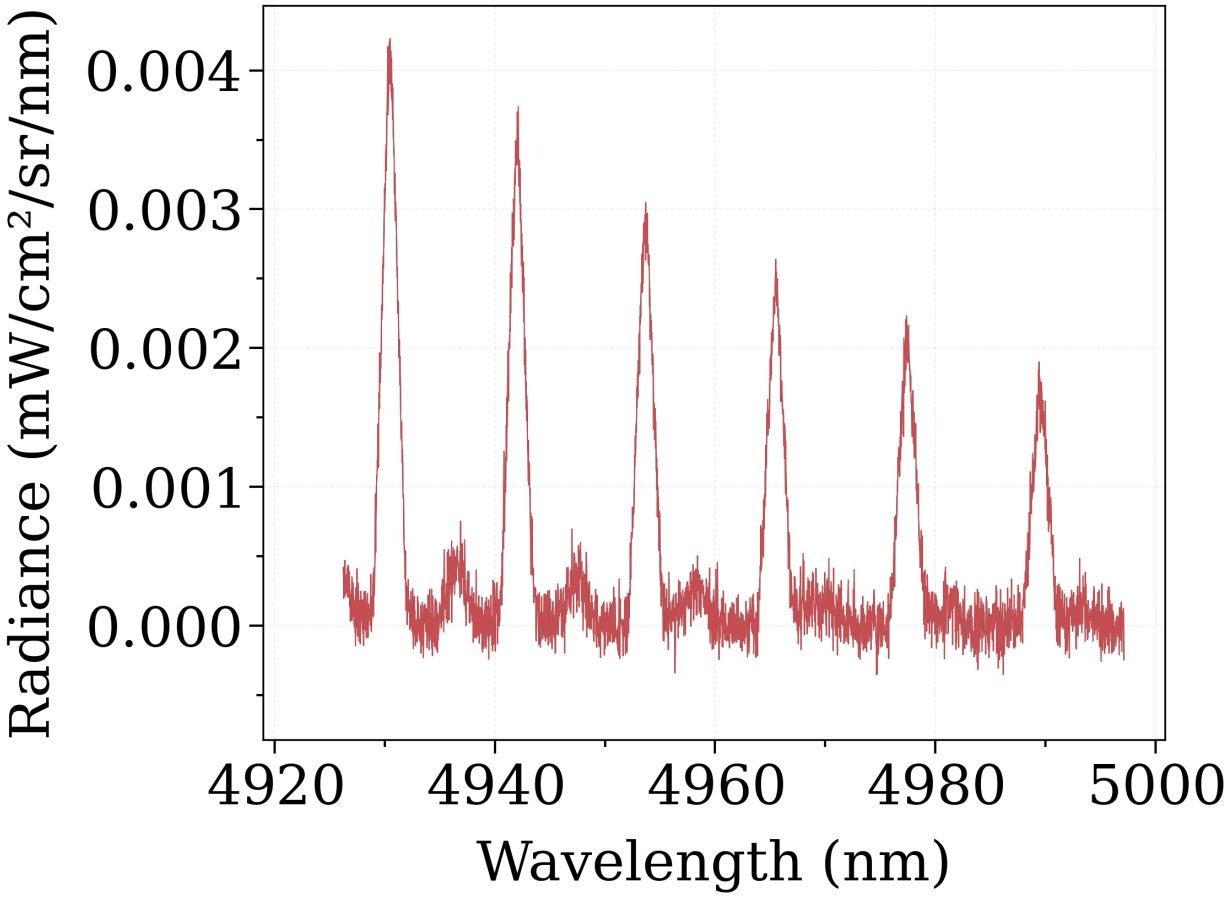
(continues on next page)

(continued from previous page)

```
                ...         ...                ...
2029.2999999999734   emission               2869
2029.3299999999733   emission               2872
2029.3899999999733   emission               2878
2005.0099999999954 absorption                440
2005.1899999999953 absorption                458
2005.9299999999946 absorption                532
2005.9599999999946 absorption                535
2009.6199999999913 absorption                901
2009.6399999999912 absorption                903
2017.0599999999845 absorption               1645
Length = 194 rows
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/tools/plot_tools.py:599: UserWarning:

Couldn't add Ruler tool (still an experimental feature in RADIS : please report the␣
↪error !)


<matplotlib.patches.Polygon object at 0x7f8467c16100>
```

Note: we can also create a RADIS spectrum object from Specutils `specutils.spectra.spectrum1d.Spectrum1D` :

```
from radis import Spectrum

s2 = Spectrum.from_specutils(spectrum)

s2.plot(Iunit="mW/cm2/sr/nm", wunit="nm")
s_exp.plot(Iunit="mW/cm2/sr/nm", wunit="nm", nfig="same")
assert s_exp == s2
```



Total running time of the script: (0 minutes 3.881 seconds)

2.6.27 Multi-temperature Fit

A method to fit an experimental spectrum directly from `SpectrumFactory`, with `fit_spectrum()`

Typical output is similar to the radis-examples Multi-temperature fit :

The method requires a fitting model. An example model is provided in `radis.tools.fitting` : `Tvib12Tvib3Trot_NonLTEModel()`. Other models can be used, such as in the *one-temperature fit example*

More advanced tools for interactive fitting of multi-dimensional, multi-slabs spectra can be found in `fitroom`. Finally, the *GPU-accelerated example* shows how to obtain real-time interactive spectra.

```
from os.path import join

from radis import Spectrum, SpectrumFactory
```

Get Fitted Data

```
from radis.test.utils import getValidationCase, setup_test_line_databases
from radis.tools.fitting import Tvib12Tvib3Trot_NonLTEModel

```

(continues on next page)

(continued from previous page)

```
setup_test_line_databases()
# Data from Dang, adapted by Klarenaar, digitized by us
s_exp = Spectrum.from_txt(
    getValidationCase(
        join("test_CO2_3Tvib_vs_klarenaar_data", "klarenaar_2017_digitized_data.csv")
    ),
    "transmittance_noslit",
    wunit="cm-1",
    unit="",
    delimiter=",",
    name="Klarenaar 2017",
)
```

```
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/spectrum/spectrum.py:4400: UserWarning:

Wavespace is not evenly spaced (0.000%) for transmittance_noslit. This may create␣
↪problems if later convolving with slit function (`s.apply_slit()`). You can use `s.
↪resample_even()`
```

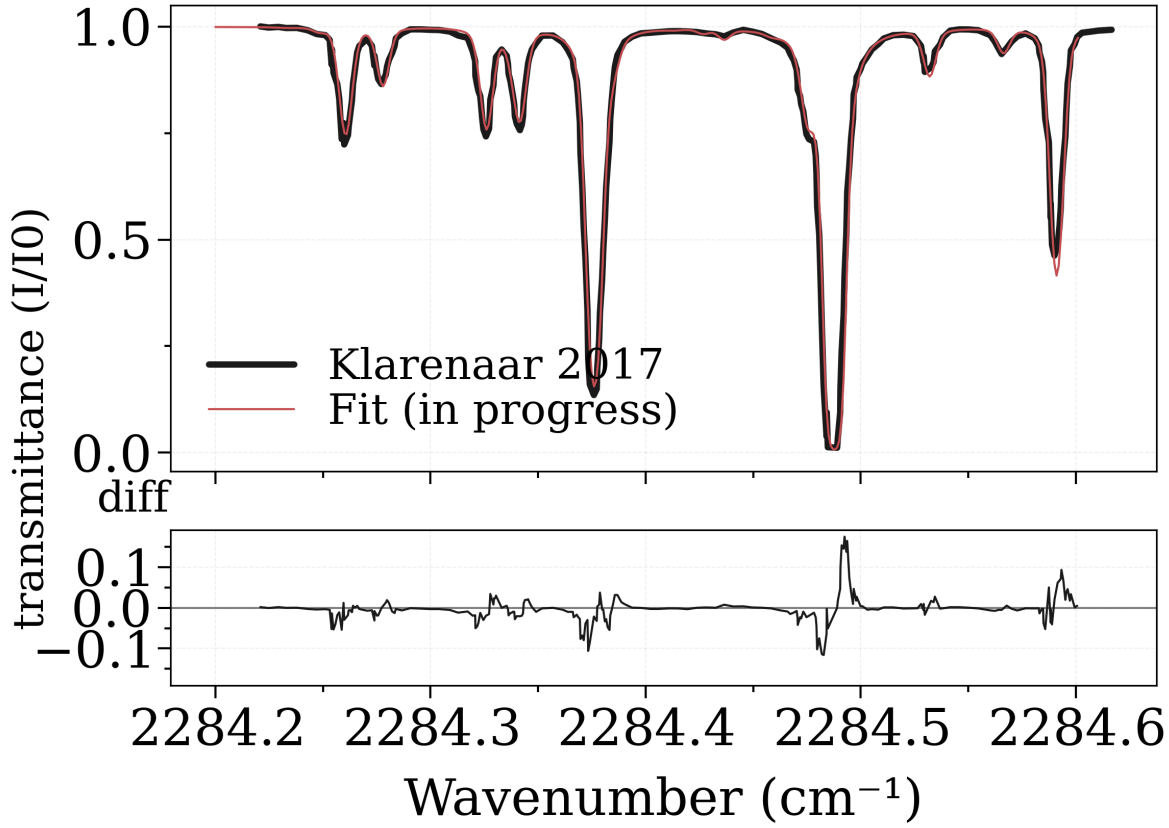
Calculate

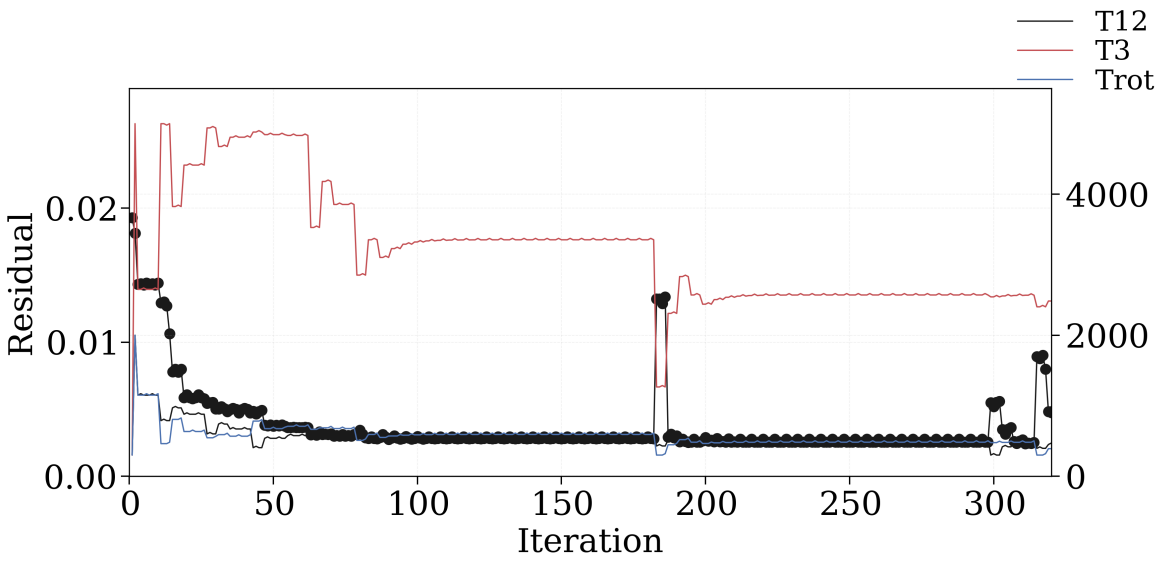
```
sf = SpectrumFactory(
    2284.2,
    2284.6,
    wstep=0.001,  # cm-1
    pressure=20 * 1e-3,  # bar
    db_use_cached=True,
    lvl_use_cached=True,
    cutoff=1e-25,
    isotope="1,2",
    path_length=10,  # cm-1
    mole_fraction=0.1 * 28.97 / 44.07,
    truncation=1,  # cm-1
    medium="vacuum",
    export_populations=None,  # 'vib',
    # parsum_mode="tabulation"
)
sf.warnings["MissingSelfBroadeningWarning"] = "ignore"
sf.warnings["PerformanceWarning"] = "ignore"
sf.load_databank("HITEMP-CO2-TEST")

s_best, best = sf.fit_spectrum(
    s_exp.take("transmittance_noslit"),
    model=Tvib12Tvib3Trot_NonLTEModel,
    fit_parameters={
        "T12": 517,
        "T3": 2641,
        "Trot": 491,
    },
    bounds={"T12": [300, 2000], "T3": [300, 5000], "Trot": [300, 2000]},
    fixed_parameters={"vib_distribution": "treanor"},
```

(continues on next page)

(continued from previous page)

```
    plot=True,
    solver_options={
        "method": "TNC",
        "maxiter": 80,  #  increase to let the fit converge
    },
)
```





```
Using database: HITEMP-CO2-TEST
'HITEMP-CO2-TEST':
{'info': 'HITEMP-2010, CO2, 3 main isotope (CO2-626, 636, 628), 2283.7-2285.1 cm-1',
 ↪'path': ['/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/
 ↪python3.8/site-packages/radis/test/files/cdsd_hitemp_09_fragment.txt'], 'format':
 ↪'cdsd-hitemp', 'parfuncfmt': 'hapi', 'levelsfmt': 'radis'}



/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
 ↪packages/radis/misc/warning.py:354: MissingReferenceWarning:

Missing doi for CDSD-HITEMP. Use HITEMP-2010?

Calculating Non-Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 491 K
   Trot                 491 K
   Tvib                 517,517,2641 K
   isotope              1,2
   mole_fraction        0.06573632856818698
   molecule             CO2
   path_length          10 cm
   pressure_mbar        20.0 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     treanor
   wavenum_max          2284.6000 cm-1
   wavenum_min          2284.2000 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-25 cm-1/(#.cm-2)
   dbformat             cdsd-hitemp
   dbpath               /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/
 ↪master/lib/python3.8/site-packages/radis...
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pseudo_continuum_threshold  0
   sparse_ldm           auto
   truncation           1 cm-1
   waveunit             cm-1
   wstep                0.001 cm-1
   zero_padding         -1
```

(continues on next page)

(continued from previous page)

```
----------------------------------------
Calculating energy levels with Dunham expansion for CO2(X1u+)(iso1)

(0s)    0.0%
(1s)    2.6%
(1s)    5.3%
(2s)    7.9%
(2s)    10.5%
(3s)    13.2%
(4s)    15.8%
(4s)    18.4%
(5s)    21.1%
(5s)    23.7%
(5s)    26.3%
(5s)    28.9%
(6s)    31.6%
(6s)    34.2%
(6s)    36.8%
(7s)    39.5%
(7s)    42.1%
(7s)    44.7%
(7s)    47.4%
(7s)    50.0%
(7s)    52.6%
(7s)    55.3%
(7s)    57.9%
(7s)    60.5%
(8s)    63.2%
(8s)    65.8%
(8s)    68.4%
(8s)    71.1%
(8s)    73.7%
(8s)    76.3%
(8s)    78.9%
(8s)    81.6%
(8s)    84.2%
(8s)    86.8%
(8s)    89.5%
(8s)    92.1%
(8s)    94.7%
(8s)    97.4%
(8s)    100.0%
Database generated up to v1=37, v2=65, v3=22, J=344
Calculating energy levels with Dunham expansion for CO2(X1u+)(iso2)

(0s)    0.0%
(1s)    2.6%
(1s)    5.3%
(2s)    7.9%
(3s)    10.5%
(3s)    13.2%
(4s)    15.8%
```

(continues on next page)

(continued from previous page)

```
(4s)    18.4%
(5s)    21.1%
(5s)    23.7%
(5s)    26.3%
(6s)    28.9%
(6s)    31.6%
(6s)    34.2%
(6s)    36.8%
(7s)    39.5%
(7s)    42.1%
(7s)    44.7%
(7s)    47.4%
(7s)    50.0%
(8s)    52.6%
(8s)    55.3%
(8s)    57.9%
(8s)    60.5%
(8s)    63.2%
(8s)    65.8%
(8s)    68.4%
(8s)    71.1%
(8s)    73.7%
(8s)    76.3%
(8s)    78.9%
(8s)    81.6%
(8s)    84.2%
(8s)    86.8%
(8s)    89.5%
(8s)    92.1%
(8s)    94.7%
(8s)    97.4%
(8s)    100.0%
Database generated up to v1=37, v2=67, v3=23, J=344
Fetching Evib & Erot. If using this code several times you should consider updating the␣
↪database directly. See functions in factory.py
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: NegativeEnergiesWarning:

There are negative rotational energies in the database

27.57s - Spectrum calculated
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: NegativeEnergiesWarning:

There are negative rotational energies in the database

------------------------------
TYPICAL FIT CALCULATION TIME:
Fit (in progress) profiler :
    spectrum_calculation      0.126s
        check_line_databank                 0.001s
        check_non_eq_param                  0.000s
```

(continues on next page)

(continued from previous page)

```
        reinitialize                        0.003s
            copy_database                       0.001s
            memory_usage_warning                0.003s
            reset_population                    0.000s
        calc_noneq_population_multiTvib     0.090s
            part_function                       0.084s
            others                              0.006s
        scaled_non_eq_linestrength          0.003s
        calc_emission_integral              0.004s
        applied_linestrength_cutoff         0.002s
        calc_lineshift                      0.001s
        calc_hwhm                           0.005s
        generate_wavenumber_arrays          0.000s
        calc_line_broadening                0.016s
            precompute_LDM_lineshapes           0.002s
            LDM_Initialized_vectors             0.000s
            LDM_closest_matching_line           0.000s
            LDM_Distribute_lines                0.010s
            LDM_convolve                        0.003s
        calc_other_spectral_quan            0.000s
        generate_spectrum_obj               0.000s
        others                              0.006s
------------------------------
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: NegativeEnergiesWarning:

There are negative rotational energies in the database

/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: NegativeEnergiesWarning:

There are negative rotational energies in the database


Now starting the fitting process:
--------------------------------

T12=1150.0,T3=2650.0,Trot=1150.0, Residual: 0.0143
T12=1170.0,T3=2650.0,Trot=1150.0, Residual: 0.0144
T12=1150.0,T3=2670.0,Trot=1150.0, Residual: 0.0143
T12=1150.0,T3=2650.0,Trot=1170.0, Residual: 0.0144
T12=1150.0,T3=2650.0,Trot=1150.0, Residual: 0.0143
T12=1170.0,T3=2650.0,Trot=1150.0, Residual: 0.0144
T12=1150.0,T3=2670.0,Trot=1150.0, Residual: 0.0143
T12=1150.0,T3=2650.0,Trot=1170.0, Residual: 0.0144
T12=789.0,T3=5000.0,Trot=462.0, Residual: 0.0129
T12=809.0,T3=5000.0,Trot=462.0, Residual: 0.0130
T12=789.0,T3=4980.0,Trot=462.0, Residual: 0.0127
T12=789.0,T3=5000.0,Trot=482.0, Residual: 0.0106
T12=969.0,T3=3825.0,Trot=806.0, Residual: 0.0078
T12=989.0,T3=3825.0,Trot=806.0, Residual: 0.0080
T12=969.0,T3=3845.0,Trot=806.0, Residual: 0.0078
```

(continues on next page)

(continued from previous page)

```
T12=969.0,T3=3825.0,Trot=826.0, Residual: 0.0080
T12=879.0,T3=4412.0,Trot=634.0, Residual: 0.0059
T12=899.0,T3=4412.0,Trot=634.0, Residual: 0.0061
T12=879.0,T3=4432.0,Trot=634.0, Residual: 0.0059
T12=879.0,T3=4412.0,Trot=654.0, Residual: 0.0058
T12=879.0,T3=4413.0,Trot=634.0, Residual: 0.0059
T12=899.0,T3=4413.0,Trot=634.0, Residual: 0.0061
T12=879.0,T3=4433.0,Trot=634.0, Residual: 0.0059
T12=879.0,T3=4413.0,Trot=654.0, Residual: 0.0058
T12=600.0,T3=4940.0,Trot=545.0, Residual: 0.0054
T12=620.0,T3=4940.0,Trot=545.0, Residual: 0.0055
T12=600.0,T3=4960.0,Trot=545.0, Residual: 0.0055
T12=600.0,T3=4940.0,Trot=565.0, Residual: 0.0050
T12=740.0,T3=4676.0,Trot=590.0, Residual: 0.0050
T12=760.0,T3=4676.0,Trot=590.0, Residual: 0.0052
T12=740.0,T3=4696.0,Trot=590.0, Residual: 0.0050
T12=740.0,T3=4676.0,Trot=610.0, Residual: 0.0048
T12=670.0,T3=4808.0,Trot=567.0, Residual: 0.0050
T12=690.0,T3=4808.0,Trot=567.0, Residual: 0.0051
T12=670.0,T3=4828.0,Trot=567.0, Residual: 0.0050
T12=670.0,T3=4808.0,Trot=587.0, Residual: 0.0047
T12=670.0,T3=4808.0,Trot=567.0, Residual: 0.0050
T12=690.0,T3=4808.0,Trot=567.0, Residual: 0.0051
T12=670.0,T3=4828.0,Trot=567.0, Residual: 0.0050
T12=670.0,T3=4808.0,Trot=587.0, Residual: 0.0047
T12=405.0,T3=4881.0,Trot=782.0, Residual: 0.0048
T12=425.0,T3=4881.0,Trot=782.0, Residual: 0.0046
T12=405.0,T3=4901.0,Trot=782.0, Residual: 0.0048
T12=405.0,T3=4881.0,Trot=802.0, Residual: 0.0049
T12=537.0,T3=4845.0,Trot=675.0, Residual: 0.0038
T12=557.0,T3=4845.0,Trot=675.0, Residual: 0.0038
T12=537.0,T3=4865.0,Trot=675.0, Residual: 0.0038
T12=537.0,T3=4845.0,Trot=695.0, Residual: 0.0037
T12=537.0,T3=4845.0,Trot=675.0, Residual: 0.0038
T12=557.0,T3=4845.0,Trot=675.0, Residual: 0.0038
T12=537.0,T3=4865.0,Trot=675.0, Residual: 0.0038
T12=537.0,T3=4845.0,Trot=695.0, Residual: 0.0037
T12=576.0,T3=4833.0,Trot=708.0, Residual: 0.0036
T12=596.0,T3=4833.0,Trot=708.0, Residual: 0.0036
T12=576.0,T3=4853.0,Trot=708.0, Residual: 0.0037
T12=576.0,T3=4833.0,Trot=728.0, Residual: 0.0036
T12=576.0,T3=4833.0,Trot=708.0, Residual: 0.0036
T12=596.0,T3=4833.0,Trot=708.0, Residual: 0.0036
T12=576.0,T3=4853.0,Trot=708.0, Residual: 0.0037
T12=576.0,T3=4833.0,Trot=728.0, Residual: 0.0036
T12=615.0,T3=3528.0,Trot=665.0, Residual: 0.0031
T12=635.0,T3=3528.0,Trot=665.0, Residual: 0.0032
T12=615.0,T3=3548.0,Trot=665.0, Residual: 0.0031
T12=615.0,T3=3528.0,Trot=685.0, Residual: 0.0033
T12=595.0,T3=4180.0,Trot=687.0, Residual: 0.0031
T12=615.0,T3=4180.0,Trot=687.0, Residual: 0.0031
T12=595.0,T3=4200.0,Trot=687.0, Residual: 0.0031
```

(continues on next page)

(continued from previous page)

```
T12=595.0,T3=4180.0,Trot=707.0, Residual: 0.0032
T12=605.0,T3=3854.0,Trot=676.0, Residual: 0.0030
T12=625.0,T3=3854.0,Trot=676.0, Residual: 0.0030
T12=605.0,T3=3874.0,Trot=676.0, Residual: 0.0030
T12=605.0,T3=3854.0,Trot=696.0, Residual: 0.0031
T12=605.0,T3=3854.0,Trot=676.0, Residual: 0.0030
T12=625.0,T3=3854.0,Trot=676.0, Residual: 0.0030
T12=605.0,T3=3874.0,Trot=676.0, Residual: 0.0030
T12=605.0,T3=3854.0,Trot=696.0, Residual: 0.0031
T12=588.0,T3=2852.0,Trot=508.0, Residual: 0.0031
T12=608.0,T3=2852.0,Trot=508.0, Residual: 0.0034
T12=588.0,T3=2872.0,Trot=508.0, Residual: 0.0032
T12=588.0,T3=2852.0,Trot=528.0, Residual: 0.0029
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=593.0,T3=3103.0,Trot=550.0, Residual: 0.0029
T12=613.0,T3=3103.0,Trot=550.0, Residual: 0.0031
T12=593.0,T3=3123.0,Trot=550.0, Residual: 0.0029
T12=593.0,T3=3103.0,Trot=570.0, Residual: 0.0028
T12=595.0,T3=3228.0,Trot=571.0, Residual: 0.0028
T12=615.0,T3=3228.0,Trot=571.0, Residual: 0.0030
T12=595.0,T3=3248.0,Trot=571.0, Residual: 0.0029
T12=595.0,T3=3228.0,Trot=591.0, Residual: 0.0028
T12=596.0,T3=3291.0,Trot=582.0, Residual: 0.0028
T12=616.0,T3=3291.0,Trot=582.0, Residual: 0.0030
T12=596.0,T3=3311.0,Trot=582.0, Residual: 0.0028
T12=596.0,T3=3291.0,Trot=602.0, Residual: 0.0028
T12=596.0,T3=3322.0,Trot=587.0, Residual: 0.0028
T12=616.0,T3=3322.0,Trot=587.0, Residual: 0.0030
T12=596.0,T3=3342.0,Trot=587.0, Residual: 0.0028
T12=596.0,T3=3322.0,Trot=607.0, Residual: 0.0028
T12=597.0,T3=3338.0,Trot=589.0, Residual: 0.0028
T12=617.0,T3=3338.0,Trot=589.0, Residual: 0.0030
T12=597.0,T3=3358.0,Trot=589.0, Residual: 0.0028
T12=597.0,T3=3338.0,Trot=609.0, Residual: 0.0028
T12=597.0,T3=3346.0,Trot=591.0, Residual: 0.0028
T12=617.0,T3=3346.0,Trot=591.0, Residual: 0.0030
T12=597.0,T3=3366.0,Trot=591.0, Residual: 0.0028
T12=597.0,T3=3346.0,Trot=611.0, Residual: 0.0028
T12=597.0,T3=3349.0,Trot=591.0, Residual: 0.0028
T12=617.0,T3=3349.0,Trot=591.0, Residual: 0.0030
T12=597.0,T3=3369.0,Trot=591.0, Residual: 0.0028
T12=597.0,T3=3349.0,Trot=611.0, Residual: 0.0028
T12=597.0,T3=3351.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3351.0,Trot=592.0, Residual: 0.0030
T12=597.0,T3=3371.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3351.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3352.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3352.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3372.0,Trot=592.0, Residual: 0.0028
```

(continues on next page)

(continued from previous page)

```
T12=597.0,T3=3352.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
```

(continues on next page)

(continued from previous page)

```
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=592.0, Residual: 0.0028
T12=617.0,T3=3353.0,Trot=592.0, Residual: 0.0029
T12=597.0,T3=3373.0,Trot=592.0, Residual: 0.0028
T12=597.0,T3=3353.0,Trot=612.0, Residual: 0.0028
T12=429.0,T3=1266.0,Trot=300.0, Residual: 0.0132
T12=449.0,T3=1266.0,Trot=300.0, Residual: 0.0132
T12=429.0,T3=1286.0,Trot=300.0, Residual: 0.0129
T12=429.0,T3=1266.0,Trot=320.0, Residual: 0.0134
T12=513.0,T3=2310.0,Trot=446.0, Residual: 0.0029
T12=533.0,T3=2310.0,Trot=446.0, Residual: 0.0031
T12=513.0,T3=2330.0,Trot=446.0, Residual: 0.0029
T12=513.0,T3=2310.0,Trot=466.0, Residual: 0.0030
T12=555.0,T3=2831.0,Trot=519.0, Residual: 0.0026
T12=575.0,T3=2831.0,Trot=519.0, Residual: 0.0028
T12=555.0,T3=2851.0,Trot=519.0, Residual: 0.0026
T12=555.0,T3=2831.0,Trot=539.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=523.0,T3=2440.0,Trot=464.0, Residual: 0.0027
T12=543.0,T3=2440.0,Trot=464.0, Residual: 0.0029
T12=523.0,T3=2460.0,Trot=464.0, Residual: 0.0026
T12=523.0,T3=2440.0,Trot=484.0, Residual: 0.0027
T12=529.0,T3=2505.0,Trot=473.0, Residual: 0.0026
T12=549.0,T3=2505.0,Trot=473.0, Residual: 0.0028
T12=529.0,T3=2525.0,Trot=473.0, Residual: 0.0026
T12=529.0,T3=2505.0,Trot=493.0, Residual: 0.0026
T12=531.0,T3=2538.0,Trot=478.0, Residual: 0.0026
T12=551.0,T3=2538.0,Trot=478.0, Residual: 0.0028
T12=531.0,T3=2558.0,Trot=478.0, Residual: 0.0026
T12=531.0,T3=2538.0,Trot=498.0, Residual: 0.0026
T12=532.0,T3=2554.0,Trot=480.0, Residual: 0.0025
T12=552.0,T3=2554.0,Trot=480.0, Residual: 0.0028
T12=532.0,T3=2574.0,Trot=480.0, Residual: 0.0025
T12=532.0,T3=2554.0,Trot=500.0, Residual: 0.0026
T12=533.0,T3=2562.0,Trot=481.0, Residual: 0.0025
T12=553.0,T3=2562.0,Trot=481.0, Residual: 0.0028
T12=533.0,T3=2582.0,Trot=481.0, Residual: 0.0025
T12=533.0,T3=2562.0,Trot=501.0, Residual: 0.0026
T12=533.0,T3=2567.0,Trot=482.0, Residual: 0.0025
T12=553.0,T3=2567.0,Trot=482.0, Residual: 0.0028
T12=533.0,T3=2587.0,Trot=482.0, Residual: 0.0025
T12=533.0,T3=2567.0,Trot=502.0, Residual: 0.0026
T12=534.0,T3=2569.0,Trot=482.0, Residual: 0.0025
T12=554.0,T3=2569.0,Trot=482.0, Residual: 0.0028
T12=534.0,T3=2589.0,Trot=482.0, Residual: 0.0025
```

(continues on next page)

(continued from previous page)

```
T12=534.0,T3=2569.0,Trot=502.0, Residual: 0.0025
T12=534.0,T3=2570.0,Trot=482.0, Residual: 0.0025
T12=554.0,T3=2570.0,Trot=482.0, Residual: 0.0028
T12=534.0,T3=2590.0,Trot=482.0, Residual: 0.0025
T12=534.0,T3=2570.0,Trot=502.0, Residual: 0.0025
T12=534.0,T3=2570.0,Trot=482.0, Residual: 0.0025
T12=554.0,T3=2570.0,Trot=482.0, Residual: 0.0028
T12=534.0,T3=2590.0,Trot=482.0, Residual: 0.0025
T12=534.0,T3=2570.0,Trot=502.0, Residual: 0.0025
T12=534.0,T3=2570.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2570.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2590.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2570.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2570.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2570.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2590.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2570.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
```

(continues on next page)

(continued from previous page)

```
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=483.0, Residual: 0.0025
T12=554.0,T3=2571.0,Trot=483.0, Residual: 0.0028
T12=534.0,T3=2591.0,Trot=483.0, Residual: 0.0025
T12=534.0,T3=2571.0,Trot=503.0, Residual: 0.0025
T12=300.0,T3=2544.0,Trot=476.0, Residual: 0.0055
T12=320.0,T3=2544.0,Trot=476.0, Residual: 0.0052
T12=300.0,T3=2564.0,Trot=476.0, Residual: 0.0055
T12=300.0,T3=2544.0,Trot=496.0, Residual: 0.0056
T12=417.0,T3=2557.0,Trot=480.0, Residual: 0.0035
T12=437.0,T3=2557.0,Trot=480.0, Residual: 0.0031
T12=417.0,T3=2577.0,Trot=480.0, Residual: 0.0035
T12=417.0,T3=2557.0,Trot=500.0, Residual: 0.0036
T12=475.0,T3=2564.0,Trot=481.0, Residual: 0.0026
T12=495.0,T3=2564.0,Trot=481.0, Residual: 0.0025
T12=475.0,T3=2584.0,Trot=481.0, Residual: 0.0026
T12=475.0,T3=2564.0,Trot=501.0, Residual: 0.0027
T12=505.0,T3=2567.0,Trot=482.0, Residual: 0.0024
T12=525.0,T3=2567.0,Trot=482.0, Residual: 0.0025
T12=505.0,T3=2587.0,Trot=482.0, Residual: 0.0024
T12=505.0,T3=2567.0,Trot=502.0, Residual: 0.0025
T12=396.0,T3=2403.0,Trot=300.0, Residual: 0.0089
T12=416.0,T3=2403.0,Trot=300.0, Residual: 0.0088
T12=396.0,T3=2423.0,Trot=300.0, Residual: 0.0090
T12=396.0,T3=2403.0,Trot=320.0, Residual: 0.0080
T12=450.0,T3=2485.0,Trot=391.0, Residual: 0.0048
T12=470.0,T3=2485.0,Trot=391.0, Residual: 0.0048
T12=450.0,T3=2505.0,Trot=391.0, Residual: 0.0049
T12=450.0,T3=2485.0,Trot=411.0, Residual: 0.0040
T12=505.0,T3=2567.0,Trot=482.0, Residual: 0.0024
T12=525.0,T3=2567.0,Trot=482.0, Residual: 0.0025
T12=505.0,T3=2587.0,Trot=482.0, Residual: 0.0024
T12=505.0,T3=2567.0,Trot=502.0, Residual: 0.0025
Best ['T12', 'T3', 'Trot'] = [ 504.58098239 2567.21733341  481.80164708]['', '', '']
↪reached at iteration 310/324
```

Total running time of the script: (1 minutes 3.959 seconds)

2.6.28 1 temperature fit

Quickly fit an experimental spectrum with a one-temperature model, directly from `SpectrumFactory`, with `fit_spectrum()`

The method requires a fitting model. An example model is provided in `radis.tools.fitting` : `LTEModel()`. Other models can be used; such as in the *multi-temperature fit example*

More advanced tools for interactive fitting of multi-dimensional, multi-slabs spectra can be found in `fitroom`. Finally, the *GPU-accelerated example* shows how to obtain real-time interactive spectra.

```
from radis import SpectrumFactory, load_spec
```

Here we get an experimental spectrum from RADIS test cases. Use your own instead.

```
from radis.test.utils import getTestFile, setup_test_line_databases

setup_test_line_databases()

# fit range
wlmin = 4167
wlmax = 4180

s_exp = (
    load_spec(getTestFile("CO2_measured_spectrum_4-5um.spec"))
    .crop(wlmin, wlmax, "nm")
    .normalize()
    .sort()
    .offset(-0.2, "nm")
)
```

Customize the `LTEModel()` for our case: we add a slit (non fittable parameter) and normalize it

```
from radis.tools.fitting import LTEModel


def LTEModel_withslitnorm(factory, fit_parameters, fixed_parameters):
    s = LTEModel(factory, fit_parameters, fixed_parameters)
    # we could also have added a fittable parameter, such as an offset,
    # or made the slit width a fittable parameter.
    # ... any paramter in model_input will be fitted.
    # s.offset(model_input["offset"], 'nm')
    s.apply_slit(1.4, "nm")
    return s.take("radiance").normalize()
```

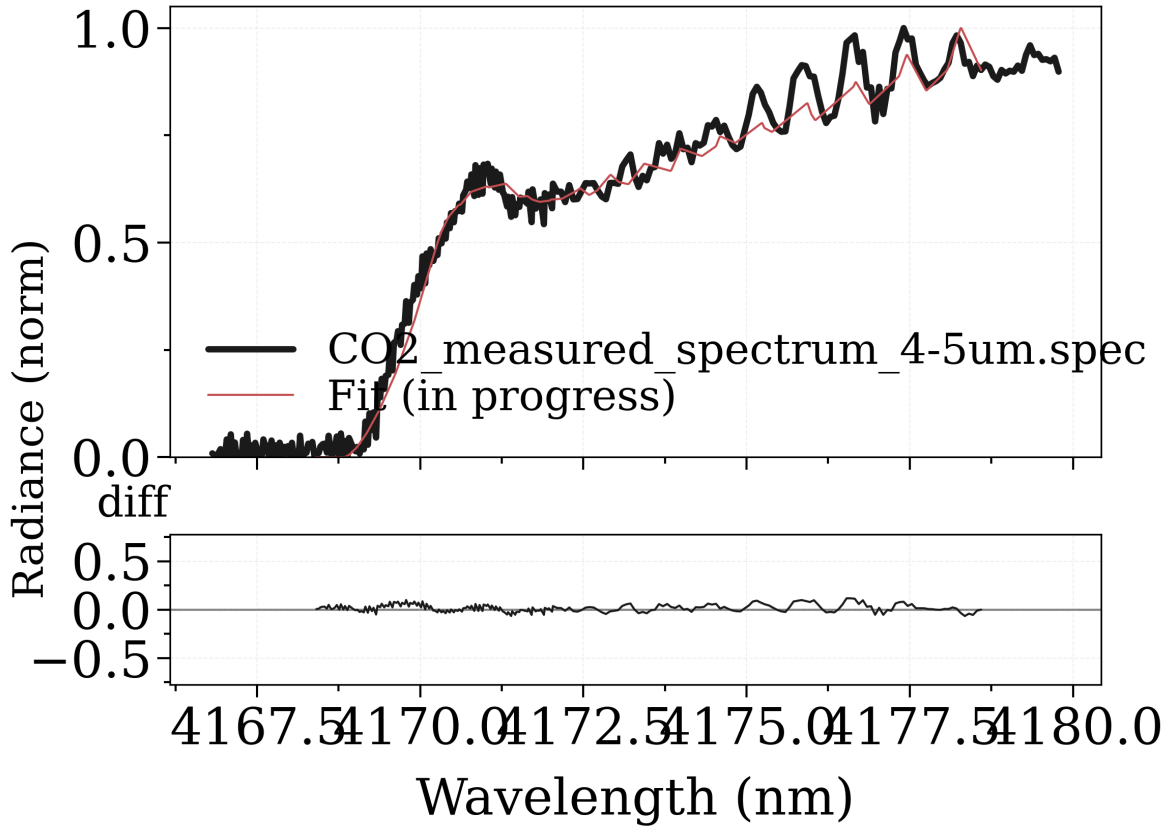
using `fit_spectrum()`

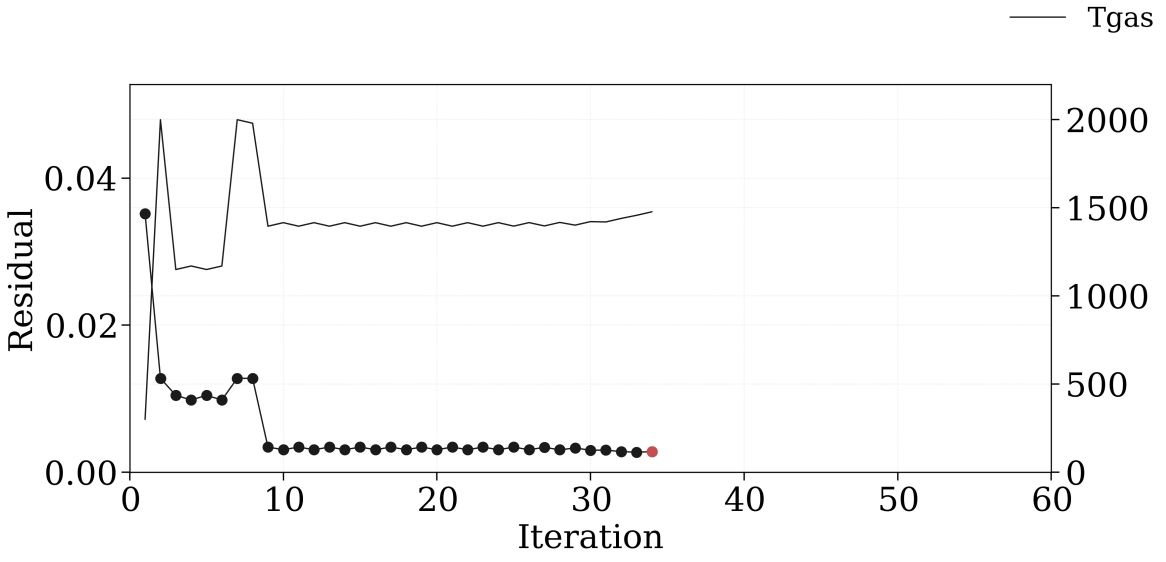
```
import astropy.units as u

sf = SpectrumFactory(
    wlmin * u.nm,
    wlmax * u.nm,
    wstep=0.001,  # cm-1
    pressure=1 * 1e-3,  # bar
    cutoff=1e-25,
```

(continues on next page)

(continued from previous page)

```
    isotope="1,2",
    path_length=10,  # cm-1
    mole_fraction=1,
    truncation=1,  # cm-1
)
sf.warnings["MissingSelfBroadeningWarning"] = "ignore"
sf.warnings["HighTemperatureWarning"] = "ignore"
sf.load_databank("HITRAN-CO2-TEST")

s_best, best = sf.fit_spectrum(
    s_exp.take("radiance"),
    model=LTEModel_withslitnorm,
    fit_parameters={
        "Tgas": 300,
        # "offset": 0
    },
    bounds={
        "Tgas": [300, 2000],
        # "offset": [-1, 1],
    },
    plot=True,
    solver_options={
        "maxiter": 15,  #  increase to let the fit converge
        "ftol": 1e-15,
    },
    verbose=2,
)
# plot_diff(s_exp, s_best)  # , show_ruler=True)
```






```
Using database: HITRAN-CO2-TEST
'HITRAN-CO2-TEST':
{'info': 'HITRAN 2016 database, CO2, 1 main isotope (CO2-626), bandhead: 2380-2398 cm-1␣
↪(4165-4200 nm)', 'path': ['/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/
↪master/lib/python3.8/site-packages/radis/test/files/hitran_co2_626_bandhead_4165_
↪4200nm.par'], 'format': 'hitran', 'parfuncfmt': 'hapi', 'levelsfmt': 'radis'}
```

(continues on next page)

(continued from previous page)

```
Generating cache file /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/
↪lib/python3.8/site-packages/radis/test/files/hitran_co2_626_bandhead_4165_4200nm.h5␣
↪with metadata :
{'last_modification': 'Sun Aug 28 21:55:18 2022', 'wavenum_min': 2380.019436, 'wavenum_
↪max': 2399.965532}

Reference databank (2391.79-2399.14cm-1) has 0 lines in range (2391.69-2399.15cm-1) for␣
↪isotope 2. Change your range or isotope options

/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/log.py:54: UserWarning:

Reference databank (2391.79-2399.14cm-1) has 0 lines in range (2391.69-2399.15cm-1) for␣
↪isotope 2. Change your range or isotope options

Calculating Equilibrium Spectrum
Physical Conditions
----------------------------------------
   Tgas                 300 K
   Trot                 300 K
   Tvib                 300 K
   isotope              1,2
   mole_fraction        1
   molecule             CO2
   overpopulation       None
   path_length          10 cm
   pressure_mbar        1.0 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   vib_distribution     boltzmann
   wavenum_max          2399.1537 cm-1
   wavenum_min          2391.6923 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   add_at_used
   broadening_method    voigt
   cutoff               1e-25 cm-1/(#.cm-2)
   dbformat             hitran
   dbpath               /home/docs/checkouts/readthedocs.org/user_builds/radis/envs/
↪master/lib/python3.8/site-packages/radis...
   folding_thresh       1e-06
   include_neighbouring_lines  True
   memory_mapping_engine  auto
   neighbour_lines      0 cm-1
   optimization         simple
   parfuncfmt           hapi
   parsum_mode          full summation
   pseudo_continuum_threshold  0
```

(continues on next page)

(continued from previous page)

```
   sparse_ldm           auto
   truncation           1 cm-1
   waveunit             cm-1
   wstep                0.001 cm-1
   zero_padding         -1
----------------------------------------
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: LinestrengthCutoffWarning:

Estimated error after discarding lines is large: 0.07%. Consider reducing cutoff

0.03s - Spectrum calculated
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: LinestrengthCutoffWarning:

Estimated error after discarding lines is large: 0.07%. Consider reducing cutoff

------------------------------
TYPICAL FIT CALCULATION TIME:
Fit (in progress) profiler :
    spectrum_calculation      0.025s
        check_line_databank              0.001s
        reinitialize                     0.002s
            copy_database                    0.000s
            memory_usage_warning             0.002s
            reset_population                 0.000s
        scaled_eq_linestrength           0.002s
        applied_linestrength_cutoff      0.001s
        calc_lineshift                   0.001s
        calc_hwhm                        0.003s
        generate_wavenumber_arrays       0.000s
        calc_line_broadening             0.011s
            precompute_LDM_lineshapes        0.003s
            LDM_Initialized_vectors          0.000s
            LDM_closest_matching_line        0.000s
            LDM_Distribute_lines             0.005s
            LDM_convolve                     0.003s
        calc_other_spectral_quan         0.003s
        generate_spectrum_obj            0.000s
------------------------------
/home/docs/checkouts/readthedocs.org/user_builds/radis/envs/master/lib/python3.8/site-
↪packages/radis/misc/warning.py:354: LinestrengthCutoffWarning:

Estimated error after discarding lines is large: 0.07%. Consider reducing cutoff


Now starting the fitting process:
---------------------------------

Tgas=1150.0, Residual: 0.0105
Tgas=1170.0, Residual: 0.0098
Tgas=1150.0, Residual: 0.0105
```

(continues on next page)

(continued from previous page)

```
Tgas=1170.0, Residual: 0.0098
Tgas=2000.0, Residual: 0.0128
Tgas=1980.0, Residual: 0.0128
Tgas=1396.0, Residual: 0.0034
Tgas=1416.0, Residual: 0.0031
Tgas=1396.0, Residual: 0.0034
Tgas=1416.0, Residual: 0.0031
Tgas=1396.0, Residual: 0.0034
Tgas=1416.0, Residual: 0.0031
Tgas=1396.0, Residual: 0.0034
Tgas=1416.0, Residual: 0.0031
Tgas=1396.0, Residual: 0.0034
Tgas=1416.0, Residual: 0.0031
Tgas=1396.0, Residual: 0.0034
Tgas=1416.0, Residual: 0.0031
Tgas=1396.0, Residual: 0.0034
Tgas=1416.0, Residual: 0.0031
Tgas=1396.0, Residual: 0.0034
Tgas=1416.0, Residual: 0.0031
Tgas=1396.0, Residual: 0.0034
Tgas=1416.0, Residual: 0.0031
Tgas=1397.0, Residual: 0.0034
Tgas=1417.0, Residual: 0.0030
Tgas=1402.0, Residual: 0.0033
Tgas=1422.0, Residual: 0.0030
Tgas=1420.0, Residual: 0.0030
Tgas=1440.0, Residual: 0.0028
Tgas=1458.0, Residual: 0.0027
Tgas=1478.0, Residual: 0.0028
Init ['Tgas'] = [1150.]['']
Final ['Tgas'] = [1458.]['']
      fun: 0.0027300270273360972
 hess_inv: <1x1 LbfgsInvHessProduct with dtype=float64>
      jac: array([3.1783589e-06])
  message: 'CONVERGENCE: NORM_OF_PROJECTED_GRADIENT_<=_PGTOL'
     nfev: 32
      nit: 4
     njev: 16
   status: 0
  success: True
        x: array([1457.50267416])
Best ['Tgas'] = [1457.50267416][''] reached at iteration 32/32
```

Total running time of the script: (0 minutes 3.451 seconds)

2.6.29 Scale Linestrengths of carbon-monoxide

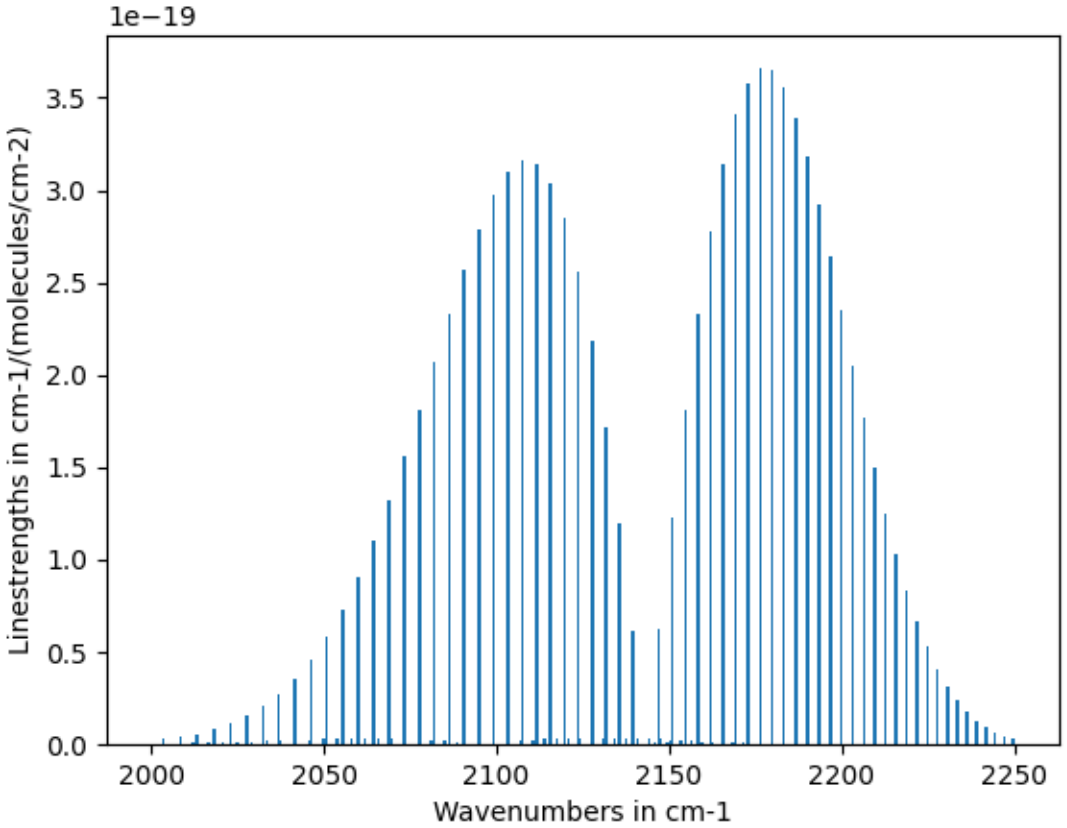
This example scales the linestrengths of CO to Tgas=300 from Tref = 296 and then plots the linestrengths against the wavenumbers. We are using `fetch_hitemp()` function to retrieve the dataframe from the HITEMP-CO databank.

References

$$S(T) = S_0 \frac{Q_{ref}}{Q_{gas}} \exp\left(-E_l\left(\frac{1}{T_{gas}} - \frac{1}{T_{ref}}\right)\right) \frac{1-\exp\left(\frac{-\omega_0}{T_{gas}}\right)}{1-\exp\left(\frac{-\omega_0}{T_{ref}}\right)}$$

See Eq.(A11) in [Rothman-1998]

Similar functions are used directly at the hearth of RADIS's SpectrumFactory in the `calc_linestrength_eq()` and `calc_linestrength_noneq()` methods




```
Scaling equilibrium linestrength
```

```
from matplotlib import pyplot as plt
from numpy import exp

from radis.db.classes import get_molecule, get_molecule_identifier
from radis.io.hitemp import fetch_hitemp
from radis.levels.partfunc import PartFuncTIPS
from radis.phys.constants import hc_k


def get_Qgas(molecule, iso, T):

    M = get_molecule_identifier(molecule)

    Q = PartFuncTIPS(M, iso)
    return Q.at(T=T)


def scale_linestrength_eq(df, Tref, Tgas):

    print("Scaling equilibrium linestrength")

    # %% Load partition function values

    def _calc_Q(molecule, iso, T_ref, T_gas):

        Qref = get_Qgas(molecule, iso, T_ref)
        Qgas = get_Qgas(molecule, iso, T_gas)

        return Qref, Qgas

    id_set = df.id.unique()
    id = list(id_set)[0]
    molecule = get_molecule(id)  # retrieve the molecule
    iso_set = set(df.iso)  # df1.iso.unique()

    Qref_Qgas_ratio = {}

    for iso in iso_set:
        Qref, Qgas = _calc_Q(molecule, iso, Tref, Tgas)
        Qref_Qgas_ratio[iso] = Qref / Qgas

    # Scaling linestrength with the equations from Rotham's paper
    line_strength = df.int * df["iso"].map(Qref_Qgas_ratio)
    line_strength *= exp(-hc_k * df.El * (1 / Tgas - 1 / Tref))
    line_strength *= (1 - exp(-hc_k * df.wav / Tgas)) / (1 - exp(-hc_k * df.wav / Tref))
    # Add a fresh columns with the scaled linestrength
    df["S"] = line_strength  # [cm-1/(molecules/cm-2)]

    # Just to make sure linestrength is indeed added
    assert "S" in df

    return df
```

(continues on next page)

(continued from previous page)

```
if __name__ == "__main__":
    Tref = 296
    df = fetch_hitemp(
        molecule="CO",
        isotope="1, 2, 3",
        load_wavenum_min=2000,
        load_wavenum_max=2250,
    )

    Tgas = 450

    df = scale_linestrength_eq(df, Tref, Tgas)
    plt.bar(df["wav"], df["S"])
    plt.xlabel("Wavenumbers in cm-1")
    plt.ylabel("Linestrengths in cm-1/(molecules/cm-2)")
    plt.show()
```

Total running time of the script: (0 minutes 5.720 seconds)

2.7 HITRAN spectra

The absorption coefficient of all HITRAN species (see `MOLECULES_LIST_EQUILIBRIUM`) is calculated in plot_all_hitran_spectra.py at 300 K, 1 atm for the first isotope:

- 1 `'H2O'` : Water (spectrum)
- 2 `'CO2'` : Carbon Dioxide (spectrum)
- 3 `'O3'` : Ozone (spectrum)
- 4 `'N2O'` : Nitrogen oxide (spectrum)
- 5 `'CO'` : Carbon Monoxide (spectrum)
- 6 `'CH4'` : Methane (spectrum)
- 7 `'O2'` : Oxygen
- 8 `'NO'` : Nitric Oxide (spectrum)
- 9 `'SO2'` : Sulfur Dioxide (spectrum)
- 10 `'NO2'` : Nitrogen Dioxide (spectrum)
- 11 `'NH3'` : Ammonia (spectrum)
- 12 `'HNO3'` : Nitric Acid (spectrum)
- 13 `'OH'` : Hydroxyl (spectrum)
- 14 `'HF'` : Hydrogen Fluoride (spectrum)
- 15 `'HCl'` : Hydrogen Chloride (spectrum)
- 16 `'HBr'` : Hydrogen Bromide (spectrum)
- 17 `'HI'` : Hydrogen Iodide (spectrum)
- 18 `'ClO'` : Chlorine Monoxide (spectrum)

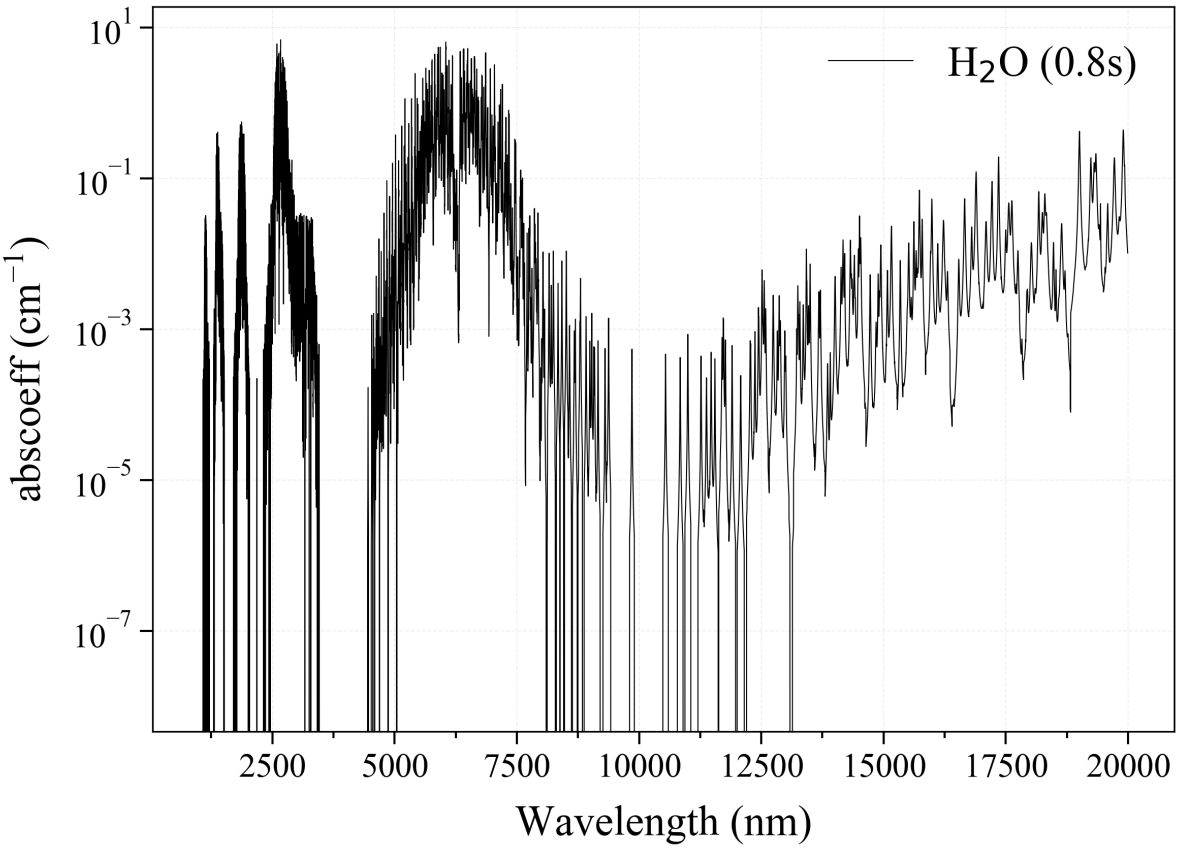
- 19 `'OCS'` : Carbonyl Sulfide (spectrum)
- 20 `'H2CO'` : Formaldehyde (spectrum)
- 21 `'HOCl'` : Hypochlorous Acid (spectrum)
- 22 `'N2'` : Nitrogen
- 23 `'HCN'` : Hydrogen Cyanide
- 24 `'CH3Cl'` : Methyl Chloride (spectrum)
- 25 `'H2O2'` : Hydrogen Peroxide (spectrum)
- 26 `'C2H2'` : Acetylene (spectrum)
- 27 `'C2H6'` : Ethane (spectrum)
- 28 `'PH3'` : Phosphine (spectrum)
- 29 `'COF2'` : Carbonyl Fluoride (spectrum)
- 30 `'SF6'` : Sulfur Hexafluoride
- 31 `'H2S'` : Hydrogen Sulfide (spectrum)
- 32 `'HCOOH'` : Formic Acid (spectrum)
- 33 `'HO2'` : Hydroperoxyl (spectrum)
- 34 `'O'` : Oxygen Atom
- 35 `'ClONO2'` : Chlorine Nitrate
- 36 `'NO+'` : Nitric Oxide Cation (spectrum)
- 37 `'HOBr'` : Hypobromous Acid
- 38 `'C2H4'` : Ethylene
- 39 `'CH3OH'` : Methanol
- 40 `'CH3Br'` : Methyl Bromide
- 41 `'CH3CN'` : Acetonitrile
- 42 `'CF4'` : CFC-14
- 43 `'C4H2'` : Diacetylene
- 44 `'HC3N'` : Cyanoacetylene
- 45 `'H2'` : Hydrogen
- 46 `'CS'` : Carbon Monosulfide
- 47 `'SO3'` : Sulfur trioxide
- 48 `'C2N2'` : Cyanogen
- 49 `'COCl2'` : Phosgene

The code to calculate each molecule is shown below:

2.7.1 1. H2O

- 1 `'H2O'` : Water absorption coefficient (opacity) at 300 K

```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='H2O',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.2 2. CO2

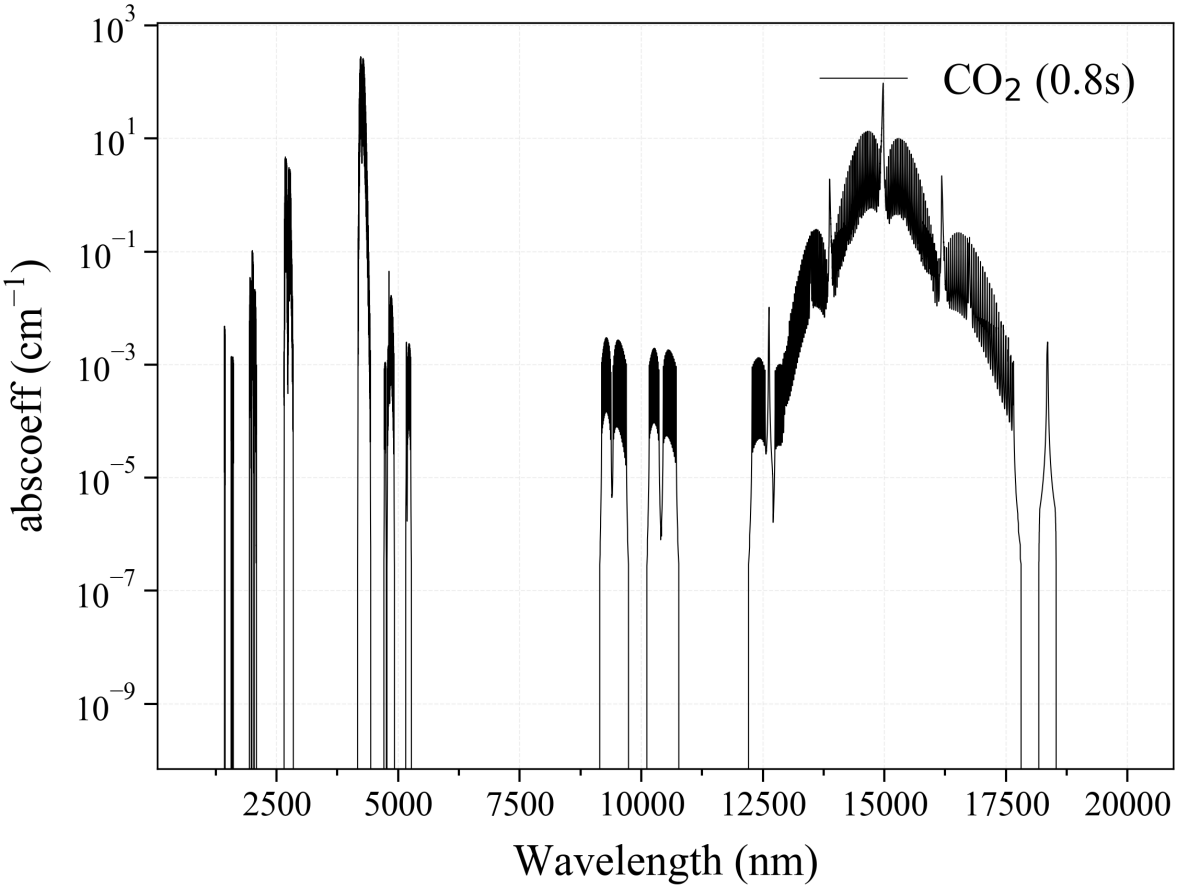
- 2 `'CO2'` : Carbon Dioxide absorption coefficient (opacity) at 300 K

```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='CO2',
                  optimization=None,
```

(continues on next page)

(continued from previous page)

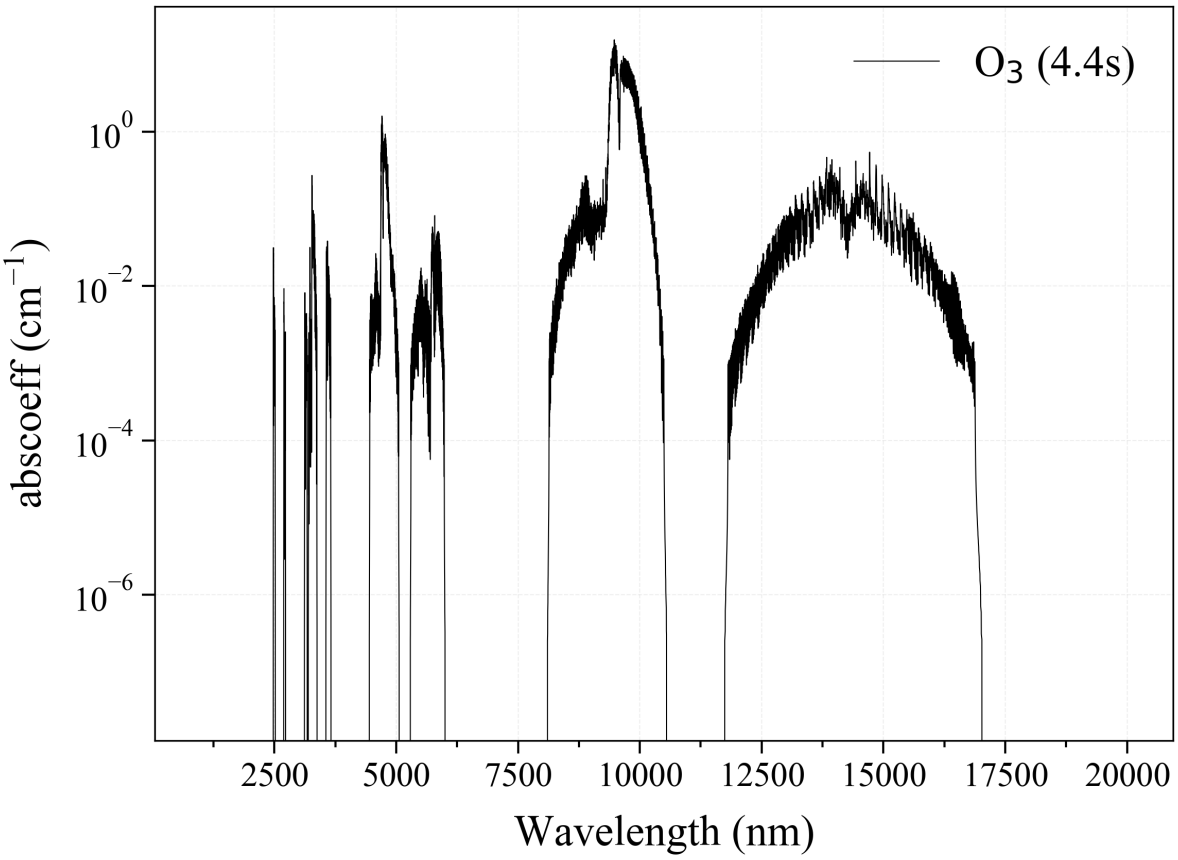
```
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



3. O3 =====-

- 3 `'O3'` : Ozone absorption coefficient (opacity) at 300 K

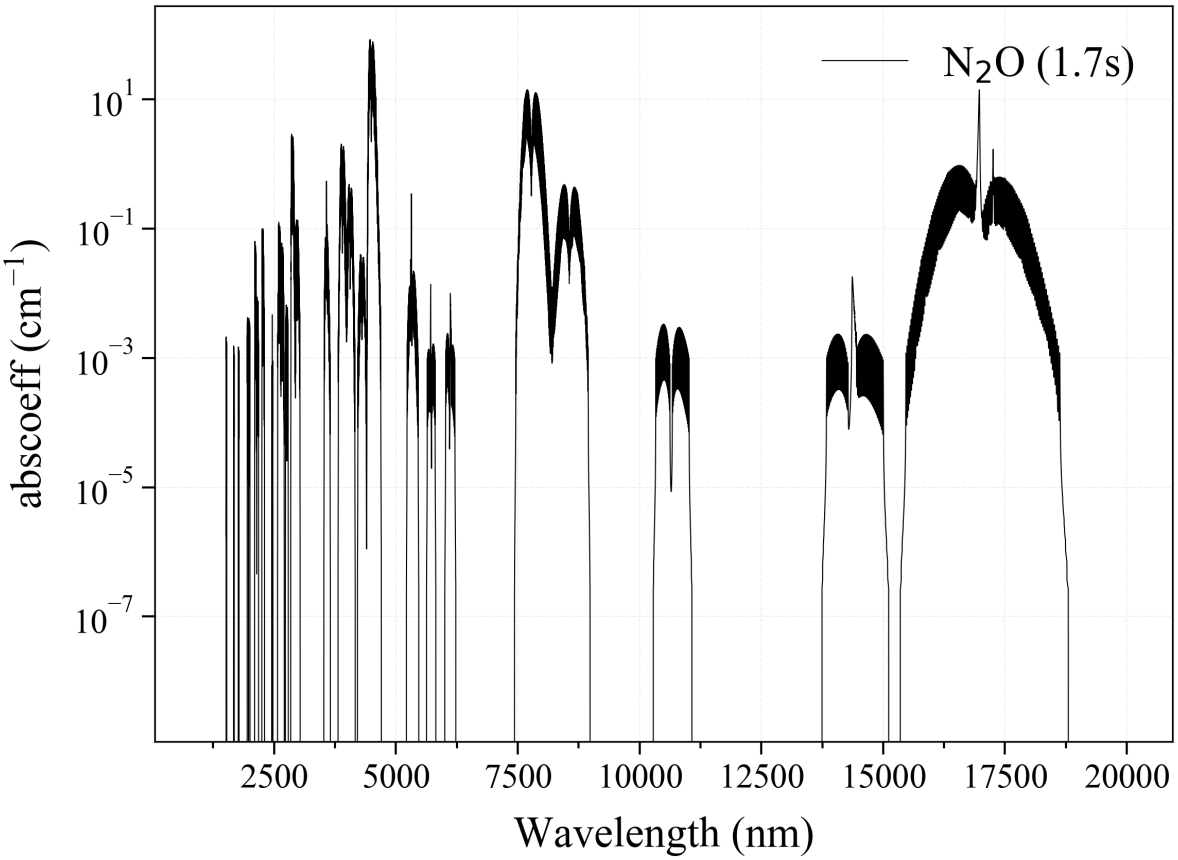
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='O3',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.3 4. N2O

- 4 `'N2O'` : Nitrogen oxide absorption coefficient (opacity) at 300 K

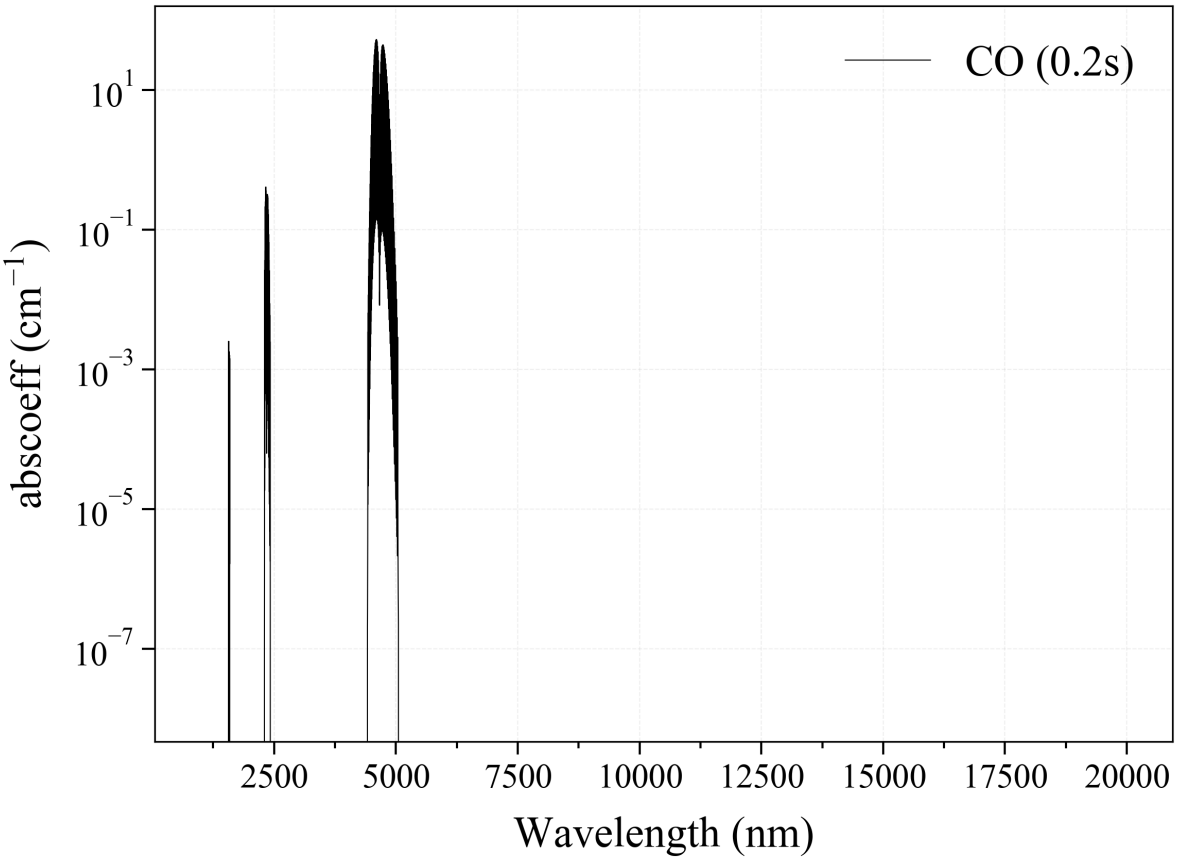
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='N2O',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.4 5. CO

- 5 `'CO'` : Carbon Monoxide absorption coefficient (opacity) at 300 K

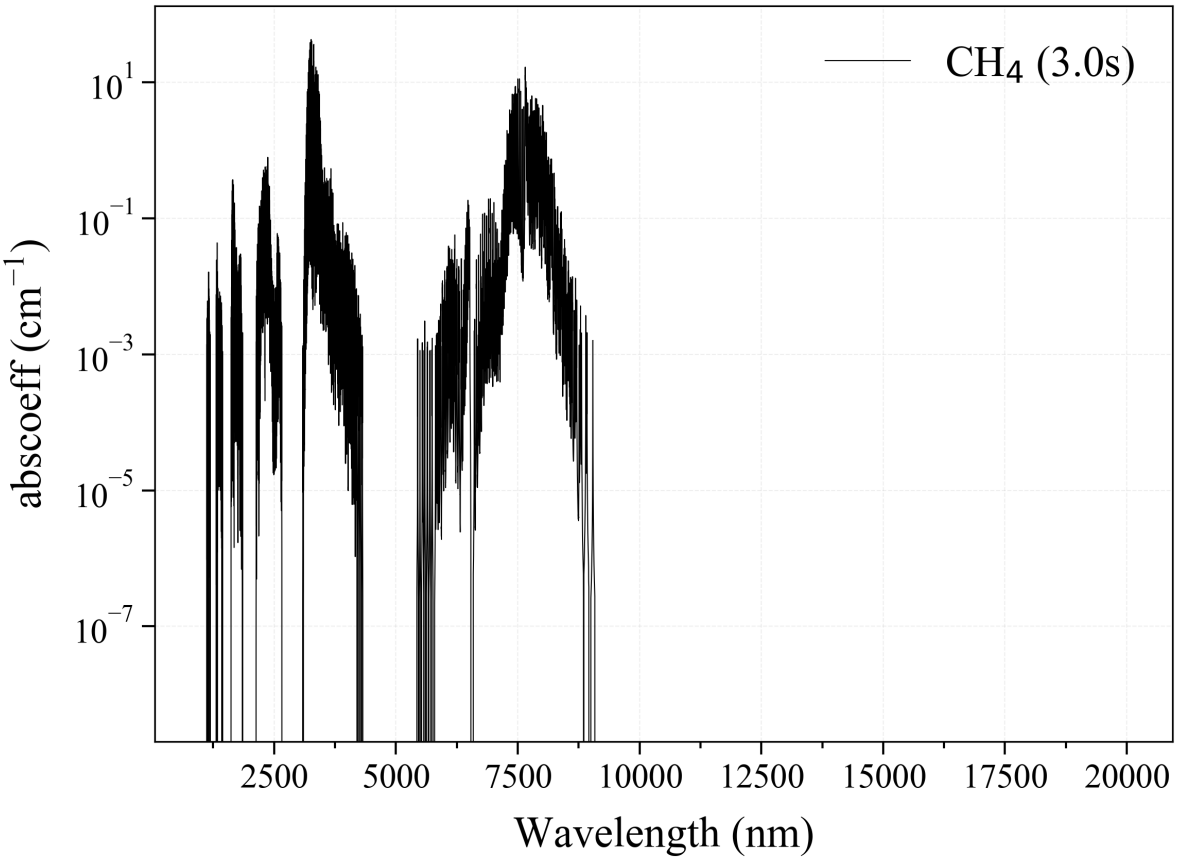
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='CO',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.5 6. CH4

- 6 `'CH4'` : Methane absorption coefficient (opacity) at 300 K

```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='CH4',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



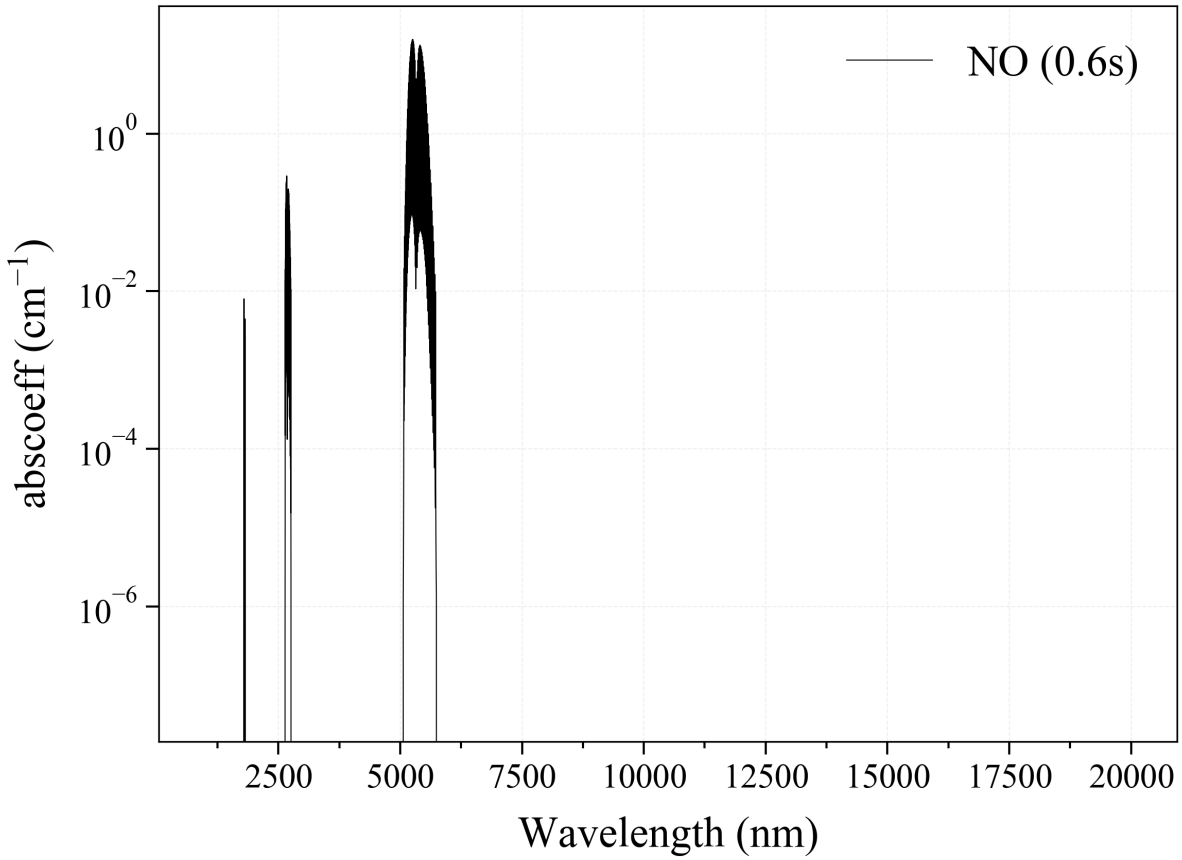
7. O2 =====-

- 7 `'O2'` : Oxygen absorption coefficient (opacity) at 300 K : no lines for `isotope='1'` (symmetric!)

2.7.6 8. NO

- 8 `'NO'` : Nitric Oxide absorption coefficient (opacity) at 300 K

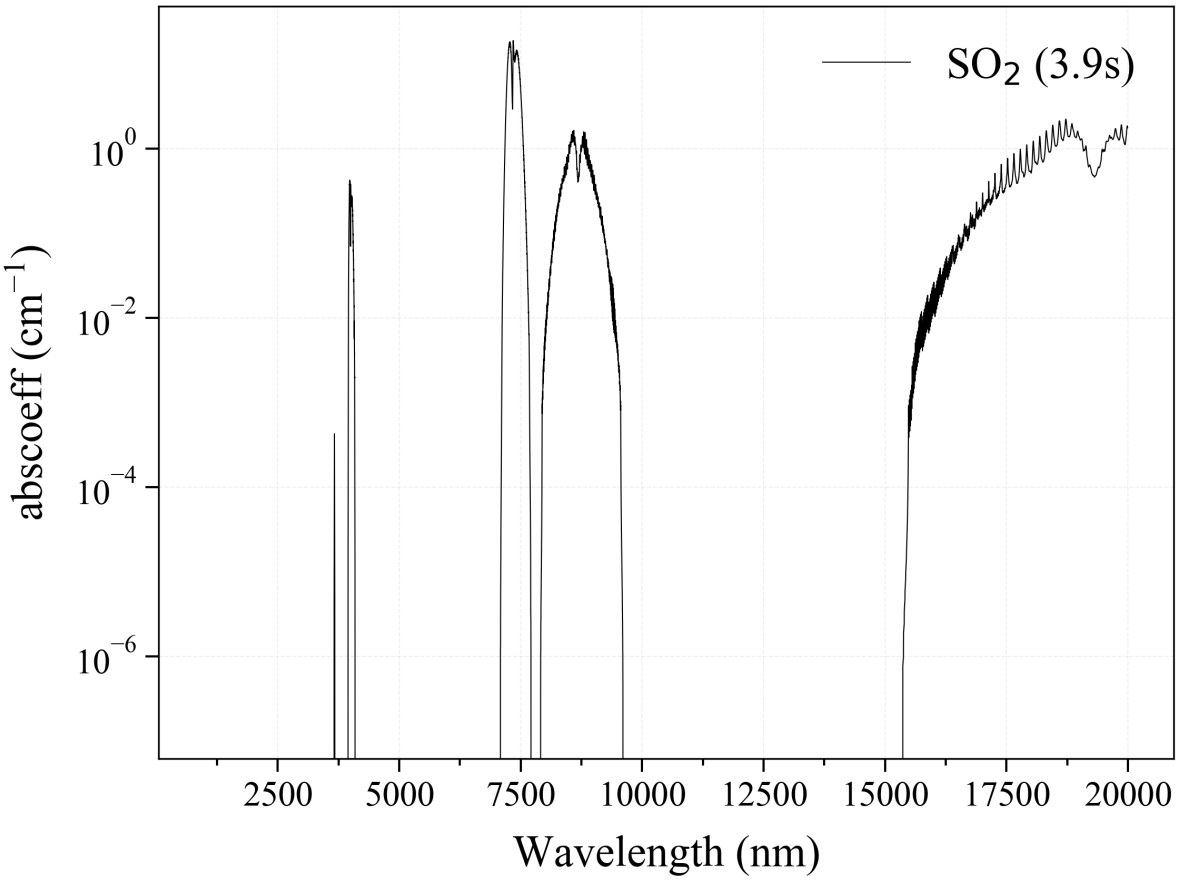
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='NO',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.7 9. SO2

- 9 `'SO2'` : Sulfur Dioxide absorption coefficient (opacity) at 300 K

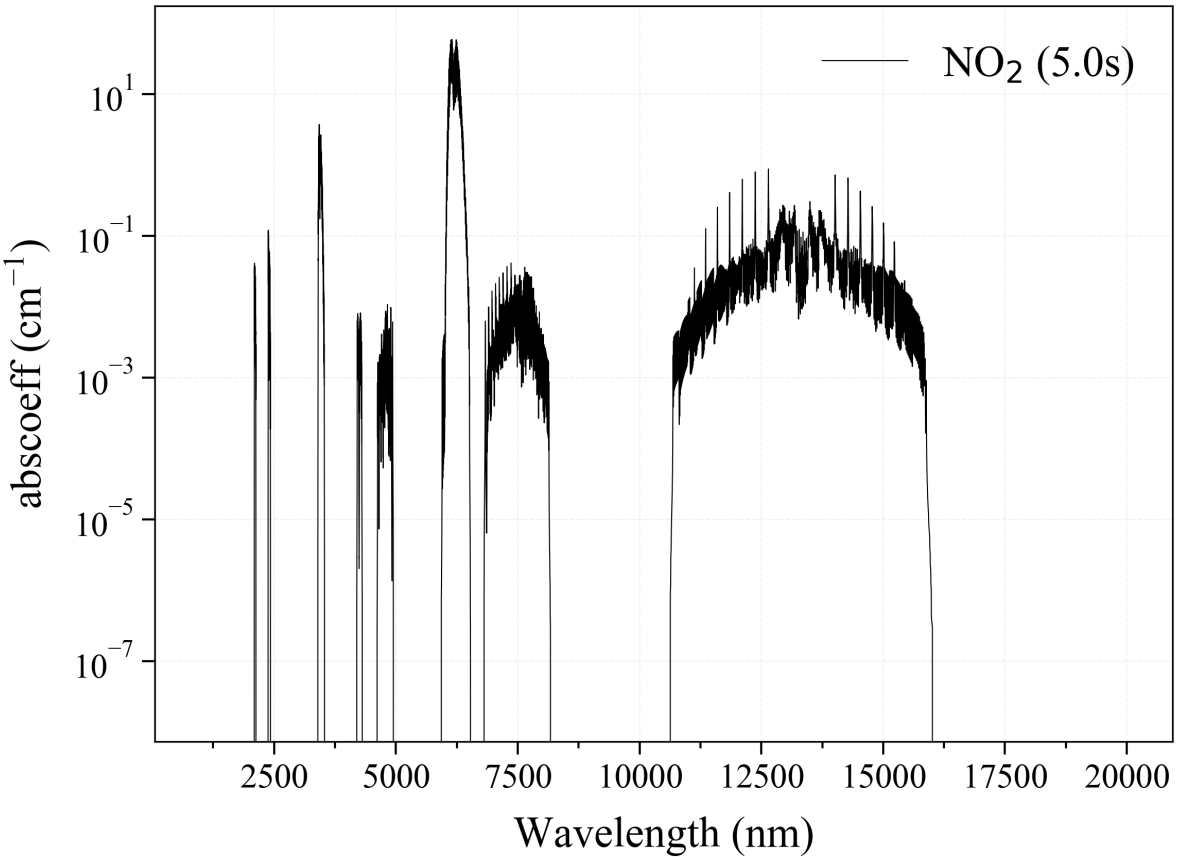
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='SO2',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.8 10. NO2

- 10 `'NO2'` : Nitrogen Dioxide absorption coefficient (opacity) at 300 K

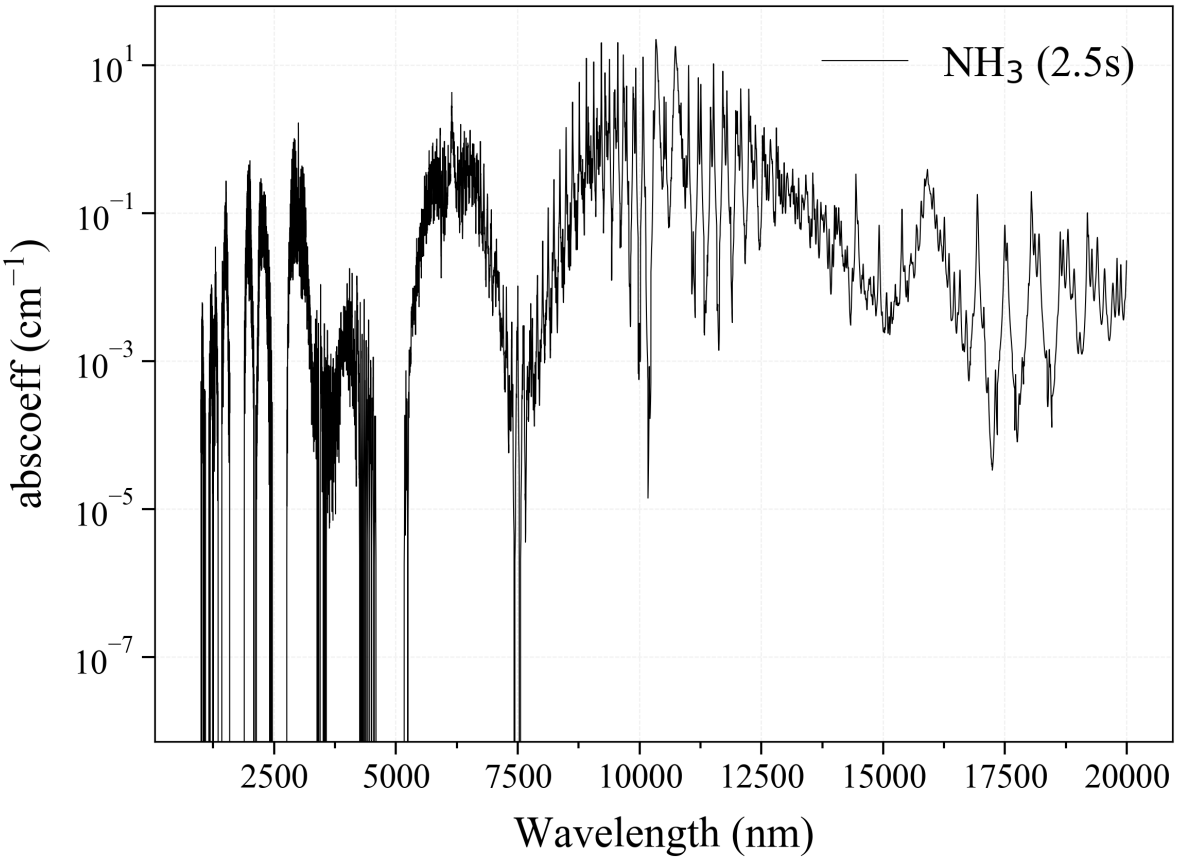
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='NO2',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.9 11. NH3

- 11 `'NH3'` : Ammonia absorption coefficient (opacity) at 300 K

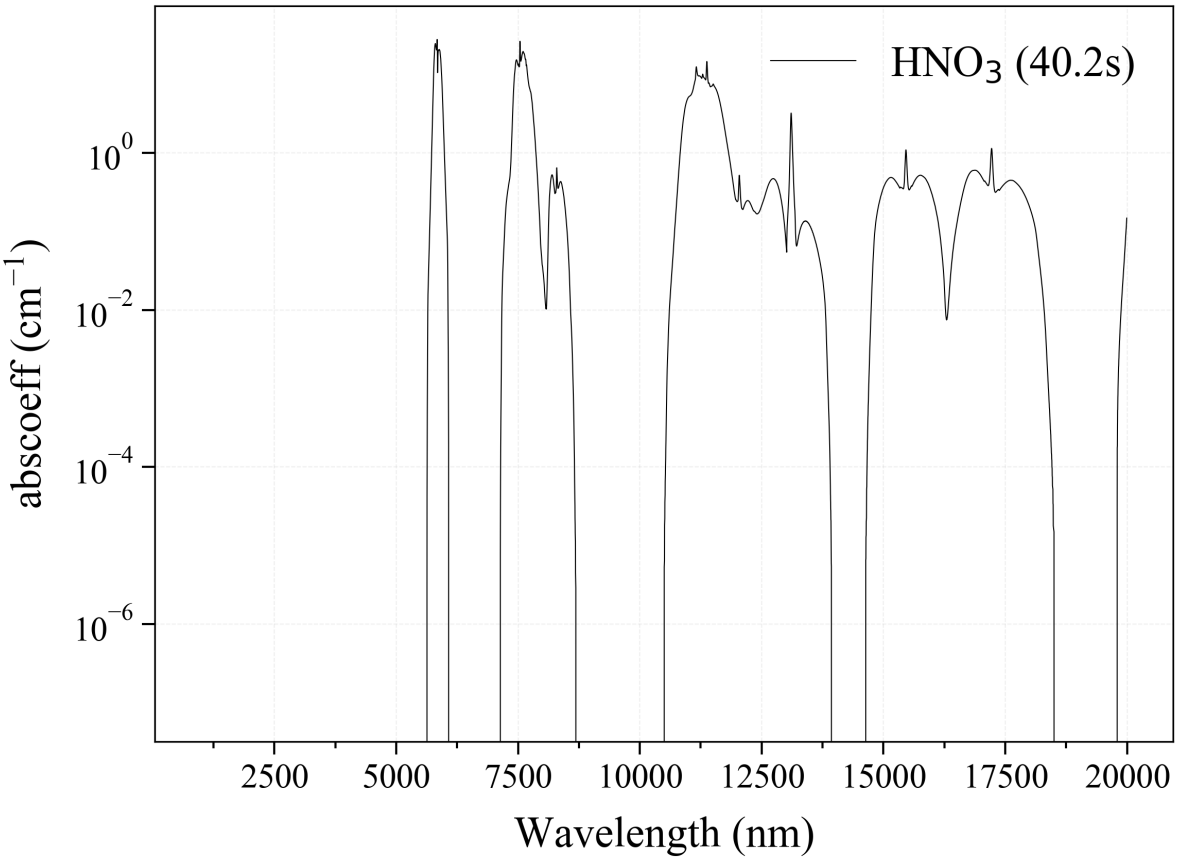
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='NH3',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.10 12. HNO3

- 12 `'HNO3'` : Nitric Acid absorption coefficient (opacity) at 300 K

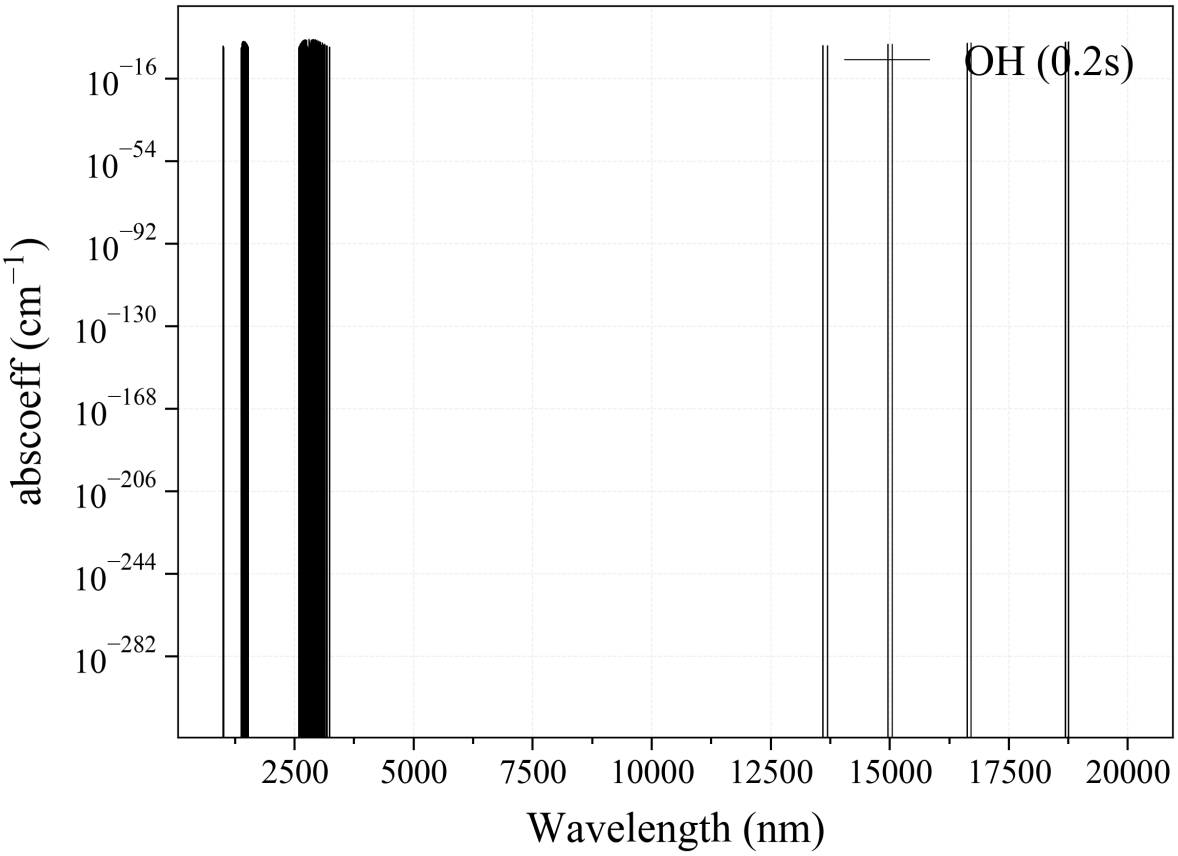
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='HNO3',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.11 13. OH

- 13 `'OH'` : Hydroxyl absorption coefficient (opacity) at 300 K

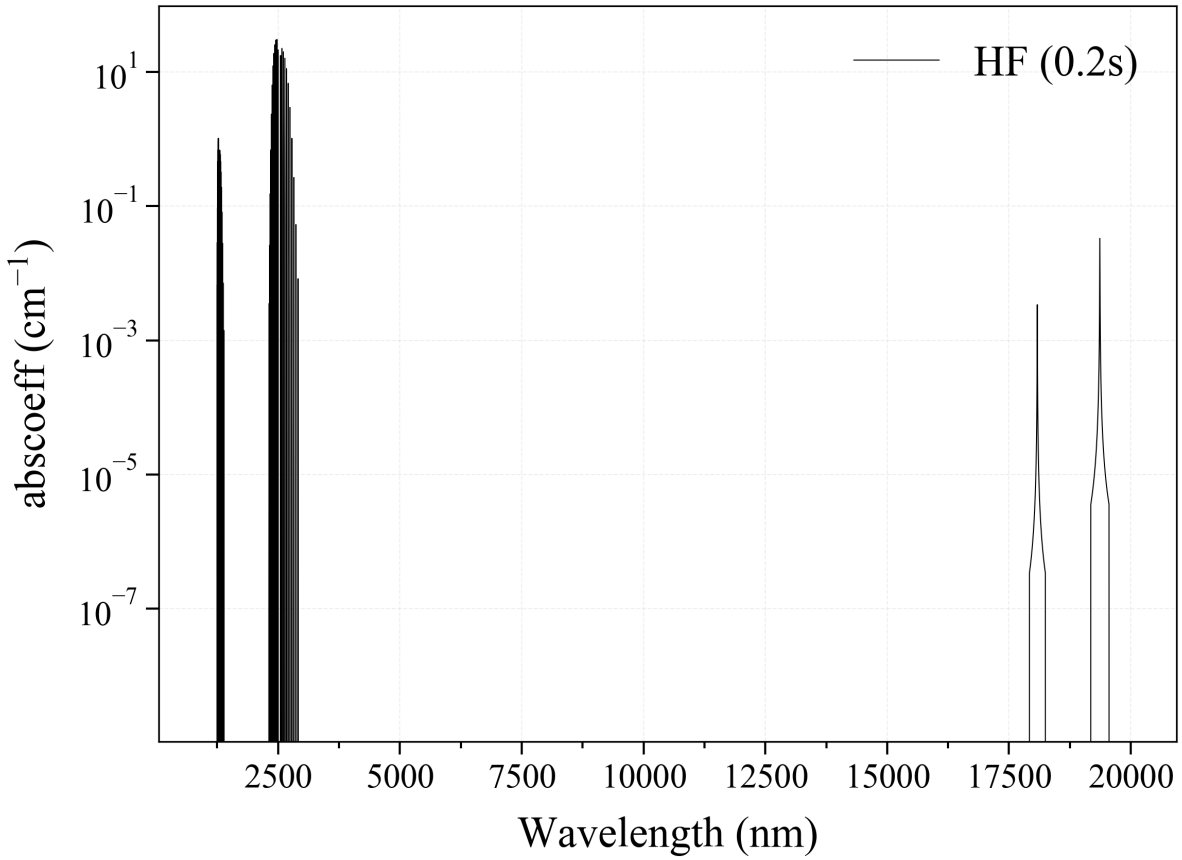
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='OH',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.12 14. HF

- 14 `'HF'` : Hydrogen Fluoride absorption coefficient (opacity) at 300 K

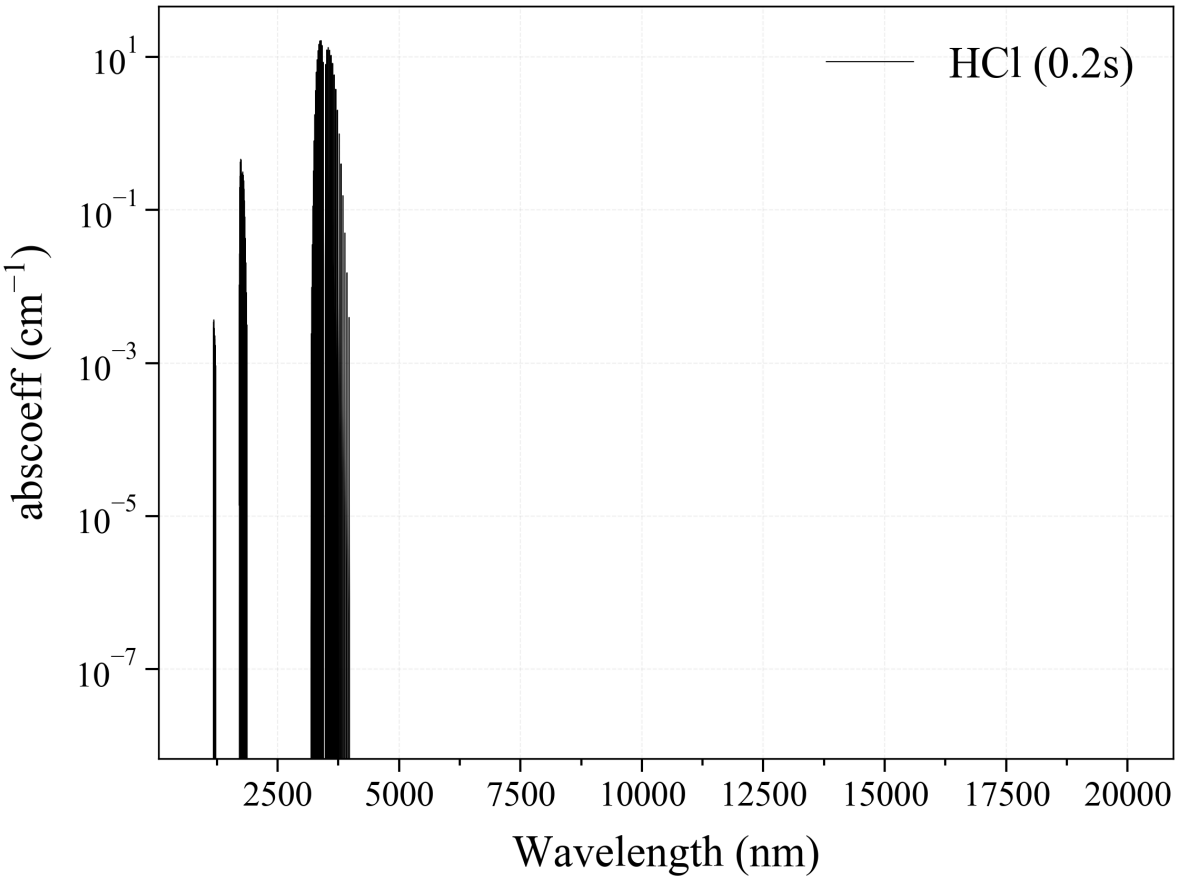
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='HF',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.13 15. HCl

- 15 `'HCl'` : Hydrogen Chloride absorption coefficient (opacity) at 300 K

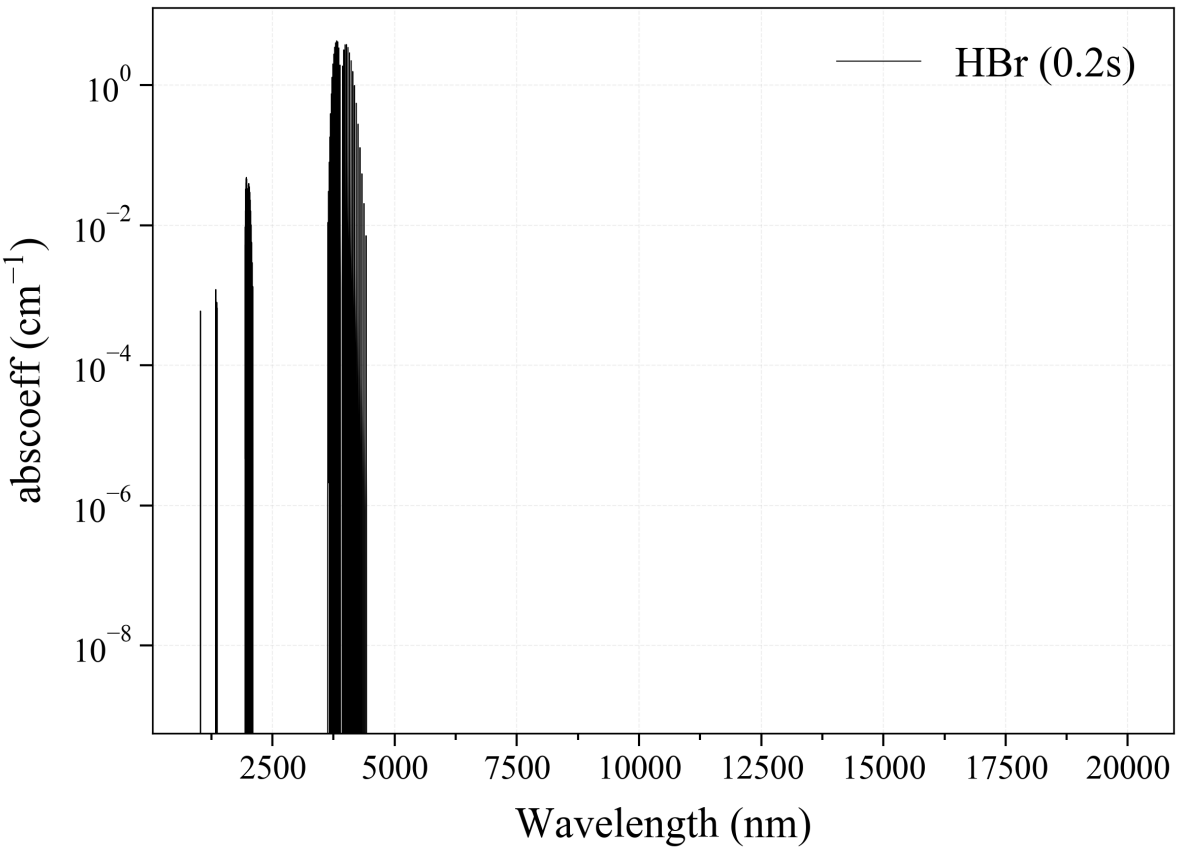
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='HCl',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.14 16. HBr

- 16 `'HBr'` : Hydrogen Bromide absorption coefficient (opacity) at 300 K

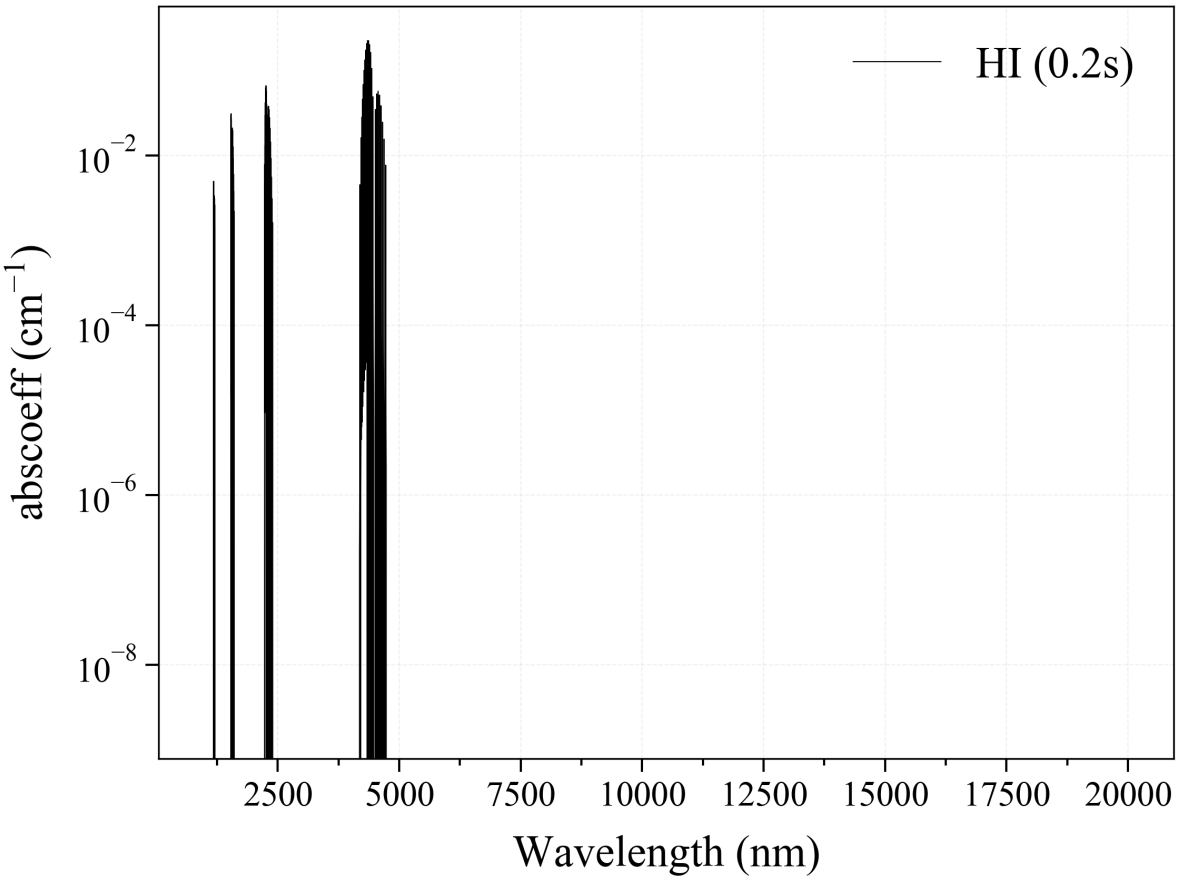
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='HBr',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.15 17. HI

- 17 `'HI'` : Hydrogen Iodide absorption coefficient (opacity) at 300 K

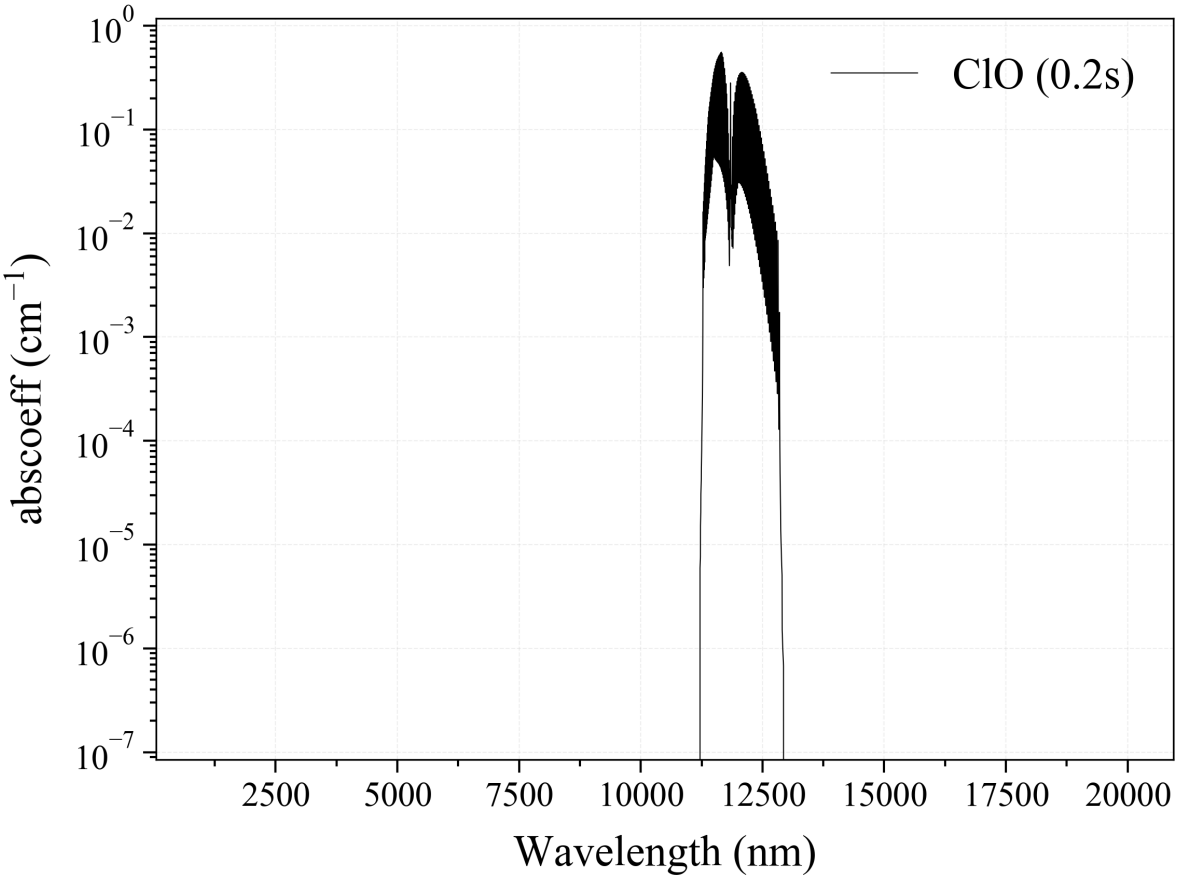
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='HI',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.16 18. ClO

- 18 `'ClO'` : Chlorine Monoxide absorption coefficient (opacity) at 300 K

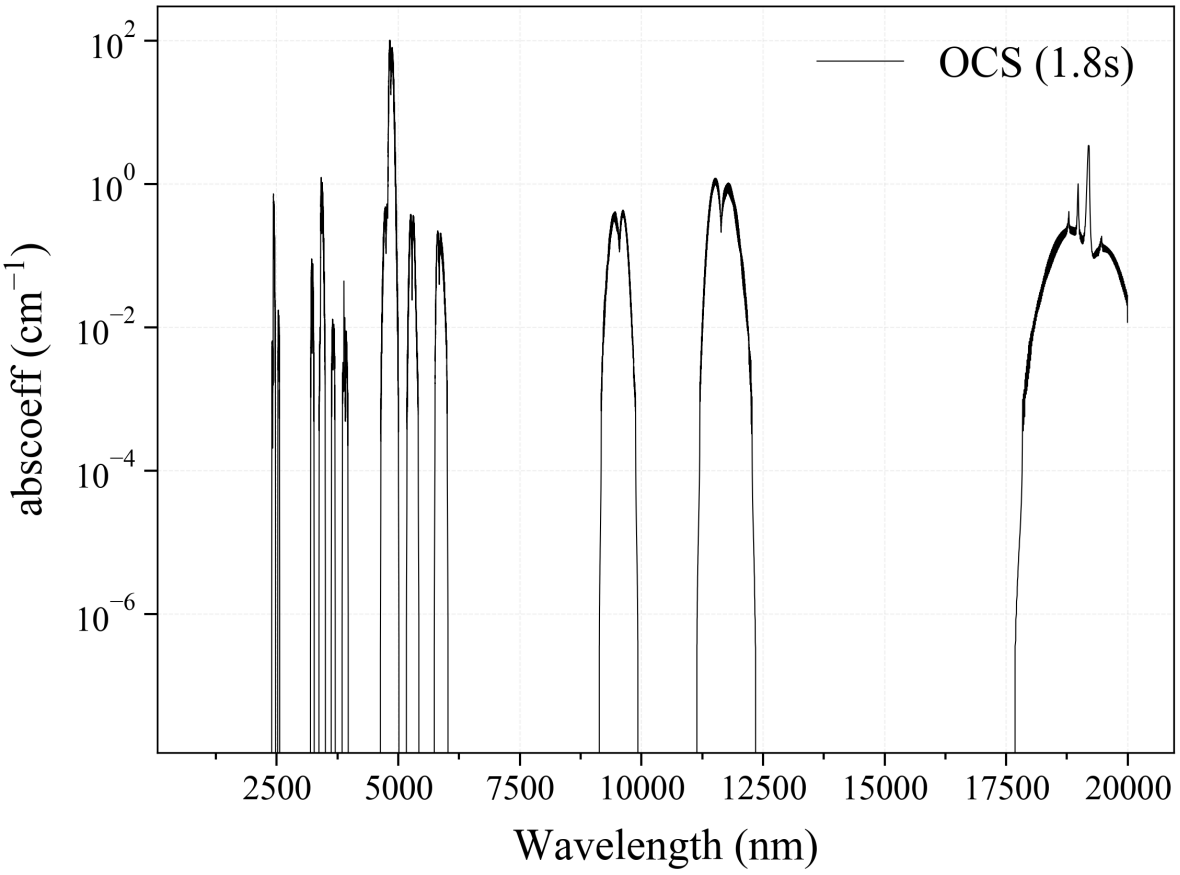
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='ClO',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.17 19. OCS

- 19 `'OCS'` : Carbonyl Sulfide absorption coefficient (opacity) at 300 K

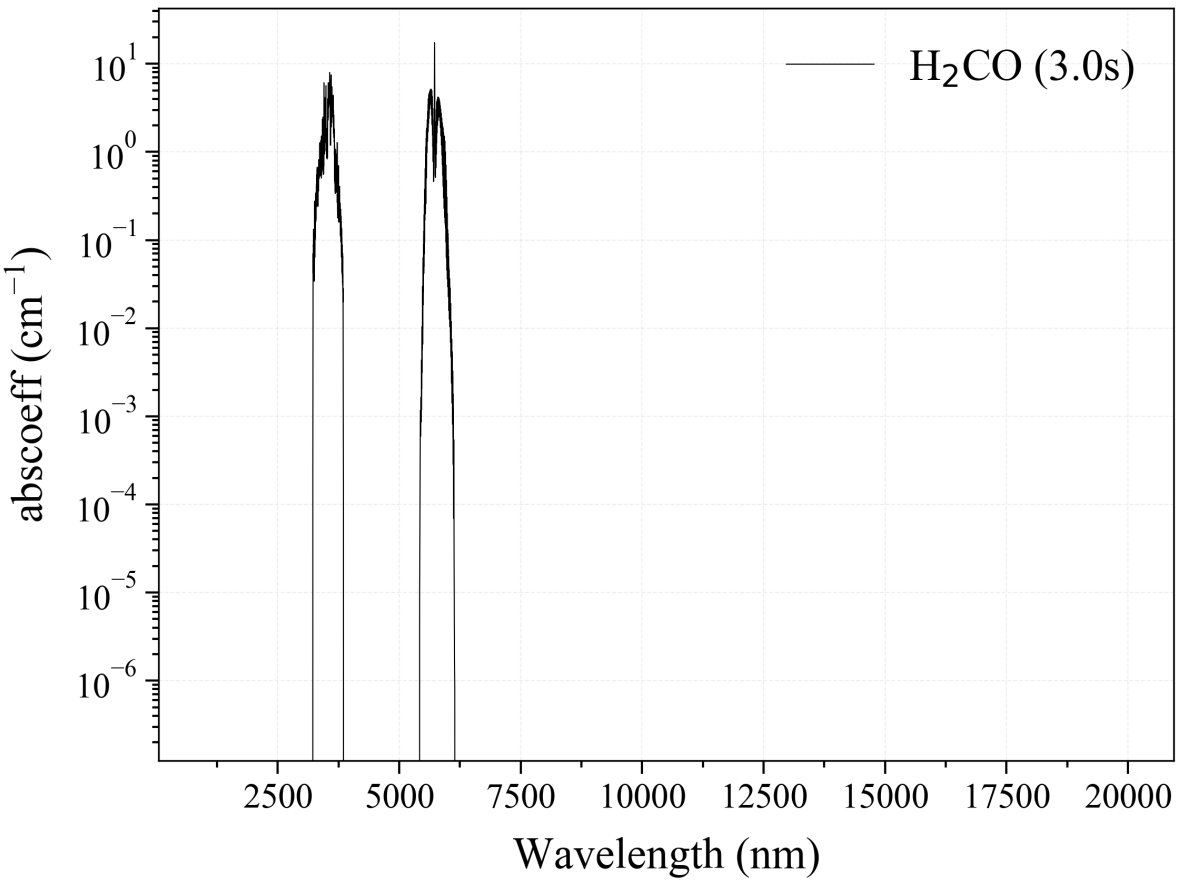
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='OCS',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.18 20. H2CO

- 20 `'H2CO'` : Formaldehyde absorption coefficient (opacity) at 300 K

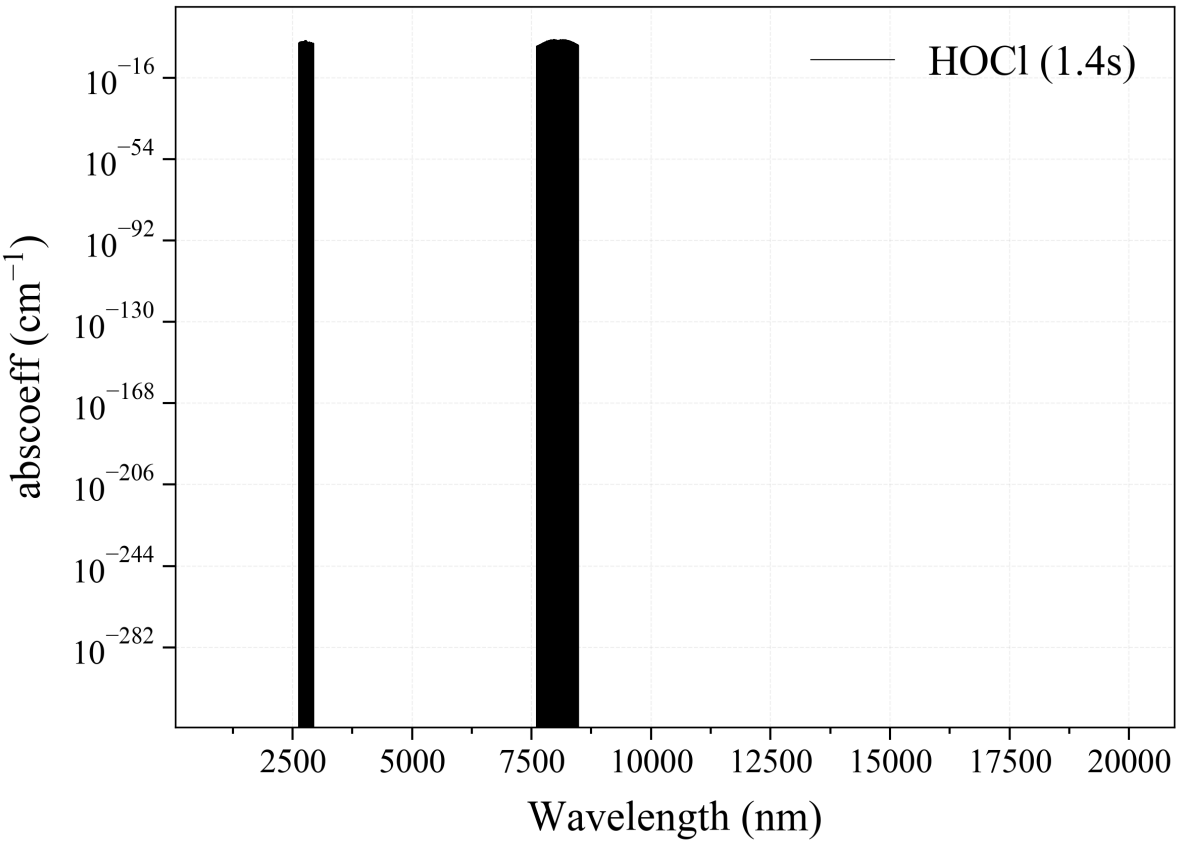
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='H2CO',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.19 21. HOCl

- 21 `'HOCl'` : Hypochlorous Acid absorption coefficient (opacity) at 300 K

```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='HOCl',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.20 22. N2

- 22 `'N2'` : Nitrogen absorption coefficient (opacity) at 300 K : no lines for `isotope='1'` (symmetric!)

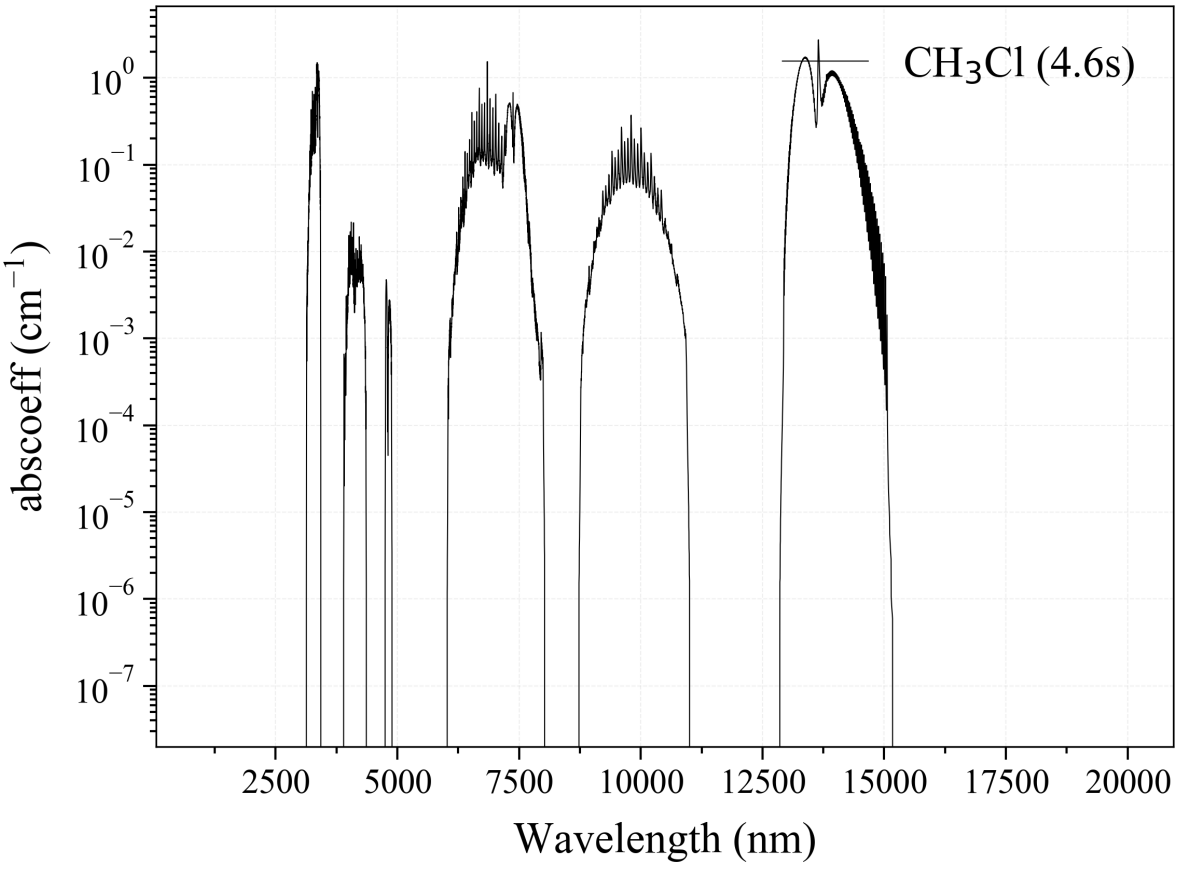
2.7.21 23. HCN

- **23 `'HCN'`**
  [Hydrogen Cyanide absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.22 24. CH4Cl

- 24 `'CH3Cl'` : Methyl Chloride absorption coefficient (opacity) at 300 K

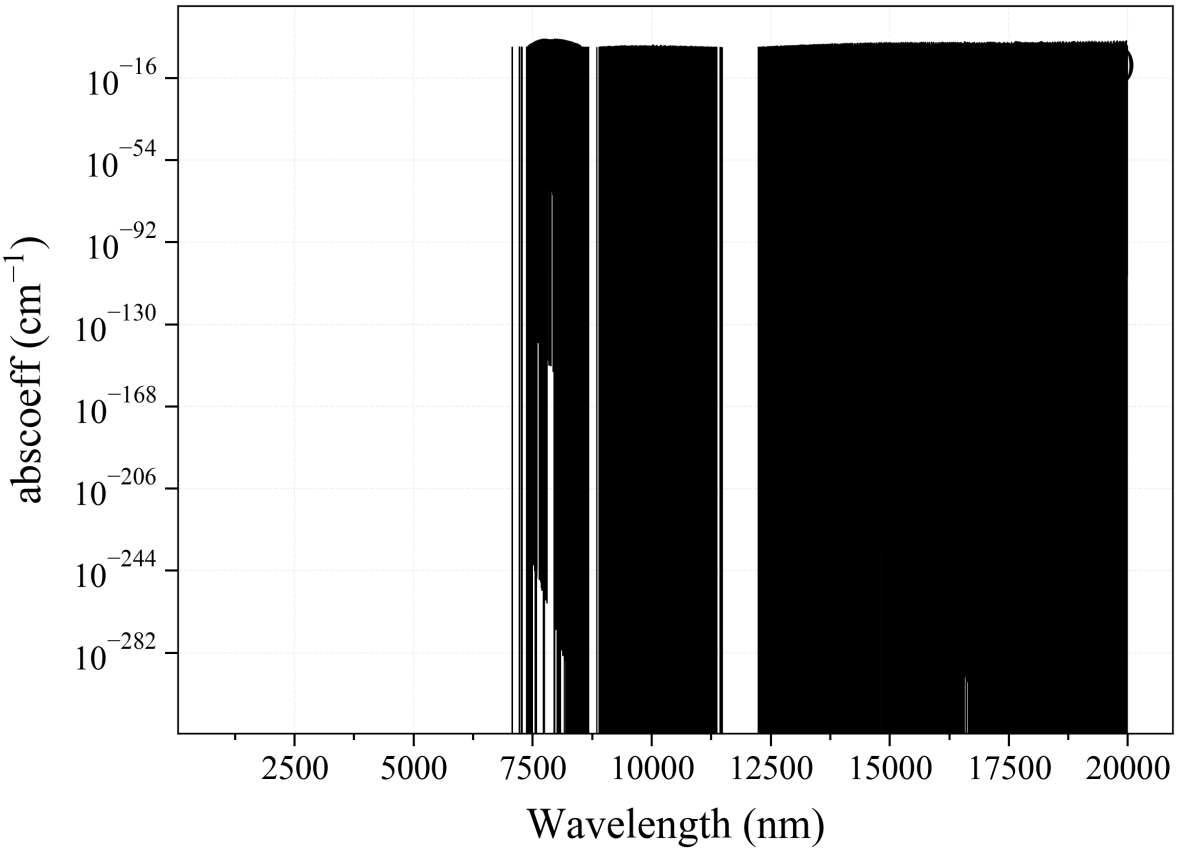
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='CH3Cl',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.23 25. H2O2

- 25 `'H2O2'` : Hydrogen Peroxide absorption coefficient (opacity) at 300 K

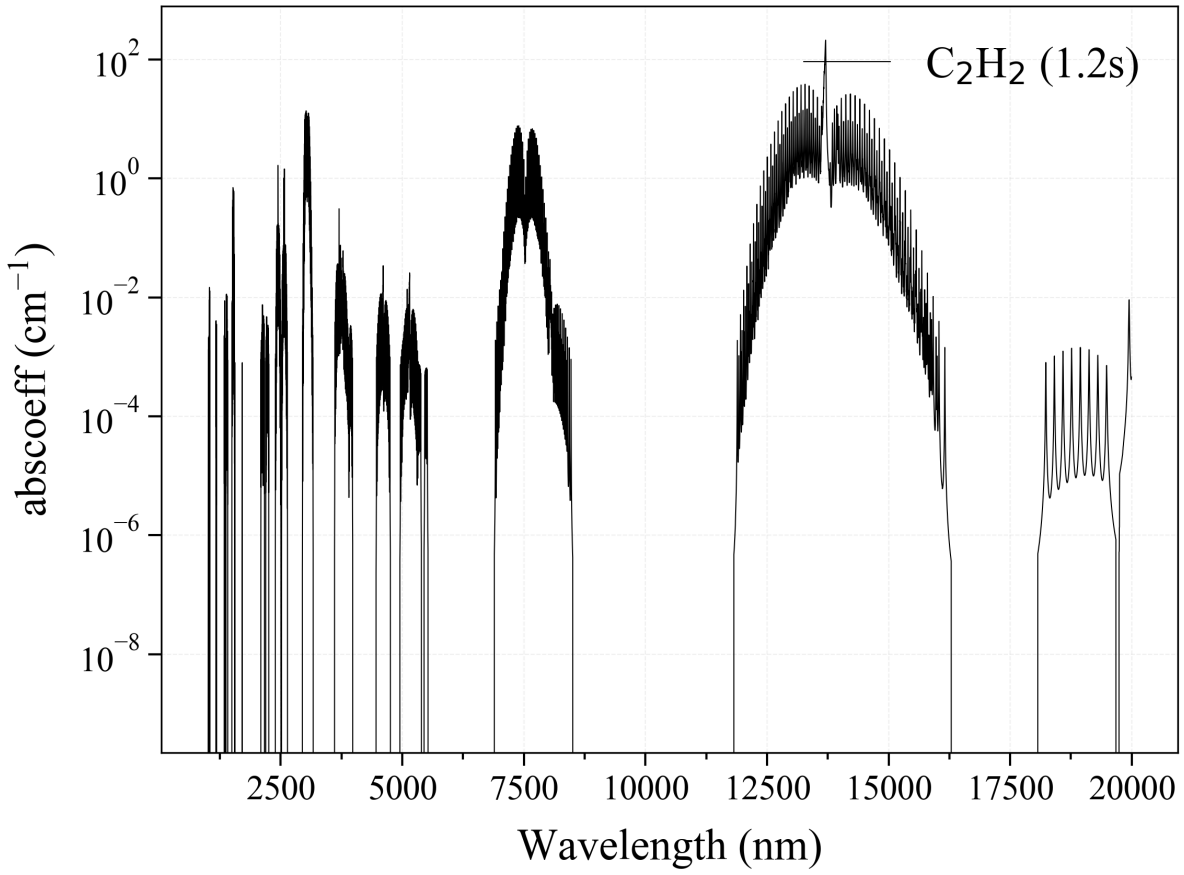
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='H2O2',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.24 26. C2H2

- 26 `'C2H2'` : Acetylene absorption coefficient (opacity) at 300 K

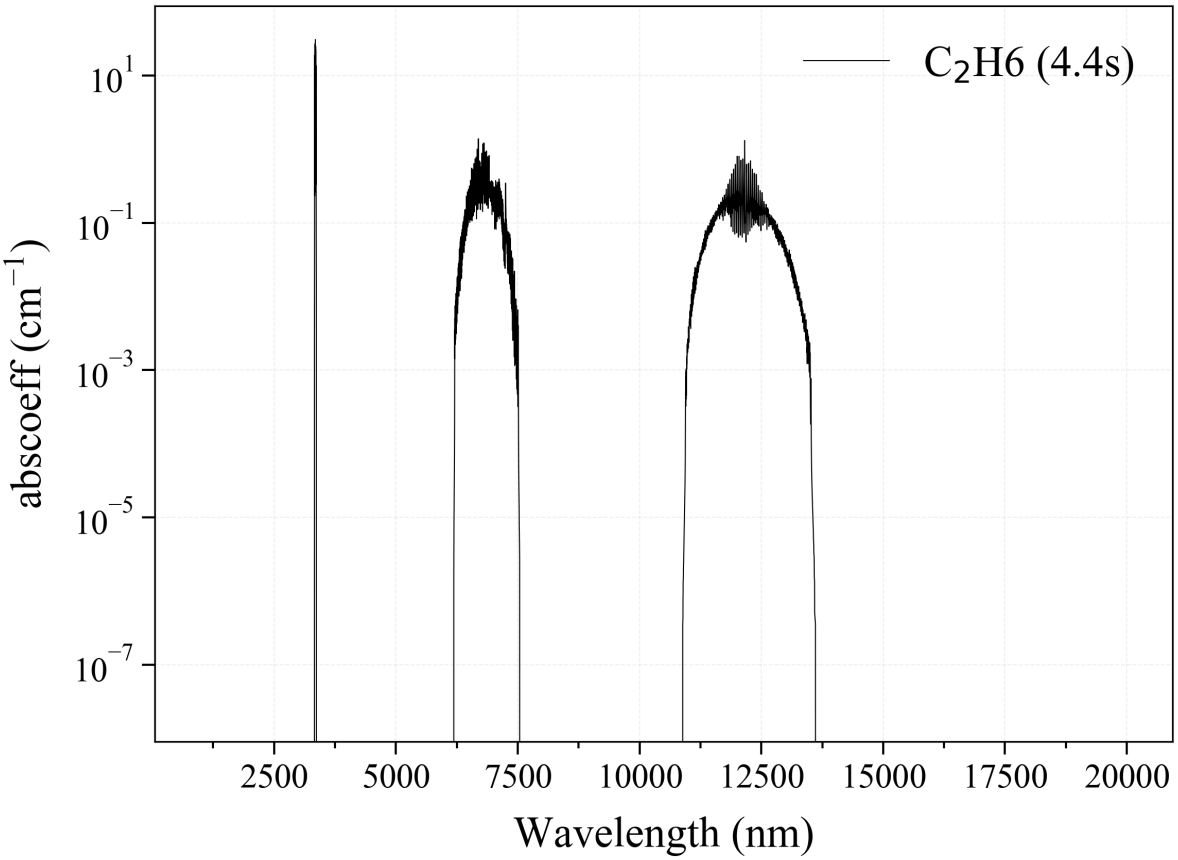
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='C2H2',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.25 27. C2H6

- 27 `'C2H6'` : Ethane absorption coefficient (opacity) at 300 K

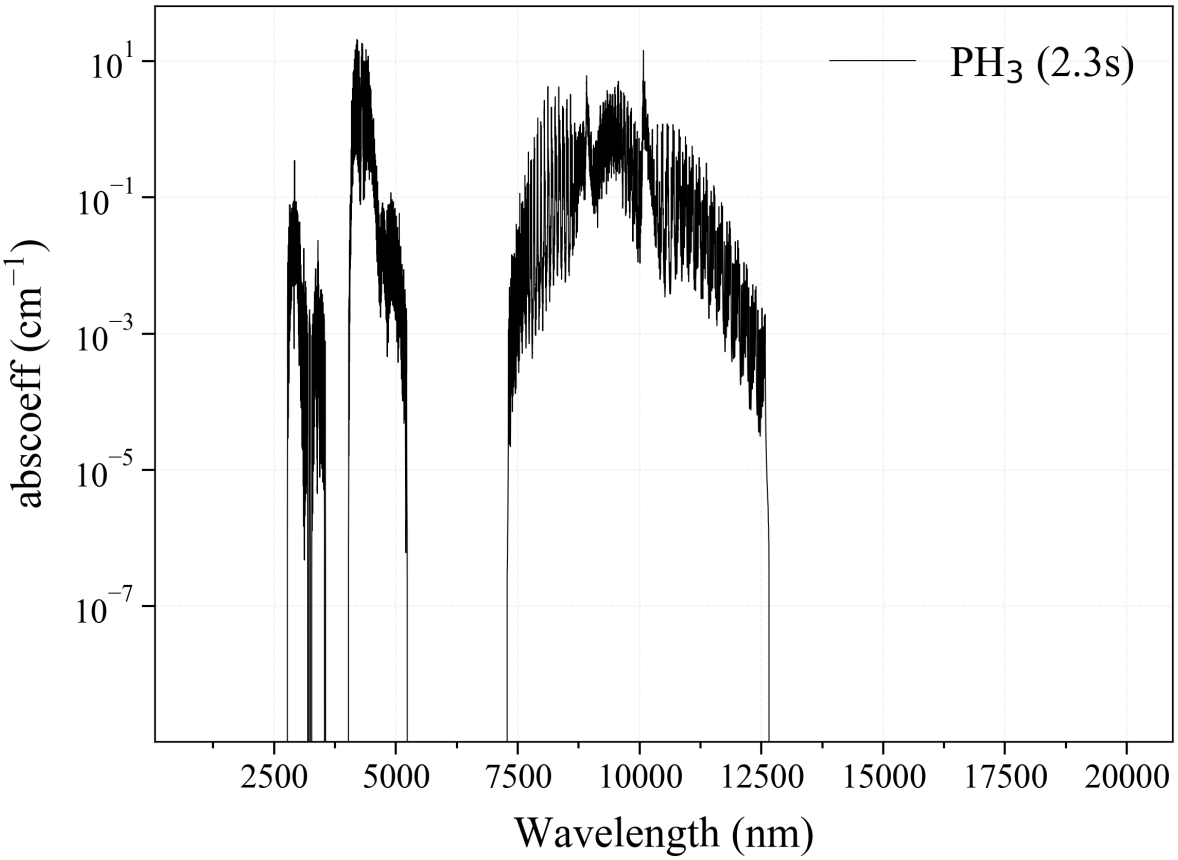
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='C2H6',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.26 28. PH3

- 28 `'PH3'` : Phosphine absorption coefficient (opacity) at 300 K

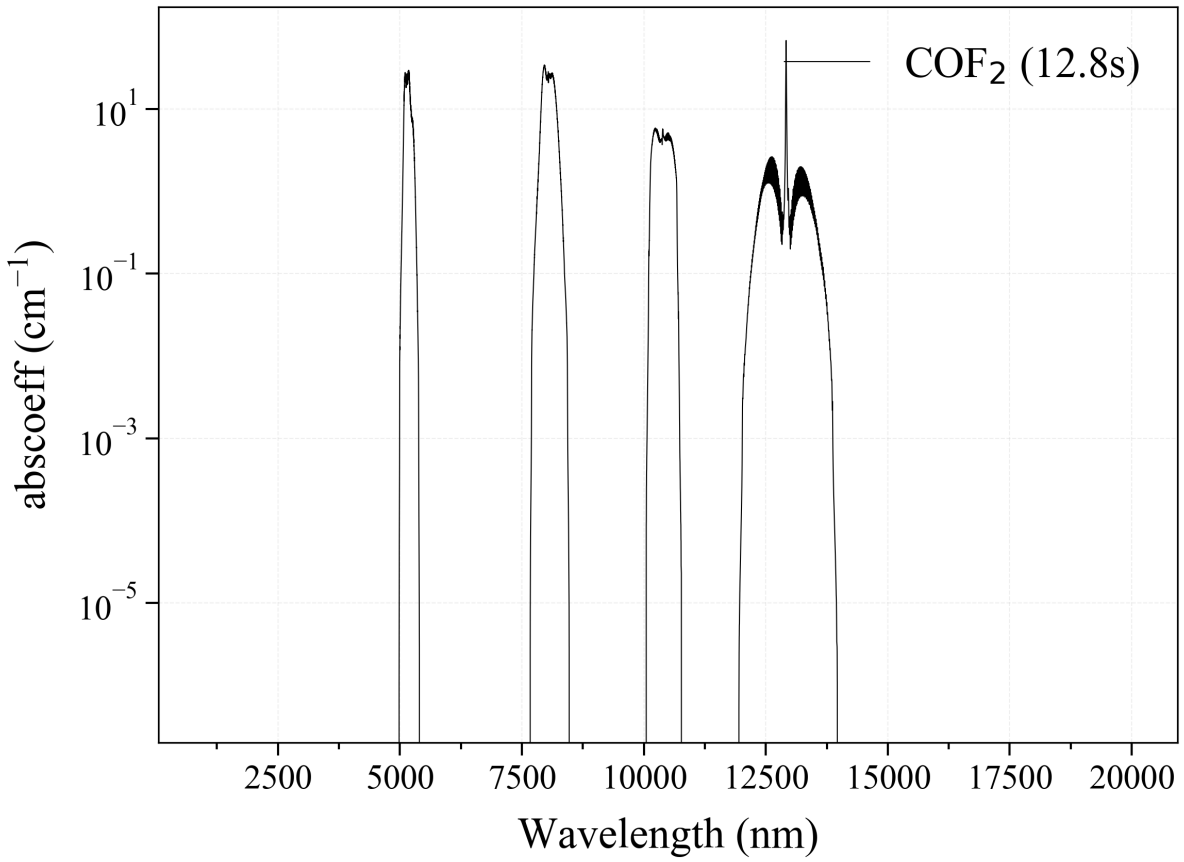
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='PH3',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.27 29. COF2

- 29 `'COF2'` : Carbonyl Fluoride absorption coefficient (opacity) at 300 K

```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='COF2',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



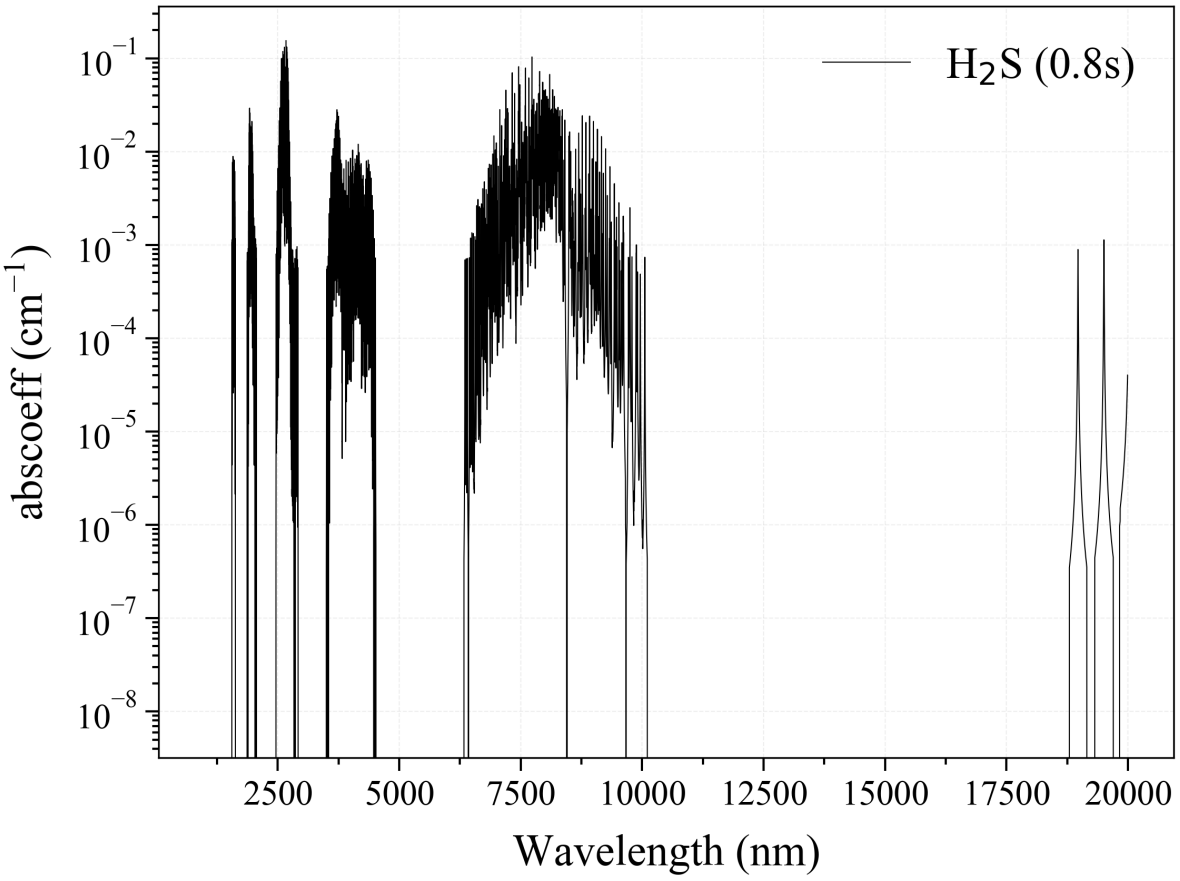
2.7.28 30. SF6

- **30 'SF6'**
  [Sulfur Hexafluoride absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.29 31. H2S

- 31 'H2S' : Hydrogen Sulfide absorption coefficient (opacity) at 300 K

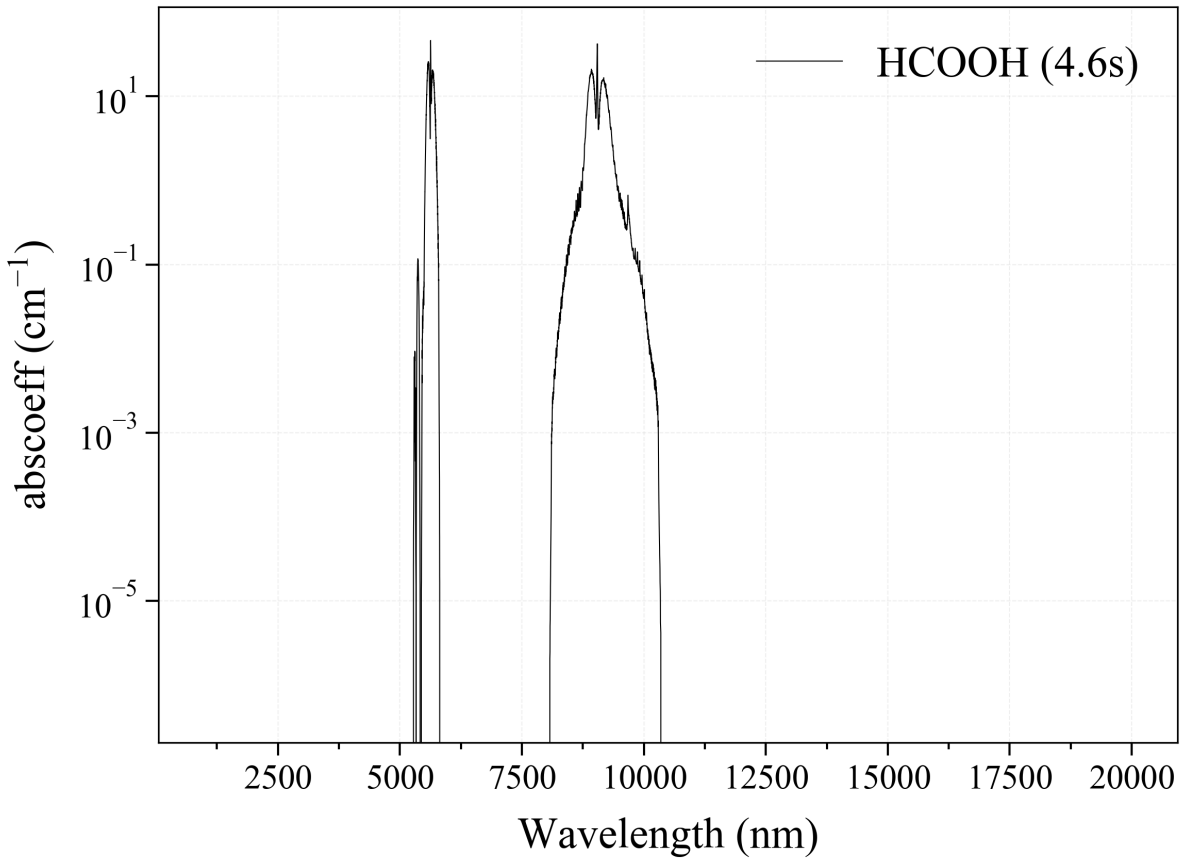
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='H2S',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.30 32. HCOOH

- 32 `'HCOOH'` : Formic Acid absorption coefficient (opacity) at 300 K

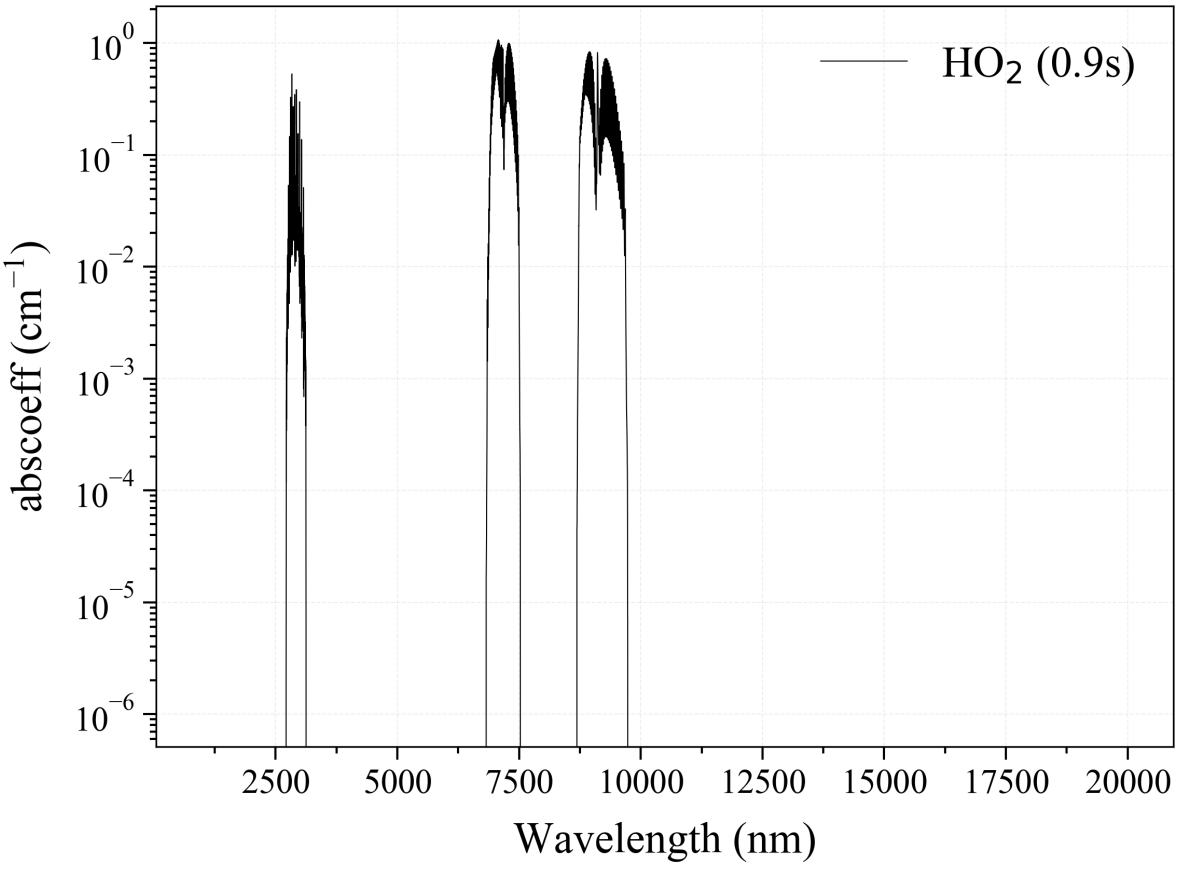
```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='HCOOH',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.31 33. HO2

- 33 `'HO2'` : Hydroperoxyl absorption coefficient (opacity) at 300 K

```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='HO2',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



34. O =====-

- **34 'O'**
  [Oxygen Atom absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

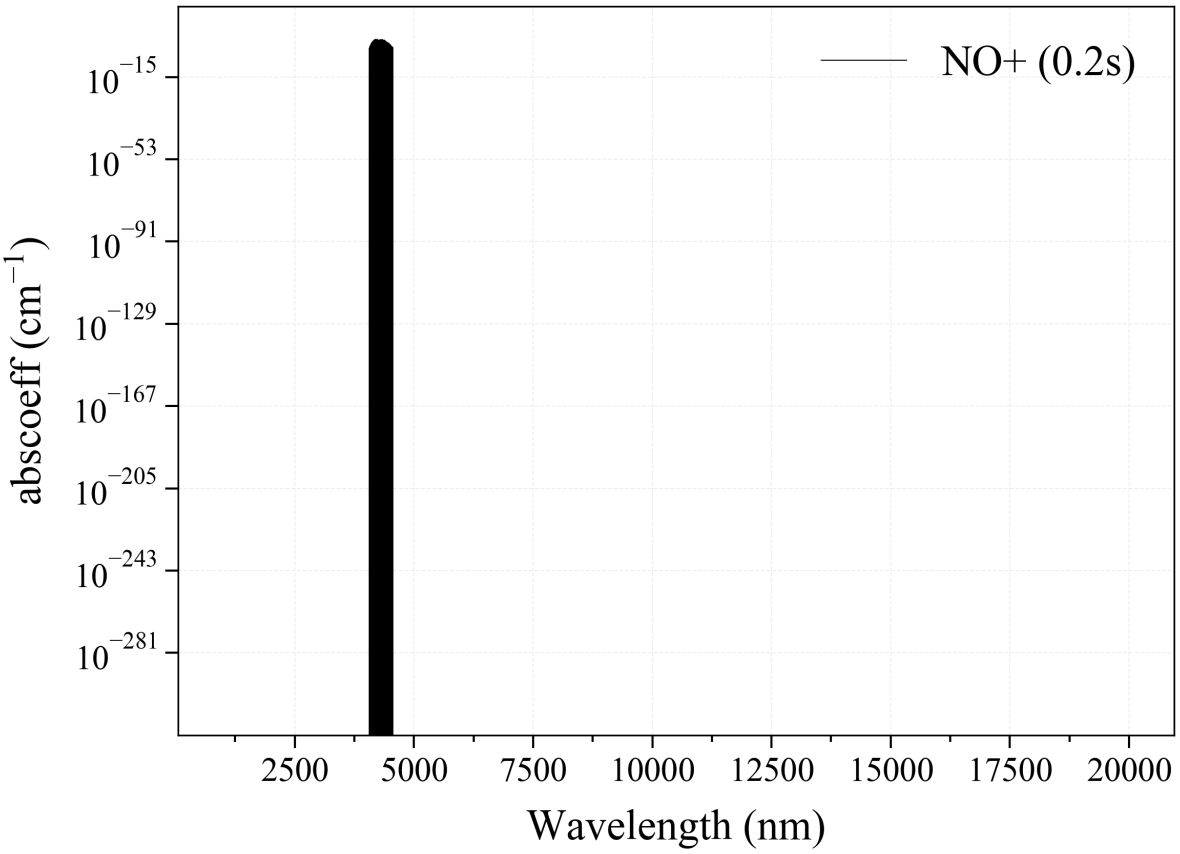
2.7.32 35. ClONO2

- **35 'ClONO2'**
  [Chlorine Nitrate absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.33 36. NO+

- 36 'NO+' : Nitric Oxide Cation absorption coefficient (opacity) at 300 K

```
s = calc_spectrum(wavelength_min=1000,
                  wavelength_max=20000,
                  Tgas=300,
                  pressure=1,
                  molecule='NO+',
                  optimization=None,
                  cutoff=1e-23,
                  isotope='1')
s.plot('abscoeff', wunit='nm')
```



2.7.34 37. HOBr

- **37 'HOBr'**
  [Hypobromous Acid absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.35 38. C2H4

- **38 'C2H4'**
  [Ethylene absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.36 39. CH3OH

- **39 'CH3OH'**
  [Methanol absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.37 40. CH3Br

- **40 'CH3Br'**
  [Methyl Bromide absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.38 41. CH3CN

- **41 'CH3CN'**
  [Acetonitrile absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.39 42. CF4

- **42 'CF4'**
  [CFC-14 absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.40 43. C4H2

- **43 'C4H2'**
  [Diacetylene absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.41 44. HC3N

- **44 'HC3N'**
  [Cyanoacetylene absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.42 45. H2

- **45 'H2'**
  [Hydrogen absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.43 46. CS

- **46 'CS'**
  [Carbon Monosulfide absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.44 47. SO3

- **47 'SO3'**
  [Sulfur trioxide absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.7.45 48. C2N2

- **48 'C2N2'**
  [Cyanogen absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

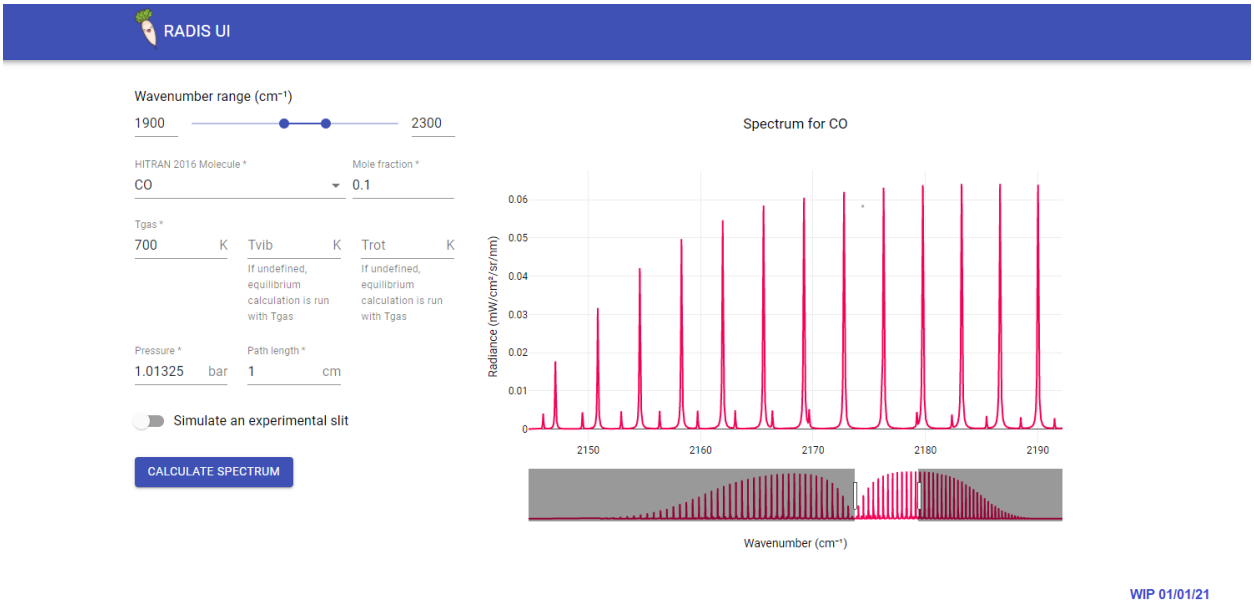
2.7.46 49. COCl2

- **49 'COCl2'**
  [Phosgene absorption coefficient (opacity) at 300 K][not calculated.] Contribute on GitHub.

2.8 Try Online

2.8.1 radis-app

A simple web-app for RADIS under development.

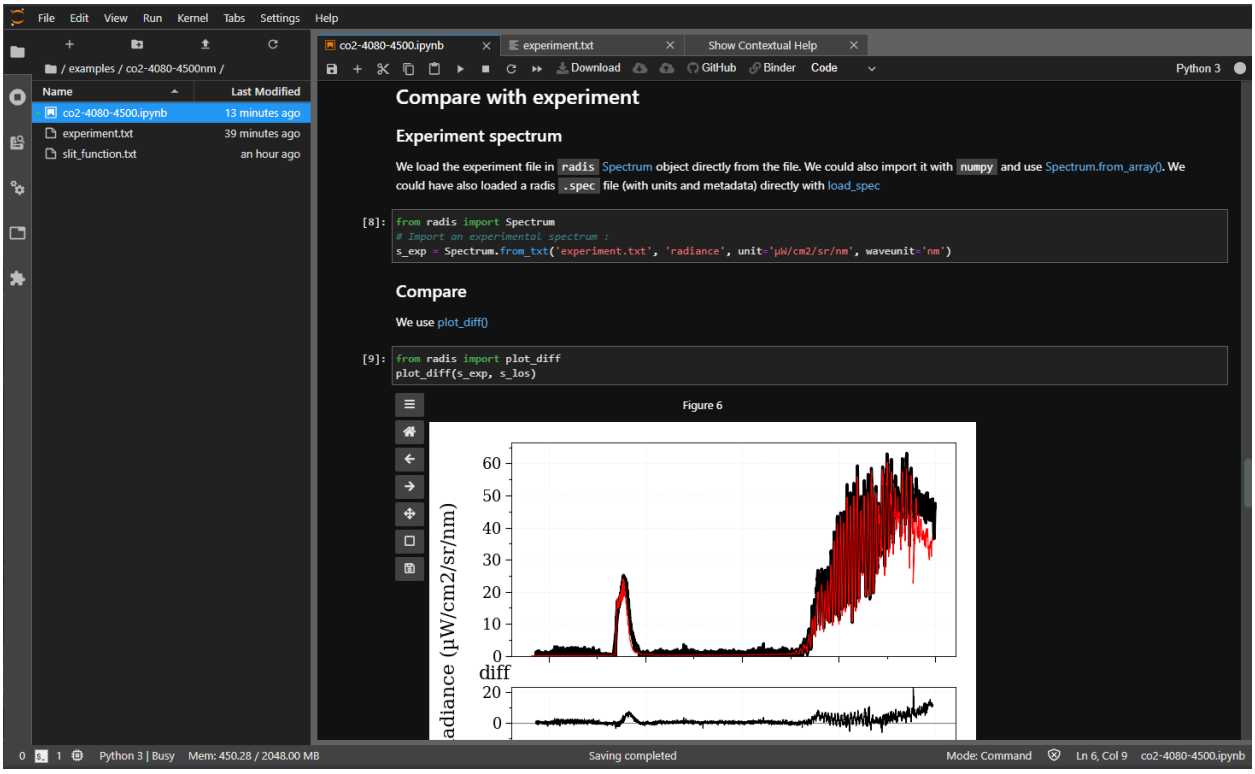


See more on GitHub

2.8.2 RADIS-lab

An online environment for advanced spectrum processing and comparison with experimental data :

- no need to install anything
- use pre-configured line databases ([HITEMP-2010])
- upload your data files, download your results !



Try it : https://radis.github.io/radis-lab/

Note: Radis-Lab allows you to share your session URL with colleagues. They will access the latest saved version of your notebook as long as your session is running.

See more on GitHub

2.9 Developer Guide

2.9.1 Contribute

RADIS is an open-source project, and therefore anyone can contribute, whether you already know spectroscopy or not.

You want to become a contributor ? Welcome ! First step is to read the Contributing Guide .

If you want to contribute and have no specific project in mind, have a look at the GitHub opened issues. For a First Time contribution, you can tackle one of the Issues displayed as Good First Issue.

Also, join the community chat, where you'll get a lot of help :

Finally, you can also suggest or vote for new features below:

2.9.2 Sources

To modify RADIS, you will need to clone the Git repository. See the *detailed installation procedure* below.

For a First Time contribution, you can tackle one of the Issues displayed as Good First Issue. Learn more in the *Contribute* section.

Install

You can either install RADIS from `pip`, the Python package manager. But if you want to access the latest features, or modify the code and contribute, we suggest that you Fork the source code from GitHub.

Use-only version : cant modify the code

In a terminal, run:

```
pip install --user radis -v
```

The 'pip' module has to be installed (by default if you've installed Python with Anaconda).

If you later want to update the package:

```
pip install radis --upgrade
```

Developer version: to modify the code and contribute to the project.

We suggest that you fork the RADIS repository and use that url for the clone step below. This will make submitting changes easier in the long term for you:

In a terminal, run:

```
git clone https://github.com/<GITHUB USERNAME>/radis.git
cd radis
pip install -e .[dev] -v
```

- The `-e` (editable) argument creates a link from the local folder `./` into Python site-packages.
- `[dev]` is optional, but installs additional developer packages that may be useful for testing and linting your code.

To make sure the install worked, run the *first example* from the Quick Start page. Then, you're all set.

For a best experience, it is recommended to install Radis on an Anaconda Python distribution, in an isolated environment. You can create a radis environment with all dependencies with:

```
cd radis
conda env create --file environment.yml
```

Then install radis in the `radis` environment:

```
conda activate radis
pip install -e .
```

Test your changes

If you want to modify the source code, you need to ensure that you don't break any of the existing tests. Refer to the *Test Section* to learn how to run the tests locally.

Update your changes

Submit a Pull Request from GitHub.

Online tests will be run automatically. They will check for:

- Physical test cases
- Code format (see Code linting below)

Code linting

Radis follows Black style for code linting to maintain consistent coding style across modules. Code style is checked using CI services which run automatically on each pull request. **Black** is automatically installed when radis is set-up in developer mode.

To format any file/files:

```
black /path/to/file/or/directory/
```

You can include Black coding style directly in some text editors

Alternatively, Black coding style can be checked automatically before each commit. For that all you need to do is to run the following command once:

```
cd radis
pre-commit install
```

On each commit, format will be fixed if it was incorrect. All you need to do is to commit a second time. Exemple:

```
$ git commit -am "test"
black....................................................................Failed
- hook id: black
- files were modified by this hook

reformatted [ALL FAILING FILES]
All done!
1 file reformatted.

$ git commit -am "test"
black....................................................................Passed
[develop XXX] test
 1 file changed, 1 insertion(+)
```

Note that pre-commit will always require you to commit again after a test was failed, because it's safer. If for any reason you want to skip formatting you can commit with the `--no-verify` argument.

Update

With Pip you can keep the package up-to-date with:

```
pip install radis --upgrade
```

If using the latest developer version (cloned from GitHub and installed with `pip install -e .[dev]`), use git to `pull` the latest changes.

Help

If you encounter any problem, please report an Issue on GitHub.

You can also ask for advice on the Q&A forum or the community chat:

2.9.3 Architecture

The RADIS modules are organized with the following flow chart

---

The upper part shows the successive calculation steps of the Line-by-Line module. These steps appear clearly in the source code of the `eq_spectrum()` and `non_eq_spectrum()` methods of the `SpectrumFactory`. See details at the end of this file.

Methods are written in Factory objects inherited with the following scheme:

`DatabankLoader` > `BaseFactory` > `BroadenFactory` > `BandFactory` > `SpectrumFactory`

The Input Conditions in the left part, and the Computation Parameters on the right part, are the input parameters of the different RADIS front-ends:

- `calc_spectrum()` for the simple cases.
- `SpectrumFactory` with `eq_spectrum()` and `non_eq_spectrum()` for the other cases. GPU calculation can be done with `eq_spectrum_gpu()`

The Input databases are either automatically downloaded from [HITRAN-2020], or defined locally in a *Configuration file*

The bottom part includes the post-processing modules of RADIS, in particular:

- The various methods associated with the `Spectrum` class.
- The *Line-of-Sight module* module
- The `LineSurvey` tool.
- The `SpecDatabase` tool.

Methods from the Flow Chart: this methods are called successively from the `radis.lbl.factory.SpectrumFactory.eq_spectrum()` and `radis.lbl.factory.SpectrumFactory.non_eq_spectrum()` methods.

- Line Database: methods of `DatabankLoader` :
  - `radis.lbl.loader.DatabankLoader.load_databank()`
  - `radis.lbl.loader.DatabankLoader.init_databank()`
  - `radis.lbl.loader.DatabankLoader.fetch_databank()`
- Partition functions: methods of `RovibParFuncTabulator` and `RovibParFuncCalculator` :
  - `radis.levels.partfunc.RovibParFuncTabulator.at()`
  - `radis.levels.partfunc.RovibParFuncCalculator.at()`
  - `radis.levels.partfunc.RovibParFuncCalculator.at_noneq()`
  - `radis.levels.partfunc.RovibParFuncCalculator.at_noneq_3Tvib()`
- Populations: methods of `BaseFactory` :
  - `radis.lbl.base.BaseFactory.calc_populations_eq()`
  - `radis.lbl.base.BaseFactory.calc_populations_noneq()`
- Line Intensities: methods of `BaseFactory` :
  - `radis.lbl.base.BaseFactory.calc_linestrength_eq()`
  - `radis.lbl.base.BaseFactory._calc_linestrength_noneq()`
  - `radis.lbl.base.BaseFactory._calc_emission_integral()`
- Line Positions: methods of `BaseFactory` :
  - `radis.lbl.base.BaseFactory.calc_lineshift()`
- Reduced line set: methods of `BaseFactory` :
  - `radis.lbl.base.BaseFactory._cutoff_linestrength()`
- Voigt Broadening: methods of `BroadenFactory` :
  - `radis.lbl.broadening.voigt_broadening_FWHM()`
  - `radis.lbl.broadening.voigt_lineshape()`
  - `radis.lbl.broadening._whiting()`
  - `radis.lbl.broadening._whiting_jit()`
  - `radis.lbl.broadening.BroadenFactory._calc_broadening_FWHM()`
  - `radis.lbl.broadening.BroadenFactory._add_voigt_broadening_FWHM()`
- Pseudo-continuum: methods of `BroadenFactory` :
  - `radis.lbl.broadening.BroadenFactory._find_weak_lines()`
  - `radis.lbl.broadening.BroadenFactory._calculate_pseudo_continuum()`
  - `radis.lbl.broadening.BroadenFactory._add_pseudo_continuum()`
- Spectral densities k, j: methods of `SpectrumFactory` :

  - `radis.lbl.factory.SpectrumFactory.eq_spectrum()`
  - `radis.lbl.factory.SpectrumFactory.non_eq_spectrum()`

- RTE (1 slab): methods of `SpectrumFactory` :
  - `radis.lbl.factory.SpectrumFactory.eq_spectrum()`
  - `radis.lbl.factory.SpectrumFactory.non_eq_spectrum()`

2.9.4 Test

Test status

Test routines are performed automatically by Travis CI whenever a change is made on the GitHub repository. See the current test status below:

It is a good practice to perform these tests locally to detect potential errors before pushing. To run the tests locally, assuming you cloned the source code (see the *Install section*), run the following command in the `radis` directory:

```
cd radis
pytest
```

The whole test procedure takes 5 - 10 min. You can use pytest filtering keys to *run specific tests only*

Code coverage

Code coverage makes sure every line in RADIS is properly tested. See the current code coverage status (click the badge for more details):

If you want to see the test coverage report locally use `codecov` that is interfaced with pytest through the `--cov=./` command:

```
pip install codecov pytest-cov
cd radis/test
pytest --cov=./
```

Performance benchmarks

RADIS performance is tested against past versions on a dedicated project : radis-benchmark.

Results can be found on : https://radis.github.io/radis-benchmark/

Select tests

You can select or ignore some tests with the `-m` option, for instance run only the fast tests with:

```
pytest -m fast --cov=./
```

The list of tags is:

- fast : each test should run under 1s
- needs_connection : requires an internet connection
- needs_python3 : a Python-3 only test
- needs_config_file: needs `~/radis.json` to be defined.

Plus some tags for user-defined [HITEMP-2010] databases, as given in the *Configuration file* section

- needs_db_HITEMP_CO2_DUNHAM : requires HITEMP-CO2-DUNHAM database in `~/radis.json`
- needs_db_HITEMP_CO_DUNHAM : requires HITEMP-CO-DUNHAM database in `~/radis.json`
- needs_db_CDSD_HITEMP_PC : requires CDSD-HITEMP-PC database in `~/radis.json`

The default test routine run on Travis CI is (see the `radis/.gitlab-ci.yml` file):

```
pytest -m "not needs_config_file" --cov=./;
```

Which ignores all test that rely on the [HITEMP-2010] databases, as they cannot (yet) be downloaded automatically on the test machine.

Write new tests

Any function starting with `test` will be picked by pytest. Use assert statements within the function to test properties of your objects. Ex:

```
def test_positive(a):
    assert a>0
```

Tips: make sure you don't open any figure by default in your test routines, else it will be stuck when called by pytest. Or, force a non-blocking behaviour adding the following lines within your test function:

```
if plot:
    import matplotlib.pyplot as plt
    plt.ion()   # dont get stuck with Matplotlib if executing through pytest
```

See: https://github.com/statsmodels/statsmodels/issues/3697

Test files

To make the errors as reproducible as possible, try to use the test files provided in the Test files are provided in the radistestfiles and radistestvalidation folders. They contain examples of line databases, spectra, or energy levels. The path to these test files can be retrieved using the `getTestFile()` and `getValidationCase()` functions, respectively.

Load a line database file

```
from radis.test.utils import getTestFile
from radis.io.hitran import hit2df
df = hit2df(getTestFile("hitran_CO_fragment.par"))

print(df)  # replace with your test code

>>> Out:

       id  iso       wav           int             A  ...  gpp  branch  jl  vu  vl
0   5    1  3.705026  2.354000e-44  2.868000e-10  ...  1.0       1   0   4   4
1   5    1  3.740024  1.110000e-38  5.999000e-09  ...  1.0       1   0   3   3
2   5    1  3.775024  9.233000e-34  1.947000e-08  ...  1.0       1   0   2   2
3   5    1  3.810028  5.706000e-29  4.130000e-08  ...  1.0       1   0   1   1
4   5    1  3.845033  3.300000e-24  7.207000e-08  ...  1.0       1   0   0   0
5   5    1  7.409906  1.815000e-43  2.726000e-09  ...  3.0       1   1   4   4
6   5    1  7.479900  8.621000e-38  5.746000e-08  ...  3.0       1   1   3   3
7   5    1  7.549901  7.177000e-33  1.867000e-07  ...  3.0       1   1   2   2
8   5    1  7.619908  4.436000e-28  3.961000e-07  ...  3.0       1   1   1   1

[9 rows x 16 columns]
```

Load a Spectrum object

```
from radis.test.utils import getTestFile
from radis import load_spec
s = load_spec(getTestFile("CO_Tgas1500K_mole_fraction0.5.spec"))

print(s)    # replace with your test code

>>> Out:

    Spectrum Name:  CO_Tgas1500K_mole_fraction0.5.spec
Spectral Arrays
----------------------------------------
   abscoeff         (37,870 points)
Populations Stored
----------------------------------------
   CO                [1]
Physical Conditions
----------------------------------------
   Tgas                 1500 K
   Trot                 1500 K
   Tvib                 1500 K
   isotope              1
   mole_fraction        0.5
   molecule             CO
   path_length          0.01 cm
   pressure_mbar        1013.25 mbar
   rot_distribution     boltzmann
   self_absorption      True
   state                X
   thermal_equilibrium  True
   vib_distribution     boltzmann
```

(continues on next page)

(continued from previous page)

```
   wavelength_max       4801.3089 nm
   wavelength_min       4401.1999 nm
   wavenum_max          2272.1077 cm-1
   wavenum_min          2082.7654 cm-1
Computation Parameters
----------------------------------------
   Tref                 296 K
   broadening_max_width  10 cm-1
   cutoff               1e-25 cm-1/(#.cm-2)
   db_assumed_sorted    True
   dbformat             hitran
   dbpath               d:/github/radis/radis/test/files/hitran_co_3iso_2000_2300cm.par
   levelsfmt            neq
   parfuncfmt           hapi
   pseudo_continuum_threshold  0
   wavenum_max_calc     2277.1104 cm-1
   wavenum_min_calc     2077.7654 cm-1
   waveunit             cm-1
   wstep                0.005 cm-1
Information
----------------------------------------
   calculation_time     0.14 s
   chunksize            10000000.0
   db_use_cached        True
----------------------------------------
```

Report errors

If you encounter any error, open an Issue on GitHub

To simplify the debugging process, provide a code snippet that reproduces the error. If you need a line database, spectrum, or energy level, try to use one of the *test files*.

Debugging

See the `printdbg()` function in `radis.misc`, and the `DEBUG_MODE` global variable.

2.10 References

2.10.1 Spectroscopic constants

CO2

Version 0.9 uses uses the CO2 spectrosopic coefficients of [Klarenaar2017] , Table 2,3 and the references therein. These constants have been compiled for Treanor distributions and may not be suited for Boltzmann distributions.

```
{
"CO2": {
    "isotopes":{
```

(continues on next page)

(continued from previous page)


```
        "1": {
            "electronic_level":{
                "001":{
                    "Te_cm-1": 0.0,
                    "dzero_cm^-1": 44600,
                    "g_e": 1.0,
                    "index": 1,
                    "name": "X1SIGu+",
                    "#REF": "Klarenaar 2017 doi/10.1088/1361-6595/aa902e Table 2,3 and␣
↪the references therein",
                    "#DOI": "10.1088/1361-6595/aa902e",
                    "we1_cm-1":  1333.93,
                    "we2_cm-1":  667.47,
                    "we3_cm-1":  2349.16,
                    "wexe1_cm-1":   2.93,
                    "wexe2_cm-1":  -0.38,
                    "wexe3_cm-1": 12.47,
                    "Be_cm-1":  0.39022,
```


See the full list of constants of the public version as text or on GitHub

In version 1.0, RADIS will use the spectroscopic constants of [Suzuki1968]

CO

CO uses the Dunham coefficients of [Guelachvili1983]


```
{
"CO": {
    "isotopes":{
        "1": {
            "electronic_level":{
                "001":{
                    "Te_cm-1": 0.0,
                    "dzero_cm^-1": 89490.0,
                    "g_e": 1.0,
                    "index": 1,
                    "name": "X1SIG+",
                    "nvib1": 17,
                    "nvib2": 83,
                    "r_equil_angstrom": 1.128323,
                    "#REF_Yij": "Guelachvili 1983 doi/10.1016/0022-2852(83)90203-5",
                    "#DOI": "10.1016/0022-2852(83)90203-5",
                    "Y10_cm-1":  0.2169813079e+04,
                    "Y20_cm-1": -0.1328790587e+02,
                    "Y30_cm-1":  0.1041444739e-01,
                    "Y40_cm-1":  0.6921598529e-04,
```


See the full list of constants of the public version as text or on GitHub

You can use *your own set of spectroscopic constants*, or precompute energy levels and use them directly (see the Energy level database).

2.10.2 References

Bibliography

List of bibliographic references used in this project:

Line Databases

Reference of supported line databases:

The latest HITRAN database version is automatically downloaded if using `databank='hitran'`.

The latest HITEMP database version is automatically downloaded if using `databank='hitran'`.

The latest ExoMol database is automatically downloaded if using `databank='exomol'`. ExoMol contains multiple sub-databases per molecule. See `fetch_exomol()`

The GEISA 2020 database is automatically downloaded with `databank='geisa'`.

For download and configuration of line databases, see the *Line Databases section*

Tools Used Within RADIS

For data retrieval :

2.10.3 Licence

The code is available for use and modifications on GitHub under a GNU LESSER GENERAL PUBLIC LICENSE (v3), i.e. modifications must remain public and under LGPLv3.

2.10.4 Cite

RADIS is built on the shoulders of many state-of-the-art packages and databases. If using RADIS for your work, **cite all of them that made it possible**.

Starting from 0.9.30, you can retrieve the bibtex entries of all papers and references that contribute to the calculation of a Spectrum, with `cite()`

```
s.cite()
```

See the *citation example*. The references usually include :

Line-by-line algorithm :

- cite the line-by-line code as [RADIS-2018]
- if using the default optimization, cite the new spectral synthesis algorithm [Spectral-Synthesis-Algorithm]
- for reproducibility, mention the RADIS version number. Get the version with `radis.get_version()` (latest version available is)

```
import radis
radis.get_version()
```

Database and database retrieval algorithms :

- cite the Line Databases used (for instance, [HITRAN-2020], [HITEMP-2010] or [CDSD-4000]).
- if downloading [HITRAN-2020] directly with `fetch('hitran')`, cite [HAPI] which is the underlying interface that makes it work !
- if running nonequilibrium calculations, mention the reference of the spectroscopic constants used to calculate the energies (for instance, the *RADIS built-in constants*)

2.10.5 Research Work

Research papers using RADIS and the associated algorithms :

- Papers citing : https://scholar.google.fr/scholar?cites=5826973099621508256
- Papers citing : https://scholar.google.fr/scholar?cites=17363432006874800849

2.10.6 Conferences

Talks presenting RADIS features and algorithms, available on the RADIS Youtube Channel :

DIT Algorithm at the ISMS 2021 Conference, by D.v.d. Bekerom :

RADIS features and updates at the ASA-HITRAN 2022 Conference, by E. Pannier :

2.10.7 Spectroscopy Tutorials

Tutorials for molecular spectroscopy can be found here at https://github.com/radis/spectro101

- Lab Spectroscopy: https://github.com/radis/spectro101/blob/main/102_lab_spectroscopy.ipynb
- Line Broadening : https://github.com/radis/spectro101/blob/main/103_lineshape_broadening.ipynb

2.10.8 Useful Links

RADIS:

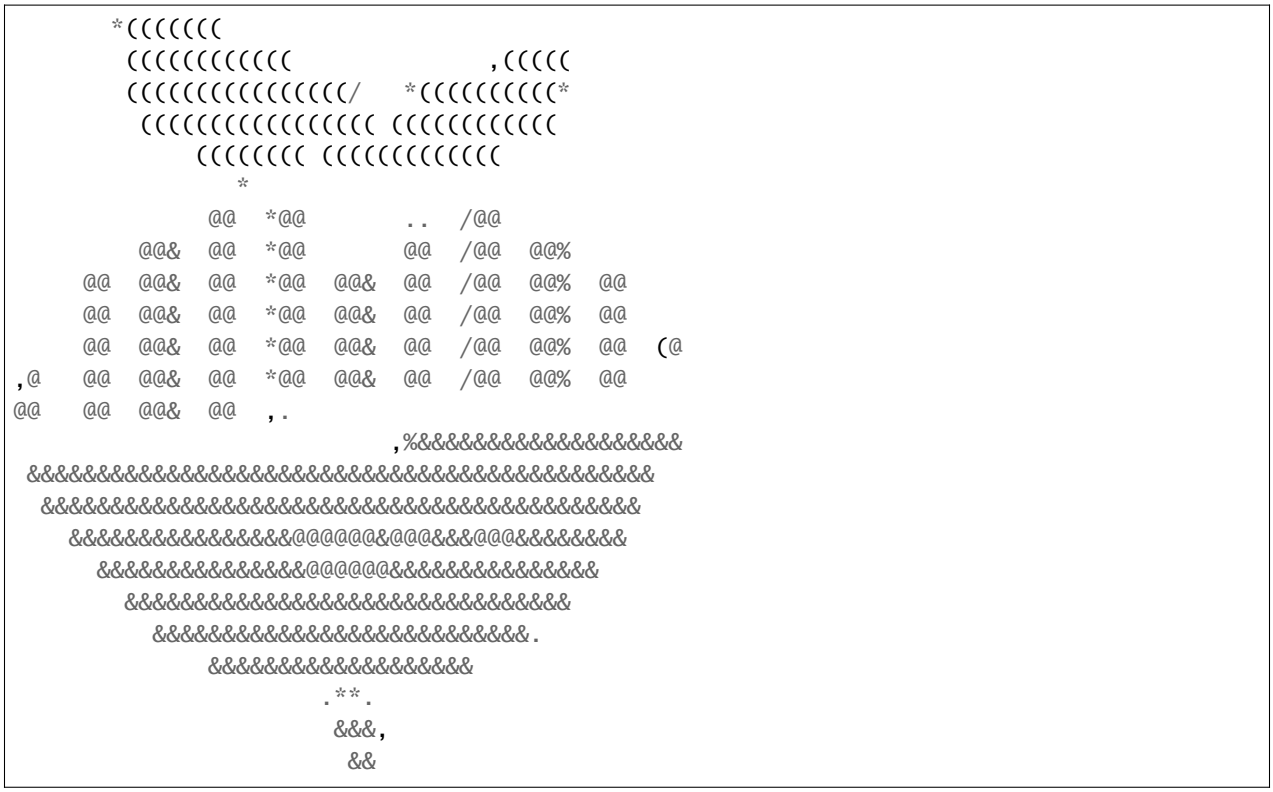
- Documentation:
- Help:
- RADIS Articles:
- Source Code:
- Test Status:
- PyPi Repository:
- Interactive Examples:
- Try online : *RADIS Lab*

Similar packages or softwares you could be interested in (please reference your own if not there!):

- A collaborative list of tools for spectroscopy

2.11 API

Documentation of each module and each function in RADIS



`radis`:

`db`	Definition of molecules and list of spectroscopic constants
`io`	Parsers for various databases
`lbl`	Core of the line-by-line calculations
`levels`	Energy levels and definitions of molecular models
`los`	Line-of-sight (multislabs) module for 1-D radiative transfer.
`misc`	Misc.
`phys`	Physical constants and conversion.
`spectrum`	Spectrum-class module for post-processing.
`tools`	Tools : database of spectra, line survey, interface with Cantera.

2.11.1 radis.db

2.11.2 radis.io

The io module provides the Python interfaces to stream handling. The builtin open function is defined in this module.

At the top of the I/O hierarchy is the abstract base class IOBase. It defines the basic interface to a stream. Note, however, that there is no separation between reading and writing to streams; implementations are allowed to raise an OSError if they do not support a given operation.

Extending IOBase is RawIOBase which deals simply with the reading and writing of raw bytes to a stream. FileIO subclasses RawIOBase to provide an interface to OS files.

BufferedIOBase deals with buffering on a raw byte stream (RawIOBase). Its subclasses, BufferedWriter, BufferedReader, and BufferedRWPair buffer streams that are readable, writable, and both respectively. BufferedRandom provides a buffered interface to random access streams. BytesIO is a simple stream of in-memory bytes.

Another IOBase subclass, TextIOBase, deals with the encoding and decoding of streams into text. TextIOWrapper, which extends it, is a buffered text interface to a buffered raw stream (`BufferedIOBase`). Finally, StringIO is an in-memory stream for text.

Argument names are not part of the specification, and only the arguments of open() are intended to be used as keyword arguments.

data:

DEFAULT_BUFFER_SIZE

> An int containing the default buffer size used by the module's buffered I/O classes. open() uses the file's blksize (as obtained by os.stat) if possible.

2.11.3 radis.lbl

2.11.4 radis.levels

2.11.5 radis.los

2.11.6 radis.misc

2.11.7 radis.phys

2.11.8 radis.spectrum

2.11.9 radis.tools

Q&A Forum

BIBLIOGRAPHY

[Klarenaar2017] B. L. M. Klarenaar, R. Engeln, D. C. M. van den Bekerom, M. C. M. van de Sanden, A. S. Morillo-Candas, O. Guaitella, "Time evolution of vibrational temperatures in a CO2 glow discharge measured with infrared absorption spectroscopy", *Plasma Sources Science and Technology* 26 (11) (2017) 115008, ISSN 1361-6595, doi:10.1088/1361-6595/aa902e.

[Suzuki1968] I. Suzuki, "General anharmonic force constants of carbon dioxide" *Journal of Molecular Spectroscopy* 25 479-500 ISSN 00222852 doi:10.1016/S0022-2852(68)80018-9

[Guelachvili1983] G.Guelachvili, D.de Villeneuve R.Farrenq, W.Urban, J.Verges, Dunham coefficients for seven isotopic species of CO doi:10.1016/0022-2852(83)90203-5

[CANTERA] D. G. Goodwin, H. K. Moffat, R. L. Speth, "Cantera: An Object-oriented Software Toolkit for Chemical Kinetics"", Thermodynamics, and Transport Processes, http://www.cantera.org, doi:10.5281/zenodo.170284, 2017.

[RADIS-2018] E. Pannier, C. O. Laux, "RADIS: A Nonequilibrium Line-by-Line Radiative Code for CO2 and HITRAN-like database species", Journal of Quantitative Spectroscopy and Radiative Transfer (2018) doi:10.1016/j.jqsrt.2018.09.027

[Spectral-Synthesis-Algorithm] D.C.M. van den Bekerom, E. Pannier, "A Discrete Integral Transform for Rapid Spectral Synthesis", Journal of Quantitative Spectroscopy and Radiative Transfer (2021) doi:10.1016/j.jqsrt.2020.107476

[Rothman-1998] L.S. Rothman, C.P. Rinsland, A. Goldman, S.T. Massie D.P. Edwards, J-M. Flaud, A. Perrin, C. Camy-Peyret, V. Dana, J.-Y. Mandin, J. Schroeder, A. McCann, R.R. Gamache, R.B. Wattson, K. Yoshino, K.V. Chance, K.W. Jucks, L.R. Brown, V. Nemtchinov, P. Varanasi "The Hitran Molecular Spectroscopic Database and Hawks (Hitran Atmospheric Workstation): 1996 Edition", Journal of Quantitative Spectroscopy and Radiative Transfer 60 (1998) 665 - 710, doi:10.1016/S0022-4073(98)00078-8

[TIPS-2020] R.R Gamache, B. Vispoel, M. Rey, A. Nikitin, V. Tyuterev, O. Egorov, I.E Gordon, V. Boudon, "Total internal partition sums for the HITRAN2020 database", Journal of Quantitative Spectroscopy and Radiative Transfer 271 (2021) doi:10.1016/j.jqsrt.2021.107713

[HITRAN-2016] I. Gordon, L. Rothman, C. Hill, R. Kochanov, Y. Tan, P. Bernath, V. Boudon, A. Campargue, B. Drouin, J. M. Flaud, R. Gamache, J. Hodges, V. Perevalov, K. Shine, M.-a. Smith, The HITRAN2016 Molecular Spectroscopic Database, Journal of Quantitative Spectroscopy and Radiative Transfer 6 (38) (2017) 3–69, ISSN 00224073, doi:10.1016/j.jqsrt.2017.06.038.

[HITRAN-2020] I.E. Gordon, L.S. Rothman, R.J. Hargreaves, R. Hashemi, E.V. Karlovets, F.M. Skinner, E.K. Conway, C. Hill, R.V. Kochanov, Y. Tan, P. Wcis{\l}o, A.A. Finenko, K. Nelson, P.F. Bernath, M. Birk, V. Boudon, A. Campargue, K.V. Chance, A. Coustenis, B.J. Drouin, J.{textendash}M. Flaud, R.R. Gamache, J.T. Hodges, D. Jacquemart, E.J. Mlawer, A.V. Nikitin, V.I. Perevalov, M. Rotger, J. Tennyson, G.C. Toon, H. Tran, V.G. Tyuterev, E.M. Adkins, A. Baker, A. Barbe, E. Canè, A.G. Cs{‘{a}}sz{‘{a}}r, A. Dudaryonok, O. Egorov, A.J. Fleisher, H. Fleurbaey, A. Foltynowicz, T. Furtenbacher, J.J. Harrison, J.{textendash}M.

Hartmann, V.M. Horneman, X. Huang, T. Karman, J. Karns, S. Kassi, I. Kleiner, V. Kofman, F. Kwabia-Tchana, N.N. Lavrentieva, T.J. Lee, D.A. Long, A.A. Lukashevskaya, O.M. Lyulin, V.Yu. Makhnev, W. Matt, S.T. Massie, M. Melosso, S.N. Mikhailenko, D. Mondelain, H.S.P. Müller, O.V. Naumenko, A. Perrin, O.L. Polyansky, E. Raddaoui, P.L. Raston, Z.D. Reed, M. Rey, C. Richard, R. T{‘{o}}bi{‘{a}}s, I. Sadiek, D.W. Schwenke, E. Starikova, K. Sung, F. Tamassia, S.A. Tashkun, J. Vander Auwera, I.A. Vasilenko, A.A. Vigasin, G.L. Villanueva, B. Vispoel, G. Wagner, A. Yachmenev, S.N. Yurchenko The HITRAN2020 molecular spectroscopic database, Journal of Quantitative Spectroscopy and Radiative Transfer (277) (2022), doi:10.1016/j.jqsrt.2021.107949.

[HITEMP-2010] L. S. Rothman, I. E. Gordon, R. J. Barber, H. Dothe, R. R. Gamache, A. Goldman, V. I. Perevalov, S. A. Tashkun, J. Tennyson, HITEMP, the high-temperature molecular spectroscopic database, Journal of Quantitative Spectroscopy and Radiative Transfer 111 (15) (2010) 2139–2150, ISSN 00224073, doi:10.1016/j.jqsrt.2010.05.001.

[CDSD-4000] S. A. Tashkun, V. I. Perevalov, CDSD-4000: High-resolution, high-temperature carbon dioxide spectroscopic databank, Journal of Quantitative Spectroscopy and Radiative Transfer 112 (9) (2011) 1403–1410, ISSN 00224073, doi:10.1016/j.jqsrt.2011.03.005

[ExoMol-2020] Tennyson et al., The 2020 release of the ExoMol database: Molecular line lists for exoplanet and other hot atmospheres, Journal of Quantitative Spectroscopy and Radiative Transfer 255, (2020), 107228, doi:10.1016/j.jqsrt.2020.107228

[ExoMol-2016] Tennyson et al., The ExoMol database: molecular line lists for exoplanet and other hot atmospheres, J. Molec. Spectrosc., 327, 73-94 (2016), doi:10.1016/j.jms.2016.05.002

[GEISA-2020] Delahaye et al, The 2020 edition of the GEISA spectroscopic database, J. Molec. Spectrosc., 380, 111510 (2021) doi:10.1016/j.jms.2021.111510

[HAPI] HAPI: The HITRAN Application Programming Interface R. Kochanov, I. Gordon, L. Rothman, P. Wcisło, C. Hill, J. Wilzewski, “HITRAN Application Programming Interface (HAPI): A comprehensive approach to working with spectroscopic data”, Journal of Quantitative Spectroscopy and Radiative Transfer 177 (2016) 15–30, ISSN 00224073, doi:10.1016/j.jqsrt.2016.03.005.

[Astroquery] astroquery: An Astronomical Web-querying Package in Python 10.3847/1538-3881/aafc33

PYTHON MODULE INDEX

INDEX