cse2rat: a script to generate RATRAN input files from UDFA circumstellar envelope models

Paul M. Woods, *Astrophysics Research Centre, Queen's University Belfast*

May 2015

`cse2rat.py` is a Python script which will take standard output files from the UDFA circumstellar envelope model[1] and produce input files suitable for RATRAN[2]. `cse2rat.py` runs on Python 2.x (tested on 2.7). It requires very little setup, needing just three inputs in order to run on your system.

Using cse2rat.py

Initial configuration

Three changes must be made to the `cse2rat.py` script in order to adapt it to the user's system. The user must specify:

1. The molecule of interest OR specify 'ALL' molecules
2. The expansion velocity (in $\mathrm{km\,s^{-1}}$) of the circumstellar envelope in question
3. The system path to the RATRAN installation.

See below for further details.

Choice of molecule

`cse2rat.py` may be run for a single molecule, or for all molecules for which there are collisional data in RATRAN's `molec` directory. Typically molecular data files for RATRAN can be downloaded from the LAMDA database[3] at Leiden. At the time of writing, there are 67 collisional data files in LAMDA, however, some species have multiple files (according to different specifications) and some species are not available in the standard UDFA Rate12 species list (e.g., DCO^+).

The user must specify the value of the `specString` variable at the start of the `cse2rat.py` script. For a single atom, ion or molecule, the user must enter the species in the exact manner in which it is present in the UDFA CSE model output file, e.g.,

```
specString = 'SiS'
```

Alternatively, the user may desire to produce RATRAN input files for all potential species, in which case the user must enter 'ALL', i.e.,

```
specString = 'ALL'
```

Note that `cse2rat.py` will generate three files per molecule.

Envelope expansion velocity

Unfortunately, the expansion velocity is not written as an output in the standard UDFA CSE model, and so the user must enter this into `cse2rat.py` manually. Units for this quantity are $\mathrm{km\,s^{-1}}$.

```
vexp = 14.5
```

The user may also wish to change the Doppler width, also expressed in $\mathrm{km\,s^{-1}}$.

```
db = 1.0
```

[1] `http://udfa.net/index.php?mode=downloads`
[2] [1] and `http://www.sron.rug.nl/~vdtak/ratran/`
[3] `http://www.strw.leidenuniv.nl/~moldata/`

Path to the RATRAN installation

The user must specify the absolute path to their RATRAN installation, e.g.,

```
ratranPath = '/Users/abc/Desktop/Ratran/'
```

This is where `cse2rat.py` will look for molecular data files (in the `molec` directory), and the place where it will write output files. Please be sure to include a trailing '/' in the path.

Running cse2rat.py

Once `cse2rat.py` has been configured to your requirements, it should be placed in the directory with the output of the UDFA CSE code. By default, `cse2rat.py` will look for the file named `csnum_rate13.out`. This of course can be modified at the beginning of `cse2rat.py`:

```
dataFile = open('csnum_rate13.out','rb')
```

The user should be sure to use the CSE model output which contains the **number density** of species, not that which contains fractional abundances.

The script can be run from the command line:

```
$ python cse2rat.py
```

or from the Python command line:

```
>>> execfile('cse2rat.py')
```

`cse2rat.py` will then generate the `*.mdl`, `*amc.inp` and `*sky.inp` files RATRAN requires.

The script will specify the electronic transitions for each molecule according to which transitions fall into the various ALMA bands from Band 3 to Band 10 (excluding Band 5). Thus for CO, `cse2rat.py` will specify that RATRAN's `sky` routine should produce profiles for the transitions 1–8 (excluding 5) of CO, which in this case correspond to the J=1–0 to J=8–7 rotational transitions of CO. Transition numbers relate to the transitions listed in RATRAN's molecular data files. If the user adds collisional data files to RATRAN (those which are not from the LAMDA database), (s)he should ensure that `cse2rat.py`'s `fileToMol` dict is updated with the filename, the species formula as present in the Rate12 database, and all possible transitions.

Running RATRAN

RATRAN can be run as normal. From the RATRAN directory:

```
$ bin/amc COamc.inp
$ bin/sky COsky.inp
```

Alternatively, one can run the 'ALL' script, which will run RATRAN for all possible molecules:

```
$ bin/amc ALLamc.inp
$ bin/sky ALLsky.inp
```

The user can, of course, edit the RATRAN input files as they normally would.

References

[1] Hogerheijde, M. R. and van der Tak, F. F. S. (2000). An accelerated Monte Carlo method to solve two-dimensional radiative transfer and molecular excitation. With applications to axisymmetric models of star formation. *A&A*, 362, 697

Acknowledgments

PMW would like to thank STFC for financial support for this work.