



User Guide to ARCiS

Michiel Min

June 4, 2019

1 Introduction

1.1 Terms of use

By using ARCiS you agree to the following:

- You are not permitted to pass (parts of) the code to anyone else. If anyone else is interested, let him/her drop me an email: M.Min@sron.nl
- You offer me co-author rights on any paper that uses results computed with ARCiS

2 Installing ARCiS

Before installing ARCiS you need:

A Fortran compiler: This can be either `gfortran` or `ifort` (any other might work but is not tested).

`cfitsio` library: This is needed to allow fortran to read and write binary fits files.

`MultiNest`: This allows ARCiS to perform Bayesian retrievals.

On a Mac the easiest is to install `cfitsio` through HomeBrew (google 'homebrew' for installation instructions. After that:

```
# brew install cfitsio
```

Next fetch `MultiNest`:

```
# git clone https://github.com/JohannesBuchner/MultiNest.git
# cd MultiNest/build
# cmake ..
# make
# sudo make install
```

Next fetch the git source code from:

```
# mkdir ARCiS ; cd ARCiS
# git clone https://github.com/michielmin/ARCiS.git ./src
# cd src
# make gfort=true multi=true
```

This creates the ARCiS binary, which you can put in any path accessible (/usr/bin or something like that).

Now you also need all the data files. These need to be stored under $\$(HOME)/ARCiS/Data/$

3 Using ARCiS

To run ARCiS you further only need an input file. On the prompt type:

```
# ARCiS inputfile.dat -o outputdir
```

which creates the output directory `outputdir` containing the output files.

There are several options included in ARCiS. These are given as keywords in the `inputfile.dat` file (or whatever you call it). Keywords are always given as `key=value` and can be anywhere in the file (order does not matter). Also, you can overwrite keywords set in the input file from the command line in the following way

```
# ARCiS inputfile.dat -o outputdir -s key1=value1 -s key2=value2
```

Any number of keys can be set on the command line. Just make sure the first argument of the command line is the name of your input file. Note that ARCiS always takes the last keyword value it encounters, first reading the input file, next the command line keywords one by one.

4 Keywords

Rp

Radius of the planet in Jupiter radii.

Mp

Mass of the planet in Jupiter masses.

Pp

Atmospheric pressure corresponding to radius Rp. Default is 10 bar.

Tstar

Temperature of the host star in K.

Rstar

Radius of the host star in Solar radii.

distance

distance to the system in parsec.

betaT

Cosine of the angle of incoming radiation.

Dplanet

Distance of the planet to the star in AU.

chemistry

Logical determining if chemistry is computed or not (either .true. or .false.)

condensates

Logical determining if condensates should be taken into account in the chemistry computations (default is .false. and most stable is to leave it like that).

pmin, pmax

Minimum, maximum pressure considered in the atmosphere

nr

Number of pressure points

lmin, lmax

Minimum, maximum wavelength considered in micron. Note that for temperature computations these must be set wide enough to ensure energy balance is properly computed.

`specres`

Spectral resolution R in $\lambda/d\lambda$

`specresdust`

Spectral resolution for computation of the solid state species in the clouds.

`cia`

Logical determining if CIA is taken into account

`maxtau`

Maximum optical depth considered for the raytracing

`compute`

Logical determining if the opacities need to be recomputed from the linelists

`scattering`

Logical determining if scattering of the thermal radiation is included

`scattstar`

Logical determining if scattering from the star is included

`TeffP`

Effective temperature of the radiation from inside the planet

`computeT`

Logical determining if the temperature structure is computed self-consistently

`maxiter`

Maximum number of iterations for the temperature structure

`COratio`

C/O ratio of the atmosphere

`metallicity`

Metallicity of the atmosphere

`planetname`

Name of the planet to read from the database. Radius, mass and distance of the planet and the star are read from the database.

5 Examples

There are two examples on the VirtualBox server, retrieval and forward modelling.