SPECIES:
Spectroscopic Parameters and atmosphEric ChemIstriEs of Stars

User Manual

Maritza Soto

August 10, 2018

Contents

Chapter 1

Introduction

SPECIES is a code written mostly in Python, that computes stellar parameters from high resolution echelle spectra. The code measures the equivalent width of the given spectra (more details on the accepted spectra in chapter 3), and solves the local thermodynamic equilibrium equation for a set of temperature, metallicity, surface gravity and microturbulent velocity. It then uses those values to compute the broadening, macroturbulent and rotational velocities, and the mass and age, as well as uncertainties for each parameter.

This present manual is intended to help users understand how to perform the calculation of the parameters with SPECIES, and understanding the results and corresponding files. For an explanation of the methods used to compute the parameters, please read Soto & Jenkins 2018.

1.1 Installation

SPECIES is publicly available in GitHub[1]. In order to install it, it is necessary to just place the corresponding folder in your preferred location. Before using it, make sure you have installed the python packages listed in the following section, as well as the ARES and MOOG codes (section 1.1.3, 1.1.4).

1.1.1 Contents

Before installing SPECIES, make sure all the necessary files and folders are included. The files are: SPECIES.py, atmos.py, calc_broadening.py, calc_errors.py, calc_params.py, ccf.py, photometric_relations.py, compute_snr.py, find_mass.py, interpol_function.py, instruments.py, run_ARES.bash, and spectra.py.

The folders, and its contents, are:

- `ARES2`
- `EW/`: `plots_spectra/`
- `MOOGFEB2017/`: MOOG files, abfind.par.
- `MOOG_linelist/`
- `Spectra/`: linelist.dat, lines_ab.dat, linelist_vsini.dat, mine.opt, `plots_ccf/`, `plots_fit`.

[1] `https://github.com/msotov/SPECIES/`

- `atm_models/`: `atlas9/`
- `binary_masks/`: G2.mas, K0.mas, K5.mas, M2.mas.
- `debugging_files/`
- `isochrones/`: `plots/`
- `output/`: `MOOG_output_files/`, `coeffs_error_files/`
- `plots_broadening/`

1.1.2 Requirements

SPECIES uses a large number of python packages in order to optimize the calculation of the parameters. Most of them are included in your average python installation, but some of them have to be installed separately. These are:

- `numpy` (`http://www.numpy.org`)
- `scipy` (`https://www.scipy.org`)
- `matplotlib` (`http://matplotlib.org`)
  For SPECIES, it is necessary that the default backend for matplotlib is `Agg`, in order to avoid problems with the `multiprocessing` package. In order to change that, first locate where your `matplotlibrc` file is. Do the following:

```
import matplotlib
matplotlib.matplotlib_fname()
```

  and in the `matplotlibrc` file change the backend:

```
backend      : Agg
```

- `astropy` (`http://www.astropy.org`)
- `astroquery` (`http://astroquery.readthedocs.io/en/latest/`)
- `PyAstronomy` (`http://www.hs.uni-hamburg.de/DE/Ins/Per/Czesla/PyA/PyA/index.html`)
- `isochrones` (`http://isochrones.readthedocs.io/en/latest/`)
  We find that with `MultiNest/PyMultiNest` (`http://astrobetter.com/wiki/MultiNest+Installation+Notes`) we obtain more accurate results for the mass and age computation than with `emcee` (`http://dan.iel.fm/emcee/current/`), therefore we recommend having MultiNest installed in your computer.
- `mwdust` (`https://github.com/jobovy/mwdust`)

1.1.3 ARES

We use the `Ares` code (Sousa et al. 2008) to compute the equivalent width for a set of lines in our spectra. It can be obtained from `http://www.astro.up.pt/~sousasag/ares/`, but a version of it is also included in SPECIES, with some minor changes from the original files. The requirements and installation are explained in the README file inside the ARES2 folder. Also, call the compiled file as `ARES_v2`. This is necessary for running SPECIES, because the script that executes ARES calls the executable `ARES_v2`. All the other installation steps should remain the same.

There has been some problems with compiling ARES using the latest version of `gsl`. A modified version for when this happens is available at `https://github.com/sousasag/ARES`, but when compiling we found a few issues, which where solved and are included in the ARES version available in SPECIES. We strongly recommend you to install that version.

Note: Please make sure that you created a link into your link directory (usually called `~/bin`, if not, create it) for ARES, by typing

`$ln -s ARES_v2 ~/bin/ARES_v2`

and making sure that the path to the `~/bin` directory is included in your bash file. This is necessary for later executing ARES.

1.1.4 MOOG

We use MOOG (`http://www.as.utexas.edu/~chris/moog.html`) in order to solve the local thermodynamic equation, for a set of parameters. A version of MOOG is included in SPECIES, from the February 2017 release. The version included in SPECIES does not require for *SM* to be installed (MOOGSILENT), and it is the one used in SPECIES. In order to install it, first delete all the .o files (executables), and modify the following files:

- Makefile.XXsilent, where XX is the type of machine you are using (ie. Makefile.rh64silent or Makefile.macdesksilent).

  Change the compilers and path to the libraries:

```
CC = cc
FC = gfortran -w

# the following lines point to some needed libraries
X11LIB = /usr/X11R6/lib
SMLIB = /Applications/scisoft/i386/Packages/sm/lib
AQLIB = /usr/local/scisoft/lib
```

  If you don't have *SM* installed, then ignore the paths to `SMLIB` and `AQLIB`.

  Also, reaching the end of the file, comment the line that says `-L$(SMLIB) -lplotsub -ldevices -lutils -L$(AQLIB) -laquaterm`, so that it will look like:

```
MOOGSILENT:  $(OBJECTS);
    $(FC) $(OBJECTS) -o MOOGSILENT -L$(X11LIB) -lX11 \
#   -L$(SMLIB) -lplotsub -ldevices -lutils -L$(AQLIB) -laquaterm
```

- Moogsilent.f

  Change the path of MOOG

```
moogpath = '/path/to/the/MOOG/folder'
```

Change the type of machine (possible values are mac, pcl and uni)

```
machine = "mac"
```

Once that is changed, type: `make -f Makefile.XXsilent`, where XX is the type of machine.

Finally, add the path to the MOOG directory to your `bash` file, so it looks something like this:

```
export PATH="/Users/maritza/SPECIES/MOOGFEB2017:$PATH"
```

Working with another version of MOOG

If you want to use a version of MOOG different from the one included in SPECIES, then in order to use it you will have to modify some other files, and check if some quantities agree with the values in SPECIES.

Files in MOOG

If you don't have *SM* installed, you'll have to modify the following files in the MOOG directory:

- `Abundplot.f`, `Binplot.f`, `Cogplot.f`, `Defcolor.f`, `Drawcurs.f`, `Fluxplot.f`, `Makeplot.f`, `Pointcurs.f`, `Specplot.f`

  Copy them into other files with different names (ie. `file_old.f`), and modify the originals so that they all look like:

```
      subroutine XXX
c*****************************************************
c     Description of the subroutine
c*****************************************************

      end
```

  with `XXX` is the name of each subroutine mentioned above. If you have questions, check those files in the `MOOG` file included in SPECIES.

- In `Makefile.XXsilent` (with XX the type of machine), reaching the end of the file, comment the line that says
  `-L$(SMLIB) -lplotsub -ldevices -lutils -L$(AQLIB) -laquaterm`, so that it will look like:

```
MOOGSILENT:  $(OBJECTS);
    $(FC) $(OBJECTS) -o MOOGSILENT -L$(X11LIB) -lX11 \
#   -L$(SMLIB) -lplotsub -ldevices -lutils -L$(AQLIB) -laquaterm
```

Files outside the MOOG directory

- `calc_params.py`: In the function `runMOOG_ab`, close to the end of the file, check that the abundances being subtracted to each elements agree with the ones in `Batom.f`, in the MOOG directory. The ones used are from Asplund et al. 2009. Currently, it looks like this:

```
    ab_NaI = abund['ab']['NaI'] - 6.24
    ab_MgI = abund['ab']['MgI'] - 7.60
    ab_AlI = abund['ab']['AlI'] - 6.45
    ab_SiI = abund['ab']['SiI'] - 7.51
    ab_CaI = abund['ab']['CaI']- 6.34
    ab_TiI = abund['ab']['TiI']- 4.95
    ab_TiII = abund['ab']['TiII']- 4.95
    ab_CrI = abund['ab']['CrI'] - 5.64
    ab_MnI = abund['ab']['MnI'] - 5.43
    ab_FeI = abund['ab']['FeI'] - 7.50
    ab_FeII = abund['ab']['FeII'] - 7.50
    ab_NiI = abund['ab']['NiI'] - 6.22
    ab_CuI = abund['ab']['CuI'] - 4.19
    ab_ZnI = abund['ab']['ZnI'] - 4.56
```

- `calc_params.py`: In the function `runMOOG` (at line $\sim$ 200), check that the abundance being subtracted to Fe agrees with the abundance for iron in `Batom.f`, in the MOOG directory. Currently, it looks like this:

```
    ab = ab - 7.50
```

- `interpol_function.py`: Close to the end of the file, check that the value being added to the iron abundance agrees with the one being subtracted in the `runMOOG` function, included in `calc_params.py`. Currently, it looks like this:

```
    logefe = 7.50 + xmetal
```

- Verify the path to the MOOG directory in the functions `runMOOG` and `runMOOG_ab`, both included in `calc_params.py`.

- Copy `abfind.par` from the MOOG directory in SPECIES, to your new MOOG directory, with its latest release.

- Finally, add the path to the MOOG directory to your `bash` file, so it looks something like this:

```
export PATH="/Users/maritza/SPECIES/MOOGFEB2017:$PATH"
```

The distribution in the results might also vary from the ones presented in Soto et al. 2018, so be careful when applying any kind of correction from the paper into your new results. The benchmark values we used to calibrate our results might also change, therefore if you're not sure about how the results will change when using another MOOG version, send me an email and I'll try to hep.

1.2 Checking the installation

In order to test if SPECIES was correctly installed, we have included one of the sun's spectra used in our calculus. It's located in `Spectra/sun01_harps.fits`. The idea is to execute SPECIES using that spectra, and see if it runs smoothly, or if there was any error or missing file. Move to the SPECIES directory and type the following in your terminal:

```
python SPECIES.py -starlist sun01 -output sun01
```

If you don't encounter problems, then the files `output/sun01.fits` and `output/sun01.dat` should have been created. They included the final results from SPECIES for that certain spectrum. A description of the input commands is described in the next section, and the files received and created by SPECIES are described in the next chapters.

If you do encounter problems, please check if MOOG and ARES were correctly installed. In your terminal, you could type `ARES_v2`, and it should recognize the command if ARES is installed (that can be typed in any directory). For MOOG, type `MOOGSILENT` in the command line. If it recognizes the command, then MOOG is correctly installed. If you cannot solve the problem, please post it on the GitHub repository for SPECIES, or send an email to `maritsoto@ug.uchile.cl`.

Chapter 2

Running SPECIES

In order to execute SPECIES, you need a high resolution echelle spectra, plus other auxiliary files, which will be better explained later.

The Spectra has to be placed in the `Spectra/` folder (or in your desired folder, please see the `-path_spectra` option below), with the following name format: `starname_inst.fits`, where `inst` is the name of the instrument used to obtain the spectra, in non-capital letters. The name of the spectra cannot be very long neither, otherwise there will be problems with MOOG.

Once that is ready, along with the auxiliary files (if needed), then execute SPECIES by typing

```
python SPECIES.py
```

Important note:

Do not try to kill SPECIES, or interrupt it, after the line

```
The computation will be carried out for the following stars:
```

was printed on your terminal screen.

This is because, after that line is printed, SPECIES will use the multiprocessing package to distribute the computation into the different cores of your system, and it doesn't behave well when trying to interrupt it. If you want to stop the computation, it is better to close the window of terminate it from your activity monitor.

2.1 Command line options

You can also add or modify some options for SPECIES, by adding them on the command line. These are:

- `-starlist`: Name of the stars to use, separated by commas. If the list is too long, retrieve it from inside the code, in the part that says

  ```
  # List of stars for which to compute the parameters
  ```

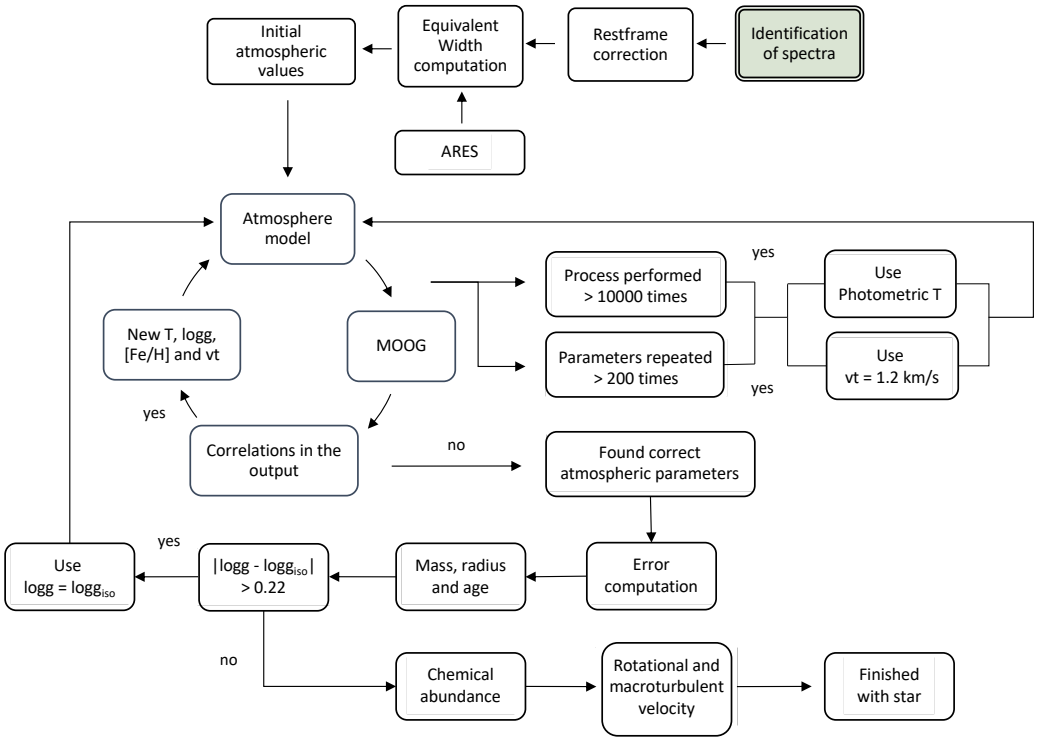

Figure 2.1: Flow diagram showing the process sequence of SPECIES.

- `-inst`: Names of the instruments you want to use, separated by commas and in non-capital letters. If "all" is included, spectra from all accepted instruments will be used. Default is "all".
- `-ncores`: Number of cores to use. Default is 4.
- `-output_format`: Format of the output file. Choices are ['Both', 'ASCII', 'FITS']. Default is 'Both'.
- `-output`: Name of the output file, without extension. Default is 'summary_output'.
- `-hold_params`: Option to set the atmospheric parameters to a certain value. When used, no argument has to be given.
- `-file_params`: Name of the file with the values for the parameters to hold. Will be used only if `-hold_params` was activated.
- `-hold_mass`: Option to set the mass, age, and photometric logg to a certain value. When used, no argument has to be given.
  DEPRECATED IN LATEST VERSION.
- `-file_mass`: Name of the file with the hold values for the mass, age and photometric logg. Will only be used if `-hold_mass` was activated.
  DEPRECATED IN LATEST VERSION.

- `-set_boundaries`: Option to change the boundaries of the atmospheric parameters. When used, no argument has to be given.
- `-file_with_boundaries`: Name of the file with the boundaries for the atmospheric parameters. Will only be used if `-set_boundaries` was activated.
- `-no_abundance`: Option to avoid the computation of atomic abundances. When used, no argument has to be given.
- `-no_error`: Option to avoid the computation of uncertainties to the atmospheric parameters. When used, no argument has to be given.
- `-no_debug`: Option to avoid the creation of a debug file for each star, where the details of the computation will be stored. When used, no argument has to be given.
- `-no_check_t`: Option to avoid using the temperature obtained when using the formulas from Casagrande et al. 2012, or Mann et al. 2015 for M-dwarfs to set boundaries to the temperature in SPECIES. When used, no argument has to be given.
- `-file_colors`: Name of the file with the colors to use to compute the temperature from Casagrande et al. or Mann et al. Used only if `-no_check_t` was not activated. If no file is given, SPECIES will try to retrieve them from a selection of catalogues in Vizier, all listed in Soto et al. 2018.
- `-use_coords`: Option to use the coordinates of the star, instead of its name, to look for the star colors. Used only if `-no_check_t` was not activated. When used, no other arguments has to be given.
- `-file_coords`: Name of the file with the coordinates of the stars (in deg). Used only if `-use_coords` was activated.
- `-mass_from_file`: Option to load the chains created during a previous mass calculation, stored in `isochrones/*.h5`
- `-vt_hold`: Option to try the computation with vt = 1.2 km/s, in the case SPECIES cannot converge to a solution.
- `-no_r_iso_logg`: Do not use the logg from isochrones interpolation to recompute the atmospheric parameters when there is disagreement with the spectroscopic one.
- `-age_limits`: Option to set limits to the parameters space for the stellar age. It has to be in the format log(age), with age in years (not Gyr).
- `-path_spectra`: Path to the directory containing the spectra. It has to be relative to the path of SPECIES, or the absolute path. Default is `./Spectra`.
- `-columns`: Columns to include in the output. The name of the star and the instrument will always be included.
- `-giant`: If the star, or list of stars, are considered to be giants. For the moment, it only modifies the boundaries of the parameter space for which SPECIES searches for the correct temperature.

For example, if I want to run SPECIES for the star HD10700, using only the HARPS spectra available, the coordinates listed in the fits header, with the temperature fixed to a certain value, and the output file to be called 'summary_HD10700', then I'd type:

```
python SPECIES.py -output out_HD10700 -use_coords -starlist summary_HD10700
    -inst harps -hold_params -file_params HD10700_params.txt
```

The format of the optional input files (`file_params`, `file_with_boundaries`, `file_colors`, `file_coords`) is given in the next chapter.

The data flow that SPECIES follows for each star is summarized in Figure 2.1, where each step is explained in detail in Soto & Jenkins 2018.

Chapter 3

Input files

As mentioned in the previous chapter, the input files needed to run SPECIES are a high resolution echelle spectra (.fits file) and a set of auxiliary file, which are not mandatory.

3.1 Input spectra

SPECIES currently accepts spectra from HARPS, FEROS, UVES, HIRES, AAT and CORALIE. The files have to be placed in the `Spectra/` directory, and are explained as follow:

- **HARPS**: The HARPS spectra can be retrieved from the Phase 3 data retrieval in the ESO archive. SPECIES uses the s1d fits file, for the A fibre, and consist of the spectra from all the orders already combined. The file has to be called `starname_harps.fits`. It also accepts spectra reduced using the `ceres`[1] pipeline. In those cases, the spectra will be divided by orders, which SPECIES will identify and treat the data accordingly (see the **FEROS** file handling).
- **FEROS**: The FEROS spectra can be retrieved from the Phase 3 data retrieval in the ESO archive, reduced by the ESO pipeline, or can be reduced data from other pipeline. The difference between both files is that, for the case of the data processed with the ESO pipeline, its format is very similar to the HARPS one, that is, all the orders are already combined. The spectra from other pipelines usually comes with each spectral order separated, in which case SPECIES has to combine them to create the 1D spectrum used by ARES. When combining the spectral orders, if two consecutive orders overlap in wavelength, then SPECIES will select the data from the red order (that is because the performance of FEROS improves in the red part of the spectrum, increasing the signal-to-noise of the data). If the input FEROS spectra has the spectral orders already combined, then the file has to be called `starname_feros.fits`. In the case when the spectral orders are separated, then the file has to be called `starname_feros_orders.fits`.
- **UVES**: The UVES spectra can be retrieved from the Phase 3 data retrieval in the ESO archive. Ii has to consist of two files, one for the blue part of the spectrum, and one for the red part. SPECIES will take both files and combine the spectra in order to create the file accepted by ARES. If there is overlap in wavelength between the red and blue parts, then SPECIES will select the data from the highest signal-to-noise (S/N) part. The files have to be called `starname_uves_red.fits` and `starname_uves_blue.fits`.

[1] `https://github.com/rabrahm/ceres`

- **HIRES**: The HIRES spectra can be obtained from the KOA data archive[2]. The spectra has to be already reduced, and without the iodine cell. Before downloading the files, make sure that the spectra is present in most of the spectral orders. It is not necessary to include the calibration files. SPECIES will combine the spectra from the different spectral orders, taken the data with the highest signal to noise (S/N) for the case when the orders overlap. The resulting spectra will be placed in `Spectra/starname_hires.fits`. The folder with the files has to be called `starname_hires/`, and be placed in the `Spectra/` directory.

3.2 Auxiliary files

These files contain additional information for SPECIES, consisting of extra information for the stellar parameters. They are described as follows:

- **file params**: This file contains information about the atmospheric parameters, as well as broadening, macroturbulence and rotational velocities. You can specify initial values for the parameters, and whether you want to keep that value uncharged through the computation of not. The file must have a header with the names of the columns, and each column has to be separated by spaces or tabs. They are as follow:

  1. `Starname`
  2. `hold`: Name (or list of names) of the parameters you wish to hold to a certain value during the computation. The names have to be comma-limited and with no spaces in between. Accepted names are `temperature`, `metallicity`, `pressure`, and `velocity`.
  3. `T`: value for the temperature. If `temperature` is not in hold, then this will be taken as the initial value. If none if given, replace the number with `no`.
  4. `logg`: value for the surface gravity. If `pressure` is not in hold, then this will be taken as the initial value. If none if given, replace the number with `no`.
  5. `met`: value for the metallicity. If `metallicity` is not in hold, then this will be taken as the initial value. If none if given, replace the number with `no`.
  6. `micro`: value for the microturbulence velocity. If `velocity` is not in hold, then this will be taken as the initial value. If none if given, replace the number with `no`.
  7. `vsini`: value for the rotational velocity. If none is given, replace the number with `no`.
  8. `vmac`: value for the macroturbulence velocity. If none is given, replace the number with `no`.

  You can also specify the uncertainty in any of the parameters, by writing them as (value,error). An example of this file is:

```
Starname    hold                        T          logg  met    micro
   vsini       vmac
HD55_harps  temperature,metallicity  4760.1,50  no    -0.72  1.123
   1.38,0.44   2.33,0.54
```

[2] `http://www2.keck.hawaii.edu/koa/public/koa.php`

- **file mass**: File with information regarding the mass, age, and photometric logg for each star. The file must have a header with the names of the columns, and each column has to be separated by spaces or tabs. They are as follow:

  1. `Starname`: `Starname_inst`, with `inst` is the instrument used to obtain the spectra.
  2. `mass`: Value for the mass (in $M_{\odot}$).
  3. `err_mass`: Uncertainty for the mass.
  4. `age`: Value for the age (in Gyr).
  5. `err_age`: Uncertainty for the age.
  6. `logg_p`: Value for the photometric logg.
  7. `err_logg_p`: Uncertainty for the photometric logg.

  For example:

```
Starname     mass   err_mass     age    err_age   logg_p   err_logg_p
HD55_harps   0.63   0.021        4.94   5.5       4.7      0.02
```

- **file with boundaries**: File with the ranges accepted for the atmospheric parameters. The file must have a header with the names of the columns, and each column has to be separated by spaces or tabs. They are as follow:

  1. `Starname`: `Starname_inst`, with `inst` is the instrument used to obtain the spectra.
  2. `temperature`: Minimum and maximum accepted values for the temperature, separated by commas. If none is given, replace it with `no`.
  3. `metallicity`: Minimum and maximum accepted values for the metallicity, separated by commas. If none is given, replace it with `no`.
  4. `gravity`: Minimum and maximum accepted values for the surface gravity, separated by commas. If none is given, replace it with `no`.
  5. `velocity`: Minimum and maximum accepted values for the microturbulence velocity, separated by commas. If none is given, replace it with `no`.

  For example:

```
Starname   temperature   metallicity   gravity   velocity
NG021676   no            no            4.0,5.0   no
```

- **file colors**: File with photometric information for each star. The file must have a header with the names of the magnitudes, and whether they have to be corrected by extinction. Each column has to be separated by spaces or tabs. They are as follow:

  1. `Starname`: Name of the star, without the instrument.
  2. photometric magnitudes: Magnitudes available for each star. If for one magnitude no information is known, replace it with `no`. Magnitudes are: B, V, R, I, J, H, K, Bt, Vt, b and y.

3. `Extinction`: `yes/no`. Whether to perform the extinction correction to the star or not. If `yes`, then specify the star's coordinates and parallax (optional).
4. `RA, DEC`: (*Optional*) Star's coordinates, in degrees. They're necessary for the extinction correction. If not known, replace it with `no`.
5. `Parallax`: (*Optional*) Star's parallax. Even though it is not crucial for the extinction correction, it will greatly improve it, as well as the estimates of mass, age, and radius. If not known, replace it with `no`.

It is important to mention that, even tho the coordinates and parallax columns are optional, if the information is given for one star, then every other star in the file has to have an entry in those columns, which can be replaced with `no` is not known, as explained before.

- **file coords**: File with coordinates for each star, in `deg`. The columns are `Starname`, `RA`, and `DEC`, all separated by spaces or tabs. The file header must include the name of the columns.

Chapter 4

Output files

This chapter explains all the files created by SPECIES, including plots and final results.

4.1 Spectral files

These are the files that involve working with the spectra alone, before computing the equivalent width for the spectral lines.

1. Combined spectra: These are the files created when it is necessary for SPECIES to combine spectral orders into only one spectrum. This is done for the feros, uves, and hires data. For the case of FEROS, the combined spectra will still be called `starname_feros_orders.fits`, and the original file will adopt the name `starname_feros_orders_original.fits`. For UVES and HIRES, the combined spectra will be called `starname_uves.fits` and `starname_hires.fits`, respectively. These files will be stored in the `Spectra/` directory.

2. Restframe spectra: This is the spectra corrected to restframe, and corresponds to the input given to ARES. The velocity correction to apply is computed by performing a cross-correlation function (CCF) between the spectra and a binary mask, consisting on all the lines between 5500 and 6050Å, for the G2 spectral type mask. The files are called `starname_inst_res.fits`, and are stored in the `Spectra/` directory.

3. Plot CCF: These plots show the CCF between the spectra and the binary mask, used to correct the spectra to restframe. The top panel shows three spectral lines (6021.8, 6024.06 and 6027.06 Å) with its position in the original spectra (grey), and in the corrected spectra (black), along with three vertical lines representing the correct restframe position (red). The bottom panel show the corresponding CCF, with its computed velocity. They are stored in `Spectra/plots_ccf/`

4. Plot lines: These plots show four spectral lines (6021.8, 6024.06, 6027.06 and 6562.8 Å) in the restframe-corrected spectrum, along with red lines corresponding to their restframe position. They are stored in `Spectra/plots_fit/`

4.2 Equivalent width files

These are the files produced by ARES, with the equivalent width (EW) information.

1. EW for the iron lines: File with the fit parameters and EW for the iron lines from the file `linelist.dat`. They are called `starname_inst.ares`, and are stored in `EW/`.

2. EW for the atomic abundances: File with the fit parameters and EW for the spectral lines from the file `lines_ab.dat`, which are used to compute the abundance of several elements. They are called `starname_inst_ab.ares`, and are stored in `EW/`.

3. Plot spectra: Plots of the 1D spectra for each star, with red lines showing the position of the lines used to obtain the EW. The inset shows the position of the H-alpha line. They are stored in the directory `EW/plots_spectra/`

4.3 Debugging files

These files describe each step of the parameters computation. They are useful for the cases when there is a problem with the code, and one wants to know in which part of the computation it happened. SPECIES is intended to be completely automatic, therefore it is possible to follow the computation for each star by studying the debugging files. These are stored in `debugging_files/`

4.4 MOOG

These files correspond to the MOOG output, when assuming an atmospheric model with the correct parameters. They correspond to:

1. `output/MOOG_output_files/starname_inst_out.test`

2. `output/MOOG_output_files/starname_inst.test`

3. `output/MOOG_output_files/starname_inst_ab_out.test`

4. `output/MOOG_output_files/starname_inst_ab.test`

4.5 Rotational and macroturbulent velocity

These are plots created during the computation of the rotational and macroturbulent velocity. They are located in `plots_broadening/`.

1. `starname_inst_a_vsini.pdf`: Shows the change of S with the abundance (left panels) and rotational velocity (right panels). Each line represents a different line.

2. `starname_inst_vsini_new.pdf`: The the line fits, created with the final values for $v \sin i$ and v_{mac}. The panels with red background are the lines that were not included in the final values, due to them not representing a good fit to the observed line.

3. `starname_inst_vsini_paper.pdf`: Combination of the two previous files, for the purposes of including it in a publication.

For a better understanding of the quantities being plotted please refer to the project paper.

4.6 Mass and age calculation

These are files produced during the computation of the mass, age, and photometric logg.

1. Chain files: These files contain the chains performed to obtain the mass, age and photometric logg. Before performing new chains, SPECIES looks for these files and uses the already computed information. The files are in
`isochrones/starname_inst_samples.h5`

2. plots CMD + distribution: These plots show the position of the star in the logg vs T_{eff} diagram, using the values already computed by SPECIES. The lines represent isochrones with the same metallicity as our star, but with different masses. The black line is the isochrone corresponding to the mass obtained in SPECIES. The other panels show the distribution of mass, age, and photometric logg obtained. These plots are found in `isochrones/plots/starname_inst_CMD_dist.pdf`

3. plots distribution: These plots show the distribution of mass, age and photometric logg found by SPECIES. The blue lines are the median for each distribution. The correlation between each quantity is also shown. These plots are found in `isochrones/plots/starname_inst_mass_age.pdf`

4.7 Final output

This file contains the final values for the parameters computed by SPECIES. It can be found in the `output/` directory, under the name and extension provided by the user. If the `-columns` option is used, then it will only include the columns selected by the user, plus the name of the star, instrument used to obtain the spectra, and exception. The standard output has the following columns:

Col 1: Star name.

Col 2: Instrument used to obtain the spectra (HARPS, FEROS, HIRES, UVES, CORALIE, AAT).

Col 3: Velocity in $\mathrm{km\,s^{-1}}$, obtained from the CCF, used to correct the spectrum to the restframe.

Cols 4-11: Atmospheric stellar parameters and their corresponding uncertainty (metallicity, temperature, surface gravity, and microturbulent velocity, respectively)

Cols 12-13: Number of FeI and FeII lines used for the computation of the atmospheric parameters, respectively.

Col 14: Exception to the atmospheric parameters. A value of one means the parameters were computed correctly. A value of two means that there were problems in the computation (parameters were repeated more than 200 times, or all of them were outside the permitted ranges), or that the code could not converge to a final value after performing over 1 million iterations.

Cols 15-18: $v \sin i$ and v_{mac} for each star, with their respective uncertainties.

Cols 19-54: Abundances for Na, Mg, Al, Si, Ca, Ti I, Ti II, Cr, Mn, Ni, Cu, and Zn, as well as their uncertainties (standard deviation from the mean) and the number of lines used.

Cols 55-56: Individual abundances for FeI and FeII, respectively.

Cols 57-70: Mass, age, $\log g_{iso}$, radius, and luminosity obtained, along with their 54% and 16% confidence levels.

Col 71: Tells whether the temperature computed from the photometric relations was used as the final temperature or not, for the cases when no convergence was reached in the atmospheric parameters.

Col 72: Tells whether the microturbulent velocity was set to 1.2 km s^{-1}, for the cases when no convergence was reached in the atmospheric parameters.

Col 73: Tells whether the surface gravity was set to be equal to $\log g_{iso}$.

Cols 74-75: Error in the microturbulence and temperature, computed using the alternatives methods described in Soto & Jenkins 2018..

Cols 76-77: Value, uncertainty, and relation used to obtain the temperature from photometry.

The output is fully explained in Soto & Jenkins 2018.

4.8 Temporary files

These are files created and used by SPECIES during the computation of the parameters, but should be deleted once the final results have been found. These are:

1. `MOOG/abfind_starname_inst.par`
2. `MOOG/abfind_starname_inst_2.par`
3. `atm_models/starname_inst_n.atm`
4. `output/starname_inst_out.test`
5. `output/starname_inst.test`
6. `MOOG_linelist/lines.starname_inst.ares`
7. `MOOG/abfind_starname_inst_ab.par`
8. `MOOG/abfind_starname_inst_ab_2.par`
9. `atm_models/starname_inst_ab.atm`
10. `output/starname_inst_ab_out.test`
11. `output/starname_inst_ab.test`
12. `MOOG_linelist/lines.starname_inst_ab.ares`

Chapter 5

Troubleshooting

Here I will try to help you with any problems when running SPECIES, most of them which I've had before.

5.1 The parameters for my star are very off

One reason for this to happen is when SPECIES couldn't converge to a solution for the atmospheric parameters. If this happens, then the `exception` column in the output should be 2. See the next section if this is your case.

Another reason for the values to be off could be because the initial values and boundaries set by SPECIES before attempting to compute the atmospheric parameters are not the correct ones for your star. You could try the following steps:

- In the debugging file, check the values for the magnitudes and parallax (first lines in the file) with what you might know about your star, or what is in Simbad. Then, check with the individual catalogues used by SPECIES to draw the photometric information (listed in the debugging files next to the corresponding magnitudes). If they're not correct, try to input them manually using the `-file_colors` option.
- Check the initial value and boundaries for the temperature. Sometimes, the relations to derive the initial temperature from photometry are not very accurate, especially for M-dwarfs and giant stars. Try running SPECIES again but with the `-no_check_t` or `-giant` options.

5.2 SPECIES didn't converge to a solution

When the `exception` column is 2, then it means there was no convergence in the atmospheric parameters, or an error occurred at some point of the computation. One of the solutions I've seen that helps many times is to use a set value for the microturbulence, as in many cases that is the parameter that won't converge. SPECIES does this when there is no convergence by using the `-vt_hold` option. If that doesn't help, then check the photometric magnitudes for the star in the debugging files, or try manually setting the value for any of the atmospheric parameters using the `-hold_params` and `-file_params` option.

SPECIES has been well tested for dwarf stars, but it might fail for giant and M-dwarfs, so that could also be the reason why SPECIES cannot converge to a solution.

5.3 SPECIES cannot find the star in a certain catalogue of magnitudes

Most of the time this happens because of issues with the connection to the Vizier service. I'd wait a day before re-running SPECIES, or contact the Vizier team at `cds-question@unistra.fr`.

5.4 I cannot install the Marshall et al. 2006 dust maps from the mwdust package

I recently got this problem as well, and it was because, at the time, I couldn't ftp to the location of the maps. I emailed Vizier with this issue and they sent me the following location for the Marshall+2006 maps: `ftp://cdsarc.u-strasbg.fr/pub/cats/J/A+A/453/635/`. My solution would be to change it in the `setup.py` file from the mwdust directory, or to install the package with the `--no-downloads` option, and download the dust maps from another source.

5.5 The computation of the mass and age seems to be stuck

This could happen because the uncertainties in any of the input values are less than 0.01. In the debugging file for your star, look for the line "`Values used for finding mass and age are:`", and "`Photometry used is`", and check the uncertainties.

5.6 I got no valid results for a certain star, or there was an error in the computation

I'd try to compute the parameters for that star only again, especially if it was part of a large sample of stars that was input to SPECIES. Sometimes the computer just doesn't want to run SPECIES for that star. If this doesn't get solved after running SPECIES again, then pay attention to the error message that will appear on the screen, and check the debugging files to see in which part of the computation it got an error.