STiC Bootcamp

A.G.M. Pietrow*

Introduction

A spectroscopic inversion is a technique to retrieve physical conditions (like temperature, velocity, and magnetic fields) from observed spectra by iteratively fitting a model atmosphere until its synthetic spectra match the observations.

Many such codes exist, but most focus on photospheric lines in local thermal equilibrium, as they are much easier to compute. A smaller sub-section can also deal with more complex lines that form in non-LTE, and form under partial reionization (PRD).

These exercises are designed to help students become familiar with the STockholm inversion Code (STiC, de la Cruz Rodríguez & Leenaarts 2019) by covering examples from synthesis and inversion.

It is assumed that the user has the code installed and that they can run the program. If this is not the case, please refer to the Github page for installation instructions.

Before starting, it is recommended that the students to familiarize themselves with the following files:

- `input.cfg`: the main input/output file where file names and *node placement* are specified.
- `keyword.input`: the RH control file. Most of the keywords from RH work, but these will rarely be touched.
- `atoms.input`: active (nNLTE) atoms and passive atoms that are used to compute the background opacity.

*Adapted from summer school document by J. de la Cruz Rodriguez

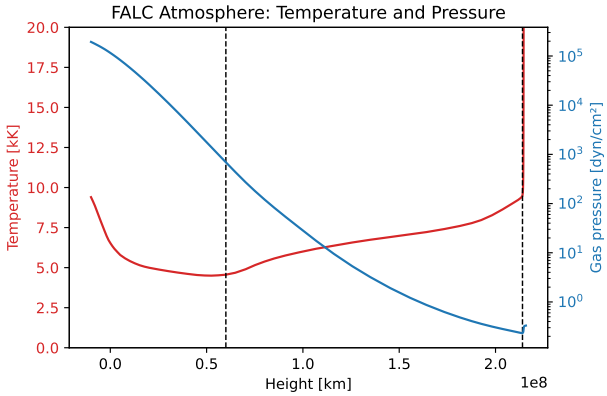

Figure 1: An overview of the FALC atmosphere showing the temperature (red) and pressure (blue) stratification of the model. The photosphere is located between z = 0 Mm and the first dashed line, the chromosphere can be found above that, and the transition region is after the second dashed line.

- `kurucz.input`: here we can specify LTE lines in Kurucz format.

1 Synthesis from a model atmosphere

The idea of this exercise is to perform a synthesis in a single pixel. This will include both LTE and non-LTE lines.

The first step is to create the input model atmosphere. An often-used atmosphere is the FALC atmosphere (Fontenla et al. 1993). This has been made available with this exercise as a text file containing six cgs[1] columns (height, temperature, line of sight velocity, microturbulent velocity, gas pressure, and electron density). In Fig. 1, the temperature and pressure stratification are shown.

1.1 input.cfg

This file is the heart of the program and contains most basic keywords for doing synthesis

[1]STiC works in cgs for all quantities.

and inversions. For this exercise, we are planning to run in *synthesis mode* with all possible types of lines that can be synthesized with STiC; LTE, nLTE CRD, and nLTE PRD.

In this case, the Mg II h&k, Ca II H&K lines, Ca II 8542 Å and Fe I 6301 & 6302 Å.

By default, RH will define a wavelength grid that is specified in the atom files, but this is rarely the wavelength grid that we want. Therefore, the `input.cfg` is used to define regions where the emerging spectra will be computed and saved to an output file.

The standard way of defining a region is as follows:

region format

region = w0, dw, nw, scl, inst, calib

where:
*w*0 is the first wavelength of the region [Å].
dw is the wavelength step [Å].
nw is the number of wavelength points.
scl is a scaling factor. The synthetic spectra are computed in cgs units[2].
inst is the spectral degradation to apply. Possible values: `none`, `spectral`, `fpi`.
calib is a file with the instrumental PSF and other parameters. For `spectral`, it should contain a PSF with an odd number of elements and not more than *nw*.

If more than one region is specified in the input file, the code will concatenate them in the same order as they appear in the input file. Example regions can be found in the config file.

Other important keywords that can influence the synthesis are:

keep_nne, which will preserve the input electron density. When set to 0, the electron density will be recomputed assuming LTE.
recompute_hydro will set the atmosphere to hydrostatic equilibrium. In the latter case, the integration will be started using the provided value of $P_{\rm gas}$ at the top of the atmosphere.

[2]If you have normalized your observations (e.g., to the quiet-Sun continuum), you can apply the same correction here. To work in cgs units just set it to 1.

1.2 atoms.input

The atoms that are marked as active will be treated in nLTE, whereas the passive atoms will be used in the computation of background opacity. If we remove all those passive atoms, you might find that you are lacking opacity in the continuum, especially in the UV. For active atoms, ZERO_RADIATION is usually a better initial solution than LTE.

1.3 keyword.input

The keyword file controls the behavior of STiC at a higher level and is almost identical to the RH keyword file. This is primarily important in inversion mode, where finite-difference-based derivatives of the intensity will be used, which requires the populations to converge further than just for synthesis (e.g., `ITER_LIMIT` = `1.0` $\times$ `10`$^{-3}$).

An important keyword is the NG acceleration, which will speed up the convergence significantly. However, it cannot be applied too early or the problem becomes numerically unstable and won't converge. The same applies for the period. The optimal set of values depends on the intrinsic properties of the atmosphere. We found that with typical setups, these values work, but if you see many errors related to singular matrix, it is better to increase the `NG_DELAY` and `NG_PERIOD` keywords.

The provided file is well commented and will not be explained further here.

1.4 Exercise instructions

For this exercise, please go over the provided python files, which have been commented. Make sure to do the following steps:

1. Create the input model atmosphere using the provided `prepare_input_model.py`. Make sure you understand what the script is doing, as modified versions will be used in the next exercises.

2. Edit `input.cfg` and add the regions that you want to compute. For nLTE lines make sure the proper atom is active in `atoms.input`, for LTE lines use the Kurucz list and place them into `kurucz.input`.

3. Make sure that mode is set to synthesis.

4. Execute the code with at least 2 cores and wait until it finishes. Use e.g. `mpiexec -n 4 ../STiC.x`

5. Plot the result with `plot_all_lines.py` and `plot_region_stokes.py`.

6. Try a few more setups where the magnetic field is changed and everything is rerun. Try the same for different atoms being active and passive.

2 Inversion of a synthetic profile

Several spectra were synthesized in the previous exercise. Now that these profiles are available, we can try to perform an inversion on those profiles. To make the work more realistic, we can also add some noise.

This exercise will show that even when all our assumptions in the radiative transfer and physics are correct, there is only so much information we can extract from our observations.

How is the agreement between the FALC model and the output from the inversion? What could not be recovered? Can you guess why?

Note: Using all lines will give a better solution, but it will also be slow. Consider redoing excercise 1 without the Ca II H&K and Mg II h&k lines, and put the atoms to passive for faster results.

2.1 Exercise instructions

1. First, execute the python script that will read the output profiles from exercise 1 and it will add noise to those profiles, creating a synthetic observation with noise.

2. Run STiC and let the inversion finish.

3. Plot the results with `plot.py` and `plot_lines.py` to see the atmosphere and fit results. Do certain lines fit better than others? What about the stokes parameters?

References

de la Cruz Rodríguez, J. & Leenaarts, J. 2019, A&A, 623, A74

Fontenla, J. M., Avrett, E. H., & Loeser, R. 1993, ApJ, 406, 319