MESA tutorial: session 1

1 What is `MESA`?

`MESA` stands for Modules for Experiments in Stellar Astrophysics. It is an open-source stellar astrophysics code consisting of different modules. Each module is responsible for a different aspect of the numerics or physics required to construct computational models for stellar astrophysics. At the core of it all is the most important module: `MESA star`. `MESA star` is a one-dimensional stellar evolution code and is designed for a wide range of stellar physics applications. You can find more information on the website `https://docs.mesastar.org/`, and also in the first `MESA` instrument paper by Paxton et al. [2011]

The advantages of `MESA` are listed on the website of the code and the key points are given below:

- Open source
- Independent modules
- Wide applicability
- High performance
- Modern techniques
- Comprehensive microphysics

2 Getting started

`MESA` has already been installed on the Science Faculty computer system of Radboud University. This section will explain how to access this. The `MESA` installation is not the newest version, but an older one named `r12778` which is found at `https://zenodo.org/record/3698354#.Yyxmg3ZBxPZ`, because this release runs stably for the computer exercises you will perform during this course. [1]

To access the `MESA` installation, you first need to log in on the Linux environment of the university computers at the Science Faculty. If you cannot manage to log in with your science account, you can find helpful information here: `https://wiki.cncz.science.ru.nl/Login`.

Once logged in, open a terminal and run the following lines of code.

```
1  #This line defines the path where MESA is located
2  export MESA_DIR=/vol/cursus/astro/ASBE/MESA/mesa-r12778
3
4  # Number of cores the computer has for MESA to use
5  export OMP_NUM_THREADS=4
6
7  # path to the MESA SDK
```

---

[1]If you want to install `MESA` on your own personal computer in your own time, follow the instructions on the `MESA` website (`https://docs.mesastar.org/en/stable/installation.html`). The corresponding `MESA` SDK (software development kit) can be downloaded from `http://user.astro.wisc.edu/~townsend/static.php?ref=mesasdk`, making sure that your SDK matches the 'MESA Version at Release' you just downloaded.

```
8  export MESASDK_ROOT=/vol/cursus/astro/ASBE/MESA/mesasdk
9  source $MESASDK_ROOT/bin/mesasdk_init.sh
```

The first line specifies the installation path for `MESA`. The second line sets the number of threads for `MESA` using the `OMP_NUM_THREADS` variable, related to OpenMP for parallel programming. We set it to 4 to match university computers. Use `lscpu` to check your machine's CPU count if unsure. The last two lines define and initialize the `MESA` SDK path.

These lines will **have to be run in every new terminal you open** to make `MESA` work. To avoid having to do this every time, you can edit your Linux shell start-up file (i.e. `~/.bashrc`) and add these lines to that file.[2]

```
1  export MESA_CACHES_DIR=/scratch/$USER/MESA/caches
```

This line defines the path where `MESA` will store its cache information, which in this case will be somewhere in the local `scratch` directory of the specific machine you are using. This directory is used instead of your own `home` directory (invoked with `$HOME`) to make `MESA` run much faster due to it being run locally instead of in the computer system where your `home` directory resides. Furthermore, there is more space available on `scratch` than on your `home` directory. If you are using a machine you have not used before, or the local `scratch` directory has been wiped, or just to be on the safe side, run the following lines of code.

```
1  mkdir /scratch/$USER
2  mkdir /scratch/$USER/MESA
3  mkdir /scratch/$USER/MESA/caches
```

These lines create a personal directory (using the `mkdir` command) named after your science account username (invoked with `$USER`) on the `scratch` directory, a directory named `MESA` inside that directory, and a directory named `caches` where `MESA` will store its cache information as was defined in the previous line with the `MESA_CACHES_DIR` path variable.

The 8 lines of code mentioned above have been put for your convenience in a text file titled `MESA_startup_script`, which is on Brightspace with this document. You can copy the entire contents of that text file and paste it in the terminal (by right clicking the terminal window and selecting paste) for it to run. This will make your terminal window ready to run `MESA`. The possible errors from the `mkdir` commands which say that the file already exists are no problem. You need to run these lines for every new terminal window you want to run `MESA` in.

2.1   Using `MESA`

It is a good idea to create a new work folder for every new `MESA` calculation you do. To do this initially, you copy the model work folder from the `MESA` installation to a new directory with a name of your choice (in this example `project1`) inside your `MESA` directory on the local `scratch` disk. This is done via the recursive copy command invoked via `cp -R`. An example line is given below:

```
1  cp -R $MESA_DIR/star/work /scratch/$USER/MESA/project1
```

For each `MESA` project, create a separate work folder in this way. To run your new `MESA` project, first go to its directory. This is done via the change directory command invoked via `cd`. An example line is given below:

```
1  cd /scratch/$USER/MESA/project1
```

---

[2]More information can be found in the subsection `Set your environment variables` on `https://docs.mesastar.org/en/release-r22.05.1/installation.html#`.

Inside this directory, you first need to assemble the code because `MESA` uses a compiled programming language called `Fortran`. This is done via a make command using the following line of code:

```
./mk
```

Afterwards, the code can be run using the following line of code:

```
./rn
```

To make changes to the `MESA` model, you will need to edit the inlist files that live inside the project directory (e.g. `inlist_project` and `inlist_pgstar`). This can be done via the file manager by accessing your work folders on the local `scratch` directory underneath the following chain: `Other Locations`, `On This Computer`, `Computer`, `scratch`. It can also be done via the terminal using the `xdg-open` command which opens a file with the machine's preferred application. An example line (in this case to open `inlist_project`) is given below:

```
xdg-open inlist_project &
```

The `&` symbol at the end allows you to keep the editor window open while typing further Linux commands in the terminal.

For more on how to use `MESA`, you can find information here: `https://docs.mesastar.org/en/release-r22.05.1/using_mesa.html`. Furthermore, more information on simulating different stellar evolution problems in `MESA` can be found here: `https://docs.mesastar.org/en/release-r22.05.1/test_suite.html#`.

2.2 Saving your work

After you have finished using `MESA` for the current session, you will need to migrate your project directories from the local `scratch` directory to prevent you from losing them. This can be done using the file manager and copying directories from `scratch` to your `home` directory, or via the terminal using the following example lines of code, which create a `MESA` directory in your `home` directory and copies your project there:

```
mkdir $HOME/MESA
cp -R /scratch/$USER/MESA/project1 $HOME/MESA/project1
```

Note that you will have to make a copy for *every* project directory you have worked on during the session! It is also important to note that you only have limited space in your `home` directory (about 7 Gb). Thus, it will be useful to migrate your `MESA` projects to a personal external storage device or to cloud storage (e.g. the OneDrive that comes with your Radboud e-mail, or a Google Drive).

3 Evolving a 1 M$_\odot$ star

1. Create a copy of the work folder and give it a suitable name, then compile the code. For example:

```
cp -R $MESA_DIR/star/work /scratch/$USER/MESA/session1
cd /scratch/$USER/MESA/session1
./mk
```

2. Change the `inlist_project` file to evolve a 1 M$_\odot$ model. Do this by changing the `initial_mass` value in `inlist_project`:

```
initial_mass = 1 ! in Msun units
```

With the default settings, the model will start on the pre-main sequence and will stop when it reaches the zero-age main sequence (ZAMS), where hydrogen burning starts. Let's evolve the model a bit further than that, to the current age of the Sun: 4.6 Gyrs. Edit `inlist_project` and change the stopping conditions. First set:

```
stop_near_zams = .false.
```

and add the lines:

```
! stop when the age of the Sun is reached
max_age = 4.6d9
```

Tip: to be able to see the plots after the run finishes, also add these lines to `inlist_project` in the `&star_job` section:

```
! stop before terminating such that you can see the plots
pause_before_terminate = .true.
```

Note: when adding new lines to your inlist files, make a habit out of also adding a comment (by using the !-sign) to explain what the parameter setting does. This will help you in the future.

(a) Now compile again (`./mk`) and run the model. First concentrate on the HR plot. You will notice the model (red circle) starts at the bottom of the plot and quickly disappears out of sight. To remedy this, change the limits of the plot window by editing the file `inlist_pgstar`. Good values for this model are $\log L$ between -1 and 3, and $\log T$ between 3.5 and 4. Save the changes to `inlist_pgstar` and you should see the changes taking effect immediately! Of course, if the model has already finished, simply run it again.

Tip: if you want to temporarily pause your model to have more time to look at a plot, you can suspend the job by pressing Ctrl + Z. To resume the model, type `fg` in the prompt. If you want to terminate your model, press Ctrl + C. If you want to terminate a paused model, first type `fg` in the terminal and then press Ctrl + C; this prevents the uncloseable `MESA` plot windows from staying open and piling up.

(b) After adjusting the HR plot, try to understand what happens. Can you explain the movement (i.e. the evolution history) of the star in the HR diagram? Is the star expanding or contracting? And how do you explain this behaviour? (Note: ignore the first horizontal part in the HR diagram where the star is still relaxing into hydrostatic equilibrium.)

(c) The `TRho_Profile` plot shows the entire stellar model, from centre to surface, as it evolves. Can you identify the centre and the surface? What is the meaning of the other (dashed) lines in the plot? Does the ideal gas law hold everywhere in the star?

(d) Colours indicate in which regions of the star convection (or some other mixing processes) occurs, and where nuclear energy is produced. Initially, almost the entire model is light blue, i.e. convective. Can you explain why? And how is this related to the star's location in the HR diagram?

(e) At later times, the central portion of the star becomes green, i.e. radiative. At the same time the effective temperature in the HR diagram increases. Can you explain this? Consult section 9.1 in the lecture notes to answer these questions, if necessary.

(f) When the model stops after 4.6 Gyrs, the star is undergoing hydrogen fusion in its core. How can you identify this from the plots? Also compare the location in the HR diagram to the current parameters of the Sun. Do they agree?

(g) You will have noticed it takes several hundreds of steps to reach the main sequence (start of hydrogen fusion), while subsequently it takes much fewer steps to reach the current age of the Sun. What would you have expected if the number of steps is proportional to the actual time the star takes to evolve? (i.e. compare to the appropriate stellar time scales.)

The reason is that the structure of the star changes rapidly when the star contracts to the main sequence, so many small time steps are needed for MESA to resolve these changes. Once the star is on the main sequence, its structure hardly changes, so the code can take much longer time steps to evolve the star through the main sequence phase.

3. Now let's evolve the model further to the red giant branch phase. Remove the stopping condition for the age in `inlist_project` and instead add the lines:

```
1  ! stop when helium core reaches this limit
2  he_core_mass_limit = 0.25
```

Do not forget to remove (or comment out) any other stopping condition that we do not need. We can also examine some other PGSTAR plots. Edit `inlist_pgstar` and add:

```
1  Summary_Burn_win_flag = .true.
2  Abundance_win_flag = .true.
```

Now restart the model from a saved 'photo' from the previous run (type `./re x***`, where the `***` are the final three digits of the previous model; you can check the exact three digits of the last photo in the `photos` directory inside your MESA model).

(a) Take some time to understand what is plotted in these panels, and identify each line in the plots. Note that the vertical scales are logarithmic in both panels, and that they represent many orders of magnitude, especially in `Summary_Burn`. You can correlate the T and ρ profiles in `Summary_Burn` with the curve plotted in the `TRho_profile` panel. Note that most of the change in density and temperature occurs in the outer few percent of mass of the star. N.B. the colours along the bottom of these panels correspond to those plotted in `TRho_profile`, and can be used to identify convective regions.

(b) As the star evolves through the main sequence, the abundance profiles change. Which processes cause these abundance changes? Consult/refresh chapter 6.4 of the lecture notes if necessary. Can you explain all the changes from the nuclear reactions taking place? (Take into account the logarithmic scale: the changes in the bottom part of the plot look dramatic but only involve small values of the mass fraction.)

(c) At what age (approximately) does the star reach the end of the main sequence? How can you tell from the PGSTAR panels? (Several plots show an indicator of this!)

(d) Observe the changes in the T and ρ profiles as the star moves from the main sequence to the giant branch. Can you explain these changes? Does the core become degenerate, and if so, in which point in the evolution? Can you explain why the temperature profile in the core becomes flat?

(e) As the star evolves along the red giant branch, convection reaches deeper and deeper in the envelope of the star. Note how this affects the abundance profiles. Which elements/isotopes are most affected? Do their surface abundances increase or decrease? This process of convection changing the surface abundances is called 'dredge-up'.

As the helium core mass increases, notice how the burning shell becomes thinner in mass. This requires MESA to take smaller time steps, so that it will take thousands of steps to

reach the end of the RGB. The observable changes between each time step become smaller and smaller. That is why we stop the model at a core mass of 0.25 M$_\odot$.

4. Many other details of the models can be shown using `PGSTAR`. Read the `MESA` website page `Using PGSTAR` (https://docs.mesastar.org/en/release-r22.05.1/using_mesa/using_pgstar.html#) to get an idea of the possibilities, and experiment with different options. Simply run the same model again, or try a different stellar mass. (In the latter case, create a new work folder for your stellar models, as in step 1 above.)

5. *Tip:* In order to be able to easily inspect the plots created in real-time by the `PGSTAR` module *after* the end of the simulation you can opt to save every N'th plot to a file in a separate directory. This can be done by adding several controls to the `inlist_pgstar` file, for example:

```
1  HR_file_flag = .true.
2  HR_file_dir =  'png'
3  HR_file_prefix = 'hr_'
4  HR_file_interval = 5
5  HR_file_width = 16
6  HR_file_aspect_ratio = 1
```

This will save a HR diagram every 5 steps into a directory `png` giving it a name starting with `hr_`. You can then combine a series of png images into a short movie using ffmpeg, for instance by executing the following in terminal when in the `png` directory (do remove the spaces between the 3 lines to make it 1 line in the terminal):

```
1  ffmpeg -f image2 -pattern_type glob -framerate 5 -i "*.png" -filter
       :v
2  scale=1680*1080 -preset:v slow -pix_fmt yuv420p -c:v libx264 -b:v 4
       M -f
3  mp4 movie.mp4
```

Note that if you want to have several graphs in one png output file (e.g. not only the HR diagram but also `TRho_profile` or a Kippenhahn plot) then you should make use of the `Grid_*` controls of the `PGSTAR` module.

References

Bill Paxton, Lars Bildsten, Aaron Dotter, Falk Herwig, Pierre Lesaffre, and Frank Timmes. Modules for Experiments in Stellar Astrophysics (MESA). *The Astrophysical Journal Supplement Series*, 192(1):3, January 2011. doi: 10.1088/0067-0049/192/1/3.