Stellar Mixing: Day 2

Pascale Garaud and Chris Mankovich

MESA Summer School, 8/13/2014

Minilab 1: Rotation Profile, ZAMS to RGB

- Download and untar `garaud_day2.tar.gz` from the Teaching Materials section of mesastar.org. This is just a single work directory. The master `inlist` includes `inlist_project` and `inlist_pgstar`.
- You'll run a 1.0 $M_\odot$ model from ZAMS, initializing rotation with a roughly solar rotation rate. To accomplish this, add in `&star_job`:

```
change_initial_rotation_flag = .true. ! change state of rotation flag on run start
new_rotation_flag = .true. ! new state = rotation ON

set_initial_surface_rotation_v = .true. ! set solid-body rotation using surface v
new_surface_rotation_v = 2 ! specify surface v, in km/s
```

- Include RGB mass loss via the Reimers prescription, with $\eta = 0.2$ (find the appropriate controls in `controls.defaults`).
- Copy `star/defaults/profile_columns.list` to your work directory and uncomment `log_omega`, `log_brunt_N`, and `am_log_nu_omega`.
- Copy `star/defaults/history_columns.list` to your work directory and uncomment `surf_avg_omega` and `center_omega`. Finally add these lines so that your Kippenhahn plot knows about mixing and burning regions:

```
mixing_regions 20
burning_regions 20
```

- Set $R/R_\odot = 6$ as your stopping criterion. (Hint: look for `photosphere_r` under "when to stop" in `controls.defaults`.)

Part One

Neglect any composition mixing except convection, and neglect any angular momentum transport except that by viscosity (this is the default—change nothing). **Can you identify what contributes to angular momentum transport throughout the star? How does your final core rotation rate compare to observations (cf. the Mosser plot)?**

Part Two

Add a constant baseline diffusion coefficient for angular momentum via

```
set_min_am_nu_non_rot = .true.
min_am_nu_non_rot = 1d0
```

Pick a random value between 1 and 6, and take 10**that to determine your value for `min_am_nu_non_rot`.

In both cases, what is your final core rotation rate? How does it compare to the typical value of $\Omega_c = 10^{-5}$ s^{-1} for mid-RGB inferred from the Mosser plot? In the second case, does your constant diffusivity provide sufficient coupling to slow the core down to observed rotation rates? At the end of part two, enter your `min_am_nu_non_rot`, final core omega, and final surface omega in the google sheet.

Minilab 2: A Physical Mechanism for Coupling?

- Take out the constant `am_nu`, but enable angular momentum transport via the Dynamical Shear Instability (DSI):

  ```
  D_DSI_factor = 1d0 ! 0 by default.
  am_nu_DSI_factor = 1d0 ! equals D_DSI_factor by default.
  ```

  For `am_nu_DSI_factor`, pick a value evenly spaced in `log10` between 10^{-1} and 10^4, determined by your -favorite- random number.

- Run again to 6 $R_\odot$. **Does DSI do the trick?**

- Uncomment `richardson_number` in your `profile_columns.list`, and modify `inlist_pgstar` to plot this column by changing the last block of controls to read

  ```
  Profile_Panels2_yaxis_name(1) = 'richardson_number' ! was 'am_log_nu_omega' previously
  Profile_Panels2_ymin(1) = 0
  Profile_Panels2_ymax(1) = 3d5 ! static axis limits for readability
  ```

- Run it again and look at the Richardson number profile, adjusting the ymax for that plot as you go so you can get an idea what the minimum value of Ri is in the shear region. **What do you conclude about the main shear region between pure He core and H/He envelope on the RGB?**

Long Lab: Shear-Driven Instabilities

Part One

- Pick your favorite angular momentum transport mechanism among Eddington-Sweet, Solberg-Hoiland, Goldreich-Schubert-Fricke, and Spruit-Tayler, and enable it via e.g.

  ```
  D_ES_factor = 1d0
  am_nu_ES_factor = XXXX ! or similarly for SH, GSF, ST in place of ES
  ```

  where `XXXX` is your favorite boost factor between 1 and 10^6.

- In `inlist_pgstar`, change `richardson_number` back to `am_log_nu_omega` and get rid of the static axis limits.

- Run again to 6 $R_\odot$ and watch what happens. **Is the core-envelope coupling enough to reproduce observed core rotation rates? Send your last saved `LOGS/profileX.data` file to cmankovich@ucsc.edu, and tell me which instability you included and with which efficiency factor.**

Part Two

Turn off all the transport mechanisms you enabled in the last part, and use the `other_am_mixing` hook to implement a simple shear-driven angular momentum transport mechanism. As usual, `star/other/other_am_mixing.f` includes a skeleton routine you can add to `run_star_extras.f` and some instructive comments (I mean it this time!).

To review, you need to

1. add `s% other_am_mixing => my_other_am_mixing` in `extras_controls`

2. turn on `use_other_am_mixing` in your inlist

3. write your routine `my_other_am_mixing`!

Implement a diffusion coefficient for angular momentum with the general form

$$\texttt{am\_nu\_extra} = C \times \frac{\kappa_T}{\mathrm{Ri}^\alpha} \tag{1}$$

for nondimensional parameters C and α. If you like, you can use the `s% x_ctrl` array to have these values set at the inlist level. Analagous to the `other_D_mix` hook, `other_am_mixing` works by just adding your new diffusivity to the existing `s% am_nu_omega` array calculated by `MESA star`. The included `run_star_extras.f` again includes the routine `get_diff_coeffs` we used in yesterday's lab, for the sake of easy calculation of the thermal diffusivity κ_T. The Richardson number is defined throughout the model and is available to you as well—where should you look?

You should end up with a loop over all zones that looks roughly like (for integer `k`):

```
do k=1, s% nz ! loop over all zones from surface to center
     ...
     [[code to calculate am_nu_extra as a function of local quantities]]
     ...

     ! add onto s% am_nu_extra as calculated in star/private/mix_info.f
     s% am_nu_omega(k) = s% am_nu_omega(k) + am_nu_extra
end do
```

In certain cases (the surface zone in particular) the Richardson number evaluated by `MESA star` can be zero, so to avoid division by zero or a very small number we suggest that you include a check to skip over tiny ($\ll 1$) Richardson numbers. It's probably also best to skip over convective zones, since the nearly rigid rotation in those regions results in a nearly zero Richardson number, and the effect of shear likely doesn't add to the diffusivity from convection.

Does your mechanism result in substantial angular momentum transport through the shear region? For what values of C and α do you succeed in slowing the core down to observed rotation rates? Any numerical problems?

Part Three (if you're looking for something to do. . .)

Apply the same diffusivity you just calculated to the mixing of *species*. You can basically copy the subroutine you just wrote and implement it as an `other_D_mix` hook. You would change the final line inside the loop in `my_chemical_mixing` so that instead of adding to `s% am_nu_omega(k)`, you are adding the new diffusivity to `s% D_mix(k)`.

Does it work?