Modelling stellar energy transport

Computational Astrophysics, UvA, 2023/2024

Adapted from the course AST3310 by B. V. Gudiksen, Institute for Theoretical Astrophyics, University of Oslo

May 7, 2024

1 Introduction

This project involves modelling the central parts of a Sun-like star, including both radiative and convective energy transport. In this project we assume the star is spherically symmetric, and we therefore assume that all the quantities are one-dimensional. The goal is to get a one-dimensional profile of the internal structure of the star, including the pressure, density, temperature, and method of energy transport. The governing equations for solving the internal structure of the radiative zone of the Sun are as follows

$$\frac{\partial r}{\partial m} = \frac{1}{4\pi r^2 \rho} \tag{1}$$

$$\frac{\partial P}{\partial m} - \frac{Gm}{4\pi r^4} \tag{2}$$

$$\frac{\partial L}{\partial m} = \varepsilon \tag{3}$$

$$\frac{\partial T}{\partial m} = \begin{cases} \nabla^* \frac{T}{P}\frac{\partial P}{\partial m} & \text{if } \nabla_{\text{stable}} < \nabla_{\text{ad}} \\ -\frac{3\kappa L}{256\pi^2 \sigma r^4 T^3} & \text{otherwise} \end{cases} \tag{4}$$

$$P = \frac{4\sigma}{3c}T^4 + \frac{\rho k_B}{\mu m_u}T \tag{5}$$

$$\rho = \left(P - \frac{4\sigma T^4}{3c}\right)\frac{\mu m_u}{k_B T} \tag{6}$$

Here we have:

- The variable of integration m is known as the mass coordinate, and is the mass at spherical shells throughout the star.
- r is the radial coordinate from the core to the surface of the star.
- ρ is the density.
- P is the pressure.
- L is the luminosity.
- ε is the total energy released from the fusion reactions in the core of the star.
- T is the temperature.
- ∇^* is the temperature gradient for convective energy transport.

- μ is the mean atomic weight, defined as $(2X + 3Y/4 + Z/2)^{-1}$, where X, Y, and Z are the fractions of hydrogen, helium, and heavier elements respectively.
- κ is the opacity
- k_B, σ, and c are the Boltzmann constant, Stefan-Boltzmann constant, and speed of light respectively.
- m_u is the atomic mass unit.

Below are the assumptions, goals, initial parameters, tasks and sanity checks, as well as instructions on the code. Read the description thoroughly before you start coding.

2 Assumptions

- Ideal gas.
- Only radiation and convection are used to transport heat (no conduction).
- $X = 0.7$, $Y = 0.29 + 10^{-10}$, $Z = 1 - (X + Y)$

3 The temperature gradient

The temperature gradient ∇ is required to determine whether a specific zone in the star is radiative or convective, which again determines the equation you use in order to model the transport of heat $\partial T/\partial m$. At each step, you need to check whether there is radiative or convective transport. To do this, you need to calculate the stable temperature gradient ∇_{stable}, which is given by

$$\nabla_{\text{stable}} = \frac{3\kappa H_p L}{64\pi r^2 \sigma T^4}, \tag{7}$$

where H_p is the scale height, defined as

$$H_p = \frac{k_B T}{\mu m_u g}, \tag{8}$$

where g is the gravitational acceleration at a given mass shell:

$$g = G\frac{m(r)}{r^2}. \tag{9}$$

Convective transport takes place if the stable temperature gradient ∇_{stable} smaller than the adiabatic temperature gradient ∇_{ad}, which, with the assumption of an ideal gas, is given by

$$\nabla_{\text{ad}} = \frac{P}{T\rho C_p}, \tag{10}$$

where C_p is heat capacity as constant pressure, defined as

$$C_p = \frac{5k_B}{2\mu m_u} \tag{11}$$

If a given mass shell is convectively stable, the temperature gradient ∇ is equal to ∇_{stable}. Otherwise, you need to calculate the convective temperature gradient ∇^*. To do this, you need to find the roots of the following equation:

$$\xi^3 + \frac{U}{l_m^2}\xi^2 + \frac{U^2\Omega}{l_m^3} + \frac{U}{l_m^2}(\nabla_{\text{ad}} - \nabla_{\text{stable}}) = 0 \tag{12}$$

using

$$U = \tfrac{64\sigma T^3}{3\kappa\rho^2 C_p}\sqrt{\tfrac{H_p}{g}}$$

$$l_m = H_p$$

$$\Omega = 4/l_m$$

This is a third-degree polynomial, so it must have at least one real root. Once you've found this, you can calculate ∇^* using

$$\nabla^* = \xi^2 = \frac{U\Omega}{l_m}\xi + \nabla_{\mathrm{ad}}. \tag{13}$$

4 Goals

Your code should produce a star that has the following properties:

- Has L, m, and r all going to 0, or at least within 5% of L_0, M_0, r_0
- Has a core ($L < 0.995L_0$) reaching out to at least 10 % of r_0.
- Has a continuous convection zone near the surface of the star. The width of this convection zone should be at least 15 % of r_0. A small radiation zone at the edge and/or a second convection zone closer to the centre is acceptable, but the convective flux should be small compared to the "main" convection zone near the surface.

5 Initial parameters

Begin with actual values of the solar surface (apart from P, which will differ since we are assuming an idea gas). $\bar{\rho}_\odot$ is the *average* density of the Sun and is given by $\bar{\rho}_\odot = 1.408 \times 10^3$ km m^{-3}. The other initial parameters are the following:

$$L_0 = 1L_\odot$$

$$R_0 = 1R_\odot$$

$$M_0 = 1M_\odot$$

$$\rho_0 = 1.42 \times 10^{-7} \cdot \bar{\rho}_\odot$$

$$T_0 = 5770 \text{ K}$$

6 Tasks

1. Create a method that reads the file opacity.txt[1] and takes T and ρ as input and returns κ. The input and output parameters must be given in SI units. You will need to use linear 2D interpolation for the common case where the input value is not exactly found in the opacity table. Your code should also be able to extrapolate if you are outside the bounds of the table (have your program output a warning when you do so). Test your method against the first sanity check below. The structure of the file is as follows:
   - The top row $\log_{10}(R)$, with $R \equiv \frac{\rho}{(T/10^6)^3}$, and ρ in units [g cm^{-3}]
   - The first column is $\log_{10}(T)$, with T being in units of K
   - The rest of the table is $\log_{10}(\kappa)$ in units of [cm^2/g]
2. Implement a similar method but for epsilon.txt. The structure is the same as the opacity table.
3. Implement methods to calculate $\rho(P,T)$ and $P(\rho,T)$.

[1]Taken from Asplund, M.;, Grevesse, N. and Sauval, A. J.; Astronomical Society of the Pacific, 2005., p.25

4. Solve the four partial differential equations numerically. You have to write your own ODE solver, but you are free to chose the method (Euler, Runge-Kutta, Simpson's method.), with the inclusion of adaptive timestepping. For the interpolation and root finding, you are free to use methods from other packages, such as scipy.

5. Include a check for convective stability at each mass shell: $\nabla_{\text{stable}} > \nabla_{ad}$ to ensure you are using the correct equation for the temperature gradient (Equation 4).

6. Ensure your code is working correctly by doing the sanity checks.

7 Sanity checks

The following sanity checks need to be implemented in your code with the option to turn them on and off.

7.1 Interpolating tables

Check that the interpolation of **opacity** values give results within 5% of the following values:

$\log_{10}(T)$	$\log_{10}(R)$ [cgs]	$\log_{10}(\kappa)$ [cgs]	$\kappa \times 10^3$ [SI]
3.750	-6.00	-1.55	2.84
3.755	-5.95	-1.51	3.11
3.755	-5.80	-1.57	2.68
3.755	-5.70	-1.61	2.46
3.755	-5.55	-1.67	2.12
3.770	-5.95	-1.33	4.70
3.780	-5.95	-1.20	6.25
3.795	-5.95	-1.02	9.45
3.770	-5.80	-1.39	4.05
3.775	-5.75	-1.35	4.43
3.780	-5.70	-1.31	4.94
3.795	-5.55	-1.16	6.89
3.800	-5.50	-1.11	7.69

Check that the interpolation of **epsilon** values give results within 5% of the following values:

$\log_{10}(T)$	$\log_{10}(R)$ [cgs]	$\log_{10}(\varepsilon)$ [cgs]	$\varepsilon \times 10^{92}$ [SI]
3.750	-6.00	-87.995	1.012
3.755	-5.95	-87.623	2.415

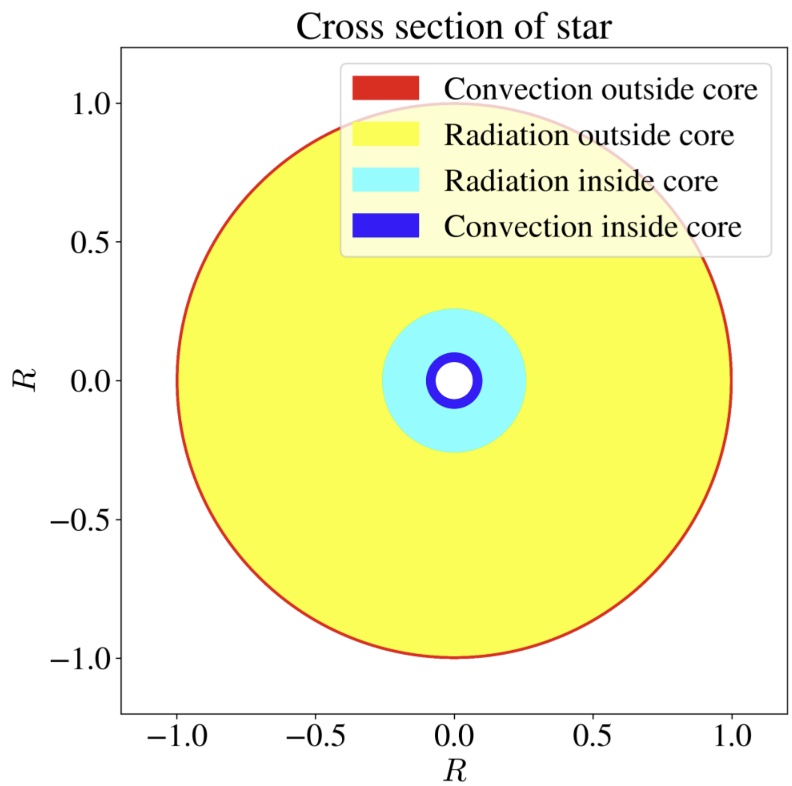
The expected values above, the values calculated by your code, and the relative errors between them should be printed when running the code. If the code does not pass the sanity check, it should print a warning!

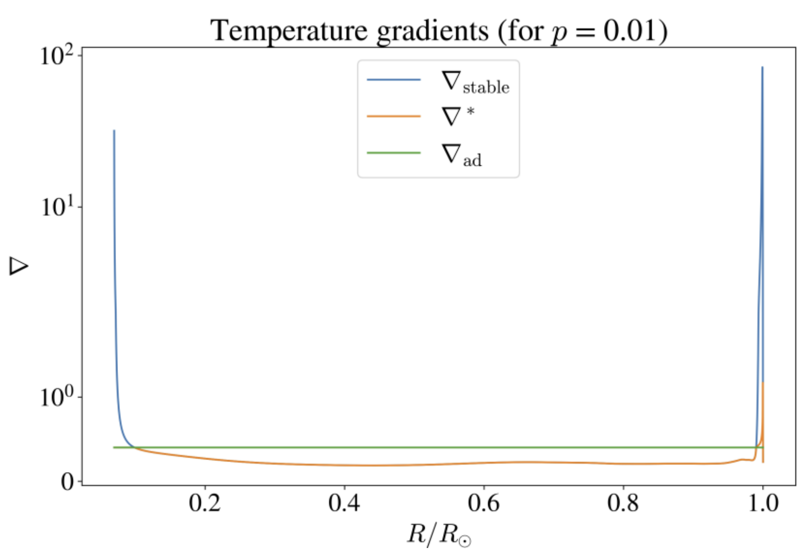
7.2 Model verification

Make a plot of the cross-section of your star, with coloured circles indicating whether the zone has convection of radiation:

- Red: Outside core ($L \geq 0.995 L_\odot$) and convection ($F_C > 0$)
- Yellow: Outside core ($L \geq 0.995 L_\odot$) and radiation ($F_C = 0$)
- Cyan: Inside core ($L \leq 0.995 L_\odot$) and radiation ($F_C = 0$)
- Blue: Inside core ($L \leq 0.995 L_\odot$) and convection ($F_C > 0$)

Your plot should look something like figure 7.2. With the given initial conditions, you should get a temperature gradient plot looking like Figure (7.2).


Cross section of star
Convection outside core
Radiation outside core
Radiation inside core
Convection inside core
1.0
0.5
0.0
−0.5
−1.0
R
−1.0
−0.5
0.0
0.5
1.0
R


Temperature gradients (for $p = 0.01$)
∇_{stable}
∇^*
∇_{ad}
10^2
10^1
10^0
0
∇
0.2
0.4
0.6
0.8
1.0
$R/R_\odot$

8 Code

The code has to be written in any programming language (preferably something the coordinators are familiar with). The code should be easy to read, well commented and logically structured. The instructors should be able to run your code.

9 Finding the best model

You should try to find the best model that satisfies all the criteria specified in **Goals**. To do this, try varying r_0, T_0, ρ_0, P_0 and T_0. You might have to change ρ_0 and P_0 by up to several orders of magnitude. The other four parameters should not be changed by more than a factor 5. For your best fit model, make plots of the main parameters. Make a cross-section figure of your best star as in Figure 7.2.

References

This project is adapted from the course AST3310 by B. V. Gudiksen as the Institute for Theoretical Astrophysics at the University of Oslo. If you want to learn more about the theory behind this project, there are lecture notes available upon request.