Draft version March 2, 2019
Preprint typeset using LaTeX style emulateapj v. 12/16/11

AST3310 PROJECT NR. 1: MODELLING A STELLAR CORE AND RADIATION ZONE

Nils-Ole Stutzer

INTRODUCTION

In order to understand the nature of stars, it is essential to understand how the energy produced in the stellar core propagates through the star's layers. The first step in researching the energy transport is to understand how the energy is transported by means of radiation through the extremely dense and hot inner part of the star, the so-called radiation zone. This part of the star is so dense and hot that it may take the energy produced in the stellar core many thousand years to travel all the way to the stellar surface, as the mean free path of a gamma photon is very small.

In this report we will focus on how we made a model for the stellar radiation zone using numerical simulations in Python. The final aim being to understand how different initial conditions at the surface of the stellar convection zone affect the stellar core and the radiation zone, so as to find initial conditions that produce a stable stellar core and radiation zone.

METHODE

To make a model of the stellar radiation zone we need to make use of the laws of physics that govern the stellar interior. But because this is quite a complex system of coupled processes we need to make some assumptions in order to obtain a solvable task. First, we assume that the radiation zone of the star is made out of thin shells, with approximate constant density. Next, we can assume that the star is in hydrostatic equilibrium. The third assumption is that all energy produced by the star comes from the fusion processes in its core. Since we are only interested in the energy transport through radiation we neglect convection. Thus, the energy only diffuses through radiation. These assumptions result in the coupled system of non-linear partial differential equations (PDEs)

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

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

$$\frac{\partial L}{\partial m} = \epsilon = \sum Q_{ik} r_{ik} \tag{3}$$

$$\frac{\partial T}{\partial m} = -\frac{3\kappa L}{256\pi^2 \sigma r^4 T^3}, \tag{4}$$

where the first equation comes from the first assumption, the second comes from the second assumption etc. Here ρ, m, L and ϵ denote the density, mass, luminosity and energy produced respectively, within a shell of radius r and temperature T. Q_{ik} and r_{ik} are the energy produced and reaction rate of the ik-th fusion reaction of the PP-chains (We do not consider high enough temperatures for CNO-cycle to be important). These where found by using the energy_production class implemented as described in Appendix C in the lecture notes of the course. The reason the free variable is chosen to be m, and not r, is that the differential equations behave more stable if m is chosen as the free variable. In addition to the four coupled differential equations we assume that we have an ideal gas

$$P_G = \frac{\rho}{\mu m_u} k_B T, \tag{5}$$

where μ is the mean molecular weight of the sun, m_u is an atomic mass unit in kg and k_B is the Boltzmann constant. The expression for gas pressure is only used to find the initial pressure so as to initialize the PDE in eq. (2). The expression for the gas pressure is also used to find the density used in the PDEs. In addition we also have the radiation pressure $P_R = \frac{4\sigma}{3c} T^4$, where σ is the Stefan-Boltzmann constant and c is the speed of light. This gives an expression for the total pressure used to initialize the differential equation (2)

$$P = P_G + P_R. \tag{6}$$

The opacity in eq. (4) is a function of both the density ρ and temperature T. This function is found by interpolating the data provided in the file `opacity.txt`.

Next, we need to calculate the mean molecular weight μ used to calculate the density and the initial pressure. For this we assume that all the hydrogen and helium is fully ionized. Thus, each hydrogen atom consists of one nucleus and one electron, and each helium atom consists of one nucleus and two electrons. They therefore contribute with two and three particles per nucleon respectively. We also assume that all the metals are fully ionized, meaning that Lithium and Beryllium contribute with four and five particles per nucleus each. Thus, the mean molecular weight

$$\mu = \frac{1}{\frac{2}{1}X_{\,1}^{\,1}\mathrm{H} + \frac{3}{3}Y_{\,2}^{\,3}\mathrm{He} + \frac{3}{4}Y_{\,2}^{\,4}\mathrm{He} + \frac{4}{7}Z_{\,3}^{\,7}\mathrm{Li} + \frac{5}{7}Z_{\,4}^{\,7}\mathrm{Be} + \frac{2}{2}Z} \tag{7}$$

$$= \frac{1}{2X_{\,1}^{\,1}\mathrm{H} + Y_{\,2}^{\,3}\mathrm{He} + \frac{3}{4}Y_{\,2}^{\,4}\mathrm{He} + \frac{4}{7}Z_{\,3}^{\,7}\mathrm{Li} + \frac{5}{7}Z_{\,4}^{\,7}\mathrm{Be} + Z} \tag{8}$$

$$\approx 0.614, \tag{9}$$

where the other metals Z contribute with two particles and are assumes to have an average mass of $2u$. The mass fractions $X_{\,1}^{\,1}\mathrm{H} = 0.7$, $Y_{\,2}^{\,3}\mathrm{He} = 10^{-10}$, $Y_{\,2}^{\,4}\mathrm{He} = 0.29$, $Z = 0.01$, $X_{\,3}^{\,7}\mathrm{Li} = 10^{-13}$ and $X_{\,4}^{\,7}\mathrm{Be} = 10^{-13}$. Also, it is important to note that it is not all to important to which degree the metals are ionized, because they are so rare, thus just assuming they are fully ionized is not a bad approximation.

It is now time to take a look at how to solve the differential equations. In theory one can use any arbi-

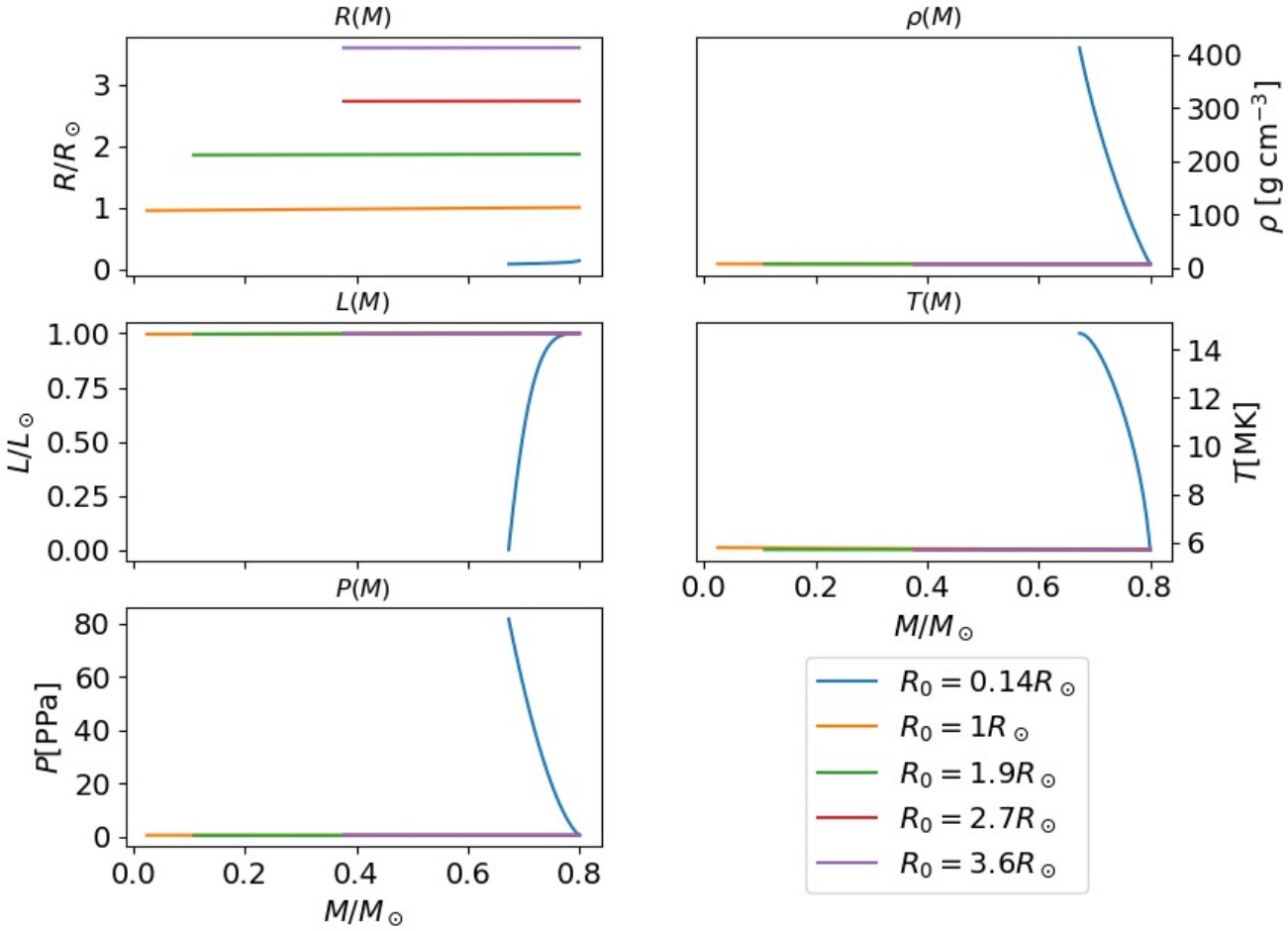

FIG. 1.— The plots show the radial distance R, density ρ, luminosity L, temperature T and pressure P for different initial radii. The colour code indicates the corresponding initial conditions.

trary integration algorithm, but we choose to use the Runge-Kutta order four (RK4) algorithm because it is a higher order method with superior precision to lower order methods like Euler's method. Meaning that the same choice of ∂m using RK4 will be quite more stable then when using Euler's method.

Further we implemented variable step length which could be turned on or off by a Boolean input argument. When using a variable step length, the local relative error is held below a given tolerance. This is done by letting

$$\partial m = -\frac{pV}{|f|}, \tag{10}$$

where p is the predetermined tolerance, V is the current value of the function to integrate and f is its derivative at the current value. Because we have several differential equations, we need to calculate ∂m for all of them, and then we use the one that is the smallest so as to ensure the best accuracy. Therefore, the precision of the RK4 algorithm will then be even further improved. Also, the integration may require less iterations, because the program chooses when it is reasonable to use big or small steps in the integration. The numerical solver of the coupled system of PDEs is done using the `StellarCore`-class found in the attached `Stellar_Core.py`-file.

After comparing the program output to the plots in the sanity check we can move on to the testing of initial values. The idea is that we change the value of each of the initial conditions, while holding the others constant. We tested different initial radii R_0, temperatures T_0 and densities ρ_0. The reason we did not test different initial pressures is that the pressure is calculated using eq. (6) which is a function of the initial temperature T_0 and density ρ_0. Thus, testing different initial pressures is not necessary, as this is essentially done by testing different initial temperatures and densities.

If we then test several different changes to each initial condition we can get a feeling how each of the changes affect the model, and which changes may compensate for others. This testing is done in several `for`-loops, where each loops through a 1/5 to 5 factor change to the initial conditions provided (See `plot-Stellar_Core.py`-file). These provided initial conditions are $L_0 = L_\odot$, $R_0 = 0.72R_\odot$, $M_0 = 0.8M_\odot$, $\rho_0 = 5.1\overline{\rho}_\odot$ and $T_0 = 5.7 \cdot 10^6$ K, which are the initial conditions of the bottom of the convection zone. The different solutions are then

plotted next to each other so as to see how the changes to the initial conditions change the simulations. Also, after getting a feeling for the behaviour of the different changes one can then explore how one change in an initial condition may affect another variable. For instance, it should be expected that the pressure will be strongly dependent on changes on the temperature and density, because the pressure is proportional to both quantities through the equation of state.

Having experimented with different initial values we are ready to try to find a model of a star where both the radius and luminosity go to zero when the mass goes to zero (within a $\pm 5\%$ tolerance of their initial values). Also, a stable solution should have a stellar core reaching out at least 10% of the initial radius used. The stellar core is the region in which 99.5% of the star's luminosity is produced. Finding a stable solution can be done in a simple trial and error strategy, where the knowledge achieved for how the change in different initial values changes the outcome of the simulations.

Another approach, which we chose, is to use the knowledge obtained by experimenting with different initial values and then loop over a grid of different initial values for which there was a significant change observed in the resulting output. More specifically we chose to loop over the two quantities that seemed to have the greatest effect on the simulation's outcome. These were found to be the initial density and radius. Further, if this strategy would not provide a stable star, one could add another loop, over for instance the temperature, to further test for more stable solutions to the simulations. The last elements of the radius, luminosity and mass arrays produced by the looped over initial values were added to their respective matrix. Also, in each of the iterations the luminosity and mass values added to their matrices were divided by $L_\odot$ and $M_\odot$, while the radius values were divided by the current looped over initial radius. These three matrices, representing a grid of tested initial values, can due to their dimensionless form, be added together into a combined matrix. Because we search for a solution where the luminosity and radius goes towards zero when the mass approaches zero, we simply need to find where the added dimensionless values $L_{\text{last}}/L_\odot + R_{\text{last}}(R_{\text{init}})/R_{\text{init}} + M_{\text{last}}/M_\odot$ approach zero the closest. Here *init* denotes the current initial value and *last* denotes the current last array value found in each loop iteration. Because we break the integration loop that solved the PDEs when the mass is negative, the luminosity and the radius should always be positive. Thus, the most stable star we obtain must be the one where the combined matrix $L_{\text{last}}/L_\odot + R_{\text{last}}(R_{\text{init}})/R_{\text{init}} + M_{\text{last}}/M_\odot$ is at its minimum. The initial values for this best model will then simply be the grid coordinates in the matrix.

The best model found can then be used to plot the values for T, L, ϵ, ρ, P and m as a function of the radius r. These plots would then give a view of how the different quantities change over the stars profile (See `BestStar.py`-file)

RESULTS/DISCUSSION

Each time a section of the program was finished, we compared their outputs to the provided sanity checks. Generally, all the sanity checks were passed without any significant errors. Comparing the κ data provided to the interpolated once, a small deviation in the second decimal of the $\log_{10}\kappa$'s was observed, but rounding up the same values as in the table were achieved. Also, when comparing the plots made by the program to the provided plots, there was basically no difference one could see with the bare eye. When trying to calculate the pressure P as a function of ρ, we got the input density ρ back when calling the density function with the argument $P(\rho)$. Thus, we can draw the conclusion that the density and pressure functions work fine, and since the other sanity checks went well we conclude that the program is ready for use.

We found that running the simulations with a ∂m found by variable step length with an error tolerance $p = 0.001$ gave a good balance between precision and run time. Thus the local relative error should be held bellow 10^{-3}. Using a higher p resulted in a choppy graph of even over- or underflows.

When trying to test different initial radii, while holding other initial values constant and equal to the once provided, we got the plots shown in Fig. (1). When looking at the plots we see that not much seems to change when using larger initial radii. But when using the lowest allowed initial radius $R_0 = 0.2 \cdot 0.72 R_\odot = 0.14 R_\odot$, the changes were quite dramatic. The pressure, density and temperature skyrocketed, while the luminosity decreased dramatically, all within a mass change of about $0.2 M_\odot$. This tells us that small radii seem to have a great effect on the outcome of the program. Specifically these changes appear to happen somewhere between $0.14 - 1 R_\odot$. Since we ultimately want to find a solution where the luminosity and the radius both approach zero (within a 5% tolerance of their tested initial values), an initial radius between $0.14 - 1 R_\odot$ could be used in order to make the luminosity drop to zero.

Next we iterated over different initial temperatures T_0 within the factor $\frac{1}{5}$ to 5 change from the provided initial conditions, and produced the plots shown in Fig. 2. In these plots the changes were generally a bit subtler on average than when changing the initial radius. In this case we can observe that the higher initial temperatures do not seem to have that great of an effect on the outcome. All higher initial temperatures seem to result in a similar behaviour. But when trying lower initial conditions, we see that the luminosity drops to zero quite quickly and almost linearly. Meanwhile the radial distance exhibits a similar behaviour, although it does not quite reach zero when the mass is zero. However, it drops quite significantly. The pressure, density and temperature behave quite similar in this case too, which is not a surprise considering they are tightly coupled through the equation of state. They all generally follow straight horizontal paths for all the higher initial temperatures. The lowest of the initial, however, cannot sustain such a high initial density, and drops fast in a matter of a small mass change, before settling into a more stable path. Also, the lowest initial temperature makes the temperature graph increase quite fast about at the same place the density does its initial drop. But generally the graph has a more increasing tendency compared to the other initial temperatures. Perhaps the most important fact to note, though, is that changing the initial temperature affects both how the radius and the luminosity look in a rather

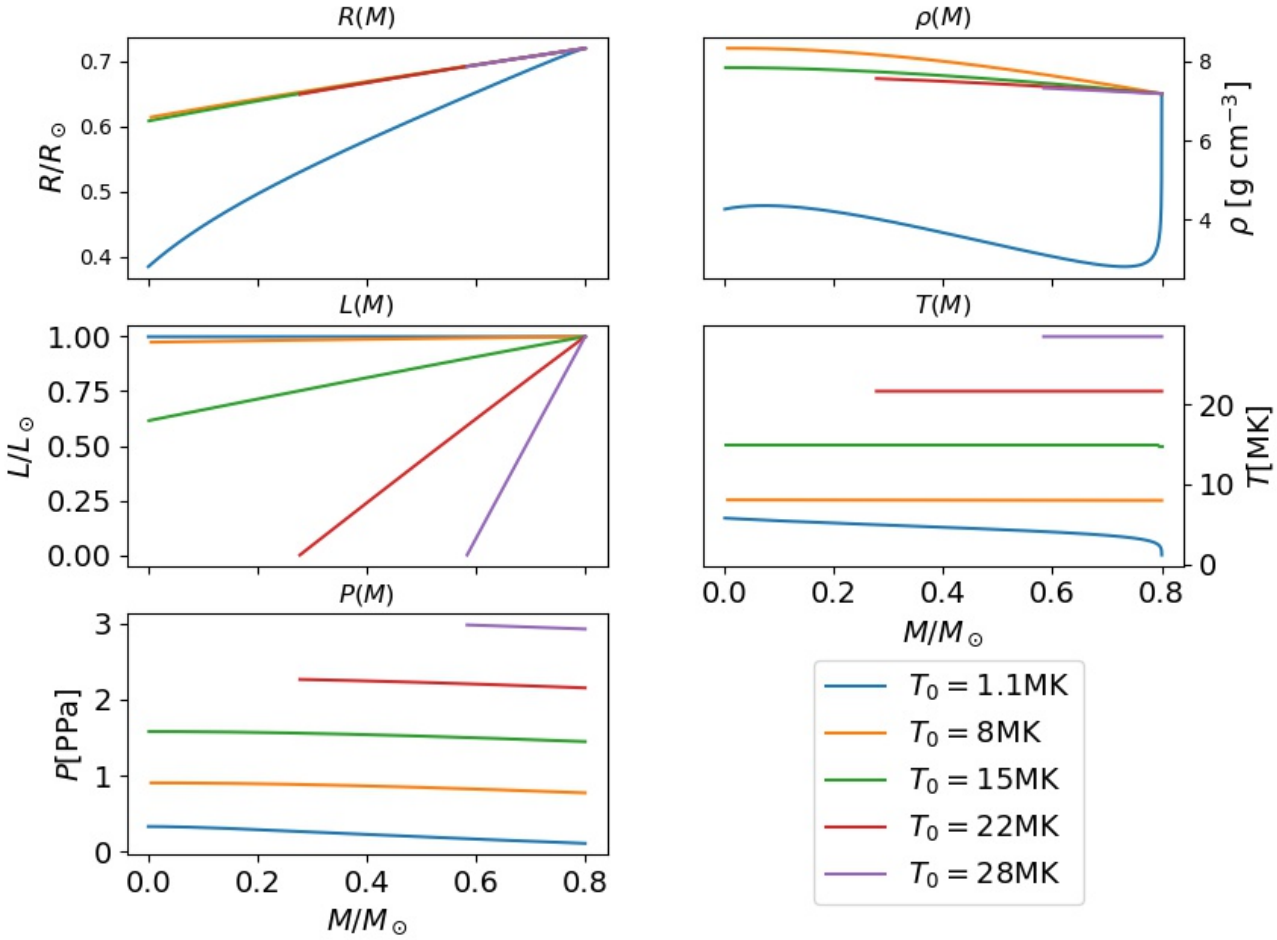

FIG. 2.— The plots show the radial distance R, density ρ, luminosity L, temperature T and pressure P for different initial temperatures. The colour code indicates the corresponding initial conditions.

more significant way than changing the initial radius and density does. The latter two affecting the radius or the luminosity significantly but not the other one.

Last the initial density was iterated over in a loop. The resulting plots can be seen in Fig. 3. In this case we see that the radial distance seems to have dropped quite a lot when reaching zero mass for a low initial density. The graph drops about $0.5R_{\odot}$ in total. The other initial densities do not affect the radius too much, as they only drop a little over the whole mass span in relatively a linear fashion. In the luminosity plot the tendency is rather different, as it is changed the leased by the different initial densities. The lowest initial density almost does not change the luminosity at all. Using the highest initial density, the luminosity is only changed by about 3% of its initial value. In the plots showing the pressure and density, all the graphs show a similar behaviour. We see that again the higher initial values tend to behave the most like straight horizontal lines, while the lowest initial density results in a slightly steeper slope. The same tendency can be observed in the temperature's behaviour, where the higher initial densities result in more or less the same slightly increasing line-like path. The lowest initial value, though, make the temperature increase about 2 MK over the whole mass span in an exponential fashion, compared to the roughly 1MK change for the higher initial densities. Again, it seems to be the lower initial values that exhibit the most interesting properties. Note that compared to the change of initial radius it is the luminosity that changes relatively little, while the radius as a function of mass decreases more significantly.

To sum up the knowledge achieved in the above-mentioned changes in initial values; The change in initial radius made the luminosity drop to zero very fast, while leaving the radial distance as a function of mass relatively steady. The change in initial temperature changed both the temperature and the luminosity quite a lot, while changing the initial density made the radial distance change more, leaving the luminosity more or less unchanged.

Thus, when choosing the two initial value changes we want to iterate over in a double loop, as mentioned in the Method section, we choose the initial radius and density. That is because they made either the luminosity or radial distance change, while leaving the other alone. In theory we can thus change the luminosity to a satisfactory path,

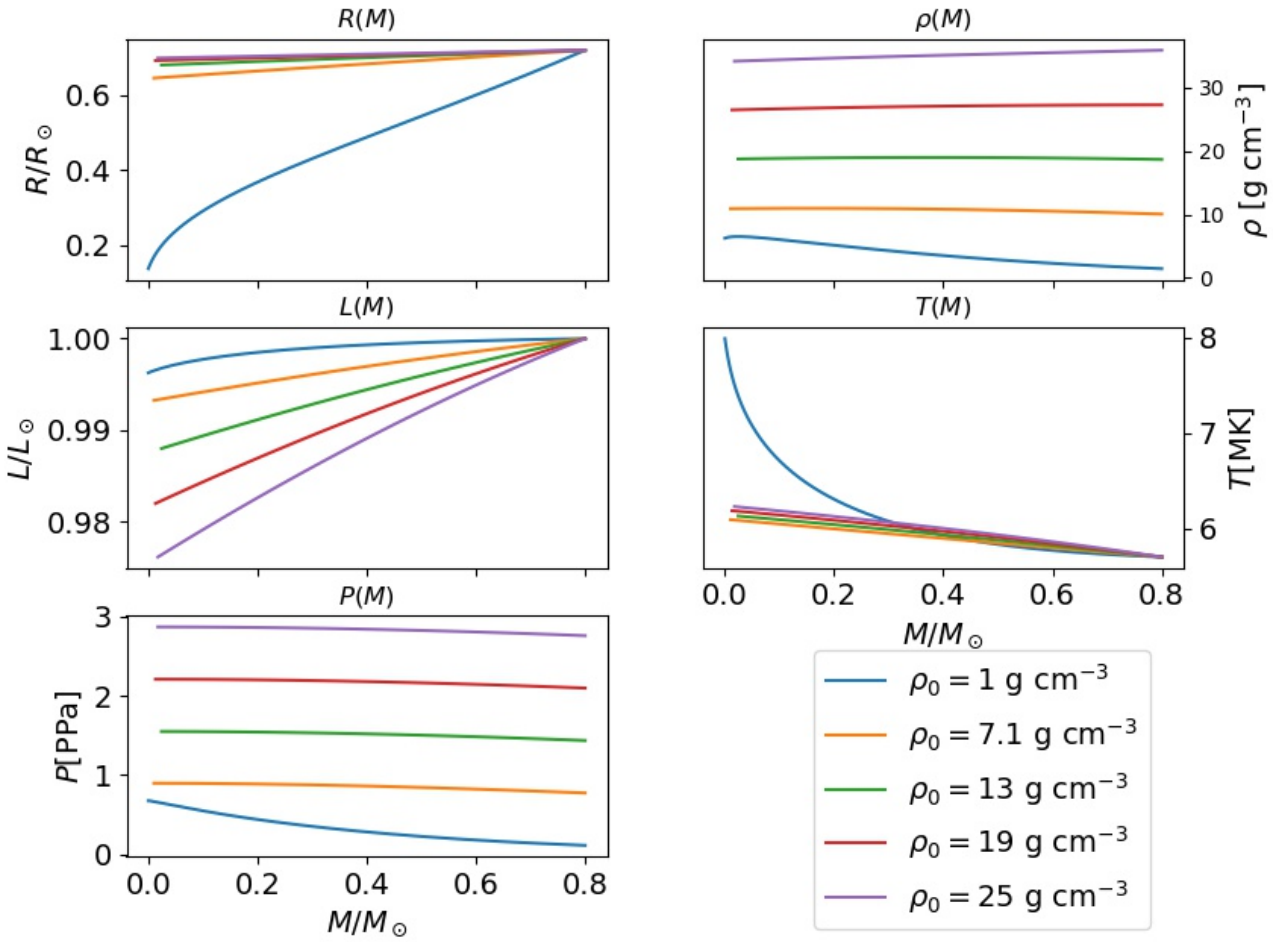

FIG. 3.— The plots show the radial distance R, density ρ, luminosity L, temperature T and pressure P for different initial densities. The colour code indicates the corresponding initial conditions.

by changing the initial radius. While the initial density can be adjusted so that the radius also approaches zero when the mass does. The colour plots produced by this algorithm can be seen in Fig. 4. We chose to iterate over a factor $1/5 - 2$ change from the provided initial conditions, with 50 iterations in each loop, thus achieving a nice level of detail of the most interesting regions seen in the colour plot. Clearly the same tendency previously observed is also seen in the colour plots. As seen in the luminosity plot in Fig. 4, luminosity tends towards zero for low initial radii, while the initial density does not have all to much of an effect. When looking at the radius plot in Fig. 4 one can again observe that the low initial density makes the radial distance go towards zero, while the initial radius only has a relatively small effect. Last, we can also see that the mass approaches zero for most initial radii and densities. Only the lowest initial radii make the mass not go to zero. In the combined plot in Fig. 4 the sum of the three others is shown. One can clearly see a region of the colour plot where all three added plots go towards zero together, that is for low initial radius and density. In this region there are probably several grid coordinates that would give a stable star.

However, the very best solution we were able to find was found by taking the minimum of the matrix of the function values in the combined plot. The grid coordinates of the minimum will thus give the initial conditions producing the solution which has the least deviation from the 5% tolerance. Specifically we found the best initial value to be $R_0^{\text{best}} \approx 0.408 R_\odot$ and $\rho_0^{\text{best}} \approx 2.756$ g cm^{-3} in addition to the other provided initial values. We found that for this model of a star the luminosity approached zero within 0.28% of $L_\odot$, the radius approached zero within 0.73% of the initial radius R_0^{best} and the mass approached zero within 2.59% of $M_\odot$, of their respective initial condition. This means that all three quantities approach zero well within the 5% tolerance. The stellar core of this solution extends about 60% of the initial radius (See red line in Fig. 5), this being well over the tolerance of 10%. All in all, this can thus be considered a successful model of a stellar radiation zone.

Having found a good model of the stellar core and radiation zone we can take a look at how it looks in detail across its cross section. In Fig. 5 one can see the mass $M(R)$, density $\rho(R)$, pressure $P(R)$, luminosity $L(R)$, temperature $T(R)$ and energy production $\epsilon(R)$ as func-

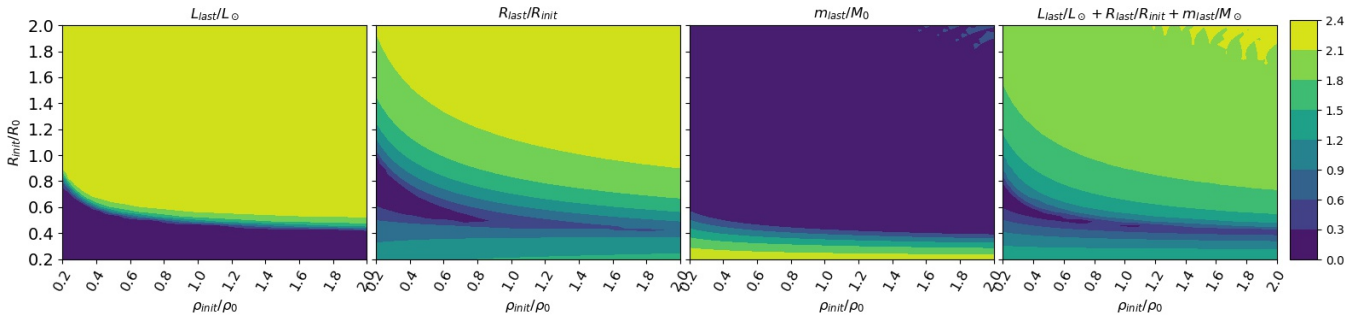

FIG. 4.— Each R, ρ-gridpoint in the three leftmost colour plots represent the last value of the luminosity $L_{last}/L_\odot$, radial distance from the core R_{last}/R_{init} and mass $m_{last}/M_\odot$ produced by solving the PDE system using the grid coordinates as initial radius and density. The rightmost plot represents the sum of the three other colour plots.

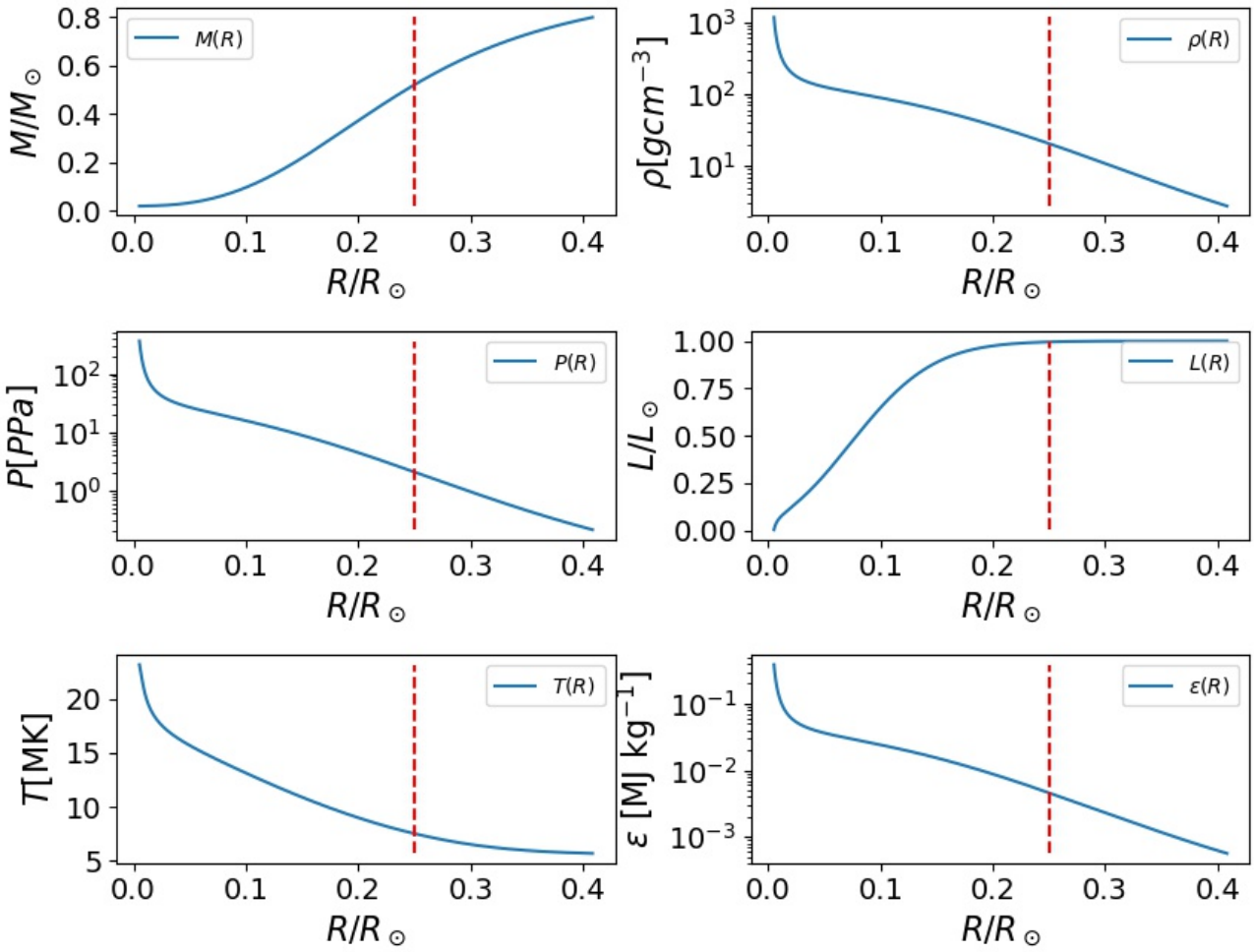

FIG. 5.— The figure shows the mass $M(R)$ (top left), density $\rho(R)$ (top right), pressure $P(R)$ (mid left), luminosity $L(R)$ (mid right), temperature $T(R)$ (low left) and energy production $\epsilon(R)$ (low right) as functions of the radial distance R from the stellar core of the best solution to the PDEs. The red dotted line indicates to where the stellar core reaches.

tions of the radial distance R from the stellar centre of the best solution found. Yet again we see that the luminosity and mass approach zero closely when the radial distance does. The form of the density and pressure are as expected a close resemblance of each other due to them being coupled by the equation of state. Also, the temperature seems to follow the general tendency of the density and pressure, which is also to no surprise as it is part of the equation of state too. Generally it makes sense that the density and pressure go upwards when approaching the stellar core, because more and more matter pushes inwards from the outer layer the further the core is approached, making the pressure and density increase. Thus, also the temperature seems to behave as it should, since a compressed gas tends to be warmed up. Since the ideal gas in this model are compressed more and more the further inwards we go, it is heated up more and more. Also considering that the initial mass is held constant for any initial value changes, it is of course no surprise that the density and pressure will be greater when using a smaller initial radius as opposed to a bigger initial radius. When looking at the energy production $\epsilon(R)$ we see that its form also seems to resemble the pressure and density. This is also consistent with the theory, as the increasing density and temperature result in a higher probability of fusion reactions happening. That is because the higher density will result in a higher number of collisions between the atomic nuclei in the plasma. While the higher temperature results in the nuclei having a higher kinetic energy making it more likely that they can overcome the coulomb potential exerted by the atomic nuclei. The cross section of the stellar core and radiation zone therefore look to be consistent with what could be expected from known physics, justifying the conclusion that this is a good result to the simulation.

CONCLUSION

To sum up, we have made a program solving a coupled system of partial differential equations in order to make a model of the stellar radiation zone. By trying different initial value for radius, temperature and density at the outermost layer we got a feeling for how they affect the simulation's outcome. This knowledge was then used to formulate a systematic method looping over a grid of initial radii and densities to find the best initial values that produced a stable stellar core and radiation zone. All in all, the best model found seems to resemble a good model of a stellar core and radiation zone. It exerts all the behaviour one would expect across its cross section, were pressure, density, temperature and energy production all grow when traveling inwards in the star. While the mass and luminosity dropped to zero when reaching the centre of the star, in addition to a big enough stellar core forming. Since the star seems to be well behaved we can conclude that we have found a stable solution to the inner parts of a star.

I thank my fellow student Bernhard Nornes Lotsberg for help and collaboration in this project.