Analysis and automation tools for the MESA stellar evolution code, implementation of rotation and atomic diffusion, toward reproduction of the Spite plateau

Author: Daniel Tickner
Student ID: 886998
Supervisor: Prof. Maurizio Salaris
Word Count: 11 2019
Date: 12 September 2021

"Can I swear? I want to swear. It's about _____ time (this report was finished)!"
Daniel Ricciardo after winning the Italian Grand Prix, 12 September 2021.

Abstract

We present a bank of resources to aid analysis of data from the stellar evolution code MESA, including conversions of output files into standardised `.csv` files that can be read by any software, fully customisable graph-plotting codes, and an automatic `inlist` generator. These resources will greatly facilitate projects requiring rapid turnaround and repeated simulations.

Simulated evolutions of low-mass, low-metallicity stars are refined to include the non-standard processes of rotation and atomic diffusion. We tailor these models toward an attempt to reproduce the *Spite plateau* for Galactic halo stars; a completion of this phase of the project was not possible, but we provide the framework to continue with this experiment.

All source material is made available through the following `GitHub` repository and may find application in similar projects involving the MESA evolution code.

https://github.com/DanTickner/MESA-analysis-tools/releases/tag/MESA

Contents

1 Introduction

1.1 Spite plateau

The *Spite plateau*, named for its discoverers, is an observational trend in which stars with effective temperature $T_{\rm eff} \gtrsim 5800$ K are measured to have roughly constant lithium abundance A(Li); below this threshold, A(Li) decreases with temperature. The trend applies to old, low-mass, halo stars (Spite & Spite, 1982). The value of A(Li) at the plateau is thought to directly correspond to the primordial lithium abundance, so accurately modelling the plateau is hoped to give strong constraints on models of big bang nucleosynthesis and the cosmological lithium problem.

Lithium is easily destroyed in stellar interiors due to its low threshold temperature for fusion, so mechanisms must be invoked to restore its surface abundance in order to explain the observed plateau. Various explanations have been proposed: among the most prolific is that constant surface lithium abundance is maintained by adequate rotational diffusion and atomic diffusion of lithium. The fact that rotation causes lower surface temperatures and enhanced surface lithium depletion, and that atomic diffusion can also enrich the surface with fusion products from the core, are strong theoretical arguments for these two effects. In this paper, we attempt to investigate whether these two processes in tandem can reproduce values of A(Li) that are consistent with a plateau.

An alternative explanation of the plateau is that mass loss (ML) effectively undoes the effect of diffusion because enhanced elements are then lost from the surface,but in evolutionary models unrealistically high ML rates are required to explain the plateau (Vick et al., 2013; Swenson, 1995). We do not consider ML in our models, especially since rotation can introduce significant additional ML if rotation rates are high enough.

1.2 Rotation

All stars rotate; this is supported by evidence from multiple independent observational techniques and theoretical considerations. However, theoretical models of rotation are highly uncertain and rely on several free parameters which are poorly constrained and have uncertain effect. Further, its implementation into one-dimensional stellar evolution codes is highly complicated, so it is usually treated as a second-order effect and neglected from models unless specifically required.

Part of the reason rotation is difficult to implement is that it introduces several chemical transport mechanisms (see Pinsonneault et al., 1989; Heger et al., 2000; Heger & Langer, 2000, and references therein). There are five main processes,[1] known as *instabilities* because the material becomes unstable to convection if the relevant condition is fulfilled; each acts on its own timescale so must be considered independently, and each requires free parameters in the theoretical development. In MESA, these free parameters take the form of "D-factors",[2] individual diffusion coefficients that the user can tweak to adjust the relative importance of each mechanism. We experimented briefly with these factors but kept them at their default values in all published models.

In general, rotational mixing acts to smooth out chemical gradients. Hence, it acts to restore H to the core as fusion progresses, increasing MS luminosity and also MS lifetime. In stars more massive than the Sun, the surface abundance of CNO elements is increased due to their production in the core (Salaris & Cassisi, 2017); we do not expect to see this effect with our stars, but we anticipate an increase in surface helium mass fraction.

The effect of rotation on the stars considered in this paper is expected to be reasonably small because low-mass stars are subject to strong magnetic braking. We test this hypothesis in §3.4. Rotating stars feature lower $T_{\rm eff}$ because centripetal force acts against gravity in the core and nuclear fusion is less intense; in turn, this extends the main sequence lifetime compared to the non-rotating case. In addition, the diffusion mechanisms mentioned above draw lithium away from the stellar surface.

[1] Namely, the dynamical shear instability, the Solberg-Hoiland instability, Eddington-Sweet circulation, the secular shear instability and the Goldberg-Schubert-Fricke instability.

[2] `am_D_mix_factor` (a global multiplier) and `D_DSI_factor` etc, `$MESA_dir/star/defaults/controls.defaults`, lines 2703-2866.

1.3 Atomic diffusion

Atomic diffusion generally causes the surface abundance of helium and metals to increase, but some species (e.g. Mg, Si, F, Fe) are expected to decrease compared to the non-diffusing case because radiation pressure pushes them into the convective envelope. The variation in surface abundances compared to the non-diffusing case peaks around the turn off; the first dredge-up almost completely undoes these effects, but we terminate our models around the turn-off and hence expect to see a difference.

Atomic diffusion is sometimes used as an umbrella term to include this process and *radiative levitation*, the movement of lighter ions (esp. H) toward the surface due to radiation pressure. However, we do not consider radiative levitation in this paper, and our usage of the term *atomic diffusion* refers solely to the former process.

1.4 The code

The stellar evolution code chosen for this project was MESA (Modules for Experiments in Stellar Astrophysics, Paxton et al., 2011, 2013, 2015, 2018, 2019). As the name suggests, this is a modular code in which the user has complete control over the choice of physical processes to be considered during the simulation. Non-standard effects such as, in our case, rotation and atomic diffusion, can easily be added to the simulation by including them in the list of commands that describes the required evolution. This list is presented as a plain text file known as an `inlist` and contains the physical properties of the star (initial mass, metallicity, element abundance, etc.) but also the choice of physical processes to include during the evolution and the relevant physical quantities associated with each process. The main executable file is `rn`, which contains the commands to run each inlist as well as instructions on where to save data outputs, making it straightforward to program sequences of multiple evolutionary models.

The control over non-standard processes and ease of scheduling multiple runs make MESA well-oriented toward the goals of this project. However, the unique dictionary of keywords required for the various physical processes, non-standard data output format, inherent need for a streamlined file system in order for repeated runs of multiple simulations to be efficiently viable, and the dearth of easily accessible official documentation, all combine to necessitate a significant investment of time and development before useful gains from the software can be made. Thus, a significant portion of this project was dedicated to developing the tools required to efficiently use MESA and extract meaningful data from it. We discuss the development of these tools in detail in §2.

1.5 The data

Once working models had been produced, we compared them to real observations in order to constrain the processes of rotation and atomic diffusion. The data chosen was from a survey by Ryan et al. (1996), which collated measurements of the effective temperature T_{eff} and metallicity [Fe/H] of several stars in the Milky Way. The stars considered in this paper are located in the Galactic halo and are near the main sequence turnoff (MSTO), so match the characteristics for stars expected to form the Spite plateau.

We compiled the data taken from Table 3 of the paper, which contained metallicity [Fe/H], effective temperature T_{eff} with associated uncertainty $\sigma(T_{\text{eff}})$, and lithium abundance $A(\text{Li})$ with associated uncertainty $\sigma(A(\text{Li}))$, for a total of 93 stars. The data is provided in Table A.1 in Appendix A.We calculated $\log_{10}(T_{\text{eff}})$ and its associated uncertainty

$$\sigma(\log_{10}(T_{\text{eff}})) \;=\; \frac{1}{\ln(10)}\,\frac{\sigma(T_{\text{eff}})}{T_{\text{eff}}} \tag{1.1}$$

from standard error propagation techniques, which allowed us to plot graphs of metallicity and lithium abundance as functions of effective temperature for this data (Figure 1.1).

The final stage of this project was intended to be a reproduction of this data using fully constrained stellar models. Somewhat inevitably, the significant outlay required to learn and use MESA hindered the time remaining to complete this stage. Somewhat inevitably, this outlay significantly hindered the amount of time available to test the initial goals of this project, namely to investigate whether the Spite plateau can be reproduced by well-constrained rotation and atomic diffusion. Instead, we focus the early part of our report on presenting the analysis tools, before going on to build the framework of a set of stars that can be evolved to test this original hypothesis.

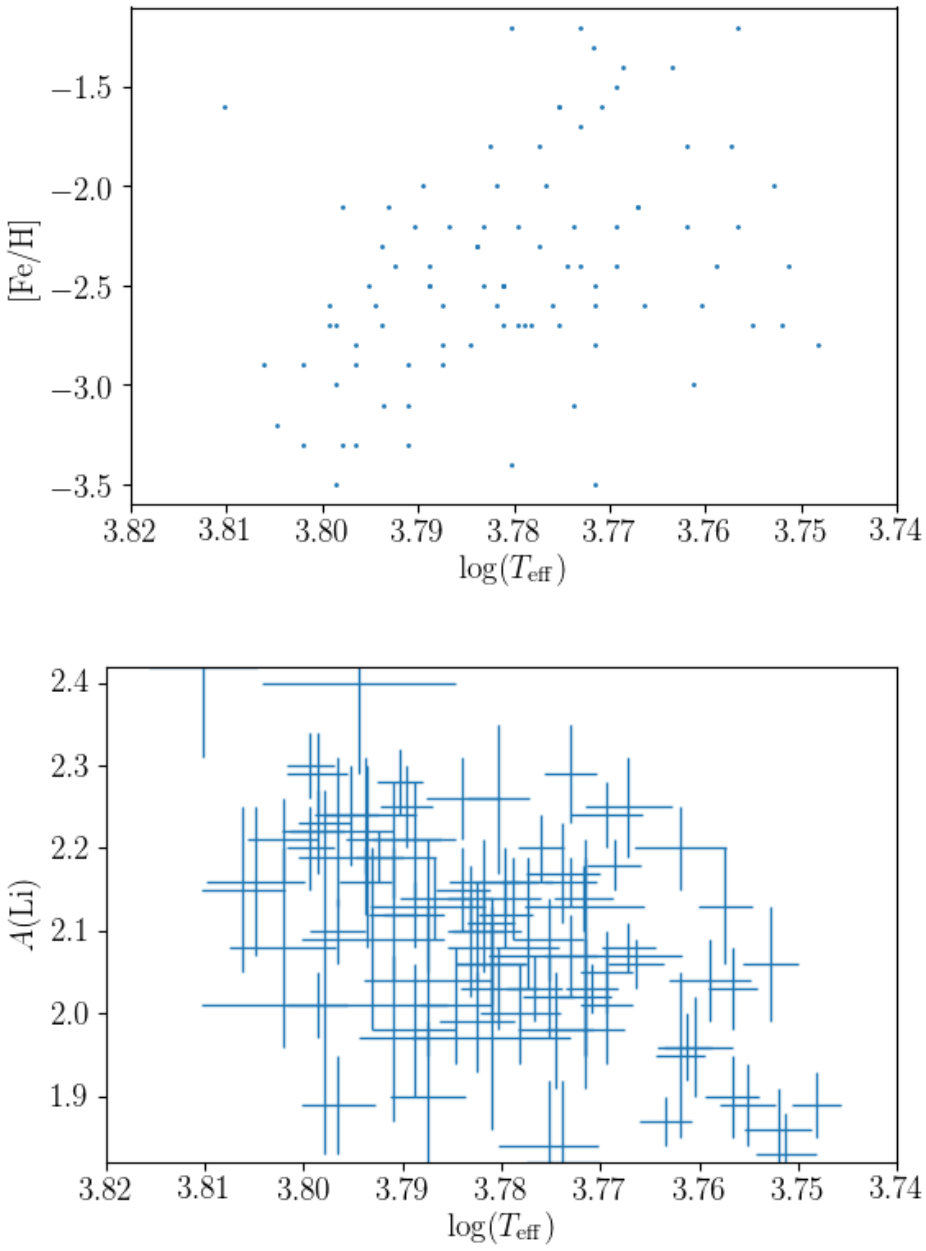

Figure 1.1: Metallicity [Fe/H] (*Top*) and surface lithium abundance A(Li) (*Bottom*) as a function of effective temperature $T_{\rm eff}$, from observational data in Ryan et al. (1996). Note that the graph of [Fe/H] does not contain error bars; this is because uncertainties in [Fe/H] were not quoted in the original paper.

Table 1.1: Acronyms and symbols used throughout this paper.

Acronym/symbol	Meaning
HRD	Hertzsprung-Russell diagram
M	Mass in solar units $M_\odot$
$M(\mathrm{X})$, $M(\,^i\mathrm{X}\,)$	Mass fraction of an element X or an isotope ^{i}X of X.
$M_{\mathrm{abs}}(\mathrm{X})$, $M_{\mathrm{abs}}(\,^i\mathrm{X}\,)$	Total mass of an element X, or an isotope ^{i}X of X, in solar units $M_\odot$
M_{tot}	Total mass of all considered elements in solar units $M_\odot$
$N(\mathrm{X})$, $N(\,^i\mathrm{X}\,)$	Number fraction of an element X, or an isotope ^{i}X of X
$N_{\mathrm{abs}}(\mathrm{X})$, $N_{\mathrm{abs}}(\,^i\mathrm{X}\,)$	Total number of atoms of an element X, or an isotope ^{i}X of X
N_{tot}	Total number of atoms of all considered elements
ML	Mass loss
MSTO	Main sequence turn-off
L	Luminosity in solar units $L_\odot$
RGB	Red giant branch
T_{c}	Core temperature in K
T_{eff}	Effective temperature in K
v_{rot}	Surface rotational velocity in km s^{-1}
X_{surf}, X_{tot}	Hydrogen mass fraction of the surface layer and of the entire star
Y_{surf}, Y_{tot}	Helium mass fraction of the surface layer and of the entire star
Z_{surf}, Z_{tot}	Metal mass fraction of the surface layer and of the entire star
ρ_{c}	Core density in g cm^{-3}
$\sigma(x)$	Statistical uncertainty in a measured value x
Ω_{crit}	Critical angular frequency for a rotating star
$\Omega_{\mathrm{initial}}$	Initial angular frequency of a protostar
Ω_{MS}	Main sequence angular frequency

2 Developing analysis codes

2.1 Translating default output to standardised format

A significant portion of the preparatory work in this project involved developing methods to translate MESA's output to a format readable by external programs for further analysis. The output of a MESA code is a directory containing multiple `profileN.data` files, were `N` is an indexing number, which provide snapshots of the star's internal structure at various points in time (i.e. properties as a function of radial position). In addition, the directory contains a single file `history.data`, which records the temporal evolution of global variables such as central density, effective temperature and surface element abundances. Scouring MESA's source material revealed no option to change the format of output files.

These `.data` files are similar to traditional `.csv` files, except that the leading five lines contain information on the running conditions of the star; the sixth line contains column headers and the desired data begins from the seventh line. A `Python` code[1] was written to convert these into `.csv` format, enabling them to be read by any program, and removing the leading five lines. The same code combined the `profileN.data` files, of which an unknown number `N` may exist, into a single `.csv` file for greatly simplified data storage. This reduces a directory with an unspecified number of `.data` files into a simple system of two files `history.csv` and `profiles.csv`. As a failsafe, the code checks the headers of each `profileN.data` file to determine whether the order of columns changes or new ones appear, and throws an error if this occurs.

2.2 Calculating additional required values

Although the requirement to translate MESA's output to useful formats adds a layer of complication, it provides the opportunity to calculate further properties of the star and append them to the same file. These calculated values are outlined below.

- Mass fractions of each element, calculated by summing the mass fractions of each isotope.
- Number fractions of each element, calculated by converting mass fractions of each element to total numbers of atoms of each element, and renormalising by the calculated total number of atoms in the star (detailed in §3.1.1).
- Lithium abundance $A(\mathrm{Li})$ calculated via (3.1).
- Hydrogen, helium and metal mass fractions for the entire star $X_{\mathrm{tot}}, Y_{\mathrm{tot}}, Z_{\mathrm{tot}}$ by application of (3.5) and Table 3.1:

$$X_{\mathrm{tot}} = \frac{M_{\mathrm{abs}}(\,^1\mathrm{H}\,) + M_{\mathrm{abs}}(\,^2\mathrm{H}\,)}{M_{\mathrm{tot}}}, \tag{2.1}$$

$$Y_{\mathrm{tot}} = \frac{M_{\mathrm{abs}}(\,^3\mathrm{He}\,) + M_{\mathrm{abs}}(\,^4\mathrm{He}\,)}{M_{\mathrm{tot}}}, \tag{2.2}$$

$$Z_{\mathrm{tot}} = 1 - X_{\mathrm{tot}} - Y_{\mathrm{tot}}. \tag{2.3}$$

[1] `P1-Read.py`. All codes written as part of this project are available for future use in the `GitHub` repository mentioned at the beginning of this paper.

– Hydrogen, helium and metal mass fractions for the surface layer,[2] whose expression is simpler because MESA outputs mass fractions fort the surface layer:

$$X_{\rm surf} = M(^1{\rm H}) + M(^2{\rm H}), \tag{2.4}$$
$$Y_{\rm surf} = M(^3{\rm He}) + M(^4{\rm He}), \tag{2.5}$$
$$Z_{\rm surf} = 1 - X_{\rm surf} - Y_{\rm surf}. \tag{2.6}$$

– Metallicity of the entire star and of the surface layer:

$$[{\rm Fe/H}]_{\rm tot} = \log_{10}\left(\frac{Z_{\rm tot}}{X_{\rm tot}}\right) - \log_{10}\left(\frac{Z_\odot}{X_\odot}\right) = \log_{10}\left(\frac{Z_{\rm tot}}{X_{\rm tot}}\right) + 1.61, \tag{2.7}$$
$$[{\rm Fe/H}]_{\rm surf} = \log_{10}\left(\frac{Z_{\rm surf}}{X_{\rm surf}}\right) + 1.61, \tag{2.8}$$

where $\log_{10}(Z_\odot/X_\odot) \approx -1.61$ is the solar metallicity (Salaris & Cassisi, 2005).

2.3 Plotting graphs from data

With the data in a readable `.csv` format, codes were written to plot the variation of any variable with time and with radial position. The codes are intended to be fully customisable, allowing the user to quickly plot any variable against any other. This rapid turnaround proved invaluable when troubleshooting the stellar models developed in this project (§3). The codes allow the user to plot arbitrarily many variables on the same graph, create a script to loop through multiple graphs, and to specify the presentation of the figure for inclusion in a formal writeup. The intention is to provide the user with an instant and consistent tool to visualise stellar models when carrying out repeated simulations or large banks of stellar models. The codes are described below.

– `P2a-HRD-TRD-Single.py` (for a single model) and `P2b-HRD-TRD-Compare.py` (to compare multiple models on the same graph) are automatic HRD and core temperature-density ($T_{\rm c}$ - $\rho_{\rm c}$) plotters. They also provide $T_{\rm eff}, L, T_{\rm c}, \rho_{\rm c}$ as separate functions of time,[3] and create a `.gif` to visualise the "motion" of the star along the HRD and $T_{\rm c}$ - $\rho_{\rm c}$ diagram.

– `P3a-History-Compare.py` plots any variable in `history.data` as a function of time, and automatically creates a second plot as a function of the `model_number` index so the user can easily identify the timestep at which a notable or problematic event occurs. A list of graphs and a separate list of stellar models, both of arbitrary length, can be specified in order to produce multiple graphs at once through nested loops.

– `P3b-SurfAbundance-Compare.py` plots the surface abundance of any element as a function of time, $T_{\rm eff}$ and `model_number`.

– `Generic-Plotter.py` is a completely generalised and flexible code which allows any two variables to be plotted against each other for a single model, `Generic-Plotter-n.py`, which iterates through a list of graphs for a single model, and `Generic-Plotter-Simultaneous.py`, which plots several similar variables (e.g. element abundances) on the same axis for a single model. allows the user to plot any variable against any other to maximise coverage of situations.

– `P4-Profile.py` plots data from `profiles.csv`. This data is effectively two-dimensional, with a full run of the variable as a function of radius (or mass coordinate, or the `zone` index) for several points in time, thus showing the evolution of the star's internal profile. This code plots any variable against radius, mass or zone and cycles through all profiles, plotting a separate graph for each point in time. The axes are normalised to be consistent across all figures. The code may easily be extended to include the `.gif` output from `P2a-HRD-TRD-Single.py`; this was not done during the project due to lack of necessity.

[2] The outermost *zone*; its exact extent is not constrained but the number of zones depends on the complexity of the internal structure.

[3] The $T_{\rm eff}$ - t and L - t graphs proved useful when validating that rotation affects $T_{\rm eff}$ but not L in Figure 3.6.

2.4 File structure

In order to efficiently and repeatably run simulations, a clear file structure is required. It is easy to become swamped in files from previous runs and to be left unsure which files to delete and which to keep once a run has finished. To aid this, MESA provides the opportunity to specify target folders to which logs are output. We recommend the following practices for large projects.

Create parent folders `z_Inlists`, `z_Logs` and `z_Photos`, where the leading `z` allows these most relevant folders to be grouped separately from MESA's own internal folders when sorting a directly alphabetically. Within each of these folders, a subfolder with the name of the current simulation may be created, into which output data will be saved. We recommend to save terminal output to a text file and keep these within their own folder `z_Terminals` to aid documentation of previous runs and to save error messages for future reference.[4]

A full working example of the file structure used during this project is given as `a-MESA` in the `GitHub` repository. The directory is missing the file `star`; this file was not modified during the project, so may be copied in from any other MESA DIRECTORY. When this is done, `mesa` should work without issue from the provided directory.

Finally, four text files,[5] written by the author of this report, are provided as basic reference tools.

2.5 Automatic inlist builder

Once a working stellar model had been produced with rotation and atomic diffusion enabled and the required initial chemical abundances (see §3), the next stage of this project involved creating a large number of stellar models. To aid this stage, a code was written to create automated sets of `inlists` for stars of increasing mass. Not only did this save time in preparing for repeated runs each of multiple stars, it also guaranteed consistency between each model - this proved necessary because the models were continually refined throughout the project.

A template `inlist`[6] was written that included the processes refined in §3. Properties which required variation between models (e.g. mass, initial abundances and filenames) were labelled with unique text signatures that could be replaced by model-specific values with an automated find-and-replace method.

A `Python` code[7] was written to copy this template, replacing the text strings by calculated values and producing an arbitrary number of `inlists` each with exact values calculated. The stellar mass, metallicity, abundance of all elements and initial rotation rate may all be specified by the user. The code also produces an executable `rn` file that includes the MESA instructions to run all of these `inlists` successively. When repeating bulk simulations that require constant tweaks to the properties of each star, this tool makes the process faster and guarantees consistency between models by applying the change to all output `inlists` when the template is modified.

[4] When running from a terminal, instead of typing `./rn`, type `./rn > z_Terminals/filename.txt`, where `filename.txt` is any name chosen to identify the run being performed.

[5] `MESA notes.txt`, `Python running instructions.txt`, `Useful history columns.txt` and `Usefule profile columns.txt`

[6] `Templates_and_data/template_inlist`

[7] `P0-Inlists.py`

3 Constraining the MESA code

In this section, we describe the process of refining the stellar models to match our requirements and to produce results in line with theoretical predictions. Atomic diffusion did not require refinement. By default, radiative levitation is not included as previously mentioned; we are not interested in this process, so we choose not to activate it.

3.1 Chemical composition

We require the initial lithium abundance of each star to match the cosmological lithium abundance $A(\mathrm{Li}) = 2.7$, regardless of the mass of the star. Unfortunately, this is difficult to implement due to MESA's inflexibility in working with mass fractions and not number fractions, and its inability to automatically scale the remaining mass fractions if one is specified. Hence, if a certain $A(\mathrm{Li})$ is required, we have no choice but to specify the fractions of all other elements along with it. In this section, we derive relations for the mass fraction of every element as a function of an initial required $A(\mathrm{Li})$ value.

3.1.1 General expression

The abundance of an element X in the photosphere of a star is denoted by [X] or $A(\mathrm{X})$ and is given by (Salaris & Cassisi, 2005)

$$[\mathrm{X}] \;=\; A(\mathrm{X}) \;\equiv\; 12 + \log_{10}\left(\frac{N(\mathrm{X})}{N(\mathrm{H})}\right), \tag{3.1}$$

where $N(\mathrm{X})$ is the number fraction of X, given by the ratio of the total number of atoms $N_{\mathrm{abs}}(\mathrm{X})$ of X to the grand total number of atoms in the photosphere N_{tot},

$$N(\mathrm{X}) \;=\; \frac{N_{\mathrm{abs}}(\mathrm{X})}{N_{\mathrm{tot}}}, \tag{3.2}$$

where[1]

$$N_{\mathrm{tot}} \;=\; \sum_{\mathrm{X}} N_{\mathrm{abs}}(\mathrm{X}). \tag{3.3}$$

The total number fraction of all elements sums to unity, and the number fraction of each element X is given by the sum of number fractions of all its isotopes ${}^{i}\mathrm{X}$:

$$1 \;=\; \sum_{\mathrm{X}} N(\mathrm{X}), \qquad N(\mathrm{X}) \;=\; \sum_{i} N({}^{i}\mathrm{X}). \tag{3.4}$$

Spectroscopic observations directly determine isotopic number abundances $N({}^{i}\mathrm{X})$ and it is these values that we wish to use (along with known relative isotope abundances). In principle, we could invert (3.1) to give $N(\mathrm{Li}) = N(\mathrm{X}) \cdot 10^{A(\mathrm{Li})-12}$ and input this $N(\mathrm{Li})$ directly to our `inlist`, but MESA only understands mass fractions. Similarly to (3.2), the mass fraction of X is

$$M(\mathrm{X}) \;=\; \frac{M_{\mathrm{abs}}(\mathrm{X})}{M_{\mathrm{tot}}}, \tag{3.5}$$

where $M_{\mathrm{abs}}(\mathrm{X})$ is the total mass of X present in the star and M_{tot} is the mass of the star. Mass fractions are given as a sum over all isotopes and the mass fractions for all elements sum to unity, similarly to number fractions.

[1] In this section, we will denote *relative* totals by their Arabic letters (e.g. N, M) and *absolute* totals with a subscript "abs" (e.g. $N_{\mathrm{abs}}, M_{\mathrm{abs}}$). We will almost exclusively be using relative values in this paper, so we reserve the simpler notation for them.

A simple relation exists between the number fraction and mass fraction of a given isotope:

$$N_{\rm abs}(\,^i{\rm X}) \;=\; \frac{M_{\rm abs}(\,^i{\rm X})}{m(\,^i{\rm X})} \;=\; \frac{M_{\rm tot}\,M(\,^i{\rm X})}{m(\,^i{\rm X})}, \tag{3.6}$$

where $m(\,^i{\rm X})$ is the mass of an atom of ^{i}X. We can turn this into an expression for each element by summing over all isotopes:

$$N_{\rm abs}({\rm X}) \;=\; M_{\rm tot}\sum_i \frac{M(\,^i{\rm X})}{m(\,^i{\rm X})} \;=\; \frac{M_{\rm tot}}{\overline{m}({\rm X})}\sum_i M(\,^i{\rm X}), \tag{3.7}$$

where we introduced the atomic weight of each element $\overline{m}$(X) because MESA does not keep track of individual isotopic masses, but approximates them all to be equal to $\overline{m}$(X) scaled by the atomic mass unit $u \approx 1.661 \times 10^{-27}$ kg. MESA uses values of $\overline{m}$(X) from de Laeter et al. (2003) and Wichers & Peiser (2020); we use these same values for consistency.[2] We must calculate $N_{\rm abs}$(X) for all elements X in the star and then compute $N_{\rm tot}$ from (3.3), before going back to compute the desired number fractions $N({\rm X}) = N_{\rm abs}({\rm X})/N_{\rm tot}$. The factors of $M_{\rm tot}$ and u cancel.

The two species we are most interested in are hydrogen and lithium. From Table 3.1, the isotopes of these elements considered by the code are ^{1}H, ^{2}H, ^{7}Li, giving

$$N({\rm H}) \;=\; \frac{M_{\rm tot}}{N_{\rm tot}}\,\frac{M({\rm H})}{\overline{m}({\rm H})} \;=\; \frac{M_{\rm tot}}{N_{\rm tot}}\,\frac{1}{\overline{m}({\rm H})}\left[M(\,^1{\rm H}) + M(\,^2{\rm H})\right], \tag{3.8}$$

$$N({\rm Li}) \;=\; \frac{M_{\rm tot}}{N_{\rm tot}}\,\frac{M({\rm Li})}{\overline{m}({\rm Li})} \;=\; \frac{M_{\rm tot}}{N_{\rm tot}}\,\frac{1}{\overline{m}({\rm Li})}\,M(\,^7{\rm Li}), \tag{3.9}$$

where $\overline{m}$(H) = 1.00794 and $\overline{m}$(Li) = 6.941.

Table 3.1: Species considered in MESA's "agb.net". The net also considers free neutrons but their surface abundance is negligible.

Element	Isotope
Hydrogen	^{1}H, ^{2}H
Helium	^{3}He, ^{4}He
Lithium	^{7}Li
Beryllium	^{7}Be
Boron	^{8}B
Carbon	^{12}C, ^{13}C
Nitrogen	^{13}N, ^{14}N, ^{15}N
Oxygen	^{16}O, ^{17}O, ^{18}O
Fluorine	^{19}Fe
Neon	^{22}Ne

[2] `$MESA_DIR/chem/public/chem_def.f90`, lines 1021-1151.

3.2 Calculating initial mass fractions

We require the initial lithium abundance of each star to match the cosmological lithium abundance $A(\mathrm{Li}) = 2.7$, regardless of the mass of the star. However, we can only directly control the initial fraction by mass of the total metallicity $z(\mathrm{Li})$. Hence, we wish to find a relation between $z(\mathrm{Li})$ and $A(\mathrm{Li})$. The mass fraction of Li is simply given by

$$M(\mathrm{Li}) \;=\; Z\, z(\mathrm{Li}), \tag{3.10}$$

where Z is the metal mass fraction of the star, one of our independent variables. Rearranging (3.1) and specialising to Li, the abundance of lithium can be calculated from $A(\mathrm{Li})$ by

$$\frac{N(\mathrm{Li})}{N(\mathrm{H})} \;=\; 10^{A(\mathrm{Li})-12}. \tag{3.11}$$

Using (3.8), (3.9), (3.10) and that $X + Y + Z = 1$, we find

$$\frac{N(\mathrm{Li})}{N(\mathrm{H})} \;=\; \frac{M(\mathrm{Li})}{M(\mathrm{H})}\,\frac{\overline{m}(\mathrm{H})}{\overline{m}(\mathrm{Li})} \;=\; \frac{Z\, z(\mathrm{Li})}{1-Y-Z}\,\frac{\overline{m}(\mathrm{H})}{\overline{m}(\mathrm{Li})}, \tag{3.12}$$

which rearranges to

$$z(\mathrm{Li}) \;=\; \frac{1-Y-Z}{Z}\,\frac{\overline{m}(\mathrm{Li})}{\overline{m}(\mathrm{H})}\,10^{A(\mathrm{Li})-12}. \tag{3.13}$$

The values used within MESA are $\overline{m}(\mathrm{H}) = 1.00794\,u$ and $\overline{m}(\mathrm{Li}) = 6.941\,u$. Our stellar models have $Y = 0.28$ and variable Z. We enforce $A(\mathrm{Li}) = 2.7$ for all models.

3.2.1 z-fractions of all other metals

This sets the initial z-fraction of Li, but we must update the z-fractions of all other metals due to the requirement that

$$\sum_{\mathrm{X}} z(\mathrm{X}) \;=\; 1. \tag{3.14}$$

This is straightforward. Consider a set of n values a_i which sum to a known value A:

$$\sum_{i=1}^{n} a_i \;=\; A. \tag{3.15}$$

Suppose that we wish to change the value of A to some other value B while maintaining the same ratio between all values a_i. The new values b_i will satisfy the sum

$$\sum_{i=1}^{n} b_i \;=\; B \tag{3.16}$$

and we can write the trivial equality

$$B \;=\; \frac{B}{A}\,A, \tag{3.17}$$

so substituting the LHS of our relations gives

$$\sum_{i=1}^{n} b_i \;=\; \frac{B}{A}\sum_{i=1}^{n} a_i. \tag{3.18}$$

We can move the constant B/A inside the sum on the RHS, giving a relation between the ith element a_i and its new value b_i:

$$b_i \;=\; \frac{B}{A}\, a_i. \tag{3.19}$$

Our situation is a sum over all metals in the net except Li. Our initial and final z-fractions $z_0(\mathrm{X}), z(\mathrm{X})$ satisfy

$$\sum_{\mathrm{X}\neq\mathrm{Li}} z_0(\mathrm{X}) \;=\; 1 - z_0(\mathrm{Li}), \tag{3.20}$$

$$\sum_{\mathrm{X}\neq\mathrm{Li}} z(\mathrm{X}) \;=\; 1 - z(\mathrm{Li}), \tag{3.21}$$

where $z_0(\mathrm{X})$ was calculated previously and $z(\mathrm{Li})$ is given by (3.13). Hence, the z-fractions we must input to our model are

$$z(\mathrm{X}) \;=\; \frac{1 - z(\mathrm{Li})}{1 - z_0(\mathrm{Li})}\, z_0(\mathrm{X}), \qquad \mathrm{X} \neq \mathrm{Li}. \tag{3.22}$$

This effect is only significant for the lowest metallicities we will use ($Z \sim 10^{-5}$) and only produces variation of z-fractions in the third significant figure. However, we will apply it for all runs.

3.2.2 Mass fractions of hydrogen and helium isotopes

A complication of specifying mass fractions of metals is that MESA also requires the mass fractions of ^{1}H, ^{2}H, ^{3}He and ^{4}He to be specified such that $X = M(^1\mathrm{H}) + M(^2\mathrm{H})$ and $Y = (^3\mathrm{He}) + M(^4\mathrm{He})$. It is important to maintain realistic abundance ratios of each isotope. We use the primordial abundance ratios (Porcelli & Ballentine, 2002; Linsky et al., 2006)

$$\frac{N(^2\mathrm{H})}{N(^1\mathrm{H})} \;\approx\; 2.3\times10^{-5}, \tag{3.23}$$

$$\frac{N(^3\mathrm{He})}{N(^4\mathrm{He})} \;\approx\; 3\times10^{-4}. \tag{3.24}$$

We require mass fractions and not number fractions of these isotopes, but from (3.8) and (3.8), the ratios are the same when we use average atomic weights $\overline{m}$:

$$\frac{N(^i\mathrm{X})}{N(^j\mathrm{X})} \;=\; \frac{M(^i\mathrm{X})}{M(^j\mathrm{X})}\,\frac{\overline{m}(^i\mathrm{X})}{\overline{m}(^j\mathrm{X})} \;=\; \frac{M(^i\mathrm{X})}{M(^j\mathrm{X})}. \tag{3.25}$$

Defining the ratios as $R_\mathrm{H} \equiv M(^1\mathrm{H})/M(^2\mathrm{H}) = 2.3\times10^{-5}$ and $R_\mathrm{He} \equiv M(^3\mathrm{He})/M(^4\mathrm{He}) = 3\times10^{-4}$, we can write the mass fractions as

$$X \;=\; M(\mathrm{H}) \;=\; M(^1\mathrm{H}) + M(^2\mathrm{H}) \;=\; M(^1\mathrm{H})\left[1 + \frac{M(^2\mathrm{H})}{M(^1\mathrm{H})}\right] \;=\; M(^1\mathrm{H})\left[1 + R_\mathrm{H}\right], \tag{3.26}$$

$$Y \;=\; M(^4\mathrm{He})\left[1 + R_\mathrm{He}\right]. \tag{3.27}$$

Our independent variables are Y and Z, so using $X = 1 - Y - Z$ and rearranging, we find expressions for each mass fraction:

$$M(^1\mathrm{H}) \;=\; \frac{1 - Y - Z}{1 + R_\mathrm{H}}, \tag{3.28}$$

$$M(^2\mathrm{H}) \;=\; R_\mathrm{H}\, M(^1\mathrm{H}), \tag{3.29}$$

$$M(^3\mathrm{He}) \;=\; \frac{Y}{1 + R_\mathrm{He}}, \tag{3.30}$$

$$M(^4\mathrm{H}) \;=\; R_\mathrm{He}\, M(^4\mathrm{He}). \tag{3.31}$$

3.2.3 Metal abundances: initial z-fractions

In order to use the above relations, we must modify the initial z-fractions because our net does not contain every element and so our fractions will not sum to unity. In normal use, this is not an issue because MESA calculates mass fractions automatically if none are specified. However, we must specify the z-fraction of Li and so the others must follow to maintain the normalisation. In this paper we choose to use the abundances quoted by Grevesse & Sauval (1998), hereafter GS98, because this is the default set of values used in MESA.[3] We quote our results to three significant figures in accordance with MESA's implementation of these values. Indeed, the abundances quoted in Grevesse & Sauval (1998) carry typical uncertainties in the third significant figure.

Stellar evolution codes do not consider every possible atomic species because each one adds an additional equation to the set of stellar structure equations, greatly increasing computational difficulty. Instead, they use a "net" of the most significant elements for the evolution of the star. We use `agb.net`, which contains the elements in Table 3.1 (file, line). Renormalising over the species contained in the net, we find the z-fractions given in Table 3.2.

Table 3.2: Rescaling of GS98 z-fractions to account for "agb.net".

Element	$z_{\text{original}}(X)$	$z_0(X)$
Li	1.42×10^{-8}	1.42×10^{-7}
Be	2.37×10^{-10}	2.37×10^{-8}
B	6.67×10^{-9}	6.67×10^{-7}
C	3.98×10^{-3}	3.98×10^{-1}
N	1.17×10^{-3}	1.17×10^{-2}
O	1.08×10^{-2}	1.08×10^{-1}
F	5.74×10^{-7}	5.74×10^{-5}
Ne	2.43×10^{-3}	2.43×10^{-1}
Total	0.0184	1.000

3.3 Stopping condition

MESA does not have the capability to recognise qualitatively the current stage of a star's evolution (e.g. the moment it enters the MS or the RGB), so we must create a stopping condition based on the value of a certain parameter.

We wish to compile a population of stars each with age 13 Gyr, so the most natural termination condition is an upper age limit. Further, MS lifetimes are sensitive to mass, luminosity and chemical composition (Ryan & Norton, 2010) and since luminosity and mass are themselves closely related and we intend to hold the chemical composition constant, in effect the age at the MSTO is uniquely determined by initial mass for the stars we consider, making this easy to implement. Although MESA supports an age condition, we encountered the unfortunate result that age conditions stop the code immediately, before any output data can be saved (analogous to an error message in general coding), rendering this condition useless for experiments.

Other possible termination conditions include the mass of the hydrogen envelope (a proxy for the progress of hydrogen fusion), the luminosity (which steadily increases along the MS) and the core density ρ_c (a monotonically increasing function of age, at least until energy production ceases). Indeed, core densities are often used as rough indicators for the activation of hydrogen and helium fusion (Cassisi & Salaris, 2013). Core density depends only on initial mass, independent of rotation or atomic diffusion unlike star age, so the same value can be used regardless of the physical processes being considered within the model.

[3] This is stated in `$MESA_dir/star/defaults/star_job.defaults`, line 342 (in the docstring of `create_pre_main_sequence_model`). The actual values are stated in `$MESA_DIR/chem/public/chem_def.f90`, lines 1304-1407.

Performing trial runs with increasing mass, it was found that a star with $M = 0.72\,M_\odot$ has certainly passed the MSTO by the time it reaches $\log_{10}(\rho_c) \sim 3.3$, where ρ_c is measured in g cm^{-3} (Figure 3.1). Since the MSTO occurs at lower densities for lower-mass stars, and those more massive than $0.72\,M_\odot$ enter the RGB before 13 Gyr so are excluded from our analysis, we can safely enforce the global termination condition

$$\log_{10}(\rho_c) \;=\; 3.3 \quad \text{(termination condition)}. \tag{3.32}$$

We discuss potential refinements to this condition in §4.2.2.

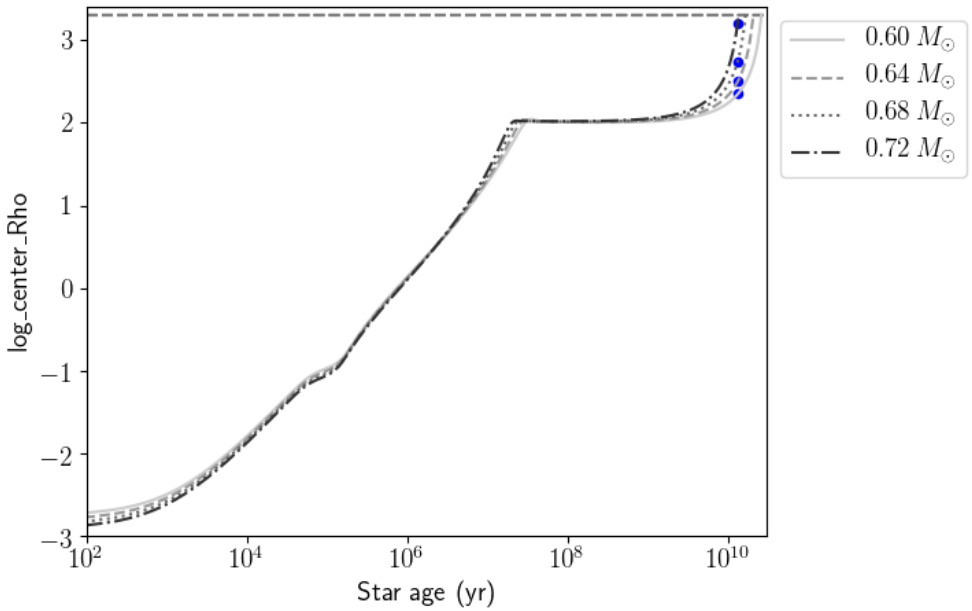

Figure 3.1: Core density ρ_c (g cm^{-3}) as a function of age (yr) for four initial masses given in the legend. The blue dots represent the position of the MSTO for each star. The dashed grey horizontal line represents the cutoff point $\log_{10}(\rho_c) = 3.3$ at which all models were terminated. All models had $Z_{\text{initial}} = 1 \times 10^{-4}$ and no rotation or diffusion.

It remains to "roll-back" each stellar model after successful termination, to recover the conditions of the star at age 13 Gyr. This was done with a `Python` code,[4] written to compile the properties of a set of models at a pre-determined evolutionary stage so that a population can be plotted against each other. The code was written to accommodate various evolutionary phases and may find future application in any of these, but in this project we are concerned with age. The code searches each star's `profile.csv` for the first model with age > 13 Gyr and records the properties of the star at that model number. More accurate age-extraction methods are discussed in §4.2.3.

However, a slight complicated is that some lower-mass stars will have passed the MSTO by 13 Gyr and began surface enhancement of He fusion products. This contaminates our population compared to the data by Ryan et al. (1996), so we must eliminate it. Although in reality the MSTO is a smooth transition and not a single instant in time, for ease of implementation we define it to be the single point in a star's lifetime of maximum T_{eff}. Then, our *Python* code will search each star's output data for the first model with age > 13 Gyr, but only add the star to the compiled list if it has not yet reached the MSTO by this age (i.e. if the maximum in T_{eff} occurs at a later age). By this process, the code compiles a list of main-sequence stars that are close to 13 Gyr.

[4] `P5a-Compile-Rows.py`, included in the `GitHub` repository.

3.4 Initial rotation rate

Initially in the development process, a two-stage `inlist` was employed in which the first `inlist` evolves a protostar until it reaches the main sequence and then a second `inlist` continues this evolution until the AGB. An advantage is that the user may control properties of the star (e.g. Ω) once it enters the MS. However, MESA carries no memory of previous `inlists`: every physical property and required process must be restated once a new `inlist` is loaded, or else the default values will be chosen. This has the potential to cause severe continuity issues in the star's evolution that may go unnoticed, so the decision was taken to perform the entire evolution with a single `inlist`. An added advantage is that construction of single `inlists` is far easier to automate, which became important later in the project.

A drawback is that direct control over the star's MS rotation rate is lost: we may only state an initial protostar rotation rate $\Omega_{\text{initial}}/\Omega_{\text{crit}}$ and rely on magnetic braking to provide a realistic MS rate. Runs were performed with $\Omega_{\text{intitial}}/\Omega_{\text{crit}}$ varying over a few orders of magnitude and the resulting rotational velocity v_{rot} and lithium abundance $A(\text{Li})$ at the MSTO were plotted (Figures 3.2 and 3.3). We slightly refined the initial mass of the control star to $M = 0.68\,M_{\odot}$ and included the automated element abundance specification outlined above.

From our graphs, magnetic braking only becomes significant for $\Omega_{\text{intitial}}/\Omega_{\text{crit}} \gtrsim 0.1$. However, in this region the MS rotation saturates at $v_{\text{rot}} \sim 400$ km s^{-1}, unrealistically high (Stauffer & Hartmann, 1986) for low-mass stars. Further, surface lithium depletion is extremely strong at these rotation rates. The region $\Omega_{\text{intitial}}/\Omega_{\text{crit}} \sim 10^{-3}$ provides $v_{\text{rot}} \lesssim 50$ km s^{-1}, more in line with observational constraints, and nonzero but non-extreme surface lithium depletion. A second experiment was carried out on this narrower interval; the value of v_{rot} depends linearly on $\Omega_{\text{intitial}}/\Omega_{\text{crit}}$, and lithium depletion is also dependent on the chosen value.

Restricting $\Omega_{\text{intitial}}/\Omega_{\text{crit}} \gtrsim 0.1$ to this window reduces variation in v_{rot} and $A(\text{Li})$ from several orders of magnitude down to a much narrower interval, but the values are still sensitive to the initial choice of rotation rate. A single value of $\Omega_{\text{intitial}}/\Omega_{\text{crit}}$ must be applied to all models before further analysis can be carried out, but we refrain from prescribing an exact value in this paper except to recommend $\Omega_{\text{intitial}}/\Omega_{\text{crit}} \sim 10^{-3}$. It may be preferable to set a hard rotation rate once the star enters the MS and gain more control over the rotating star, but this entails a restructuring of the entire code as previously mentioned.

To avoid significant deviations from the nonrotating case, we ensured that rotation rates never exceeded Ω_{crit} and rotational mass loss never occurred. Due to the choice of `inlist` construction, we did not have direct control over the main sequence rotation rate Ω_{MS}, only the initial rotation rate of the collapsing protostar on the pre-main sequence Ω_{PMS}. This was of little initial concern because low-mass stars are thought to undergo significant magnetic braking before the main sequence, slowing their rotation rates to a low value that is quite independent of the protostar rotation rate. Initial runs appeared to support this hypothesis, but it is not true in general. We will discuss potential arguments to this claim later in the paper.

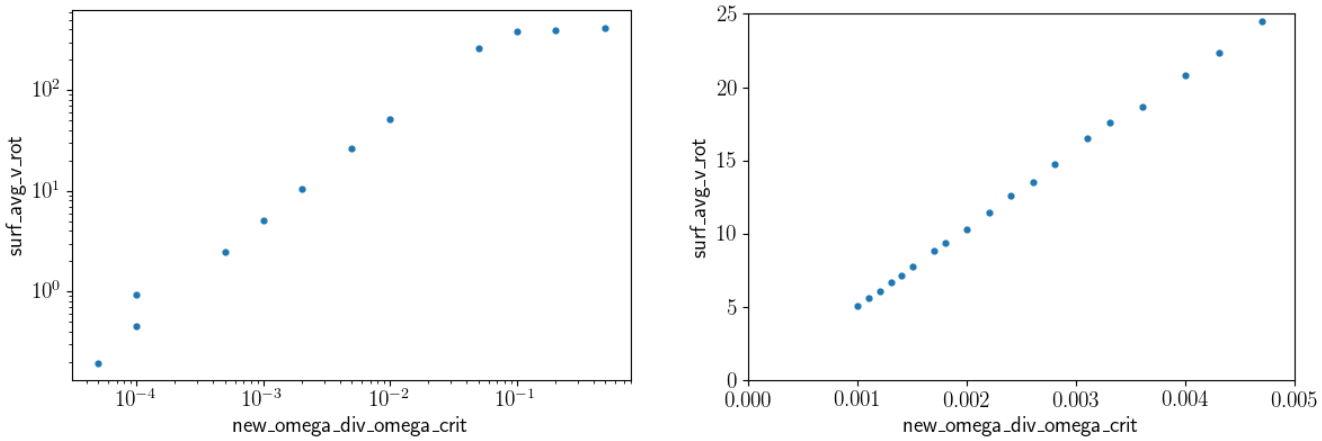

Figure 3.2: Average surface rotational velocity v_{rot} (km s^{-1}) as a function of initial angular frequency over critical frequency $\Omega_{\text{intitial}}/\Omega_{\text{crit}}$, for a $0.68\,M_{\odot}$ star at the MSTO. MESA internal variables label the axes.

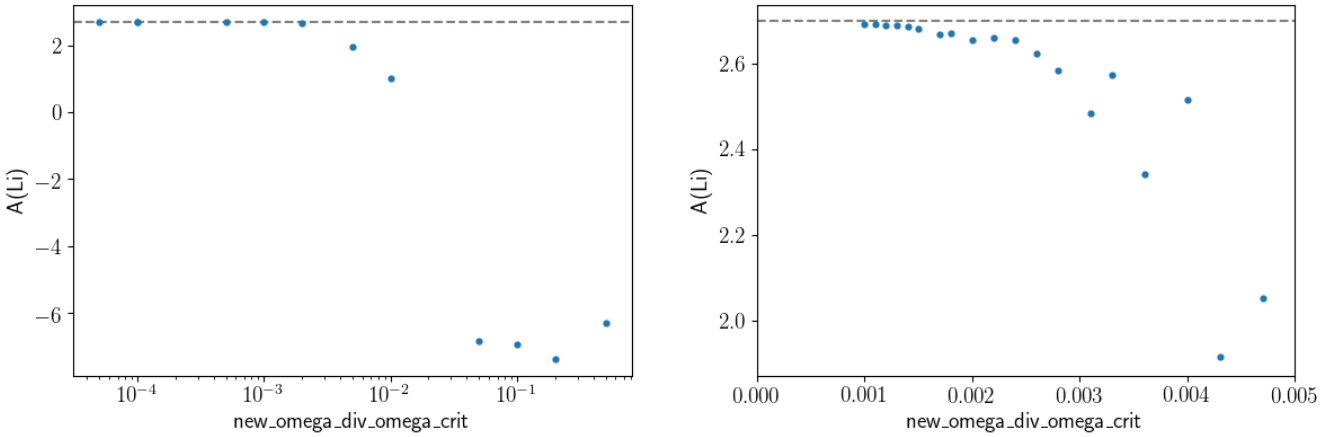

Figure 3.3: Surface lithium abundance $A(\mathrm{Li})$ as a function of initial angular frequency over critical frequency $\Omega_{\mathrm{intitial}}/\Omega_{\mathrm{crit}}$, for a $0.68\,M_{\odot}$ star at the MSTO. MESA internal variables label the axes. Initial lithium abundance $A(\mathrm{Li})_{\mathrm{initial}} = 2.7$ indicated by dashed grey line.

In this project, it was important to avoid mass loss (ML) triggered by rotation above some critical frequency Ω_{crit}.[5] Above this frequency, the sum of centripetal force and radiation pressure overcome gravity and the outer layers of the star are expelled. However, since low-mass stars are typically slow rotators and ML necessarily removes any changes in surface abundance, we aim to avoid it in our code. To test this, the rotation frequency was plotted across the evolution of a stellar model for various initial values $\Omega_{\mathrm{initial}}/\Omega_{\mathrm{crit}}$ (Figure 3.4). From the graph, we require $\Omega_{\mathrm{initial}}/\Omega_{\mathrm{crit}} \lesssim 5 \times 10^{-4}$ to guarantee that rotational ML never occurs. A stricter bound of $\Omega_{\mathrm{initial}}/\Omega_{\mathrm{crit}} \lesssim 5 \times 10^{-5}$ will prevent rotation from even reaching 10% of the critical value.

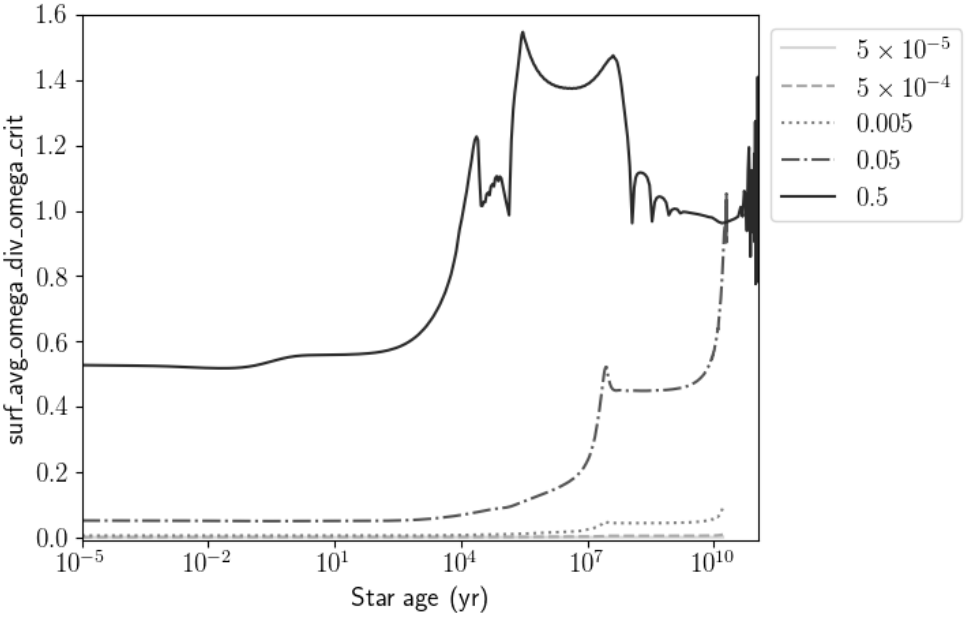

Figure 3.4: Angular frequency relative to critical value for mass loss $\Omega/\Omega_{\mathrm{crit}}$ as a function of star age (yr), for five initial rotation rates $\Omega_{\mathrm{initial}}/\Omega_{\mathrm{crit}}$ given in the legend. All models had $M = 0.68\,M_{\odot}$ and $Z_{\mathrm{initial}} = 1 \times 10^{-4}$.

[5] An expression for Ω_{crit} is given in §6.4 of Paxton et al. (2013) and on line 1238 of `$MESA_DIR/star/defaults/star_job.defaults`.

3.5 Comparing the effects of rotation and atomic diffusion

With rotation and atomic diffusion both constrained and the initial chemical composition of the star held constant, we can now consider the effects of rotation and atomic confusion compared to a standard star without these processes enabled.

To do this, we evolved a star in four separate cases: standard (no rotation or diffusion), diffusion only, rotation only, and both processes enabled. All other variables were kept the same between the stars, so any deviations from the standard star could be directly attributed to rotation or to diffusion. The star was chosen to be roughly representative of the old halo stars to be compared against: we chose mass $M = 0.76\,M_{\odot}$, initial metal mass fraction $Z_{\text{initial}} = 1 \times 10^{-4}$ and mixing length parameter $\alpha_{\text{ML}} = 1.8$. An initial rotation rate $\Omega_{\text{initial}}/\Omega_{\text{crit}} = 5 \times 10^{-3}$ was chosen. We plot a HRD of the four stars in Figures 3.5 and 3.6, and compare the surface element abundances in Figures 3.7, 3.8 and 3.9.

The HRD of all four models is plotted in Figure 3.5. Note that the choice of low $\Omega_{\text{intitial}}/\Omega_{\text{crit}} = 5 \times 10^{-3}$ means the effects of rotation are small compared to the standard model. Even this modest rotation rate yields mixing strong enough to deplete surface Li by 0.8 dex by the MSTO (Figure 3.7). Atomic diffusion causes even stronger depletion, although halfway through the MS surface Li begins to be replenished followed by an even sharper decline. The net effect when both processes occur together is a very strong decrease in surface Li.

In the non-rotating standard model, all element abundances remain at their initial values for the entirety of the evolution up to the MSTO. Hence, any changes in element abundance in this experiment are purely due to rotation and/or atomic diffusion. The qualitative effect of atomic diffusion on surface abundances is consistent across all three figures. Early on the MS, the depletion of surface elements is very strong. A temporary increase begins at 3-4 Gyr; we postulate that the convective zone reaches the core at this time and atomic diffusion causes fusion products to move toward the surface, but timing constraints prevented us from attempting to verify this claim. An extremely sharp decline occurs around 7 Gyr but abundances are rapidly replenished afterward, so that the star leaves the MS with surface elements strongly depleted compared to the protostar but more abundant than at 7 Gyr.

Rotation, at the low rate considered here, does not significantly affect [Fe/H] or Y compared to the standard star. However, its potential effect is still important because, when diffusion occurs, rotation acts as a stabilising factor to reduce the overall depletion of elements except Li. Indeed, the traces with rotation and diffusion show less depletion than those with diffusion only. This is to be expected since rotational diffusion is known to act in the direction opposing a changing concentration gradient Salaris & Cassisi (2017).

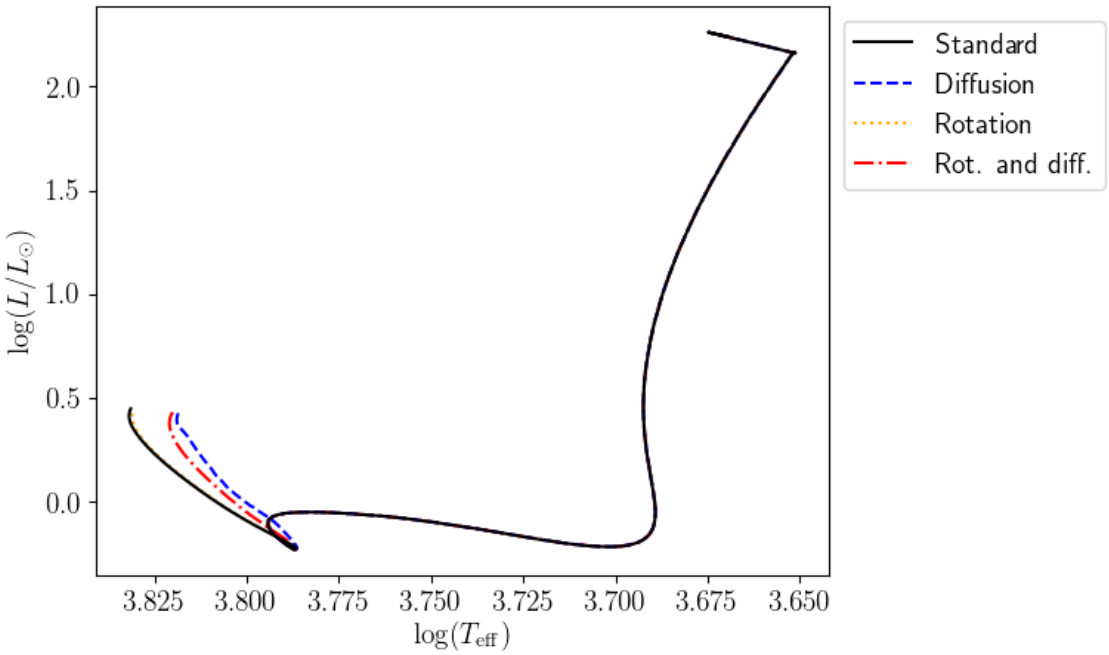

Figure 3.5: Hertzsprung-Russell diagram for a $0.76\,M_\odot$ star with $Z = 1 \times 10^{-4}$ evolved from the pre-MS (point of maximum L, near top right) until $\log_{10}(\rho_c) = 3.3$, with activation of rotation and atomic diffusion given in the legend. Rotating models had $\Omega_{\rm initial}/\Omega_{\rm crit} = 5 \times 10^{-3}$.

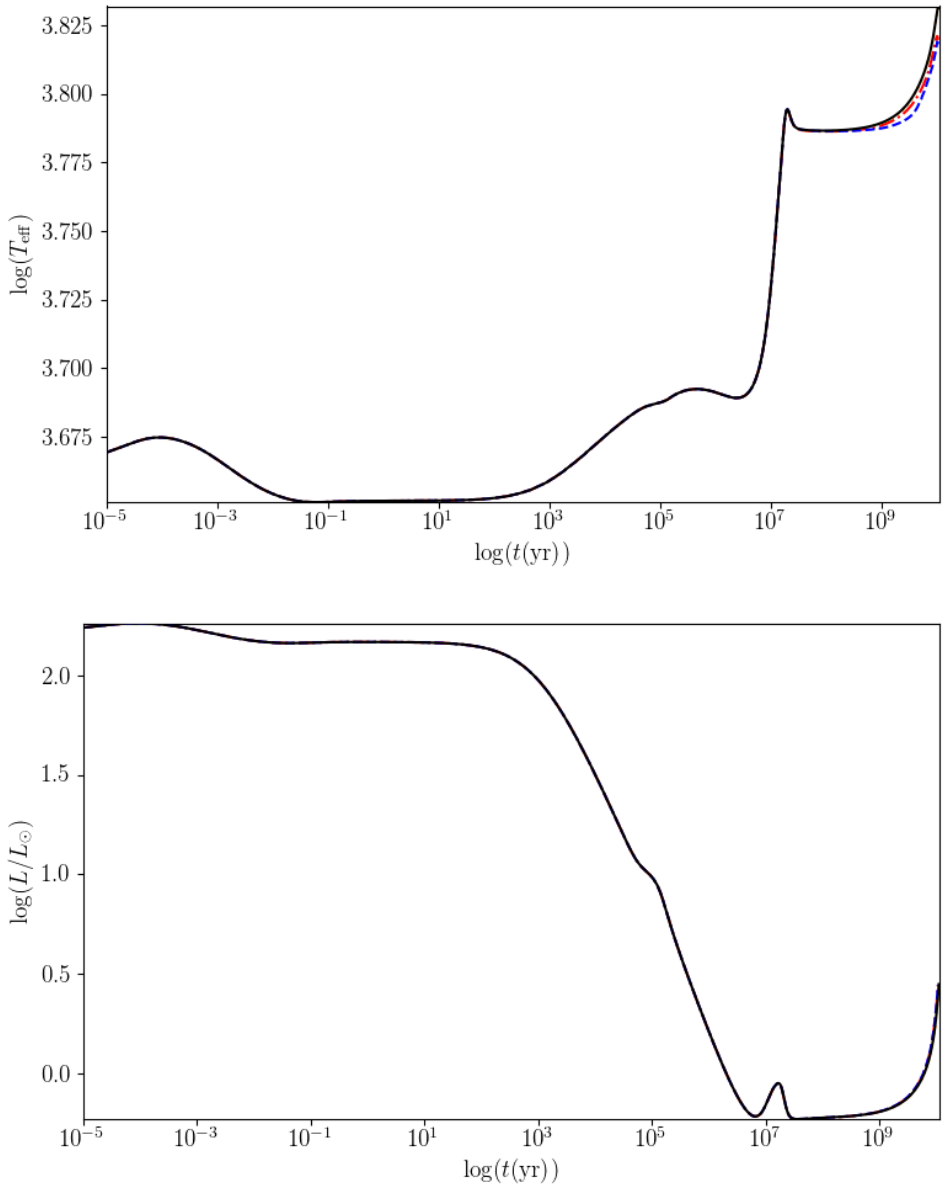

Figure 3.6: Effective temperature $T_{\rm eff}$ (Top) and luminosity L (Bottom) as a function of time for the four stellar models in Figure 3.5.

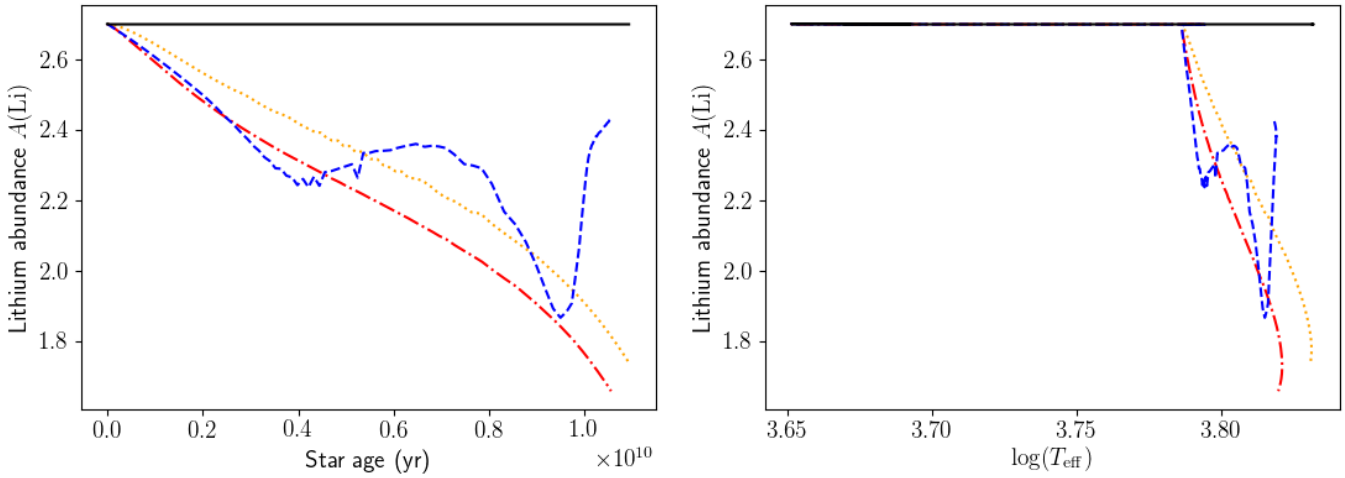

Figure 3.7: Lithium abundance with time (Left) and $T_{\rm eff}$ (Right) for the four models in Figure 3.5.

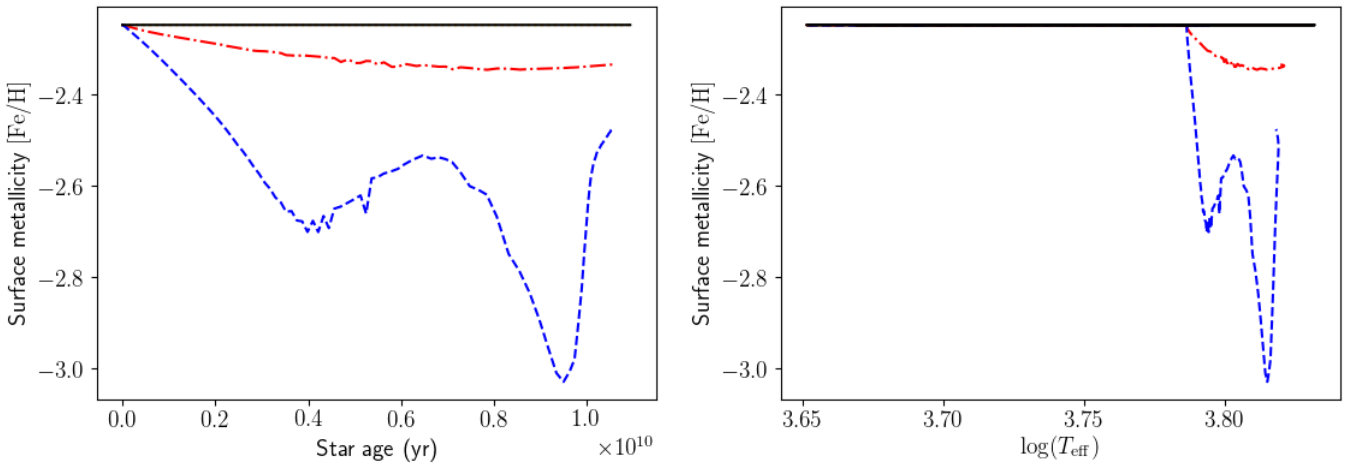

Figure 3.8: Lithium abundance with time (Left) and $T_{\rm eff}$ (Right) for the four models in Figure 3.5.

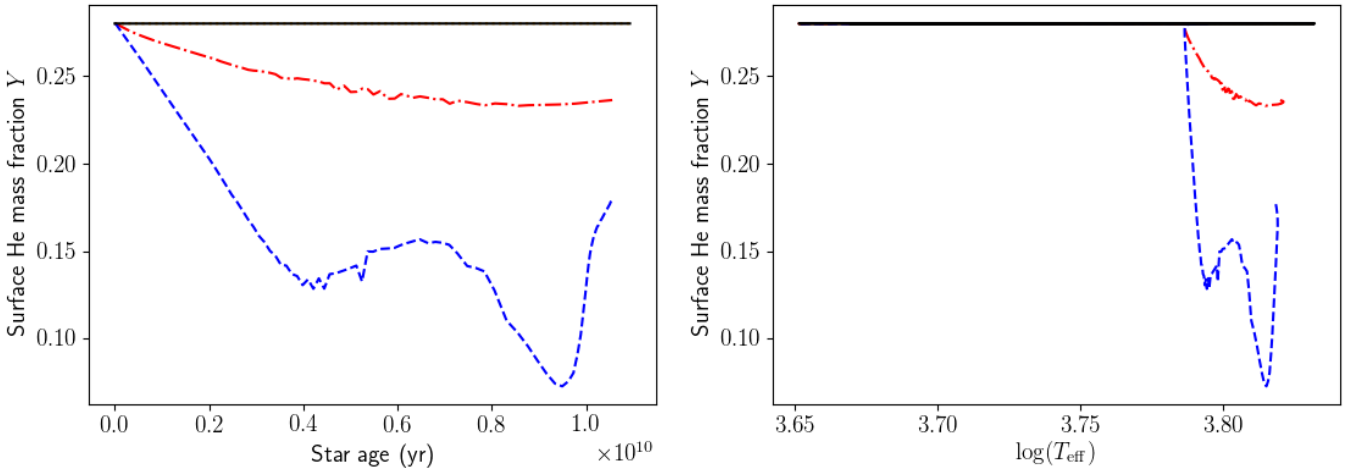

Figure 3.9: Lithium abundance with time (Left) and $T_{\rm eff}$ (Right) for the four models in Figure 3.5.

4 Discussion

4.1 Conclusion

We have described the development of a set of tools which greatly improve the ease and efficiency of analysing output from the stellar evolution code MESA. We have developed codes to convert its output from a `.data` format to a universally understood `.csv` format and to calculate additional properties unsupported by MESA's output, appending these additional values to the same `.csv` file. We developed a range of graphing codes to enable quick turnaround of simulations and aid formal presentation of results. We constructed an automated tool for producing `inlists` for arbitrarily large numbers of stars to be simulated sequentially. Our codes were designed to be fully customisable toward any application, not just that of the current report.

We built a MESA `inlist` for a low-mass, low-metallicity star featuring the non-standard processes of rotation and atomic diffusion, and refined it produce a star that emerges at the MSTO with the properties expected.

The final goal of this project was to use the above models to assemble a population of old halo stars around the MSTO of varying masses, and to refine this population until their lithium abundance as a function of effective temperature reproduces the Spite plateau. Unfortunately, the processes of learning MESA, developing the tools needed to properly analyse its output, and refining simulations of low-mass stars, proved too time-consuming to allow for a genuine attempt at reproducing the Spite plateau. Exploratory efforts were made but minor issues hindered the results obtained (§4.2.1). Any interpretations of these efforts would be rushed and not fully considered, so they have been omitted from the conclusion in the interest of avoiding potentially misleading results.

An attempt to reproduce the Spite plateau would have brought this project to its intended endpoint and perhaps shed light on the plateau's elusive origin. It is regretful that it could not have been properly tested, but we hope that by providing these tools, the search for an explanation of the Spite plateau may finally come to an end.

4.2 Suggestions for improvement

4.2.1 Continuation of attempt to replicate the Spite plateau

The code in this report is able to evolve a population of low-mass stars with varying masses toward the MSTO and extract the properties of each star at 13 Gyr, discarding any models that have progressed to the RGB and whose surface composition will significantly alter. In principle, all that is required to reproduce the Spite plateau is to plot A(Li) vs $T_{\rm eff}$ for these stars and overlay it onto the observational data from Figure 1.1 (Bottom).

We attempted to plot an A(Li) - $T_{\rm eff}$ graph (Figure 4.1) but the models appeared to show a strong dependence of A(Li) on initial mass (Figure 4.2). The issue is that a star's convective envelope recedes during the MS, but around the MSTO it is restored to a large extent of the star and atomic diffusion can then transfer a large amount of fusion products (especially He) to the surface (Salaris et al., 2000). This greatly enhances $Y_{\rm surf}$ and all other abundances fall as a direct result.[1] If the higher mass (faster-evolving) members of the population reach the MSTO and experience this surface He enrichment but lower-mass members do not, a great inconsistency is introduced in the Li abundance of these stars and a direct comparison across the population becomes difficult to judge.

The extent and significance of MSTO He enrichment was not explored during the project, but it must be understood and quantified in order for meaningful conclusions to be drawn. It is for this reason that we refrained from quoting any results based on the Spite plateau. It is hoped that an understanding of this process will quickly lead to a situation in which the plateau can be reproduced.

[1] Recall that the sum of all abundances is normalised to unity, so if $Y_{\rm surf}$ increases then all other abundances must decrease.

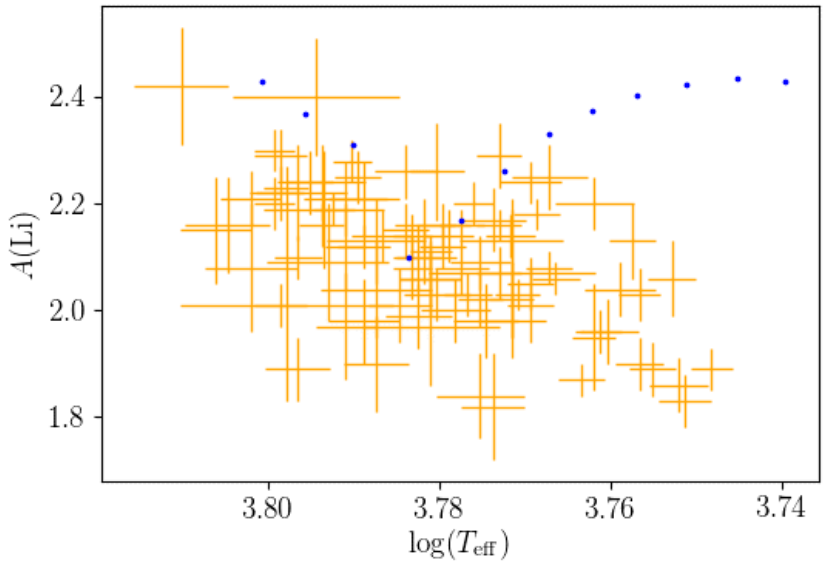

Figure 4.1: Lithium abundance A(Li) as a function of effective temperature $T_{\rm eff}$ for a population of stars ranging from $0.60\,M_\odot$ to $0.71\,M_\odot$ in $0.01\,M_\odot$ increments (blue dots). All models featured $\Omega_{\rm initial}/\Omega_{\rm crit} = 5 \times 10^{-5}$ and $Z_{\rm initial} = 1 \times 10^{-4}$. Overlaid (orange points with error bars) is the data from Ryan et al. (1996), Figure 1.1 (Bottom).

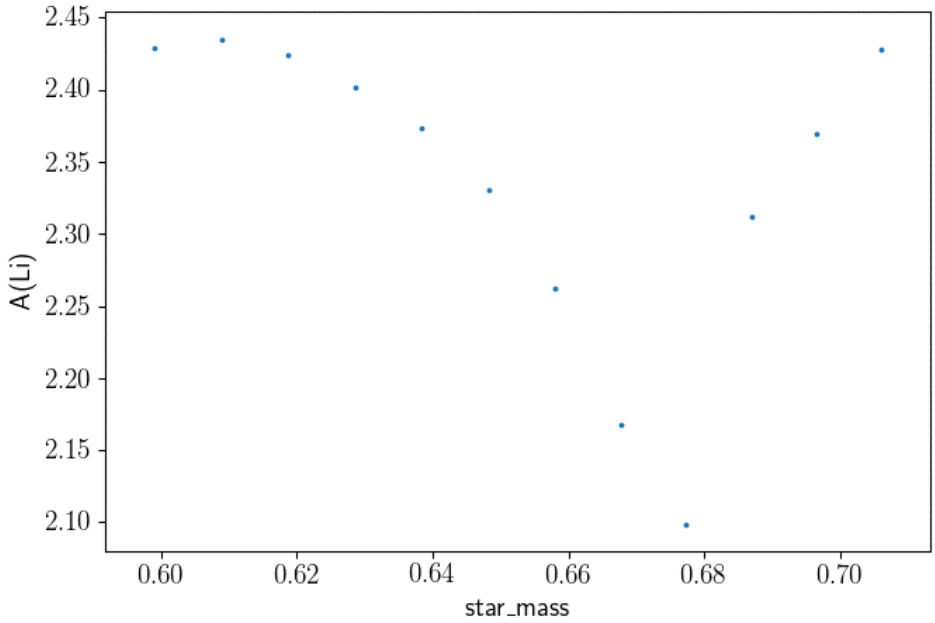

Figure 4.2: A(Li) as a function of initial mass in solar units, for the same data as in Figure 4.1.

4.2.2 Variable termination condition

We stated in §3.3 that all stars were terminated at $\log_{10}(\rho_c) = 3.3$ regardless of initial mass. This will provide us with a star that has certainly reached the MSTO, regardless of the mass considered, within the range of masses considered by this project. In principle, we could introduce a variable condition in which the density limit is smaller for lower-mass stars; this would slightly boost efficiency by spending less time in unneeded evolutionary phases, but in practice the evolution finished quickly enough (of order one minute per star) for such a modification not to be needed. The time saved in execution would not be worth the extra development time and added `inlist` complexity.

4.2.3 Extraction of properties at a given age

We stated in §3.3 that stellar properties at 13 Gyr were determined by selecting the first model with age higher than this value. This introduces the potential of "age overshoot", in which all chosen stars are actually older because the probability of a model being taken at *exactly* 13 Gyr is infinitesimal.

If more accuracy is required, the analysis could be extended with two possible methods. 1) Enforcing a shorter maximum timestep[2] to guarantee that a model is recorded within, say, 0.1 Gyr of the target age. This necessarily increases the number of models required to evolve a star to a given point and increases code runtime. 2) Interpolation between the two profiles straddling the target age. In a particular, linear interpolation is very easy to implement but it implicitly assumes that all recorded properties ($T_{\rm eff}$, $v_{\rm rot}$, surface abundances etc) scale roughly linearly with time between these points; this may not be true.

In our simulations, the overshoot was of order 0.2 Gyr, small enough that we did not observe qualitative changes in the star's properties compared to the values it *would* have had at 13 Gyr. However, if the analysis progressed to an attempted replication of the Spite plateau, linear interpolation would have been implemented in an attempt to improve accuracy. The provided analysis code can be easily extended to include it if necessary.

4.2.4 Choice of a single `inlist`

In §3.4, we explained the decision to evolve the star with a single `inlist` instead of stopping once the model reaches the MS and enforcing the required properties. Part of the reasoning was that magnetic braking should yield the same MS rotation rate regardless of our choice of $\Omega_{\rm initial}/\Omega_{\rm crit}$, but simulations showed that magnetic braking only takes effect once the rotation rate is already unrealistically high. Thus, our models are in fact sensitive to $\Omega_{\rm initial}/\Omega_{\rm crit}$, and from Figure 3.7, the this strongly affects the value of A(Li).

In future runs, it is vital to ensure that a value of $\Omega_{\rm initial}/\Omega_{\rm crit}$ is decided upon, and no deviations are made from this choice. If it is deemed necessary to specify $\Omega_{\rm MS}/\Omega_{\rm crit}$, the `inlist` will need to be transferred to a two-stage code. This will require a reworking of the file structure of the working directory provided and an extension of the automated inlist builder, but is not beyond the realms of feasibility should it be required.

4.3 Code and data availability

All `Python` code and all `inlists` produced as part of this project were written entirely by the author of this report. They have been made freely available in a `GitHub` repository:

https://github.com/DanTickner/MESA-analysis-tools/releases/tag/MESA

Please note that, in respect of the hard deadline necessitated by an MSc report, the final commit to the repository was made before 23:59 BST on Sunday 12 September 2021. This can be verified by mousing over the text `DanTickner released this (time) ago`, which reveals the final upload time `12 Sept 2021, 22:59 BST`. The repository will not be adjusted in any way after this time, except for the future inclusion of this report once grades have been finalised. In this respect its contents may be judged as part of the project itself, while still respecting the submission deadline.

[2] `max_years_for_timestep` in the `controls` section of `inlist`.

A Appendix: observational data from Ryan et al.

Table A.1: Observational data from 93 old halo stars within the Milky Way galaxy. Taken from Table 3 of Ryan et al. Ryan et al. (1996), with logarithm of effective temperature $\log(T_{\text{eff}})$ and associated uncertainty $\sigma(\log(T_{\text{eff}}))$ calculated as part of the analysis in this paper. The data is available in the file `Data-Ryan+1996.csv` in the `GitHub` repository associated with this paper.

Enumeration	Metallicity [Fe/H]	Effective temperature T_{eff}	Eff. temp. uncertainty $\sigma(T_{\text{eff}})$	Lithium abundance $A(\text{Li})$	Li. abund. uncertainty $\sigma(A(\text{Li}))$	Logarithm eff. temp. $\log_{10}(T_{\text{eff}})$	Log. eff. temp. uncertainty $\sigma(\log_{10}(T_{\text{eff}}))$
1	−2.4	5740	55	2.04	0.05	3.759	0.004
2	−2.6	5840	38	2.06	0.03	3.766	0.003
3	−2.3	5990	44	2.08	0.05	3.777	0.003
4	−2.9	6340	52	2.21	0.05	3.802	0.004
5	−1.8	5780	34	1.95	0.10	3.762	0.003
6	−1.2	6030	44	2.26	0.09	3.780	0.003
7	−2.3	6080	52	2.26	0.05	3.784	0.004
8	−2.9	6260	40	2.10	0.04	3.797	0.003
9	−2.0	5980	38	2.03	0.04	3.777	0.003
10	−2.7	6290	40	2.22	0.05	3.799	0.003
11	−2.3	6080	38	2.15	0.05	3.784	0.003
12	−1.6	5960	36	2.02	0.05	3.775	0.003
13	−2.6	6300	34	2.30	0.04	3.799	0.002
14	−2.7	6020	44	2.16	0.04	3.780	0.003
15	−2.8	6090	51	2.01	0.07	3.785	0.004
16	−2.4	6150	44	2.12	0.04	3.789	0.003
17	−2.2	6070	32	2.14	0.04	3.783	0.002
18	−1.8	5720	36	2.13	0.07	3.757	0.003
19	−2.3	6220	38	2.16	0.04	3.794	0.003
20	−2.6	6050	51	2.10	0.05	3.782	0.004
21	−2.2	6120	50	2.14	0.05	3.787	0.004
22	−1.5	5880	36	2.01	0.07	3.769	0.003
23	−2.1	5850	59	2.25	0.06	3.767	0.004
24	−2.0	6050	49	2.16	0.05	3.782	0.004
25	−2.2	6170	32	2.28	0.04	3.790	0.002
26	−1.6	6460	82	2.42	0.11	3.810	0.006
27	−2.4	6200	34	2.19	0.03	3.792	0.002
28	−2.2	5940	52	2.17	0.06	3.774	0.004
29	−2.5	6040	34	2.11	0.03	3.781	0.002
30	−3.1	6180	34	2.12	0.08	3.791	0.002
31	−3.7	6060	54	1.99	0.06	3.782	0.004
32	−2.0	5660	36	2.06	0.07	3.753	0.003
33	−3.3	6260	54	1.89	0.06	3.797	0.004
34	−2.7	6220	73	2.24	0.07	3.794	0.005
35	−2.9	6130	82	2.13	0.08	3.787	0.006

Table A.1: Continued.

Enumeration	Metallicity [Fe/H]	Effective temperature $T_{\rm eff}$	Eff. temp. uncertainty $\sigma(T_{\rm eff})$	Lithium abundance A(Li)	Li. abund. uncertainty $\sigma(A(\mathrm{Li}))$	Logarithm eff. temp. $\log_{10}(T_{\rm eff})$	Log. eff. temp. uncertainty $\sigma(\log_{10}(T_{\rm eff}))$
36	−2.6	5910	54	1.98	0.07	3.772	0.004
37	−2.6	6130	54	1.90	0.09	3.787	0.004
38	−2.7	6300	34	2.20	0.05	3.799	0.002
39	−2.8	6260	82	2.22	0.09	3.797	0.006
40	−2.5	6150	59	2.21	0.07	3.789	0.004
41	−2.5	6150	59	1.98	0.08	3.789	0.004
42	−3.2	6380	73	2.16	0.09	3.805	0.005
43	−3.4	6030	54	2.03	0.05	3.780	0.004
44	−2.5	6040	112	1.97	0.11	3.781	0.008
45	−2.5	5910	82	2.13	0.08	3.772	0.006
46	−3.1	5940	50	1.82	0.10	3.774	0.004
47	−2.9	6400	62	2.15	0.10	3.806	0.004
48	−3.3	6280	180	2.01	0.18	3.798	0.012
49	−3.3	6340	79	2.08	0.12	3.802	0.005
50	−3.3	6180	50	1.97	0.10	3.791	0.004
51	−3.1	6217	100	2.19	0.11	3.794	0.007
52	−2.8	6130	92	2.04	0.09	3.787	0.007
53	−2.4	5930	36	2.07	0.05	3.773	0.003
54	−1.3	5912	40	2.14	0.04	3.772	0.003
55	−2.1	6210	103	2.09	0.11	3.793	0.007
56	−2.6	6230	140	2.40	0.11	3.794	0.010
57	−1.6	5960	49	2.09	0.05	3.775	0.004
58	−2.4	5640	40	1.83	0.05	3.751	0.003
59	−2.6	5970	32	2.20	0.04	3.776	0.002
60	−3.5	6290	44	2.29	0.05	3.799	0.003
61	−3.0	6290	44	2.01	0.04	3.799	0.003
62	−2.7	5690	36	1.89	0.05	3.755	0.003
63	−2.4	5950	52	1.98	0.07	3.775	0.004
64	−2.7	6010	40	2.14	0.05	3.779	0.003
65	−2.2	5880	49	2.24	0.04	3.769	0.004
66	−2.2	5710	32	2.03	0.05	3.757	0.002
67	−2.8	5600	32	1.89	0.04	3.748	0.002
68	−2.5	6240	40	2.24	0.06	3.795	0.003
69	−2.7	5650	44	1.86	0.05	3.752	0.003
70	−2.8	5910	133	2.07	0.12	3.772	0.010
71	−2.5	6040	50	2.06	0.07	3.781	0.004
72	−2.7	5960	70	1.84	0.08	3.775	0.005
73	−2.7	6000	56	2.00	0.06	3.778	0.004
74	−2.7	6040	50	2.06	0.05	3.781	0.004
75	−2.9	6180	69	2.21	0.07	3.791	0.005
76	−3.0	5770	32	1.96	0.04	3.761	0.002
77	−2.5	6070	32	2.08	0.06	3.783	0.002
78	−2.6	5760	50	1.96	0.06	3.760	0.004
79	−3.5	5910	36	2.02	0.07	3.772	0.003
80	−1.2	5930	36	2.29	0.06	3.773	0.003

Table A.1: Continued.

Enumeration	Metallicity [Fe/H]	Effective temperature $T_{\rm eff}$	Eff. temp. uncertainty $\sigma(T_{\rm eff})$	Lithium abundance A(Li)	Li. abund. uncertainty $\sigma(A(\mathrm{Li}))$	Logarithm eff. temp. $\log_{10}(T_{\rm eff})$	Log. eff. temp. uncertainty $\sigma(\log_{10}(T_{\rm eff}))$
81	−2.2	6020	38	2.12	0.04	3.780	0.003
82	−2.1	5850	36	2.08	0.03	3.767	0.003
83	−2.2	5780	61	2.20	0.05	3.762	0.005
84	−1.8	6060	36	2.10	0.06	3.782	0.003
85	−2.1	6280	38	2.23	0.04	3.798	0.003
86	−1.7	5930	36	2.16	0.03	3.773	0.003
87	−1.8	5990	36	2.16	0.03	3.777	0.003
88	−2.4	5880	36	2.05	0.05	3.769	0.003
89	−1.2	5710	36	1.90	0.05	3.757	0.003
90	−1.6	5900	36	2.03	0.03	3.771	0.003
91	−1.4	5800	36	1.87	0.03	3.763	0.003
92	−2.0	6160	38	2.25	0.05	3.790	0.003
93	−1.4	5870	36	2.18	0.03	3.769	0.003

Bibliography

Cassisi S., Salaris M., 2013, Old Stellar Populations: How to Study the Fossil Record of Galaxy Formation. Wiley-VCH, Weinheim, DE, https://ui.adsabs.harvard.edu/abs/2013osp..book.....C

Grevesse N., Sauval A. J., 1998, Space Sci. Rev., 85, 161

Heger A., Langer N., 2000, ApJ, 544, 1016

Heger A., Langer N., Woosley S. E., 2000, ApJ, 528, 368

Linsky J. L., et al., 2006, ApJ, 647, 1106

Paxton B., Bildsten L., Dotter A., Herwig F., Lesaffre P., Timmes F., 2011, ApJS, 192, 3

Paxton B., et al., 2013, ApJS, 208, 4

Paxton B., et al., 2015, ApJS, 220, 15

Paxton B., et al., 2018, ApJS, 234, 34

Paxton B., et al., 2019, ApJS, 243, 10

Pinsonneault M. H., Kawaler S. D., Sofia S., Demarque P., 1989, ApJ, 338, 424

Porcelli D., Ballentine C. J., 2002, Rev. Mineral Geochem., 47, 411

Ryan S. G., Norton A. J., 2010, Stellar Evolution and Nucleosynthesis. Cambridge University Press, Cambridge, GB, https://ui.adsabs.harvard.edu/abs/2010sen..book.....R

Ryan S. G., Beers T. C., Deliyannis C. P., Thorburn J. A., 1996, ApJ, 458, 543

Salaris M., Cassisi S., 2005, Evolution of Stars and Stellar Populations. John Wiley & Sons, West Sussex, GB, https://ui.adsabs.harvard.edu/abs/2005essp.book.....S

Salaris M., Cassisi S., 2017, R. Soc. Open Sci., 4, 170192

Salaris M., Groenewegen M. A. T., Weiss A., 2000, A&A, 355, 299

Spite F., Spite M., 1982, A&A, 115, 357

Stauffer J. B., Hartmann L. W., 1986, Publ. Astron. Soc. Pac., 98, 1233

Swenson F. J., 1995, ApJL, 438, L87

Vick M., Michaud G., Richer J., Richard O., 2013, A&A, 552, A131

Wichers E., Peiser H. S., 2020, Encyclopedia Britannica,

de Laeter J. R., Böhlke J. K., De Bièvre P., Hidaka H., Peiser H. S. amd Rosman K. J. R., Taylor P. D. P., 2003, Pure Appl. Chem., 75, 683