Stars for Schools
2022

The following document is our "Stars for Schools" course. Inside, we will introduce you to the physics at work in stars. Stars are very large – the radius of the Sun is 100 times that of the Earth – but their structure and evolution rely on microscopic quantum, particle and nuclear physics. You will learn to make your own models of stars using the *Window to the Stars* software, and you will gain valuable experience of analysing the data you generate to understand the astrophysics at work in stars. Part I introduces the ideas of stellar astrophysics and computer modelling of stars. Part II contains ideas and exercises to help you understand the astrophysics of stars and suggests concepts for independent project work.

Robert Izzard, Rahul Kakaiya, Sabana Ghale, Natalie Rees, Arman Aryaeipour and Andy Brittain

University of Surrey, United Kingdom

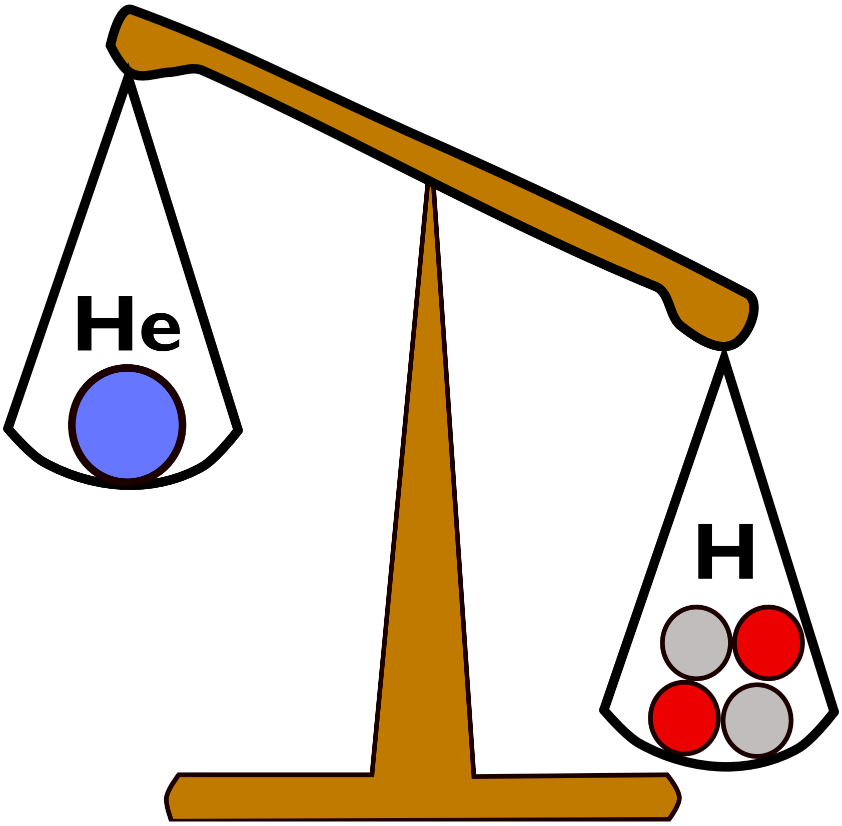
email: r.izzard@surrey.ac.uk

UNIVERSITY OF SURREY

Contents

Part I

Stellar astrophysics and modelling of stars



1 *What are stars?*

What are the stars? When you look up in the sky on a clear night, what do you see? Hundreds, perhaps thousands, of bright points, surrounded by the black of space. These are the stars – but what *are* they? This question perplexed humans for most of recorded history. What are stars? How did they form? What are they made of? How do they shine?

Have a think about the above questions before we start this course. Do you know how stars work – or have you taken them for granted? Stars were the tools that, for millennia, allowed humans to navigate the seas. They inspired art, music, literature and poetry. In the last two centuries, and continuing now, they are the best laboratories we have for testing our theories of nuclear and particle physics, quantum mechanics, special and general relativity, and plasma physics. We need to understand how they are born, live, interact and die to understand the basic physical processes. Understanding basic science is key to new ideas, innovation and technology, and this course will hopefully provide you with some understanding of fundamental science in the most extreme laboratories in the Universe: the centres of the stars.

More information...

- https://www.universetoday.com/25156/history-of-stars/
- https://www.nationalgeographic.com/science/space/universe/stars/
- https://science.nasa.gov/astrophysics/focus-areas/how-do-stars-form-and-evolve
- https://en.wikipedia.org/wiki/Star
- https://www.naturalnavigator.com/find-your-way-using/stars/
- https://www.wikihow.com/Navigate-by-the-Stars

1.1 *The chemical composition of stars*

One might think that the matter that makes up stars has a similar elemental composition to the Earth, which is about 50% oxygen, 28% silicon, 8% aluminium and 5% iron (Lutgens & Tarbuck, 1989). Or, perhaps, you think stars should be like the Earth's atmosphere, 78% nitrogen and 21% oxygen (Minzner, 1977). How would you check such assumptions? This problem existed for most of human history. We cannot physically go to the stars, but they do send one thing to us: light. This light can be investigated by *spectroscopy*.

Light can be split into its constituent colours, called a *spectrum*, using a prism. If one looks closely at this spectrum, it contains lines. These correspond to the chemical elements in the material that emits the light. In the laboratory or at home, you see *emission lines* in gas-discharge lamps, such as the orange sodium lamps used as street lamps. Perhaps less obviously, gas can also absorb light leading to dark absorption lines in a spectrum. When looking at the spectrum of the Sun, such lines are visible throughout (Fig. 2), and by comparing the colour, or wavelength, of the lines to those emitted from hot gases in the laboratory, we can tell what the Sun is made of. We can do similarly with the light from other stars, but of course it is more difficult because they are further away.

It was not until the doctoral thesis of Cecilia Payne, in 1925, that we finally understood that stars are mostly made of hydrogen and helium, the two lightest elements (hydrogen contains one proton, helium contains two protons and two neutrons). She showed that the relative amounts of

Figure 1: Young stars in the LH95 star-forming region of the Large Magellanic Cloud, a dwarf galaxy close to our Milky Way (ESA/Hubble). This image was taken by the Hubble Space Telescope.

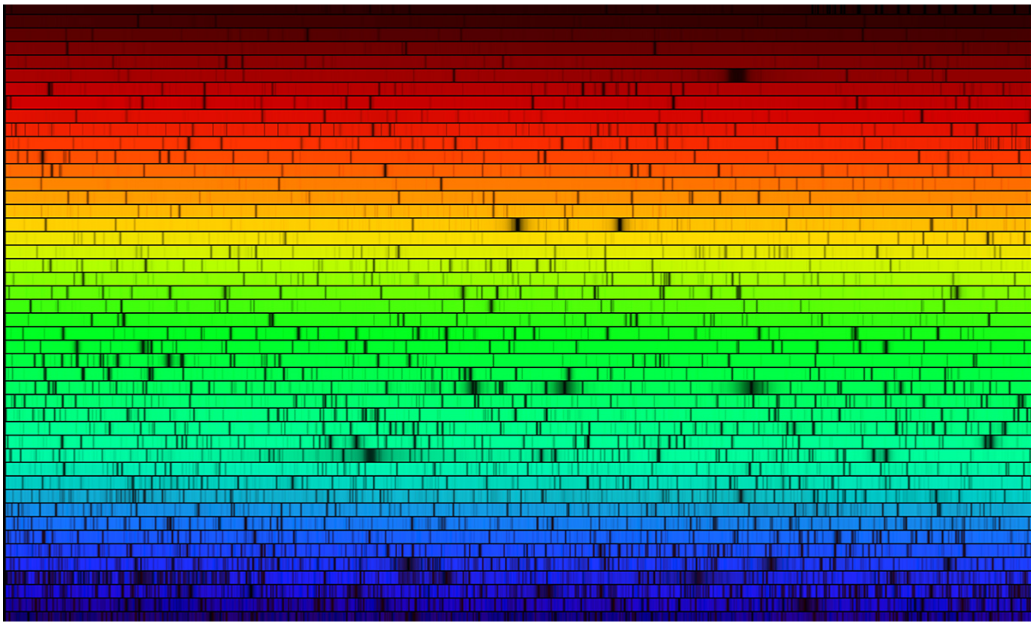


Figure 2: High-resolution Solar spectrum in the wavelength range 4000 to 7000 Å, where 1 Å = 10^{-10} m. Note the many dark absorption lines. By comparing their wavelength to spectra of gases in the laboratory, we can calculate the composition of the Sun. Taken from https://solarsystem.nasa.gov/resources/390/the-solar-spectrum/ (N.A. Sharp, NOAO/NSO/Kitt Peak FTS/AURA/NSF).

heavier elements, like carbon and silicon, were similar to those found on Earth, but that the total amount of these "metals"[1] is a little more than 1%, by mass, in the Sun (Payne, 1925).

More information...

- https://en.wikipedia.org/wiki/Spectroscopy
- https://en.wikipedia.org/wiki/Gas-discharge_lamp
- https://en.wikipedia.org/wiki/Sunlight
- https://solarsystem.nasa.gov/resources/390/the-solar-spectrum/
- https://www.youtube.com/watch?v=7v9AUnnrQ5w

1.2 Timescales in stars

It helps greatly to understand the basic timescales on which stars evolve and die. We use the Sun as an example, which has radius $R = \mathrm{R}_\odot = 6.957 \times 10^8\,\mathrm{m}$, mass $M = \mathrm{M}_\odot = 1.989 \times 10^{30}\,\mathrm{kg}$ and luminosity $L = \mathrm{L}_\odot = 3.85 \times 10^{26}\,\mathrm{J\,s^{-1}}$, where the $\odot$ symbol means "the Sun". Remember that the gravitational constant is $G = 6.67408 \times 10^{-11}\,\mathrm{m^3\,kg^{-1}\,s^{-2}}$.

The dynamical timescale is the time scale on which material can move around in a star. Remember that, subject to a constant acceleration a and with a stationary start, material moves a distance,

$$s = \frac{1}{2}at^2 , \tag{1.1}$$

in a time t. In a star, a typical distance is the stellar radius, so we set $s = R$, and the acceleration due to gravity is GM/R^2. The square of the timescale is then,

$$t^2 = 2\frac{s}{a} = 2 \times R \times \frac{R^2}{GM} = \frac{2R^3}{GM} , \tag{1.2}$$

hence, ignoring the factor $\sqrt{2} = 1.41 \cdots \approx 1$, and taking square roots to find the timescale,

$$t \approx \sqrt{\frac{R^3}{GM}} . \tag{1.3}$$

This is the *dynamical* timescale, also called the "free-fall" timescale because it is the time it takes to free fall, under constant gravitational acceleration. In the Sun, this time is about

$$\tau_{\mathrm{dyn},\odot} = \sqrt{\frac{\mathrm{R}_\odot^3}{G\mathrm{M}_\odot}} \approx 600\,\mathrm{s} , \tag{1.4}$$

which is 10 minutes. This is the time the Sun takes to react to interior and surface movements, and its core vibrates on this timescale in reaction to the ejection of solar flares, for example.

[1] In astronomy, anything that is not hydrogen or helium is called a "metal". Indeed, the "metallicity" of a star is the fractional amount of mass in the star that is not hydrogen or helium. We call the mass fraction of hydrogen X, the mass fraction of helium Y and everything else, the metallicity, Z, such that $X + Y + Z = 1$.

The thermal timescale, also called the *Kelvin-Helmholtz timescale*, is the time it takes a star to radiate its energy into space, assuming it only contains thermal energy. The thermal energy in a star, in the absence of any other energy source, comes from the collapse of its parent gas cloud which releases the gravitational energy as heat. Roughly, the gravitational energy, hence thermal energy, is,

$$E_{\mathrm{th}} = E_{\mathrm{grav}} = mgh = M \times \frac{GM}{R^2} \times R \approx \frac{GM^2}{R}, \tag{1.5}$$

and the star radiates, i.e. loses energy, at a luminosity L. The thermal timescale is then

$$\tau_{\mathrm{th}} = \frac{\text{thermal energy}}{\text{energy loss rate}} = \frac{E_{\mathrm{th}}}{L} \approx \frac{GM^2}{2RL}, \tag{1.6}$$

where the factor 2 appears if we perform a more careful analysis, e.g. https://en.wikipedia.org/wiki/Kelvin%E2%80%93Helmholtz_mechanism. In the Sun, this timescale is about 3×10^7 years, i.e.,

$$\tau_{\mathrm{th},\odot} = \frac{\mathrm{GM_\odot}^2}{2\mathrm{R_\odot L_\odot}} \approx 30\,\mathrm{Myr}. \tag{1.7}$$

This is the lifetime the Sun would have if no other sources of energy were available than the energy from gravitational collapse.

The nuclear timescale is the time it takes a star to exhaust its nuclear fuel. To a very good approximation, we can consider only hydrogen burning which is the simplest nuclear burning cycle and which powers both the Sun and most stars. The following reaction converts four hydrogen nuclei, or protons, to one helium nucleus,

$$4\,{}^1_1\mathrm{H} \rightarrow {}^4_2\mathrm{He}. \tag{1.8}$$

Four protons have a mass,

$$4m_{\mathrm{p}} = 4 \times 1.6726 \times 10^{-27}\,\mathrm{kg} = 6.6904 \times 10^{-27}\,\mathrm{kg}, \tag{1.9}$$

while a helium nucleus has a mass

$$m_{\mathrm{He}} = 6.64477 \times 10^{-27}\,\mathrm{kg}. \tag{1.10}$$

The difference, called the *mass defect*, is,

$$4m_{\mathrm{p}} - m_{\mathrm{He}} = 4.563^{-29}\,\mathrm{kg}, \tag{1.11}$$

as was shown by Francis Aston using a mass spectrograph, and for which a Nobel prize was awarded in 1922.

We can use Einstein's famous equation, $E = mc^2$, to convert this to the energy released by nuclear fusion of hydrogen to helium,

$$\Delta E = (4m_{\mathrm{p}} - m_{\mathrm{He}})\,c^2 = 4.563^{-29}\,\mathrm{kg} \times \left(299792458\,\mathrm{m\,s^{-1}}\right)^2 = 4.1 \times 10^{-12}\,\mathrm{J}, \tag{1.12}$$

where Δ means "change in". This is not much energy, but remember the mass of a star is huge so this happens many times every second in the core of a star.
We can calculate the *burning efficiency* of hydrogen fusion,

$$\frac{\Delta E}{E} = \frac{\left(4 \times 1.6726 \times 10^{-27}\,\mathrm{kg} - 6.64477 \times 10^{-27}\,\mathrm{kg}\right) c^2}{\left(4 \times 1.6726 \times 10^{-27}\,\mathrm{kg}\right) c^2} \approx 0.007 = 0.7\%. \quad (1.13)$$

This means that when the hydrogen in a star burns to helium, it loses 0.7% of its mass to energy. The total energy available in the Sun is thus,

$$E_{\odot} = 0.007 \times f \times \mathrm{M}_{\odot} c^2, \quad (1.14)$$

where $f \approx 0.1$ is the mass fraction of the Sun that burns from hydrogen to helium in its lifetime. Given the rate at which this is being lost is about the Solar luminosity, $\mathrm{L}_{\odot} = 3.85 \times 10^{26}\,\mathrm{J\,s^{-1}}$, the *nuclear timescale* is then the time it takes to lose all this energy at this rate,

$$\tau_{\mathrm{nuc}} = \frac{E_{\odot}}{\mathrm{L}_{\odot}} = \frac{0.007 \times f \times \mathrm{M}_{\odot} c^2}{\mathrm{L}_{\odot}} = \quad (1.15)$$

$$\frac{0.007 \times 0.1 \times 1.989 \times 10^{30}\,\mathrm{kg} \times \left(299792458\,\mathrm{m\,s^{-1}}\right)^2}{3.85 \times 10^{26}\,\mathrm{J\,s^{-1}}} = 3.25 \times 10^{17}\,\mathrm{s}. \quad (1.16)$$

We can then use the approximate rule[2] that $1\,\mathrm{yr} \approx \pi \times 10^7\,\mathrm{s}$, and then that $3.25/\pi \approx 1$, so

$$\tau_{\mathrm{nuc}} \approx 10^{10}\,\mathrm{yr}\,. \quad (1.17)$$

So, the Sun has the following timescales, from short to long,

1. Dynamical $\tau_{\mathrm{dyn},\odot} = 10\,\mathrm{mins}$,
2. Thermal $\tau_{\mathrm{th},\odot} \approx 30\,\mathrm{Myr}$, and,
3. Nuclear $\tau_{\mathrm{nuc},\odot} = 10^{10}\,\mathrm{yr}$.

These are *very* different timescales. They mean that the Sun reaches a dynamically stable, "dynamical equilibrium" state within a few minutes, so it is now in dynamical equilibrium. Similarly, the Sun burns its nuclear fuel for about ten billion years before it runs out. This is a very long time: about twice the estimated age of the Earth. Because the Sun has not reached an equilibrium state, it is *still* burning nuclear fuel at its core. This is fortunate for us: without nuclear burning in the Sun, the Earth would be a frozen planet and humans would not exist.

- Exercise: How much hydrogen is converted to helium in the Sun every second?

More information...

- Timescales http://www.mit.edu/~iancross/8901_2019A/lec011.pdf
- Mass defect https://www.youtube.com/watch?v=fVilEOSBII8
- Age of Earth https://en.wikipedia.org/wiki/Geological_history_of_Earth
- Fusion https://www.youtube.com/watch?v=7E-0j90Cwpk

[2] Actually, one year is 3.155324×10^7 s, which means this approximation is accurate to about 0.4% and is very easy to remember.

2 Modelling stars on your PC

A personal computer, or PC, is not just for games. It can be used to make models of physical objects, which in this course are stars. You could also use your PC to model fluid flow, for example around aeroplanes or rockets, or electrical circuits, or the stability of buildings. Modern PCs are amazingly powerful and in this course we will be using just your PC, or even a small PC like a *Raspberry Pi*, to make professional-quality models of whole stars. First, we will look at how we model physics on computers, then we will look at how we model stars.

2.1 How do we model physical processes on a computer?

You have studied physics for some time now, so you know you use mathematics to model physical processes. The real question then is how is mathematics modelled on a computer. This is a big topic, so we'll just scratch its surface, but you can try this yourself with a simple programming language like *Python*.

Let's start with a simple model to generate a sequence of numbers called the Fibonacci series. These are 0, 1, 1, 2, 3, 5, 8, ... How is one number related to the previous number? It's actually related to the previous *two* numbers: $0+1=1$, $1+1=2$, $1+2=3$, $2+3=5$ and so on. So how will we calculate this on a computer? We use a programming language, in this case *Python*[3].

Numbers are stored in Python's *variables*. These are just pieces of the computer's memory that contain a number. You can think of them as pieces of paper with numbers written on them if you like. We then store the first two numbers of the series, 0 and 1, in two variables which we name `n1` and `n2`. We could then write code like the following to calculate the first five Fibonacci numbers, storing the next three in `n3`, `n4` and `n5`. The `print(n5)` shows the result, stored in `n5`, on the screen.

```
n1=0
n2=1
n3=n1+n2
n4=n2+n3
n5=n3+n4
print(n5)
```

But what if we want, say the 100th Fibonacci number? We could defined `n6`, `n7` etc. but this would take a long time to write and, because the code will be long, it will likely contain bugs that make it crash[4].

We can instead use a "while" loop to automate the process so we do not have to define 100 variables. At each step in the while loop, the next value in the series is calculated in `next`, then `n1` and `n2` are updated as required by the Fibonacci's rule. The `count` is updated so that when it hits `maxcount`, which means `count < maxcount` is false, the `while` loops stops. We set `maxcount=100` as required.

```
n1=0
n2=1
count=0
```

[3]You can try Python online at https://www.programiz.com/python-programming/online-compiler/ or install it on your PC for free https://www.python.org/downloads/.

[4]A "crash" is when a code stops for reasons that were not predicted. No doubt you have seen this: in Windows it is called the "blue screen of death" and even simpler devices, like mobile phones, crash from time to time.

```
maxcount=100
print(n1)
print(n2)
while count < maxcount:
    next = n1 + n2
    n1 = n2
    n2 = next
    count = count + 1
    print(next)
```

This is a simple example, you can see more advanced forms of the above at https://www.programiz.com/python-programming/examples/fibonacci-sequence and https://www.geeksforgeeks.org/python-program-for-program-for-fibonacci-numbers-2/. However, the basic ideas shown above are exactly what we do when modelling any physical system on a computer, including the evolution of a star. We know the state of the system at some time t, say $t = 0\,\mathrm{s}$, and want to update it to some later time, for example at $t = 60\,\mathrm{s}$. We have some constraints on how the system behaves, such as the physical laws of the system like Newton's laws or the laws of thermodynamics, so the system's properties and their rates of change are predictable using variables which can be stored and changed by your program (your "code"). This is exactly what happens when stars are modelled on a computer.

More information...

- https://repl.it/languages/python3

2.2 *What is stellar modelling?*

The standard computational tool of anyone interested in understanding stars is a stellar evolution code — a piece of software that can construct a model of the interior of a star, and then evolve it over time. Evolution codes allow us to check and refine the various physical theories that together compose stellar astrophysics (e.g., atomic physics, nuclear physics, fluid dynamics, thermodynamics); they provide laboratories for performing experiments on stars (e.g. discovering what factors contribute to the formation of red giants); and, they shed light on stages of stellar evolution that may be too fleeting to observe directly in the Universe.

2.2.1 Star to shells: counting mass or radius

Stars are spheres to a very good approximation, for example our Sun is spherical to an accuracy of 99.9997. The slight deformation is caused by its rotation, but this is slow and can be ignored in most stars, so we will also ignore it.

To model a spherical star, we divide the star into spherical "shells" and then stack these shells from the centre to the surface (Fig. 3). Each spherical shell has a radius, r, which is its distance from the centre of the star (because the shell is a sphere, all points in the shell are the same distance from the centre). Also, each shell has a mass co-ordinate, m_r, which is the mass *interior* to the shell[5]. These shells are filled with gas plasma: they are not solid like egg shells! This means they

[5] This means that, in a $1\,\mathrm{M}_\odot$ star, with the same mass as the Sun, the shell with $m_r = 0.5\,\mathrm{M}_\odot$ has half the total mass inside it and half outside it. The shell with $m_r = 0.9\,\mathrm{M}_\odot$ has 90% of the mass inside it and 10% outside it.

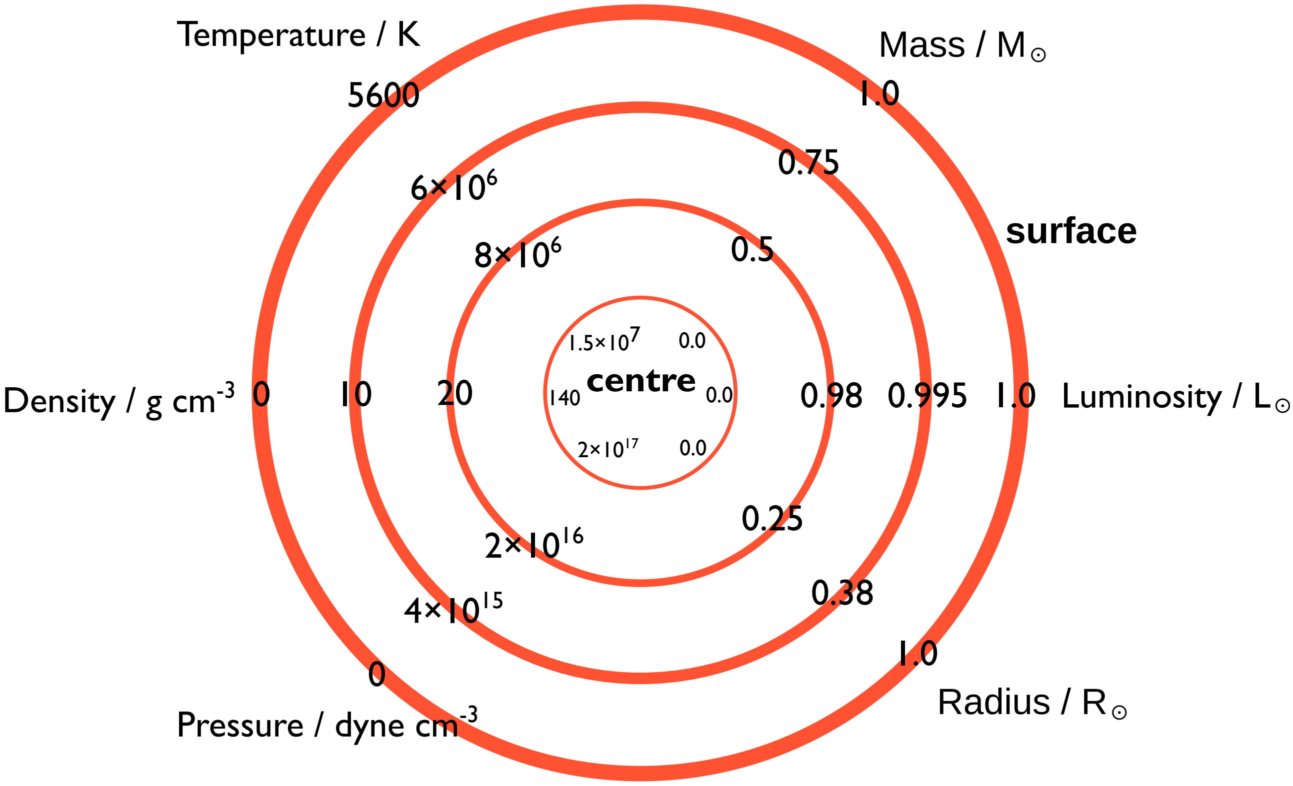


Figure 3: Shell structure in a stellar model, with the shells at 0 $M_\odot$ (numbers at the centre), 0.25 $M_\odot$ (no numbers), 0.5 $M_\odot$, 0.75 $M_\odot$ and 1 $M_\odot$ (the surface). At each shell, the temperature, mass, luminosity, radius, pressure and density are marked in units of K, $M_\odot$, $L_\odot$, $R_\odot$, dyne (10^{-5}N) and g cm^{-3} respectively. The numbers are taken from a real stellar model of the Sun made with *Window to the Stars*. Note that most of the luminosity is generated in the centre where the temperature and density are highest. The surface contains all the mass at 1 $M_\odot$, and has no pressure or density because it is the vacuum of space. Our real stellar models, as made by *Window to the Stars*, contain hundreds of shells, not just the four shown here.

can expand and contract if necessary, and they will do this during the life of a star. Remember: conditions like the temperature, pressure and chemical composition are the same everywhere in a shell, be it on the polar axis or in the equatorial plane.

- At the centre, we have $r = 0$ cm and $m_r = 0$ kg.
- At the surface we have $r = R$ and $m_r = M$ where R and M are the total mass radius and mass of the star, respectively. In the Sun, we have $M = M_\odot$ and $R = R_\odot$.

The volume inside a shell is calculated from the formula for the volume, V_r, of a sphere,

$$V_r = \frac{4}{3}\pi r^3 . \quad (2.1)$$

You can calculate the average density, ρ, inside any shell from,

$$\rho = \frac{m_r}{V_r} . \quad (2.2)$$

- Exercise: show that the average density inside a shell at radius r and mass m_r is $\rho = 3m_r/4\pi r^3$. What then is the average density of the Sun, given that its mass is $M_\odot = 1.989 \times 10^{30}$ kg and its radius is $R_\odot = 6.957 \times 10^8$ m?

2.2.2 Pressure and temperature

The gas inside the Sun is, to a very good approximation, an ideal gas. In any small volume in the star, including in each spherical shell, the pressure, P, and temperature, T, are related by the ideal gas law,

$$PV = n\mathcal{R}T\,, \tag{2.3}$$

where V is the volume of the shell that contains n mol of gas[6], and $\mathcal{R} = 8.314\,\mathrm{J\,K^{-1}\,mol^{-1}}$ is the gas constant. We can divide Eq. 2.3 by V,

$$P = \frac{n}{V}\mathcal{R}T\,, \tag{2.4}$$

and multiply the right hand side by $\mu/\mu = 1$, where μ is the mass per particle, to obtain,

$$P = \frac{\mu n}{V}\frac{\mathcal{R}T}{\mu}\,. \tag{2.5}$$

The first fraction is just the density, because μn is the mass of all the particles in the volume V, so,

$$\frac{\mu n}{V} = \rho\,, \tag{2.6}$$

and hence we can rewrite the ideal gas law in a more useful form that relates the pressure to the density and temperature,

$$P = \frac{\mathcal{R}}{\mu}\rho T\,. \tag{2.7}$$

The factor $\mathcal{R}/\mu$ is roughly constant in a star (we will investigate this later). This equation is the form we will use below. Note that if we double the density or the temperature, we double the pressure. This simple rule governs the fundamental structure of stars.

- Exercise: do you think m_r would be related to V_r by a straight line, that is, do you expect the density of the Sun to be constant?
- Exercise: what is the relation between mass in a shell, δm, and its thickness, δr?
- Exercise: how do you know the density increases as we go deeper into the Sun?

[6] Remember, 1 mol of gas is $N_A = 6.02214076 \times 10^{23}$ gas particles (https://en.wikipedia.org/wiki/Mole_(unit)).

2.2.3 Hydrostatic equilibrium

Stars are usually in what is called *hydrostatic equilibrium*. Hydrostatic comes from two words: *hydro* meaning water or, in our case, something that flows (the plasma in the star), *static* meaning *not moving*. This implies that the force on any shell, hence its acceleration, is zero and the shell does not move, that means its radius, r, is fixed. You are familiar with hydrostatic equilibrium from life on Earth. To a very good approximation the atmosphere of the Earth is in hydrostatic equilibrium. It does not fly off into space, but neither does it sink into the ground.

- Why does an atmosphere around a planet, or core of a star, not just fall into the ground by the force of gravity?
- When you put a book on a table, what forces act on the book and why does it not fall through the table?

The forces on a shell of mass δm, and with thickness δr, in a star are caused by two things with which you are familiar: gravity and pressure. We consider these forces below. Note that a *positive* force points out of the star while a *negative* force points toward the centre of the star.

1. The gravity of all the mass interior to the shell pulls it towards the centre of the star with a force $-G\,\delta m\,M_r/r^2$ where $G = 6.674 \times 10^{-11}\,\mathrm{m^3\,kg^{-1}\,s^{-2}}$ is Newton's gravitational constant.
2. The pressure of the gas below the shell tries to push it away from the centre of the star. This pressure is P_{bottom} corresponding to a force $F_{\mathrm{bottom}} = P_{\mathrm{bottom}} \times 4\pi r^2$ because the area of the shell's sphere is $4\pi r^2$.
3. The pressure of the gas on the top of the shell tries to push it into the centre of the star. This is not quite the same as the pressure on the lower surface: it is $-F_{\mathrm{top}} = -P_{\mathrm{top}} \times 4\pi r^2$.

The total force on the shell is then,

$$F = -G\frac{\delta m M_r}{R^2} + F_{\mathrm{bottom}} - F_{\mathrm{top}}\,. \tag{2.8}$$

When a shell is in *hydrostatic equilibrium* the forces cancel out. You know that force and acceleration are related by Newton's law,

$$F = \delta m \times a\,, \tag{2.9}$$

so the acceleration must be zero because the force is zero and the mass in the shell is not zero.

- How do you know that the surface of the Sun is in approximately in hydrostatic equilibrium?

More information...

- http://www.mezzacotta.net/100proofs/archives/215
- https://aty.sdsu.edu/explain/thermal/hydrostatic.html
- https://en.wikipedia.org/wiki/Plasma_(physics)
- https://www.youtube.com/watch?v=lkg1p173TAc
- https://www.youtube.com/watch?v=qGv2dpbkCzo

2.2.4 Stellar Structure: pressure equilibrium, energy flow

Every shell in the star has a luminosity because its gas is hot. At the shell or radius r, or mass m_r, this is written L_r. At the surface, we have $r = R$, $m = M$ and $L_r = L$. At the centre, $r = 0\,\mathrm{m}$, $m_r = 0\,\mathrm{kg}$ and $L = 0\,\mathrm{W}$ because there is no mass inside the centre.

L_r is power, its unit is the Watt ($\mathrm{J\,s^{-1}}$, symbol W). Because $L > 0\,\mathrm{W}$ at the surface of the star, which we know because stars shine, energy must be flowing outwards from the centre (where $L = 0\,\mathrm{W}$) to the surface, and from the surface into the rest of the Universe.

Where does all this energy come from? What is powering the star? This was a mystery for a long time. We know, however, that energy moves from the centre of the star to the surface. The surface is then so hot that it glows and radiates out into the rest of the Universe.

Energy flow in stars is not a simple subject. You probably know that heat is transferred by conduction, convection or radiation. In most stars, including those like the Sun, conduction is so slow that it is irrelevant. However, both radiation and convection *are* important. Which occurs depends on the temperature gradient, that is how fast the temperature changes as one moves outwards from the centre of the star. If the gradient is very large, convection sets in, just like in a thunderstorm on a hot day[7]. When the gradient is small, radiative transport is efficient enough to transmit all the energy generated by the core to the surface. Most stars contain zones that are "radiative" and zones that are "convective". Our Sun has a radiative core and a convective envelope near the surface. We need stellar evolution modelling to discover which stars contain which zones, and where.

- Why does a hot star radiate into the Universe?
- From where do you think the energy in a star comes?

2.2.5 Stellar evolution: nuclear burning

Material at the centre of stars is so hot that the nuclei in the plasma can merge, a process called *nuclear fusion*. Fusion happens only when two particles come close enough. In stars, this means forming atomic nuclei, which have sizes of femtometres, or $10^{-15}\,\mathrm{m}$. This is *very* close. It is very difficult to force two particles to get this close when they have the same charge. In stars, the particles are protons each of which has a charge $q = +1e = +1.602 \times 10^{-19}\,\mathrm{C}$.

The force between two particles of charge q_1 and q_2 is the *Coulomb force*,

$$F = \frac{1}{4\pi\epsilon_0}\frac{q_1 q_2}{r^2}, \tag{2.10}$$

where $\epsilon_0 = 8.885 \times 10^{-12}\,\mathrm{F\,m^{-1}}$ is the vacuum permittivity. Because nuclei are positively charged, both q_1 and q_2 are positive so the force is positive, i.e. *repulsion*.

It turns out that fusion starts in stars at temperatures around $10^6\,\mathrm{K}$, while the centre of the Sun is at about $1.5 \times 10^7\,\mathrm{K}$. At their hottest, during supernova explosions, stars can reach tempeartures of a few GK ($1\,\mathrm{GK} = 10^9\,\mathrm{K}$). This is the probably hottest place in the current Universe!

[7] On a hot day, the Earth's surface is heated by the Sun, but the atmosphere is mostly transparent so heats very little. The temperature gradient is proportional to the difference between the ground and air temperatures, which becomes large. If large enough, the air near the ground expands enough that its density drops and it starts to rise. This is convection, and in the case of the Earth causes rain and thunderstorms. Stars lack the water to cause the rain, but convection remains the most efficient way to get energy out of a star.

- Why is there no "cold fusion", i.e. why do nuclei not merge at temperatures lower that those found in stars?
- What are the only nuclear fusion events to occur on Earth?
- What is the energy, i.e. nuclear-burning, efficiency of the Sun, as measured by power per unit volume in $\mathrm{W\,m^{-3}}$? Do a similar calculation for 1) lightbulb and 2) a compost heap. Are you more energy efficient than the Sun? Do your answers change if you consider power per unit mass?

2.2.6 Want to learn more?

The above is an introduction only. To properly understand stellar evolution, one really needs to know some calculus, i.e. derivatives, differential equations and integrals. If you know some of this, try reading up on the *stellar structure equations* which are a set of differential equations that quantify the principles we discussed above: mass conservation, hydrostatic equilibrium and nuclear burning. By solving the equations of stellar structure at each point (shell) in the star, we can model the evolution of the star. The differential equations are solved numerically throughout, and often the choice of solution method is important – this is a whole branch of mathematics.

More information...

- https://en.wikipedia.org/wiki/Stellar_structure#Equations_of_stellar_structure
- https://www.mathsisfun.com/calculus/differential-equations.html
- https://en.wikipedia.org/wiki/Differential_equation
- http://www.astronomy.ohio-state.edu/~depoy/courses/lecture.notes/transport.html

3 *Installing and running* Window to the Stars

The *Window to the Stars* software is packaged for the *Linux* operating system. You have probably only ever used *Windows* or, if you are luckier, *MacOS*. *Linux* is similar to *Windows*, but because of its reliability and open-source nature it is the standard choice for most people working in astrophysics. We have provided the *Window to the Stars* software on *Linux* set up for both PCs and the *Raspberry Pi*.

Please note: where you need a login and password, e.g. when using the virtual machine or *Raspberry Pi* image, these are "wtts" and "wtts".

3.1 *Installation on a PC*

You can run *Window to the Stars* on a PC in three ways.

3.1.1 Virtual machine

The easiest way is to use a "virtual machine", which allows you to run a virtual operating system inside your existing operating system. In this way you can run *Linux* in *Windows*. We use the *Virtualbox* software to do this, which can be freely downloaded at https://www.virtualbox.org/. You can then download the *WTTS* virtual machine from https://figshare.com/articles/dataset/Window_to_the_Stars_2_1/8114201 and load this into *Virtualbox*. Once running, you will have *WTTS* in its own private operating environment and it should be as fast as any native application.

More information...

- https://docs.oracle.com/cd/E26217_01/E26796/html/qs-import-vm.html

3.1.2 Docker

The *Docker* tool allows you to "dock" an application inside an operating system in your current operating system. This is similar to using a virtual machine but requires less memory and other system resources, so is ideal for older PCs. You can install *Docker*, for free, by following instructions at https://docs.docker.com/get-docker/. Then, run the following from a terminal.

```
docker volume create wtts_volume
docker run \
       --name=wtts2 \
       -it \
       --rm \
       -e DISPLAY \
       --ipc=host \
       --mount source=wtts_volume,target=/home/wtts/wtts_rundir\
       -v /tmp/.X11-unix:/tmp/.X11-unix \
       -v $HOME/.Xauthority:/home/wtts/.Xauthority \
       robizzard/wtts:latest
```

3.1.3 Native install

If you know how to install software on *Linux*, you can install *WTTS* natively by running the following in a *bash* shell.

```
wget http://personal.ph.surrey.ac.uk/~ri0005/code/wtts2/linux_installer
chmod +x linux_installer
./linux_installer
```

This should work well on any modern *Linux*, for example the *Ubuntu* (it has been tested up to *Ubuntu 18*).

3.2 Installation on the Raspberry Pi

Window to the Stars also runs on the *Raspberry Pi*. This is a small and powerful, but inexpensive, computer that is the size of your hand. You can install *WTTS* on a microSDHC card: we recommend a minimum 16GB card, these are currently less than £5 each from online vendors.

First you should download the WTTS "image" file, currently the 1.8GB file *WTTS2.13-raspbian-buster.img.gz*, from https://zenodo.org/record/3627232. You will need to unzip this file (e.g. with the *gunzip* or *winzip* tool) because it is compressed. The resulting *.img* file should then be installed on an SDHC card following the instructions at https://www.raspberrypi.org/documentation/installation/installing-images/.

Once the image is written on the SDHC card, you can insert this into the Raspberry Pi and it will boot.

Please note: the above image is based on *Raspbian Buster* for the *Raspberry Pi 4 model B* which is currently available. It will probably not work on an older *Raspberry Pi* model 3, but there is nothing to stop you following the instructions of section 3.1.3 on such a system.

3.3 Website and help

The *Window to the Stars* website is at http://personal.ph.surrey.ac.uk/~ri0005/window.html and help is available by emailing r.izzard@surrey.ac.uk. Please note that while I can help with *Window to the Stars* and its installation, I cannot help with other problems on your PC and I may point you to external websites where solutions can be found. Often, you can find these solutions yourself with the help of a search engine such as duckduckgo.com – this is how I do it!

More information...

- *WTTS* homepage http://personal.ph.surrey.ac.uk/~ri0005/window.html
- Raspberry pi https://www.raspberrypi.org/
- Linux https://en.wikipedia.org/wiki/Linux
- Ubuntu https://ubuntu.com/

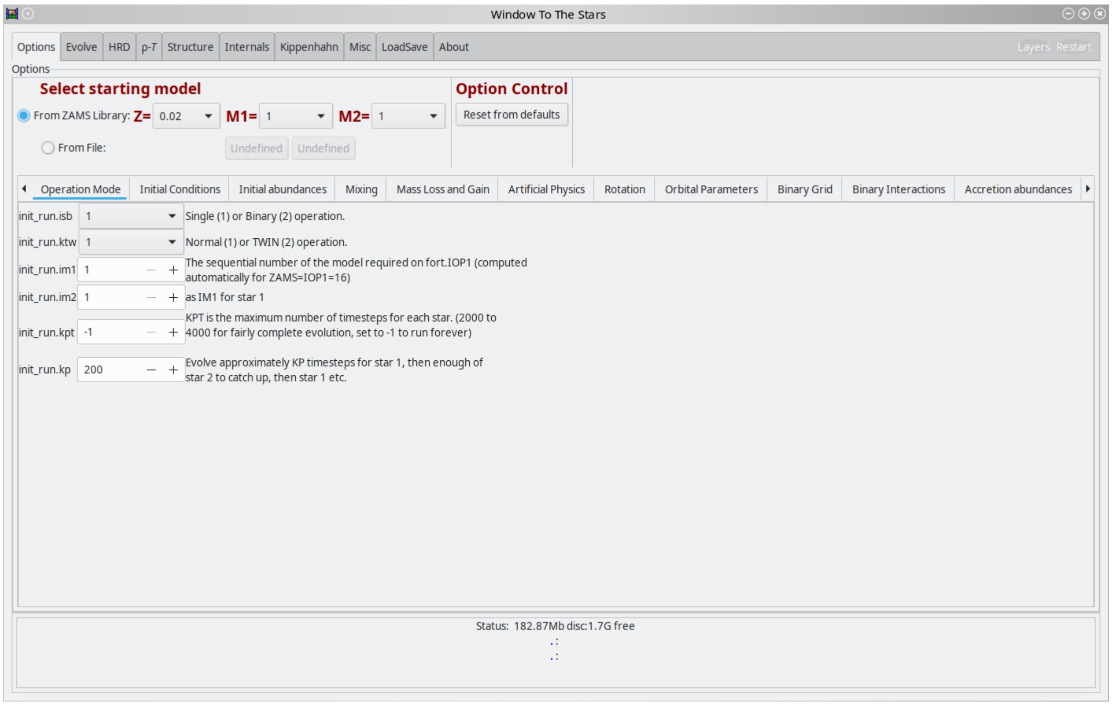


Figure 4: When you first run Window to the Stars, you are given the Options tab. Here you can choose the initial conditions of your star, i.e. its mass, M, and metallicity, Z, and change the physical prescriptions that are used when modelling the star.

4 *The Window to the Stars software*

Window to the Stars, or *WTTS*, is software that acts like a web browser for stellar-evolution modelling. It is a graphical user interface that makes it easy to make stellar models and provides you with the tools required to interpret these models so you can understand the physics at work and make predictions about what stars will do. Your job is to choose the starting conditions of the star, this means you need to decide its initial mass (in solar masses) and chemical composition. You can also change the physical recipes that are used.

4.1 *Running* WTTS

WTTS is like a tabbed web browser. When you run *WTTS*, it opens a window that looks like that shown in Fig. 4. The various tabs – Options, Evolve, HRD, $\rho - T$, Structure, Internals, Kippenhahn, Misc, LoadSave and About - are aligned along the top of the window. You can click on the tabs to move between them.

In the top-right of the window are the *Layers* and Restart buttons. Layers are a feature that allows you to run several stars at once, you may find this useful. The Restart button closes down *WTTS* and restarts it – this is useful if things go wrong. The software is not perfect, so if things go wrong you can try this.

The general idea is that your workflow goes from left to right through the various tabs. We briefly describe the tabs below.

4.2 *Data organisation in* WTTS

WTTS starts in its own directory, but you need to run it in a new directory if you want a new star. Note that a "directory" is the same thing as a "folder" in computer language. How you run *WTTS* from a new directory depends on which system you are using.

- You could run *WTTS* from a terminal, the old-fashioned way, and WTTS defaults to using whatever directory it is in when it starts. Try launching a terminal and running the following, changing "my_directory" to the name of your choice.

```
mkdir my_directory
cd my_directory
wtts2-gtk3
```

- You can also change the directory in which you save files in the "Layers" box. Click the "Layers" button, in the top right corner, then double click the "." at the beginning of the line under *Dir*. "." is the current directory, but you want to use a new one. Enter a new directory name, then hit return. Next, right click on that line and click "Mkdir <directory>" (where <directory> is whatever you called it). This makes the directory for you and is where the output from *WTTS* is placed.

Note that your directories (folders) take up space on your system, perhaps many gigabytes. If you run out of space, or to remain well organised, you should remove the folders that you no longer need. Note that you will *not* be able to get them back, so back them up, e.g. on a USB stick.

4.3 *Numbers on computers*

You are used to writing numbers in the form 0.1234 or, equivalently 1.234×10^{-1}. On a computer, this is written $1.234E-1$ where the E means $\times 10$. More generally, xEy is $x \times 10^{y}$. You may, in older software, see a D instead of an E.

More information...

- https://en.wikipedia.org/wiki/Scientific_notation#E_notation
- https://www.bbc.co.uk/bitesize/guides/zsnbr82/revision/4
- https://en.wikipedia.org/wiki/Computer_number_format

4.4 *Options*

In this tab you choose the mass, metallicity and physics that applies to your stellar model. This should be done before you try to evolve the star.

4.4.1 Initial conditions

The main two starting variables are the mass of the star, M, and its chemical composition as measured by the metallicity, Z. Remember $Z = 1 - X - Y$ where X is the mass fraction of hydrogen and Y is the mass fraction of helium (section 1.1). You can choose these in the "Select starting model" section (Fig. 4). Note that *WTTS* can run both single and binary-star systems. Binary stars contain two stars that orbit each other, much like the Earth orbits the Sun. There are thus two masses M_1 and M_2. In this course, we only need to set M_1 because we are making only single stars.

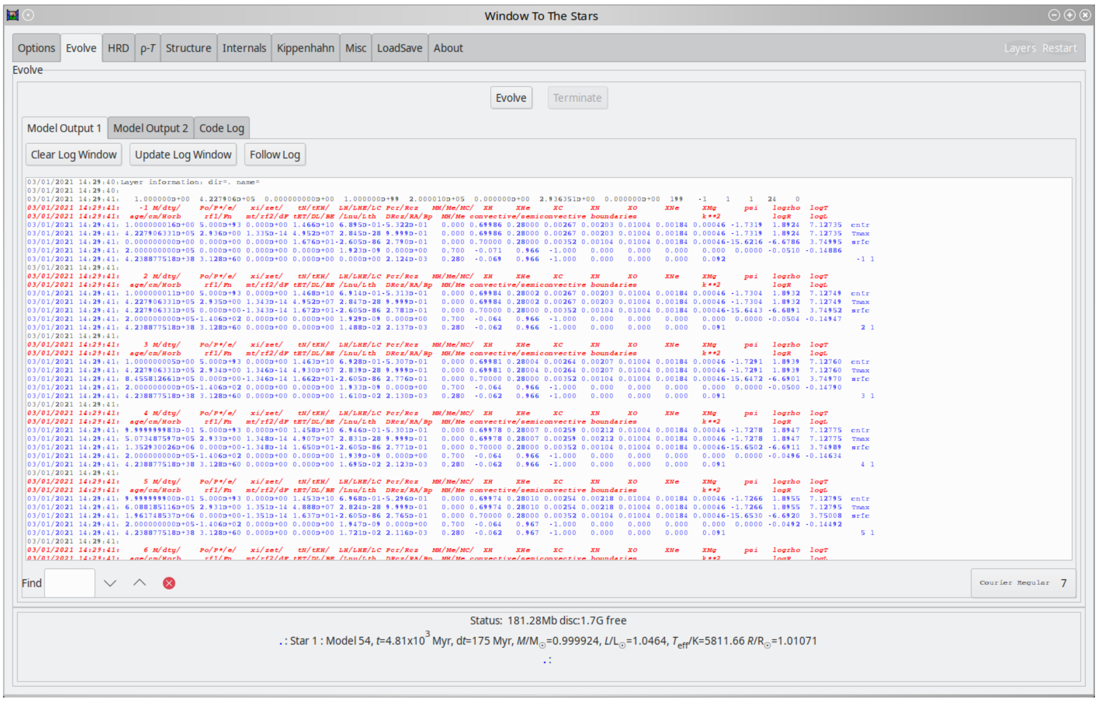


Figure 5: The *Evolve* tab of *Window to the Stars*. Most useful to you are the *Evolve* and *Terminate* buttons which start and stop evolution of your star. The text below that, with red and blue colours, is the technical information output by the stellar evolution code. This is really for experts, but is useful to look at when things go wrong.

4.5 Evolve

This tab allows you to start and stop the evolution of your stellar model, and view the log files that are the result of running the model (Fig. 5). These log files are quite technical and are really meant for people who are experts in stellar evolution. Bear in mind that, until *WTTS* was developed, these were the *only* way to understand what was going on in the stars, and that when the stellar evolution code goes wrong, this is the place to look for reasons.

4.6 HRD

This is the Hertzsprung-Russell (HRD) diagram, the most famous diagram of stellar evolution (Fig. 6). The HRD shows the logarithm of the effective temperature on the abscissa (x-axis) and the logarithm of the stellar luminosity on the ordinate (y-axis) as the star evolves. This is often called a "stellar evolution track". The effective temperature, T_{eff}, is calculated from the equation,

$$L = 4\pi\sigma R^2 T_{\mathrm{eff}}^4, \qquad (4.1)$$

where L is the stellar luminosity, R its radius and $\sigma = 5.670373 \times 10^{-8}\,\mathrm{J\,s^{-1}\,m^{-2}\,K^{-4}}$ is the Stefan-Boltzmann constant. This is the temperature the star has if it radiates as a perfect black body, which is a very good approximation in most stars.

You can choose to map the colour of the track to properties of the star. By default, the colour of the track is designed to be close to the true stellar colour. Of course, I have had to make it a bit darker because the Sun is white and so is the background of the diagram: you would not see it! You can also label points on the track by choosing from the *Label with* menu.

4.6 HRD

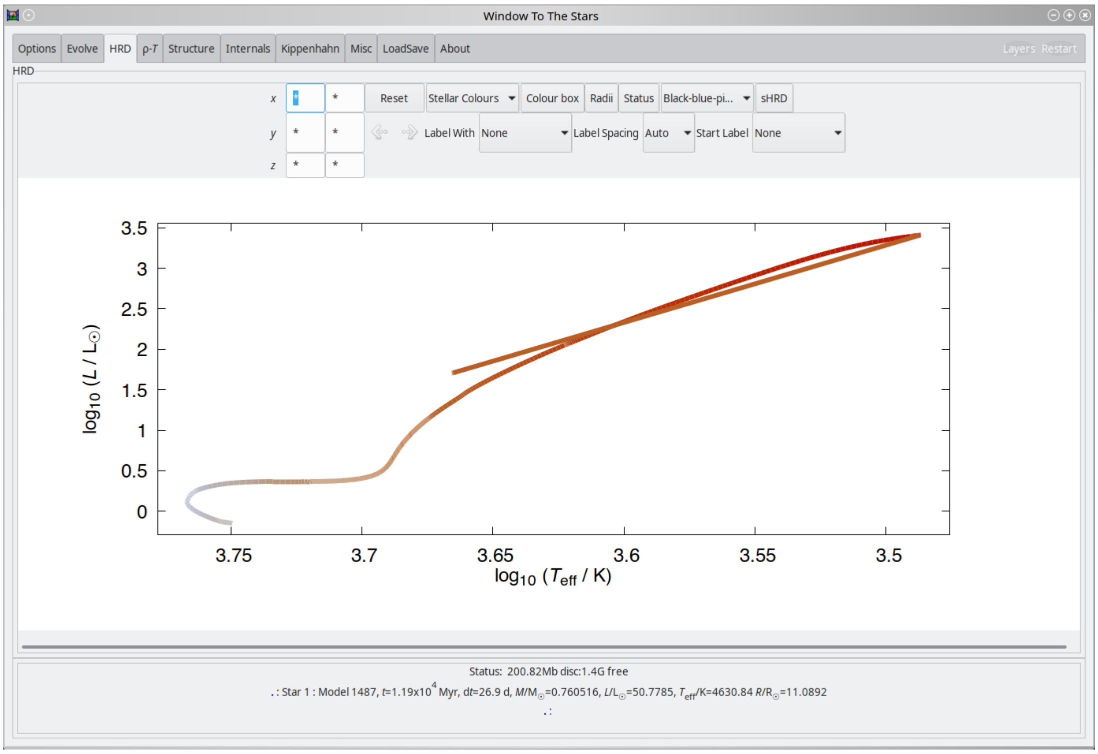


Figure 6: The *HRD* tab of *Window to the Stars*. This shows the most famous diagram in stellar astrophysics with the logarithm of the effective temperature in Kelvin, T_{eff}, on the abscissa (x-axis) and the logarithm of the luminosity in Solar units, $L/\mathrm{L}_\odot$, on the ordinate (y-axis). The stellar model track shown above is that of a $1\,\mathrm{M}_\odot$ star of solar metallicity. Because the luminosity is measured in Solar units, the model most similar to our Sun has $L = \mathrm{L}_\odot$ which means $L/\mathrm{L}_\odot = 1$ so $\log_{10}(L/\mathrm{L}_\odot) = \log_{10} 1 = 0$. You can find this location on the track and hence calculate the effective temperature. The colour of the track is designed to be close to reality.

More information...

- https://en.wikipedia.org/wiki/Hertzsprung-Russell_diagram
- https://usq.edu.au/academic-success-planner/logarithms-exponentials/introduction-logarithms/step1
- https://www.mathsisfun.com/algebra/logarithms.html
- https://en.wikipedia.org/wiki/Effective_temperature
- https://en.wikipedia.org/wiki/Stefan-Boltzmann_law

4.7 $\rho - T$: density–temperature

This tab contains the density–temperature plot, represented by the symbols ρ (Greek *rho*) and T, usually of the centre of the star (Fig. 7). The x-axis is the logarithm of the density and the y-axis is the logarithm of the temperature. You can use the track to see that, as a star evolves, its centre typically heats up and becomes denser. You can also see lines, selected using the "NucZones" button, showing the boundaries of nuclear burning, for example hydrogen, helium, carbon burning etc. By clicking the "MaxT" button, you can see the track at the location of maximum temperature in the star. Normally this is the centre, but there are times when it is not.

Because the long-term evolution of a star is dictated by the conditions at its core, the $\rho - T$ diagram is a vital tool to understand what is happening in the star.

4.8 Structure

The Structure tab allows you to make graphs of one property of the star vs another over its whole evolution or a portion of its evolution. Often, you wish to plot something vs time: Fig. 8 shows the stellar radius (its size) as a function of time for a 1 $M_{\odot}$ star.

The structure variables are shown in a list. To choose one as an x data type, left click on one of the "Off" labels in the "X" column, and change this from "Off" to "Linear" or "log10". Do the same for the "Y" column. You should see a plot appear.

Note that in addition to linear and logarithmic plots, you can plot "logmod", which is $\log_{10}|x|$ where $|x|$ is the modulus of x, or 10^x which is the inverse of the logarithm. Thus if you plot $10^{\log_{10}(R/R_{\odot})}$ this is just *linear* $R/R_{\odot}$.

4.9 Internals

Each star that you evolve is a series of models, with each one following from the previous. Each model corresponds to a particular *stellar age*, and the change in age from one model to the next is called the *timestep*. The timestep is *not* a constant: it varies depending on how smooth the evolution is. If the star changes at a slow rate, the timestep can be long. If the star changes very rapidly, the timesteps have to be very short. You can think of the timestep as a "non-linear time": often it is much easier to visualise very rapid evolution as a function of model number rather than real time.

You can plot the properties of the star at each timestep in the *Internals* tab. Use the "On/Off" column to choose which timesteps you want to plot, then in the "X" and "Y" columns choose what you would like to plot on the x and y axes.

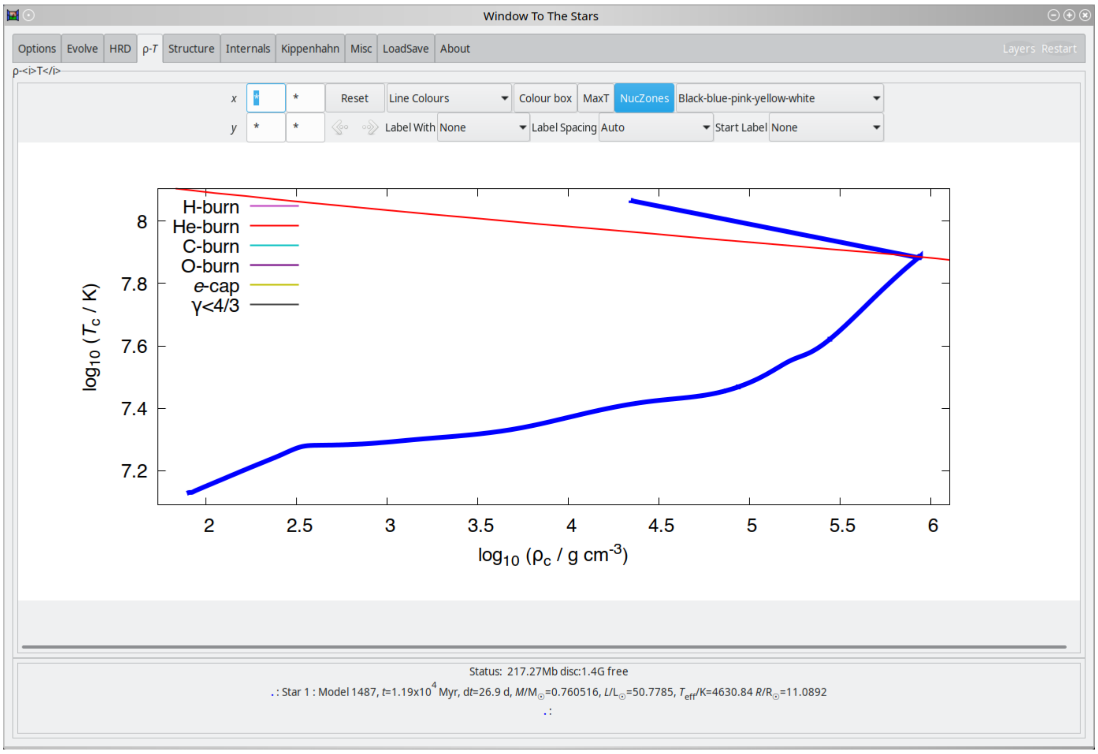


Figure 7: The $\rho - T$ tab of *Window to the Stars*. This shows the density and temperature usually at the centre of the star, i.e. ρ_c and T_c. In the above, we show a $1\,M_\odot$ star in the blue track. It starts at $\log_{10}(\rho_c/\,g\,cm^{-3}) = 1.89$, $\log_{10}(T_c/K) = 7.12$, the conditions at the birth of a $1\,M_\odot$ star, and moves towards the top right as the star evolves. When the track crosses the "He-burn" line, helium burning starts and the centre heats rapidly while at the same time its density drops.

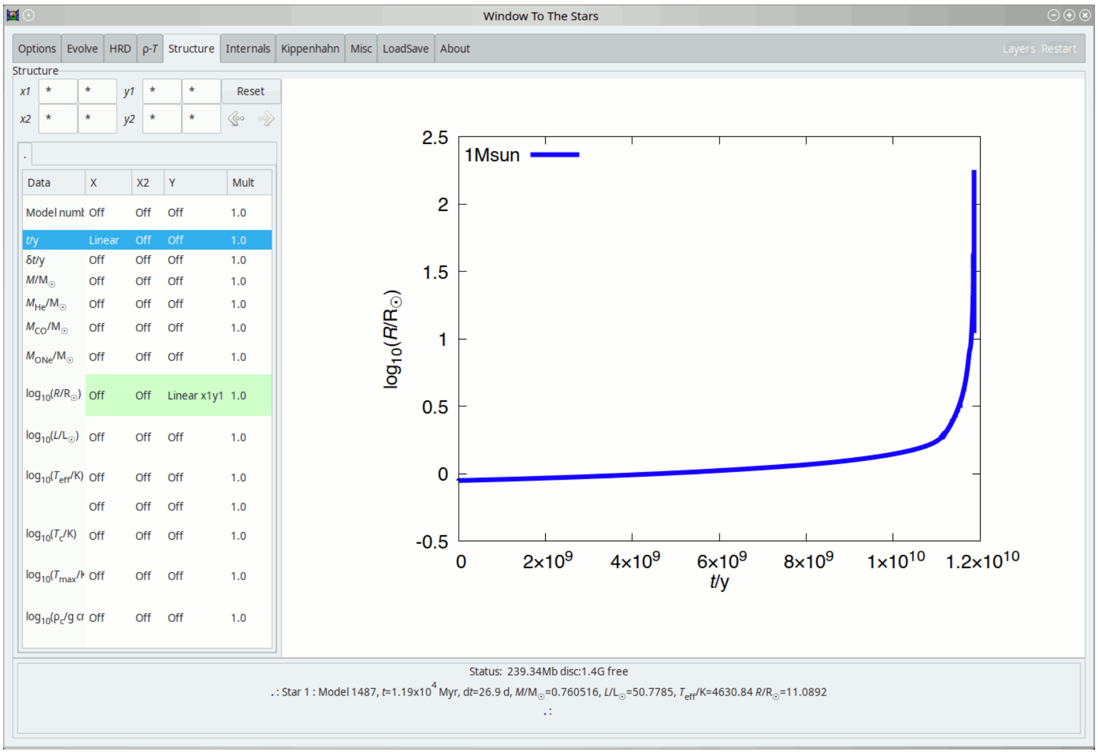


Figure 8: The *structure* tab of *WTTS*. You can use this tab to plot any structure variable, like the luminosity, radius, central chemical abundance, etc. as a function of any other.

4.10 Kippenhahn

The *Kippenhahn* tab makes three-dimensional (3D) surface-colour plots of data, named after the German astronomer Rudolf Kippenhahn who used such diagrams in his textbook *Stellar Structure and Evolution* (Kippenhahn et al., 2012). On the x and y axes you can plot any structure variables, like time, model number, mass coordinate, while as a colour (the z coordinate) you can choose any internal property of the star. This allows you to plot, in a compact 3D way, the evolution of a star.

More information...

- https://en.wikipedia.org/wiki/Rudolf_Kippenhahn

4.11 Misc, LoadSave, About

The last three tabs, *Misc*, *LoadSave* and *About*, are not used in this course. In the *About* tab you can see a photo of *Peter Eggleton* who wrote the *TWIN* code used to do all your stellar evolution calculations.

More information...

- https://www.corpus.cam.ac.uk/people/dr-peter-eggleton

- make a few stellar tracks: M=0.1,1,10 to see what changing the mass does

4.11 Misc, LoadSave, About

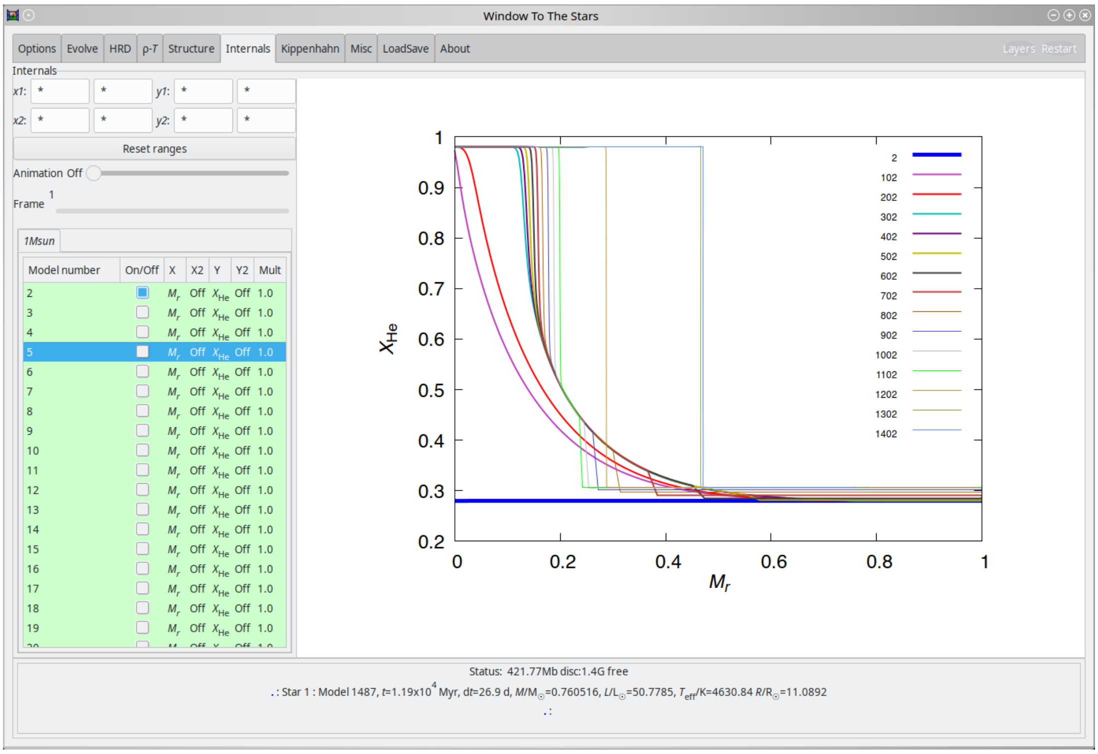


Figure 9: The *internals* tab of *WTTS*. You can use this tab to plot any internal structure of any model that is calculated. In the case shown, the helium abundance by mass fraction, X_{He}, is plotted as a function of mass coordinate, M_r, for every 100th model of a 1 $\mathrm{M}_{\odot}$ evolutionary model set. It is clear that at the beginning of the evolution (model 2) the star is about 27% helium, while by model 1402 the core, from 0 $\mathrm{M}_{\odot}$ to 0.47 $\mathrm{M}_{\odot}$, is 98% helium.

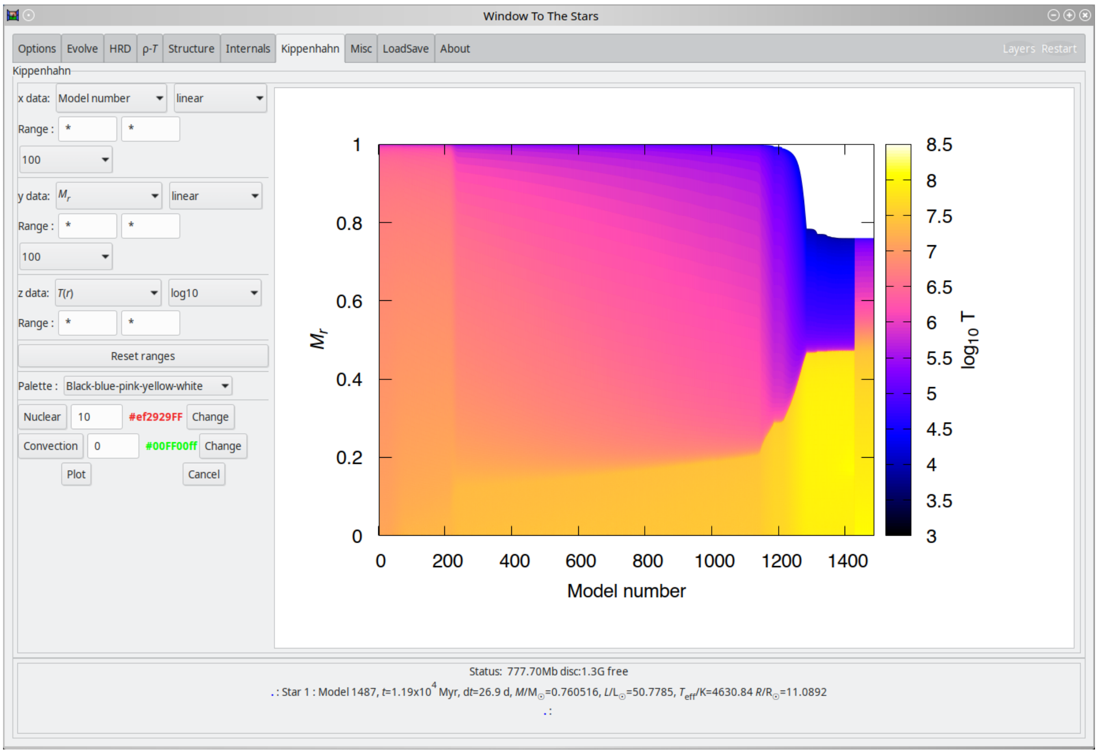


Figure 10: The *Kippenhahn* tab of *WTTS*. You can use this to show a 3D plot of three variables. In the above, we show model number on the x-axis, mass coordinate M_r on the y-axis, and the colour is the logarithm of the temperature, so yellow is hot while blue and black are cold. You can see that up to just after model 200, during core hydrogen burning, the star hardly changes. After this, until about model 1200, the core heats while the envelope cools: the star is becoming a red giant. From model 1200 the star has a hot core and a very cool envelope, until just after model 1400 helium ignites in the core.

- Exercise to calculate the lifetime of stars

4.12 Images

Tabs that contain images have common features. These include:

Range boxes to choose the x, y and, if necessary, z (colour) axis ranges. You can also zoom in by selecting regions of the plot or using your mouse scroll wheel. The "Reset" button resets the ranges to automatic settings, while the arrows below this allow you to choose the previous (or next) range that was chosen.

Saving an image If you right click on the image you will see a number of further options. Most important to you is "Save image" which allows you to save the plot for later use, "Copy to clipboard" which allows you to paste the image into another application (e.g. to email it).

Plot options Right click and choose "Gnuplot options" to change things like the line width, line type, label fonts etc.

Data Viewer You can also right click to see the "data viewer". This allows you to see the raw data that is used to make the plot. You can also use this to save the data for use in other applications, e.g. for your own data analysis.

4.13 If WTTS goes wrong

You may find that things go wrong and *WTTS* may freeze or crash. There are things you can do in this situation.

1. First, try to stop running your star by going to the *Evolve* tab and pressing *Terminate*. It may take a few seconds to do this: please be patient! If you are running stars in several layers, you can go to the *Layers* window, right click on a layer's details and select "Terminate evolution of all layers". Again, this will take a few seconds.

2. You can try right clicking on the title bar of the *WTTS* window and choosing "Close".

3. If you have run *WTTS* from a terminal, or a terminal was launched when *WTTS* was run, you can go to that terminal and press CTRL-C (the *CTRL* and *c* keys at the same time). This should kill *WTTS*.

4. If *WTTS* is really frozen up, you need to kill it through the operating system. If it does not respond, you will need to open a terminal and run the following command:

```
ps aux |grep wtts|grep perl|gawk "{print \$2}"
```

   This should give you a number, called the *process ID*. This is the number by which the operating system identifies the *WTTS* task. Now type the following, replacing <pid> with your process ID number.

```
kill <pid>
```

   If this fails to kill *WTTS*, try

```
kill -9 <pid>
```

4.13 If WTTS goes wrong

This cannot fail unless the *Linux* kernel has a problem, in which case you need to restart your system.

More information...

- https://en.wikipedia.org/wiki/Terminal
- https://www.linux.com/training-tutorials/how-kill-process-command-line/

5 *Layers in* WTTS

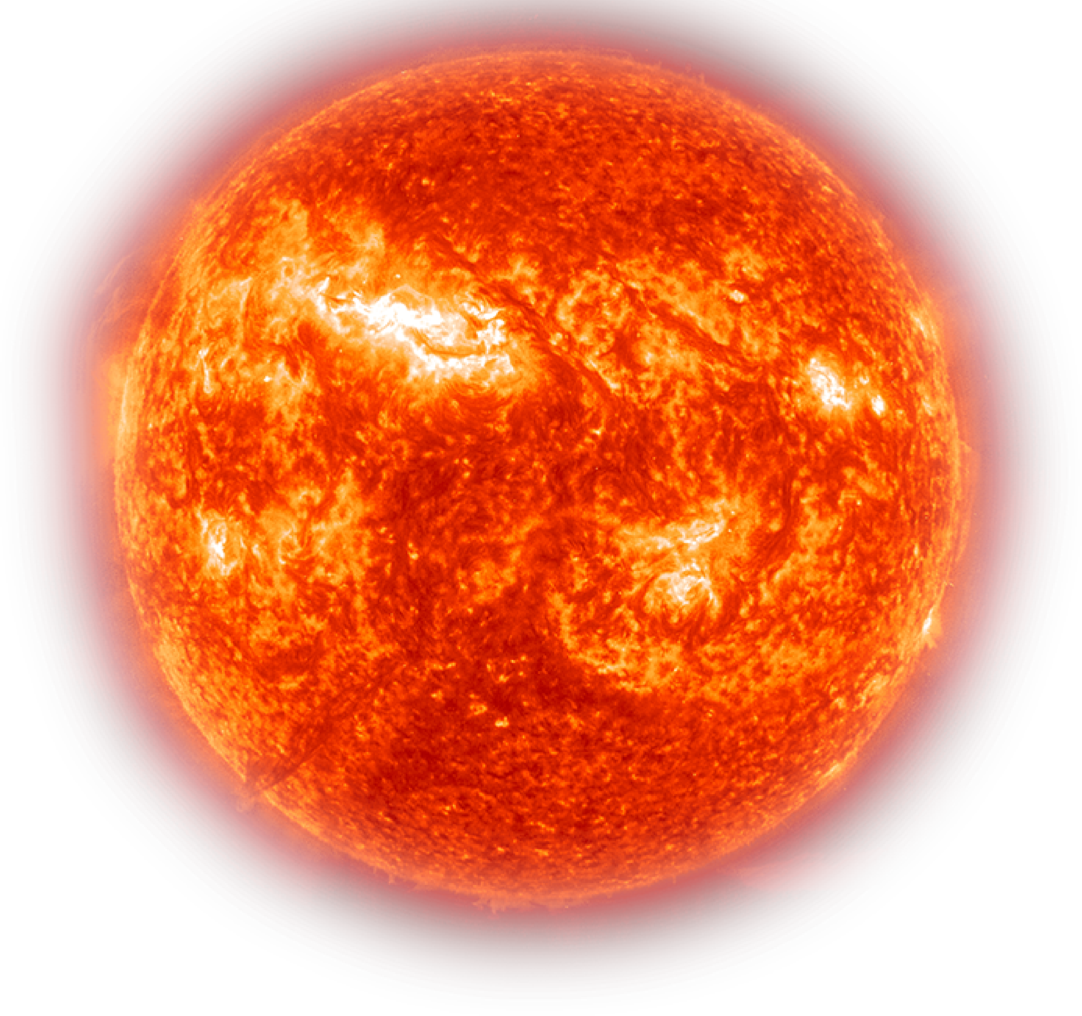
The *Layers* window is an advanced, experimental, tool in *WTTS* that allows you to run multiple stars at once and compare their properties easily. Each "layer" contains a different star, and you can layer these stars on top of each other in plots. For this activity, this is exactly what you want to do. Try the following:

1. In a new directory, open *WTTS* afresh and click on the "Layers" tab. You always have at least one layer.
2. Click "Add New Layer". You should now see two lines in the layers list. Both have their "Dir" column, the folder containing data, set to ".", the current folder. You want to change the "." in the second line to "10". Do this by double clicking on the second "." and enter "10", then press return. Now right click on the "10" and select "Mkdir 10". This has now made you a new folder called "10".
3. Select the "10" layer by clicking the "main" toggle button in the "10" row. The main *WTTS* window is now in control of layer "10". Select "10" from the "M1=" menu in the main *WTTS* window. Next, *only* in the $10\,M_\odot$ layer, go to the "Mesh" tab and change the "Pressure weight" value to "5.0E-001" (which means 5×10^{-1}, more easily written as 0.5)[8].
4. Go back to the "Layers" box. Right click on any line and select "Start evolution of all layers". Now your stars, $1\,M_\odot$ in the first layer and $10\,M_\odot$ in the second layer, are evolving – they take quite some time so you need patience! The stars will evolve as far as the stellar evolution code can take them – this is not to their end, but until it can no longer calculate a structure.
5. You can now go to the HRD tab to watch the stars evolve. Note that there are two tracks: the lower (dimmer and cooler) one is the $1\,M_\odot$ star and the upper (brighter and hotter), is the $10\,M_\odot$ star. To see which is which, in the "Start label" drop-down list, select "Mass".
6. All the plotting tools you are used to now have multiple layers in them. You can turn plots on and off just as before, but you can also turn stars on and off in plots in the "On/Off" column in the "Layers" tab.
7. If you have trouble running stars, remember that you should start in a new folder and that you must "mkdir" ("make directory") the folders from the layers tab so that each star has its own space.

[8] We need to do this because the "mesh" functions in the stellar evolution code TWIN specify where the software must do its calculations, more specifically where the "mass shells" should be located in the star. In a $10\,M_\odot$ star the structure is different to a $1\,M_\odot$ star, and we change the "pressure weight" to tell the software about this in advance. If you forget, you'll never get the $10\,M_\odot$ star to become a red giant.

Part II

Stellar projects



Our Sun.
Image credit: SOHO (ESA and NASA).

6 The nuclear thermostat

If stars are made of mostly hydrogen and helium, what powers them? Why do they shine and why for so long? It turns out that stars are *thermostatic* and that they have a *negative heat capacity*. In this part of the course, we will look at these peculiar properties.

6.1 Pressure and temperature in the core

When a star forms, mass falls from a large distance to near the centre of the star, a process called *star formation*. You know how mass reacts in a gravitational field: it speeds up as it falls. Gravitational energy is converted into kinetic energy. You also know that when this mass hits a surface and is stopped, like when you drop something on the floor, the kinetic energy is converted into other forms, such as sound and heat. Even the sound quickly dissipates and becomes heat. So the net result is that gravitational energy becomes heat. This is why, when it forms, the centre of a star is *hot*. The gravitational energy of the molecular cloud, from which the star formed, is converted into heat.

The centre of a star is also under great *pressure* because it supports the material above it against gravitational collapse. The material in a star like the Sun is, to a very good approximation, an ideal gas. As more mass is piled on top, the pressure and density increase to resist the collapse of the star. As a result, the centre of a star is hot, dense and has a very high pressure.

The high temperature, density and pressure at the centre of the star are sufficient to support the star and make it shine *as long as there is sufficient energy at the core*. If the star is shining, it is losing energy, so it cannot shine forever. In project 8 you will investigate how long a star can survive without a source of fuel in its core. Real stars *do* have fuel in their cores: they are made of hydrogen which, when dense and hot enough, undergoes nuclear fusion reactions that convert it to helium, as described in section 1.2. Nuclear reactions keep the centre of the star hot and allow it to live for a very long time.

- Use *Window to the Stars to* run a model of the Sun from its birth to helium ignition. Investigate the central temperature and density in the $\log T - \log \rho$ plane as the star evolves.

6.2 The stellar thermostat

Why is the temperature, both at the core and the surface, of a star so constant for most of its lifetime? The Sun has had roughly the same luminosity and temperature, based on biological evidence such as fossils, for hundreds of millions of years. That the Sun is so stable is because of two things. First, it contains a lot of hydrogen fuel so it takes a long time to burn all this fuel, as discussed in section 1.2. Secondly, the Sun is actually a thermostat meaning that if you heat it up it cools itself, and if you cool the Sun it reacts by warming itself.

Consider the following. If you could magically reduce the temperature and pressure at the centre of the Sun, how would it react? Well, first the Sun would shrink. The pressure supporting the layers above has dropped, so these layers would accelerate towards the centre. The density at the centre of the Sun would then increase, causing the temperature to increase. Nuclear reactions go faster at higher temperature, so the rate of nuclear burning increases and the centre of the star heats up. We are now back to where we started before you applied your magic.

This kind of behaviour is what we call a system of *negative feedback*. The system is stabilized: any small change you make is compensated by the feedback. This means that our Sun is a very stable system: it does not want to change and, as long as it has sufficient nuclear fuel – which it does for 10^{10} yr – it will not change very much. This is fortunate for us on Earth!

- Use *Window to the Stars to* run a model of the Sun while applying artificial energy loss.

More information...
- https://www.youtube.com/watch?v=b-6RCIuNU10
- https://en.wikipedia.org/wiki/Negative_feedback

6.3 *The specific heat of stars*

You are used to the temperature of an object increasing as you pump energy into it. When you turn on a kettle you pump energy into its water, hence its temperature increases. The specific heat capacity of everyday objects is thus positive. Water, at 20 °C = 293 K, has a specific heat capacity of about 4182 J K^{-1}, so for every 4.182 kJ you pump into 1 kg of water, its temperature increases by 1 K.

- Use *Window to the Stars to* run a model of the Sun while applying artificial energy gain.

Stars are not ordinary, everyday objects. As we argued above, if I pump energy into the centre of a star, the star *decreases* its temperature. This means its specific heat capacity is *negative*. It turns out that this is a property shared by objects that are bound by gravity, like stars and galaxies.

Even black holes have a negative specific heat. The energy of a black hole is given by Einstein's formula,

$$E = Mc^2 , \tag{6.1}$$

where M is the mass of the black hole and c is the speed of light. The temperature of a black hole is,

$$T = \frac{\hbar c^3}{8\pi k_B G M} \,\mathrm{K} \approx \left(\frac{\mathrm{M}_\odot}{M}\right) \times 6 \times 10^{-8}\,\mathrm{K}, \tag{6.2}$$

where $\hbar = 1.0545718 \times 10^{-34} \mathrm{m}^2\,\mathrm{kg}\,\mathrm{s}^{-1}$ is Planck's constant divided by 2π, k_B is the Boltzmann constant and G is the gravitational constant (Hawking, 1974). We can combine equations 6.1 and 6.2 to find,

$$T = \frac{\hbar c^5}{8\pi k_B G E} .$$

Now, if we double the energy E of the black hole, the temperature changes by a factor 1/2 – which means when you add energy, the temperature drops. Similarly, if we decrease the energy of the black hole, its temperature increases. This is another example of *negative heat capacity*.

More information...
- https://en.wikipedia.org/wiki/Specific_heat_capacity
- https://www.youtube.com/watch?v=Hs5x0-IU2F4

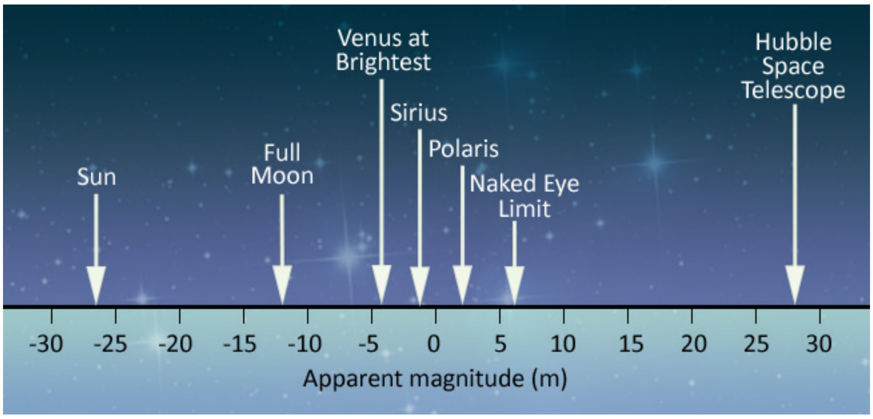


Figure 11: The magnitude scale: https://lonewolfonline.net/astronomical-magnitude-scale/.

7 The distance to the stars

7.1 Magnitudes and the distance modulus

Astronomers use a system called magnitudes to describe how bright stars are as seen from Earth. Before telescopes, classification was done using the naked eye. The brightest stars were labelled first magnitude, slightly fainter stars were second magnitude and this continues until the faintest stars the eye could see which were labelled sixth magnitude. So, somewhat counter-intuitively, a smaller magnitude corresponds to a brighter star. It turns out our eyes work on a logarithmic scale, and can detect about a factor of 100 difference in brightness among stars, so a 1st magnitude star is about 100 times brighter than a 6th magnitude star. Once telescopes were invented, even fainter stars could be observed and the scale was extended to larger magnitudes. Even brighter objects than the brightest stars have magnitudes less than 1 and the scale extends to negative numbers. For example, the Sun has a magnitude $m = -26.74$. Figure 11 gives some more examples of where different objects lie on the magnitude scale.

When we observe a star on Earth, it's apparent brightness depends on two things: the intrinsic brightness of the star and the distance the star is away from us. An object with a given intrinsic brightness will appear fainter the further away from us it is placed. You learnt previously that the intrinsic brightness of a star is described by its luminosity, L. For example, more massive stars tend to be more luminous and so will appear brighter in the sky. The **apparent magnitude, m** of a star is defined as the brightness of a star as we observe it on Earth. Thus, it depends on both the luminosity and the distance. Another useful quantity is the **absolute magnitude, M** which is defined as the brightness of a star we would observe if it was placed at a distance of 10 parsec ($1\,\text{pc} = 3 \times 10^{16}$ m) from Earth. In this definition the distance is fixed, so the absolute magnitude of a star only depends on its intrinsic brightness and thus the luminosity of the star.

The difference between the apparent magnitude and the absolute magnitude is known as the **distance modulus,** and is related to the distance to the star through a logarithm,

$$m_{\text{mag}} - M_{\text{mag}} = 5 \log_{10}(d) - 5, \tag{7.1}$$

where d is the distance to the star in parsecs (pc). So you can observe the apparent magnitude and if you also know the distance to the star you can deduce the absolute magnitude and thus the star's luminosity.

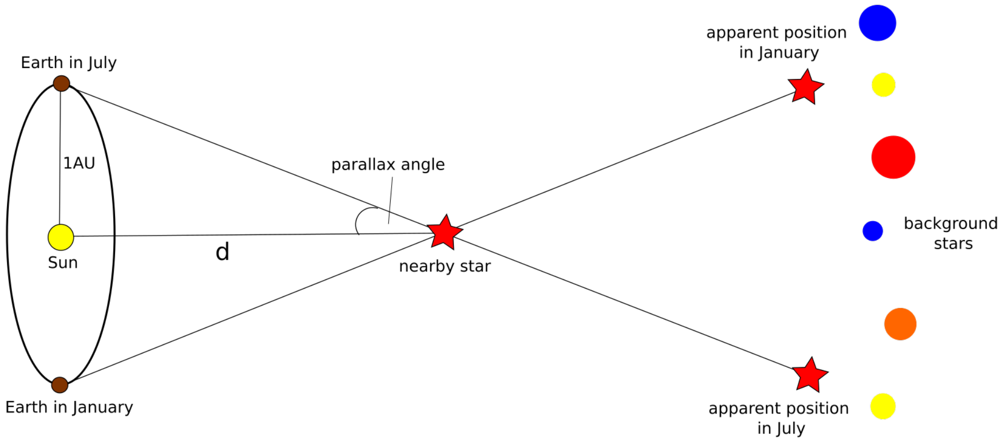


Figure 12: Parallax due to the Earth's motion around the Sun

To consolidate your understanding watch the following YouTube videos on the topic of magnitudes and the distance modulus,

- Absolute magnitudes: https://www.youtube.com/watch?v=yfsUhOPCMaM,
- Using the distance modulus formula: https://www.youtube.com/watch?v=aRx9WLRRmNE,
- More distance modulus calculations: https://www.youtube.com/watch?v=MEnD7LWXHpA.

7.2 Distances with Gaia

One method of measuring distances is using parallax. When the Earth orbits around the Sun, relatively nearby stars appear to move position with respect to the background of distant stars due to the difference in our line of sight (see figure 12). **Parallax** is the apparent change in an object's position due to the Earth's orbit around the Sun. With the parallax, and knowing the distance from the Earth to the sun is approximately 1AU, you are able to to deduce the distance to the star. The Gaia mission will measure parallaxes for around a billion stars, enabling distance measurements to these stars.

More information...

- The Gaia mission and spacecraft: https://sci.esa.int/web/gaia
- Parallaxes and Gaia: https://www.gaia.ac.uk/science/parallax

7.3 Distances with Cepheids

Cepheid variables are stars that brighten and dim periodically. An example is shown in figure 13. Classical Cepheids are massive, evolved stars undergoing very regular pulsations in their brightness. In 1908 Henrietta Swan Leavitt discovered a direct proportionally between their period and luminosity known as the period-luminosity relation. This relation is extremely useful as measuring the periods of these stars allows us to deduce their luminosities (an analogue of the absolute magnitude). As we can measure the apparent magnitudes of these stars from earth, and we can calculate their absolute magnitudes using their periods, we can take a final step and calculate the distance to these stars using the distance modulus equation. Because of this, these stars are very useful for measuring distances to stellar systems. You will investigate how we can use these stars to measure the distance to the LMC in the following exercise.

Star	Period, P/d	Apparent magnitude, m_{mag}	Absolute magnitude, M_{mag}	Calculated Distance (parsecs)	Gaia Parallax, (mas)	Gaia Distance (parsec)
HV 2353	3.10	14.9				
HV 12231	4.75	14.0				
HV 12700	8.14	15.0				
HV 2301	9.51	13.1				
HV 2260	13.0	14.5				
HV 2324	14.5	13.3				
HV 2261	17.3	12.6				
HV 1013	24.1	14.3				
HV 2540	28.1	12.5				

Table 1: Data of Cepheid variable stars in the Large Magellanic Clouds (LMC).

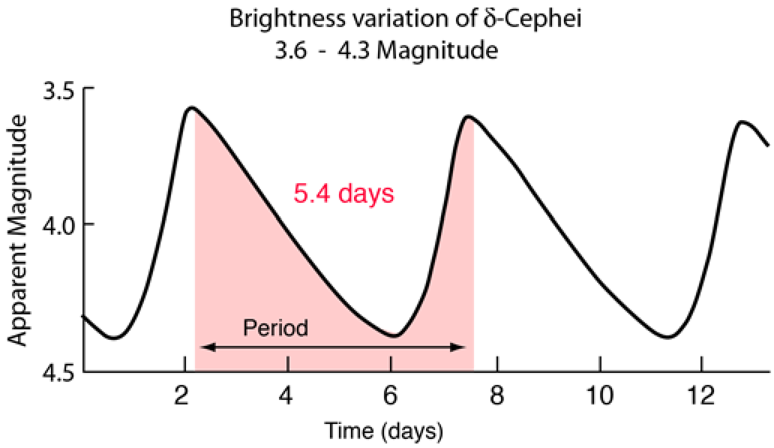


Figure 13: The Light curve of delta Cephei, a classical cepheid variable star showing the regular increase and decrease in its brightness. From http://hyperphysics.phy-astr.gsu.edu/hbase/Astro/cepheid.html.

7.4 Cepheids Exercise

In this exercise, you will calculate the distance to the Large Magellanic Cloud (LMC) using data about the cepheids within it. The LMC is one of the satellite dwarf galaxies orbiting the Milky Way, shown by figure 14. You can read more about the LMC at https://en.wikipedia.org/wiki/Large_Magellanic_Cloud.

1. The absolute magnitude, M_{mag} of a Cepheid variable star can be calculated using the relation,

$$M_{mag} = -2.2\log_{10}(P/d) - 2.05\,,$$

where P is the Cepheid period in days. Using this relation calculate the absolute magnitudes of the stars in table 1 and write them in the appropriate column. This relation was found by Groenewegen (2018) by analysing the parallax angles of 248 classical Cepheids from the Gaia mission data.

2. Using the absolute magnitudes you just calculated and the apparent magnitudes provided, calculate the distance of the cepheids from the earth by using the distance modulus equation in section 7.1. Once you have done this, calculate the average distance of the stars.

3. Now we will use the computer programming language *Python* to prove Henrietta Swan Leavitt's period-luminosity relation. Open the website https://trinket.io/embed/python3. Copy and paste the code from the file *distance-determination.py* to the left hand side of the website interface. This code will be used to create a period-luminosity graph from the Cepheid variable-star data in Table 1.

4. From the table, which variable would be on which axis on the graph? Once you've decided, label each axis in the code. In the example below, the x-axis would be labelled pigeons, and the y-axis elephants:

```
plt.title('A Graph of Pigeons and Elephants')
plt.xlabel('pigeons')
plt.ylabel('elephants')
```

   Check with your teacher before moving on to the next step.

5. Now we will input the axis data for the graph. In the code, input the data for the x axis by writing it in between the square brackets of: `x = np.array([])` and separate each data point with a comma. Similarly, input the data for the y axis by writing it in between the square brackets of: `y = np.array([])` and separate each data point with a comma. An example of these lines filled in would be:

```
x = np.array([1,2,3,4,5,6,7,8,9])
y = np.array([1.1,2.2,3.3,4.4,5.5,6.6,7.7,8.8,9.9])
```

6. Click `run` and allow the code to execute, it may take a few seconds so be patient. A new graph will be created on the right hand side. Can you clearly see Henrietta Swan Leavitt's period-luminosity relationship? The gradient and y-intercept of this relationship is displayed on the top. What is the equation of the line?

7. Now we will compare the distance that we have calculated to the LMC with the distance calculated by the Gaia mission. Open the Gaia archive website through the link https://gea.esac.esa.int/archive/, and click on search. We will now change a few of the options. Change the 'Radius' from 5 to 1. From the 'Search in:' drop down select 'gaiadr3.gaia_source'. And finally, press the 'Display columns', deselect all the options, and then select only the 'source_id' and 'parallax' options.

8. Now one at a time, type the names of the stars in table 1 in the 'name' box and press 'submit query'. Note down the stars parallax in table 1. Go back to the search window by pressing 'Basic' and search the next star until you have the parallaxes of all the stars. Some of the Gaia parallaxes may be negative, this is just a result of the measurement method and not to be worried about! Just note down the parallax as a positive number.

9. Gaia parallaxes are given in milliarcseconds. 1 milliarcsecond = 0.001 arcsecond. Parallax is converted into distance d, in parsec, by the equation:

$$d = \frac{1}{p},$$

where p is the parallax angle in arcseconds. Do the appropriate unit conversion for the Gaia parallaxes and then calculate the Gaia distances to the stars. Write these distances in the appropriate table column. Calculate the average for the Gaia distances, how does this compare with the average distance you calculated earlier, are they similar? What can we do to improve our results in the future?

Figure 14: The Large Magellanic Cloud captured by the European Space Observatory's VISTA telescope in Chile. From https://www.eso.org/public/news/eso1914/.

7.5 Supplementary work for programming skills

- If you're interested in learning simple mathematical computer programming from the ground up, check out https://projecteuler.net/ which has an archive of exercises mathematical computer programming exercises from beginner level all the way up.
- Another place to learn useful problem solving skills is https://isaacphysics.org/alevel. The skills that these two websites can help develop are critical in science and scientific research, and it is all available for free!

8 The beginning of nuclear astrophysics

At the beginning of the 20th century, the reason why stars shine and what fuelled them was unknown. Why do you think the Sun shines? Is it a chemical reaction, like a candle burning? Is it a light bulb powered by a giant battery? Is it a piece of glowing iron in the sky? I was once told this by someone who genuinely believed it.

In this project you will conduct experiments similar to those who asked the above questions, with one big advantage: you have a computer program that can model stars.

Aim: You will be modelling stars based on the level of knowledge a physicist would have had at the beginning of the 20th century. You do not (yet) know about nuclear burning, but you do understand about how gas moves (its dynamics) and gravity, hence you understand concepts like *hydrostatic equilibrium* and *radiation*.

Tasks: You will model a Sun that has no nuclear burning (see below for suggested steps) and consider, based on the evidence, whether this is our Sun.

- How does the chemical profile, that is the chemical abundances as a function of depth in the star, change with time?
- How long can this "no-nuclear" Sun shine if it looks roughly like our Sun, i.e. has $L \sim \mathrm{L}_\odot$ and $R \sim \mathrm{R}_\odot$?
- How does the structure of this "no-nuclear" Sun compare to our real Sun? Is its core hotter or cooler, is its radius typically larger or smaller, is its surface hotter or cooler?
- How hot does a "no-nuclear" Sun become in its core, and how old is the star when its core is at its hottest?
- What happens to the "no-nuclear" Sun after a Hubble time (about 13.7×10^9 yr)?
- If you live in a Universe in which there is no hydrogen burning, but there *is helium burning*, how many times faster does the helium burning have to be, compared to reality, to fuel the Sun by this burning chain?
- What were historical explanations for the Sun's luminosity prior to the discovery of nuclear burning, and what arguments other than stellar modelling suggest an age for the Sun?

Remember to explain *how* you answer these questions, explain your *method*, *why* your answer is what it is, say *what* evidence you find, be *quantitative* in your reporting, and try to link your answers to sources you find in books or on the internet (without copying the material from these sources – use your own words or quote the sources with citations).

Making a Sun without nuclear burning

To make a Sun without nuclear burning, you simply turn it off. Fortunately, this is simpler in a simulation[9] than in real life.

1. Launch *WTTS*. Select metallicity $Z = 0.02$ and mass $M_1 = 1$, which should be chosen by default.

[9] Or science fiction, e.g. https://memory-alpha.fandom.com/wiki/Trilithium.

2. In the *Options*, find the *Nuclear Network* tab. Set *cnucx*, *cnucy* and *cuncz* to 0.0. These are the hydrogen, helium and carbon-burning nuclear reaction rate multipliers. By setting them all to zero, you multiply all nuclear-reaction rates by zero, hence there is no nuclear burning in the star.

3. Evolve the star. You can watch it evolve in the *HRD* tab and analyse its structure in the *Structure*, *Internals* and *Kippenhahn* tabs. You will want to investigate stellar properties like the hydrogen-burning luminosity, L_{H}, the helium-burning luminosity, L_{He}, and, especially in the *Kippenhahn* tab, the nuclear-burning energy generation rate, ϵ_{nuc}.

9 *Making the chemical elements in stars*

The Universe was made in the Big Bang. For about 20 minutes, the whole Universe was hot enough to fuse protons, the nuclei of hydrogen atoms, into alpha particles, the nuclei of helium atoms. Because the Universe was expanding during this time, and had a roughly fixed amount of total energy in it, it was also cooling. Eventually, it cooled below the temperature threshold for nuclear burning (around 10^7 K or ten-million Kelvin) and changes in the abundance, by mass fraction[10], of hydrogen (X) and helium (Y) stopped[11] at,

$$X = 0.75\,, \tag{9.1}$$
$$Y = 0.25\,. \tag{9.2}$$

- What is the relative abundance of hydrogen and helium by *number of nuclei* of each type?

Nuclear burning happens through two cycles in the centre of stars.

pp chain The proton-proton, or "pp", chain involves direct collision of protons to make heavier elements. Fig. 15 shows how protons combine to make ^{2}H, called deuterium, which is a proton combined with a neutron. The deuterium then captures another proton, to make ^{3}He, called helium-3, and then two helium-3s combine to make ^{4}He and two free protons. The network can thus be represented by the following equation,

$$6p \rightarrow {}^4\mathrm{He} + 2p\,, \tag{9.3}$$

or, subtracting $2p$ from both sides to give the net reaction equation,

$$4p \rightarrow {}^4\mathrm{He}\,. \tag{9.4}$$

Also released by the nuclear reactions are gamma-rays, written γ-rays, which carry some of the released nuclear energy into the star and are eventually emitted at the star's surface, as well as very light particles called *neutrinos*.

CNO cycle The CNO cycle is an alternative way that stars generate nuclear energy. Protons are captured and emitted by carbon (C), nitrogen (N) and oxygen (O) isotopes, as shown in Fig. 16. C, N and O act as catalysts for the reactions, and the net effect is, as in the pp-cycle,

$$4p \rightarrow {}^4\mathrm{He}\,. \tag{9.5}$$

- Which type of fundamental force is involved in each of the steps of the pp chain?
- Do you expect the pp-chain or CNO-cycle to dominate at a) low and b) high temperatures?
- What happens to the total number of C, N and O particles?
- How much energy is generated by the pp-chain and CNO cycle? Which cycle more efficiently heats a star?

[10] The total mass fraction is 1.

[11] https://en.wikipedia.org/wiki/Big_Bang_nucleosynthesis

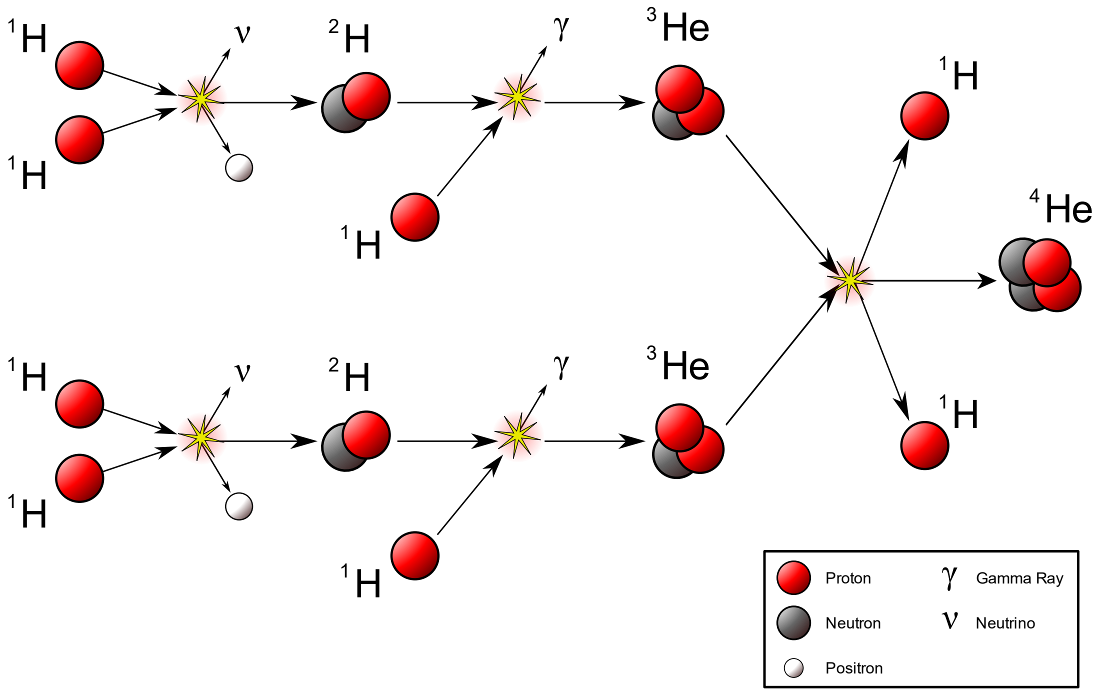


Figure 15: The pp chain. Red balls are protons, grey balls are neutrons, white balls are positrons, ν are neutrinos, γ are gamma rays, and the yellow flash points are where reactions happen. The reaction proceeds from left to right.

Activities

- Evolve a 1 $M_\odot$ and a 10 $M_\odot$ star, each as far as you can. You will need to make separate directories for each mass and run WTTS twice, or use the "Layers" tab to make two layers, one for each mass (see Section 5 for a description of how to use the *Layers* window).
- Compare the rate of pp and CNO burning at the birth time of the star, called the "zero-age main sequence"[12]. Do this by going to the *internals* tab, click "on/off" for the first model (usually model 2), in the same row select M_r (mass coordinate) in the X column, and either R_{pp} (the pp-chain burning rate) or R_{pC} (the CNO cycle rate, or rather the rate of the $^{12}\mathrm{C} + p$ reaction). Which reaction dominates in the Sun? Which dominates in a 10 $M_\odot$ star?
- How different could hydrogen burning rates be for life to survive on Earth?
  You can change the hydrogen burning rate, by either pp-chain or CNO, by going to the "Options" → "Nuclear network" tab. The CNUCX variable multiplies the hydrogen burning rate. Try multiplying it by 0.1 or 10, see what happens to the star on the main sequence. Does the star get too hot or cold? Too bright or too dim? What range of temperature and luminosity of the Sun allows life to be supported on Earth?

[12]The "main sequence" is the hydrogen-burning stage at the beginning of a star's life. Our Sun is in this phase of evolution. "Zero-age" means "birth" in English – i.e. age $t = 0$. I do not know who invented this name, it is not one I would have recommended!

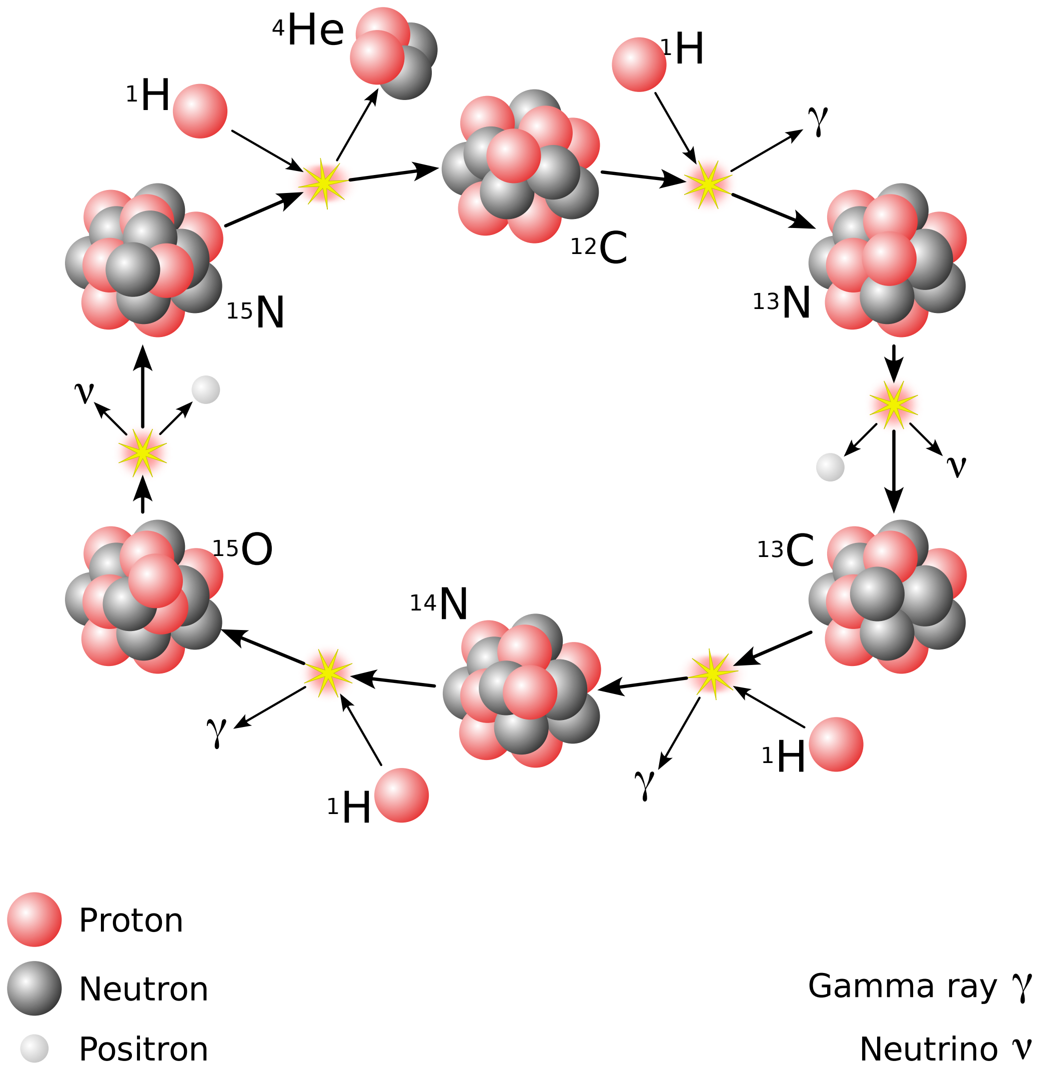


Figure 16: The CNO cycle. Red balls are protons, grey balls are neutrons, white balls are positrons, ν are neutrinos, γ are gamma rays, and the yellow flash points are where reactions happen.

More information...

- https://en.wikipedia.org/wiki/Proton-proton_chain_reaction
- https://en.wikipedia.org/wiki/CNO_cycle
- https://en.wikipedia.org/wiki/Nuclear_force
- https://en.wikipedia.org/wiki/Strong_interaction
- https://en.wikipedia.org/wiki/Weak_interaction
- https://websites.pmc.ucsc.edu/~glatz/astr_112/lectures/notes8.pdf

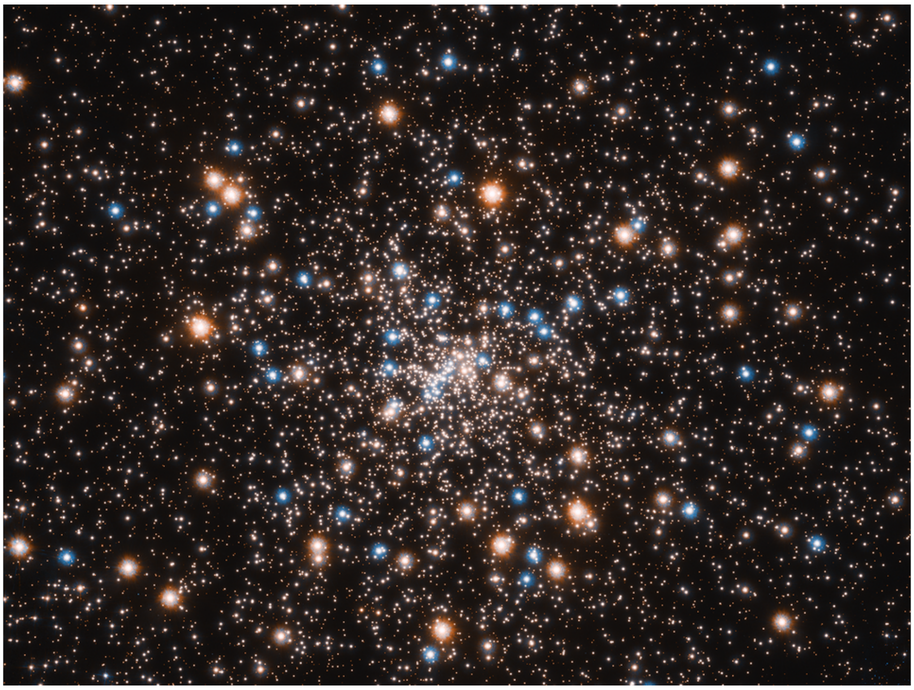


Figure 17: Hubble Space Telescope image of globular cluster NGC 6397 (https://hubblesite.org/image/4145/gallery).

10 Dating star clusters

Star clusters are large groups of stars that are gravitationally bound. The globular cluster NGC 6397 is shown in figure 17. It contains about 400,000 stars and is located about 7,800 light years away from us, making it one of the closest globular clusters.

10.1 Evolution of star clusters

All the stars within a cluster form at approximately the same time and so have the same age. However they form with a variety of masses. More massive stars burn through their nuclear fuel faster causing them to evolve faster and live shorter lives. The main sequence lifetime of a star is how long it spends on the main sequence, burning hydrogen in it's core, before it evolves off the main sequence and towards the giant branch. The more massive stars in the cluster have shorter main sequence lifetimes and will move off the main sequence before the less massive stars in the cluster. This leads to a main sequence turn-off mass in the HR diagram of a cluster. Stars more massive than this turn-off mass have evolved off the main sequence and are seen as giants and beyond whilst stars less massive than the turn-off mass are still seen on the main sequence. Figure 18 shows the HR diagram of the globular cluster NGC 6397 from Husser et al. (2016). You can see that the main sequence is truncated at the turn-off mass where the more massive stars have evolved up the giant branch. Stars at the turn-off point are just about to evolve off the main sequence and so the age of the cluster must be equal to the main sequence lifetime of these stars. In the following exercise you will estimate the age of this cluster by first finding the turn off point mass.

To consolidate your understanding watch this *YouTube* video:
https://www.youtube.com/watch?v=Owa9JRLhbhI

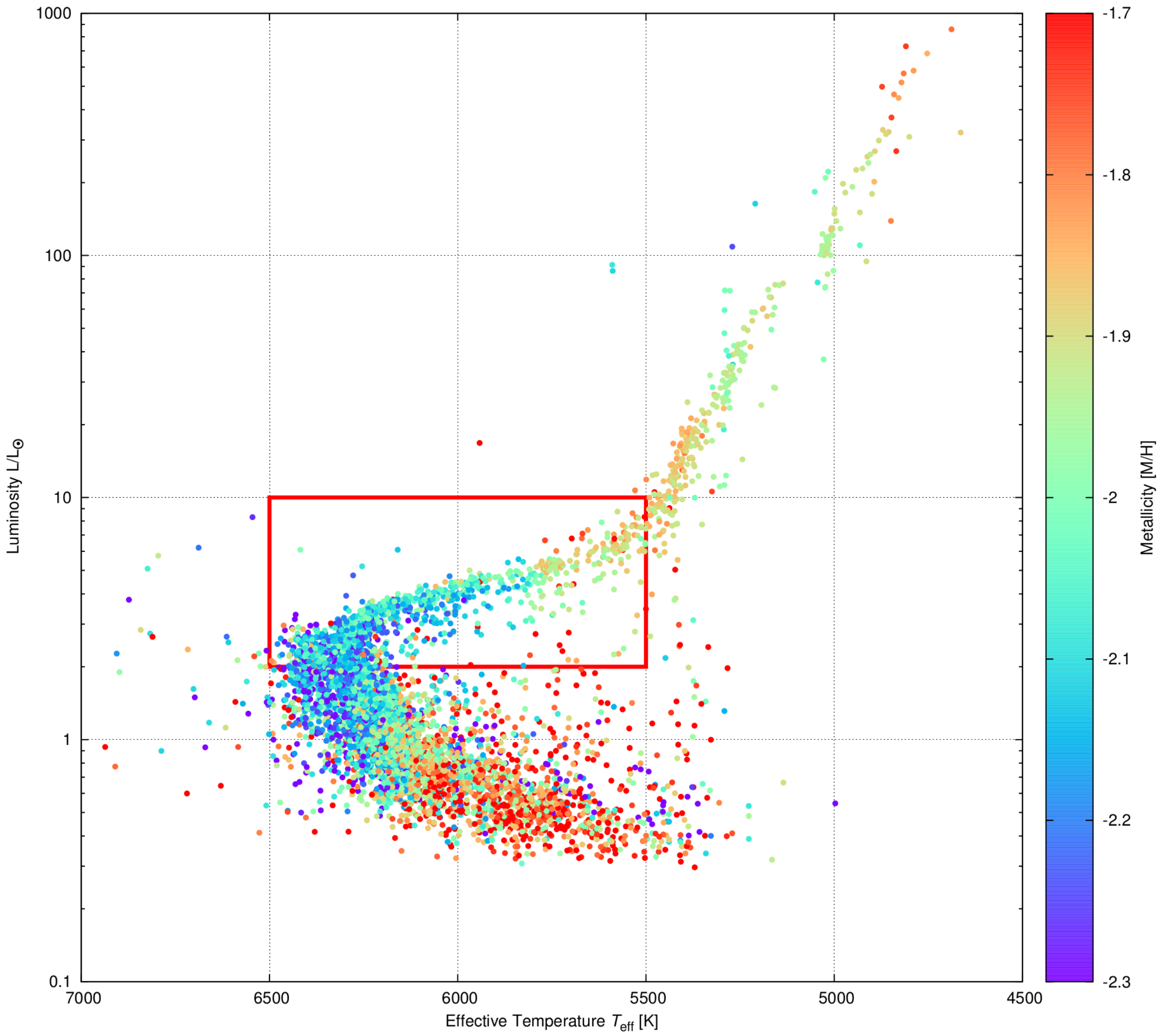


Figure 18: Hertzsprung-Russell (HR) diagram of the globular cluster NGC 6397 taken from Husser et al. (2016).

10.2 Estimating the age of NGC 6397

1. First using figure 18 estimate the effective temperature, T_{eff} and the luminosity, L of the turn-off point. Note this plot has linear axes but *WTTS* has logarithmic axes. Don't worry about the red box, it's not important for this context.

2. Now using WTTS evolve stars with different masses. The stars in NGC 6397 have low metallicity so set $Z = 0.0001$ for all your models. You can use the layers window to run multiple stars on the same HR diagram, instructions for using layers can be found in Section 5. Try to replicate the shape of the turn off in NGC 6397.

3. Which star mass gives you a T_{eff} and L most similar to what you found in step 1? This is the turn-off mass.

4. Now find the main sequence lifetime of the turn-off mass. Remember the main sequence ends when the star moves almost horizontally to lower T_{eff}. You can set time as a label on the HR diagram. This should be approximately the age of the cluster.

5. Find out the known age of the cluster and compare your estimate. This was a simple estimate so don't expect it to be too close, within the same order of magnitude is good. A more accurate result could be found by fitting evolutionary tracks to the data.

6. How does this compare to the age of the Universe? Can you now explain why the stars in NGC 6397 have low metallicity?

7. NGC 6397 is a globular cluster. Another type of cluster is an open cluster. Do some research into open and globular clusters, how do they differ?

11 Project reporting

In science, there are three main ways to report your results. You will be advised which of these is most suitable for you.

Written reports are the most common form of communication, used both formally and informally. The main astronomy journals are Monthly Notices of the Royal Astronomical Society, Astronomy and Astrophysics and the Astrophysical Journal. Dissertations written at the end of a degree, such a Masters or PhD, are also forms of written reports. Learning to write properly is a key skill required in almost every area of life. If you want to learn to write for science, I recommend reading the book *Eloquent Science* (Schultz, 2009, ISBN 978-1878220912).

Posters are used to communicate messages quickly and efficiently to many people. They are often made for conferences, where participants may peruse them during coffee breaks, or to put in the corridors of buildings so that interested people can view them as they walk by. While they are less formal than written reports, they should still contain the science and be made accurately and efficiently. Remember: a poster sells the work, there must be a big title and headlines.

Presentations are usually talks in lecture halls, similar to classes in schools. Their duration can be a few minutes to an hour, and usually involve a slideshow on a laptop presented on a big screen, e.g. with software like *Openoffice Impress* or *Powerpoint*. Recently, online talks have become more common because conferences are moving to online form (e.g. using *Zoom*, *Skype* or similar software). These often are placed online, e.g. on *YouTube*. Talks are an excellent way to communicate science and, because they allow questions from the audience, foster superior understanding and even may be considered a form of entertainment.

More information...

- https://academic.oup.com/mnras
- https://www.aanda.org/
- https://iopscience.iop.org/journal/0004-637X
- https://eloquentscience.com/
- https://www.qmul.ac.uk/spa/outreach/in-school/school-activities/research-in-schools/how-to/scientific-posters/
- https://www.youtube.com/watch/AwMFhyH7_5g
- https://www.openoffice.org/product/impress.html
- https://www.youtube.com/user/robizzard

References

Eggleton, P. P. 1971, MNRAS, 151, 351
Groenewegen, M. 2018, A&A, 619, A8
Hawking, S. W. 1974, Nature, 248, 30
Henyey, L. G., Forbes, J. E., & Gould, N. L. 1964, ApJ, 139, 306
Husser, T.-O., Kamann, S., Dreizler, S., et al. 2016, A&A, 588, A148
Kippenhahn, R., Weigert, A., & Weiss, A. 2012, Stellar Structure and Evolution
Lutgens, F. K. & Tarbuck, E. J. 1989, Essentials of geology, 3rd edn. (Columbus : Merrill Pub. Co), includes index
Minzner, R. A. 1977, Reviews of Geophysics, 15, 375
Payne, C. H. 1925, PhD thesis, Radcliffe college
Schultz, D. 2009, Eloquent science: a practical guide to becoming a better writer, speaker, and atmospheric scientist (American Meteorological Society)

A How stellar modelling works on a PC

We often talk of a "stellar evolution code". This is a computer program that models the equations of stellar structure using variables, just like our Fibonacci example above. Such a code requires data which describes mass as a function of temperature, density and chemical composition. These data include the equation of state, e.g. the ideal gas law, nuclear reaction networks and rates, and "opacity tables" which describe how much radiation matter absorbs. The stellar evolution code then uses a "starting model", which is a profile of temperature and density at the beginning of the star's life (when its composition is just that of the molecular cloud from which it is born), and evolves this forward in time. As a star evolves, its central abundance of hydrogen decreases even though at the surface it looks the same. Because a stellar evolution code models the changes in composition throughout the star, caused by either nuclear burning or mixing of layers in the star, it can model stars throughout their lives.

Stellar evolution codes trace their ancestry back half a century, to a seminal paper by Henyey et al. (1964). With the advent of electronic computers, these authors devised a way to solve the partial differential equations governing stellar structure and evolution. They imagined a series of shells centres on the centre of mass of the star. At each shell a set of equations was developed to describe the physics active inside the shell. Solutions of these equations give properties such as pressure and density at the shell's location, hence throughout the star.

Many evolution codes have been written based on the *Henyey method*, and various improvements to the method have been introduced over time. Among the most well-known historical codes are those by Eggleton, Kippenhahn and Paczynski — quite a few modern codes are essentially heavily modified versions of these. The Eggleton (1971) code was particularly innovative, in that it introduced an algorithm for automatic redistribution of shell locations. This allows stars to be evolved further into their developmental phases with only a few hundred shells in total, compared to thousands with other codes, which makes the code very fast.

Sometimes, when two stars are close to each other their gravity causes them to orbit each other. We call these binary stars. When they are **really** close, they can exchange mass (Roche Lobe Overflow), and evolve in ways single stars can't! One descendant of the original Eggleton code is the *TWIN* code. It allows simulation of binary evolution as well as single-star evolution. While the physics in TWIN is quite up to date, it is rather user-unfriendly. To address this issue, Robert Izzard and Evert Glebbeek have developed *Window To The Stars* (WTTS), a graphical interface to *TWIN*, which will be used in this course. All relevant information regarding this course and the *WTTS* software is available for free online at: http://personal.ph.surrey.ac.uk/~ri0005/window.html.

More information...

- https://users.monash.edu.au/~johnl/M4111/Lectures-1-2.pdf
- https://en.wikipedia.org/wiki/Roche_lobe