THE EFFECTS OF METALLICITY ON CONVECTIVE OVERSHOOT BEHAVIOUR IN MODELS OF δ SCUTI STARS

BY

VERONIKA DORNAN

A thesis submitted to the

Department of Physics

Mount Allison University

in partial fulfillment of the requirements for the

Bachelor of Science degree

with Honours

May, 2019

ABSTRACT

Delta Scuti variables are stars which exhibit periodic changes in their luminosity through radial and non-radial pulsations of their surfaces. The aim of this research is to gain a better understanding of these stars' behaviours through an expansion of the findings of Lovekin and Guzik [3], which found a strong correlation between Delta Scuti stars' pulsation constant (Q) as a function of effective temperature and the amount of convective overshoot within the star. However, only models with metallicities of $Z = 0.02$ were examined, prompting the question of if this relation is dependent on chemical composition. I collected the data in this research through creation of a grid of stellar models using the open-source coding language `MESA` [4], and analyzed the models pulsation properties using the complementary coding language `GYRE` [10]. By varying the models mass, rotation speed, convective overshoot, and metallicity it was determined that the relationship found by Lovekin and Guzik does hold for stars of other metal compositions and may have a quantifiable metallicity dependence, although statistically weak. In addition, a second relation between Q as a function of effective temperature and convective overshoot was discovered at higher effective temperatures as well. The cause of this new relation was found to be due to the deepest surface convection zone depth of the models.

ACKNOWLEDGMENTS

I would like to thank the Mount Allison University Physics Department for providing me with the support and skills necessary to tackle this thesis, and NSERC for funding my research over the previous summer that has been the basis of my thesis. I would also like to extend my deepest thanks of course to my supervisor Dr. Catherine Lovekin, who has been instrumental not only in providing me guidance in this thesis, but also being an incredibly valuable mentor to me in my final year at Mount Allison.

Contents

List of Figures

List of Tables

INTRODUCTION

1.1 Theory

δ Scuti Variables refer to a class of variable stars characterized by their position on the HR diagram. These stars' luminosities vary periodically, which can be photometrically observed from Earth. They achieve these luminosity variations through radial and non-radial pulsations of their surface. That is to say, δ Scuti variables' physical size varies slightly, causing their luminosity to vary in the same way, since their surface area is increasing and decreasing as their energy production remains constant. Radial pulsations occur when the star's surface expands and contracts uniformly, as shown in Figure 1.1. Non-radial pulsations occur when portions of the star's surface expand while others contract and visa versa, as shown in Figure 1.2.

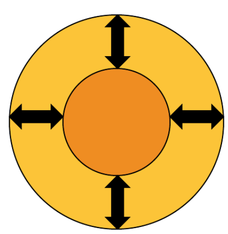

Figure 1.1: An example of radial pulsations; like inflating a basketball.

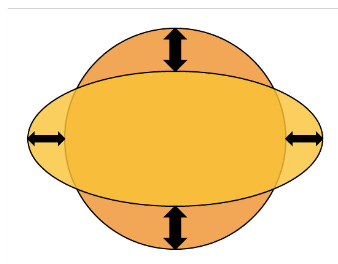

Figure 1.2: An example of non-radial pulsations; like a bouncy ball hitting a surface.

The pulsation periods of δ Scuti Variables are relatively short compared to other types of well-known variable stars such as Cephied or RR Lyrae variables. δ Scutis' periods can range from about 30 minutes to 5 hours, with typical brightness fluctuations between 0.003 to 0.9 magnitudes in the visual band of light, which is about 1% to 4% of the star's magnitude [3]. They are distinguished from other variable stars by their position on the HR diagram, where the instability strip crosses the zero-age main sequence. This position is shown in figure 1.3. Stars in this area have spectral classes between F8 and A2, and luminosity classes between V and III. δ Scutis' masses can range anywhere from 1.4 to 2.5 solar masses. This mass range acts as a transition region between the dominant form of energy transport within the star.

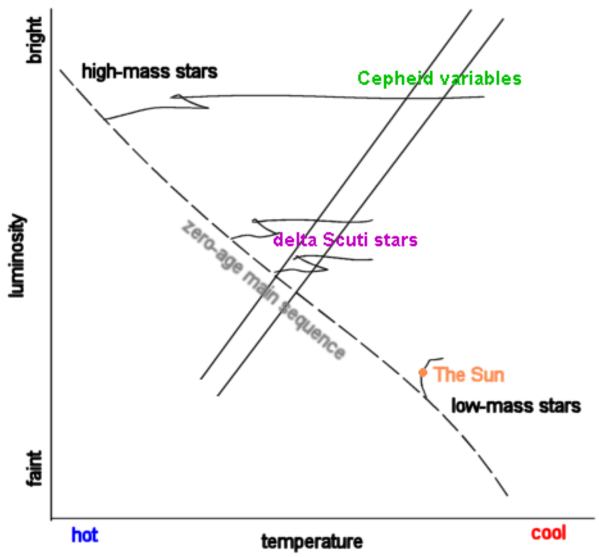

Figure 1.3: δ Scuti Variable stars' position as shown on a simple HR diagram, where the two parallel lines represent the instability strip. Image from [9].

There are two main methods for energy to be transported in stellar interiors; through radiation, or through convection. Radiation refers to energy being transported by photons outward from the core to the surface through random walks, and the resulting temperature gradient can be described by Equation 1.1.

$$\frac{dT}{dr} = \frac{3\rho(r)\kappa(r)L(r)}{64\pi\sigma_{SB}T(r)^3r^2} \tag{1.1}$$

In this equation $\rho(r)$ is the density, $\kappa(r)$ is the opacity, $L(r)$ is the luminosity, and $T(r)$ is the temperature of the star, all of which are functions of the star's radius, r.

In this equation σ_{SB} refers to the Stefan-Boltzmann constant [8].

On the other hand, convection refers to energy being transported by bulk motion of the fluid, going from the inner region of the star to the outer region and back again. The temperature gradient via convection can also be described mathematically:

$$\frac{dT}{dr} = \left(1 - \frac{1}{\gamma}\right)\frac{T(r)}{P(r)}\frac{dP}{dr} \tag{1.2}$$

Here γ is the adiabatic constant for an ideal gas, $T(r)$ is the temperature as a function of radius, and $P(r)$ is the pressure as a function of radius. $\frac{dP}{dr}$ is the pressure gradient, given by the equation of hydrostatic equilibrium;

$$\frac{dP}{dr} = -\frac{GM(r)\rho(r)}{r^2} \tag{1.3}$$

where G is the gravitational constant, $M(r)$ is the cumulative mass inside the star as a function of radius, r, and $\rho(r)$ is the density of the star, which is also a function of radius. This equation describes how equilibrium is achieved when the force of gravity acting on a volume of fluid is equal to the outward pressure of the fluid. For most of a star's evolution and lifetime it is in hydrostatic equilibrium so it does not collapse due to gravity or have its atmospheres evaporate due to pressure [8]. It should be of note that Equations 1.1, 1.2, and 1.3, are all one-dimensional equations and assume that the star they are describing is spherically symmetric, which is reasonable.

Both convective and radiative transport occur in specific regions within the star, and the position of these regions inside the star is determined by the star's mass. For example, stars of low mass (less than 0.35 solar masses) have no radiative zone, and transport all their energy convectively. Stars of slightly higher masses, like our Sun, have a radiative core surrounded by a convective envelope. However, much more massive stars transport their energy through a convective core and a radiative envelope instead. Figure 1.4 illustrates this relationship between mass and interior structure. In stellar models, convective zones are calculated based on a numerical stability criterion. In these models, the Ledoux criterion is used, which states that a layer of material within a star is stable against convection if:

$$\nabla_{rad} < \nabla_{ad} + \frac{\phi}{\delta}\nabla_{\mu} \tag{1.4}$$

where T is the temperature, P is the pressure, ρ is the density, μ is the chemical composition, and

$$\nabla_{rad} = \left(\frac{dln(T)}{dln(P)}\right)_s, \nabla_{ad} = \left(\frac{dln(T)}{dln(P)}\right)_s, \phi = \left(\frac{\delta ln(\rho)}{\delta ln(\mu)}\right)_{P,T},$$

$$\delta = \left(\frac{\delta ln(\rho)}{\delta ln(T)}\right)_{\rho,\mu}, \nabla_{\mu} = \left(\frac{dln(\mu)}{dln(P)}\right)_s$$

with s indicating the surroundings and e indicating an element of the layer [12].

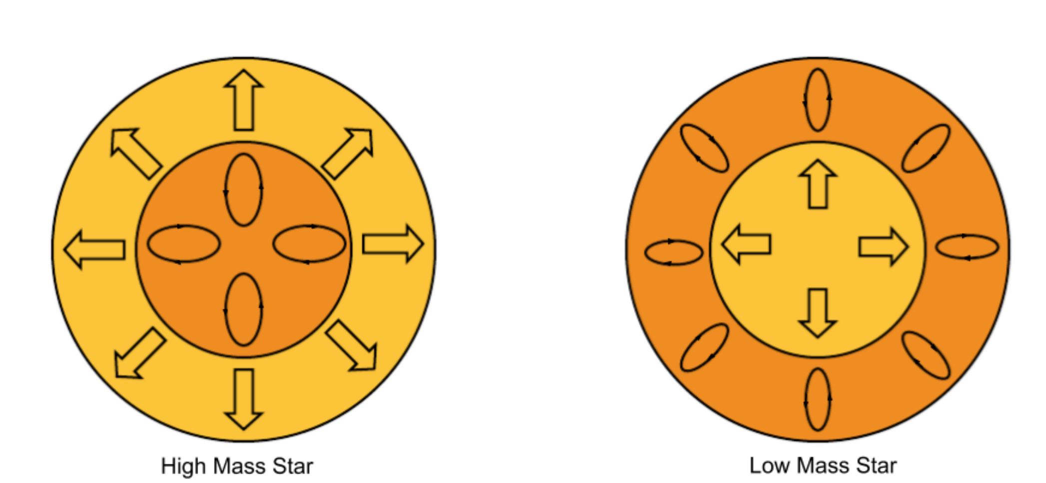

Figure 1.4: A cross section through representative stellar interiors. The arrows represent radiative transport, and the ovals represent convective transport in this case.

The region in the star where the radiative zone and the convective zone meet is called the convective boundary. However, the fluid being moved by the convective cells does not necessarily stop at the convective boundary. This is because although the acceleration of the material may have gone down to zero by the time it reaches the boundary, it still has momentum bringing it past the boundary and into the next zone of the star's interior before coming back. The amount of fluid that goes past the boundary into the radiative zone is referred to as the star's convective overshoot. The amount of convective overshoot a star has can affect its lifetime if it has a convective core, since the area where convective overhooting occurs can be treated as an extension of the core itself.

δ Scuti Variables' pulsations are caused by the κ mechanism, in which κ refers to the star's radiative opacity. In δ Scuti stars, this mechanism occurs in the He II Ionization Zone of the stellar atmosphere. As the helium-rich atmosphere of the star heats, its helium eventually becomes ionized, which causes the opacity of the atmosphere to increase, blocking photons from escaping. As the opacity of the layer of the star's atmosphere increases, it becomes more difficult for the energy being produced in the interior to escape, resulting in a build-up of energy, and as a result, an increase in heat. This creates a type of positive feedback loop, since as the energy within the star builds up, the He II ionization zone becomes more ionized and less energy is allowed to escape. As the energy and heat within the star increases, the pressure also increases until it needs to be released, causing the layer of the ionized helium atmosphere to expand and cool. Eventually the layer will cool, the opacity will drop, and the layer will sink back to its original position and opacity, and the process will start again. This mechanism only affects variable stars, since their ionization zone is at just the right depth to drive these pulsations [11].

Many types of variable stars' pulsations, including δ Scutis, are known to adhere to a period-mean density relation. Through this relation the period of a variable star's pulsations and the star's mean density can be related to yield a pulsation constant which can help compare stars of similar size and structure. This relation is shown in equation 1.5 [2].

$$Q = \Pi\left(\frac{\overline{\rho}}{\rho_{\odot}}\right)^{\frac{1}{2}} \tag{1.5}$$

In this equation Q represents the pulsation constant, Π represents the period of the star's pulsation in days, $\overline{\rho}$ represents the mean density of the variable star, and $\overline{\rho_{\odot}}$ represents the mean density of the sun. Equation 1.5 is used to calculate the pulsation constants in the research described in this thesis.

1.2 Original Research

The research presented in this thesis is an expansion upon the work of Lovekin and Guzik in 2017 [3]. Although models of both δ Scuti and γ Doradus variable stars were used originally, for the purposes of my work I have focused on this research's applications in a better understanding of δ Scuti stars specifically. In their research, Lovekin and Guzik created a grid of stellar models of masses ranging from $1.2\odot - 2.2\odot$, rotation rates from $\Omega/\Omega_c = 0.0 - 0.4$, and convective overshoot parameters from $0.0 - 0.1$, and allowed these models to evolve along the main sequence and then analyzed their pulsation properties. They found a strong correlation between δ Scuti stars' pulsation constants as a function of effective temperature and their convective overshoot. This relationship can be seen in Figure 1.5, and is best illustrated by the trend of the first radial order, represented by the top, blue curve.

It was found that the behaviour of the pulsation constant as a function of the effective temperature in log-log space could be described as a piece-wise function, with a break point at $\log(T_{eff}) \approx 3.84$. For temperatures higher than this point the slope of the function remains flat, while for temperatures lower than this point the slope of the function, which will be referred to as m_{low}, varies with the convective overshoot of

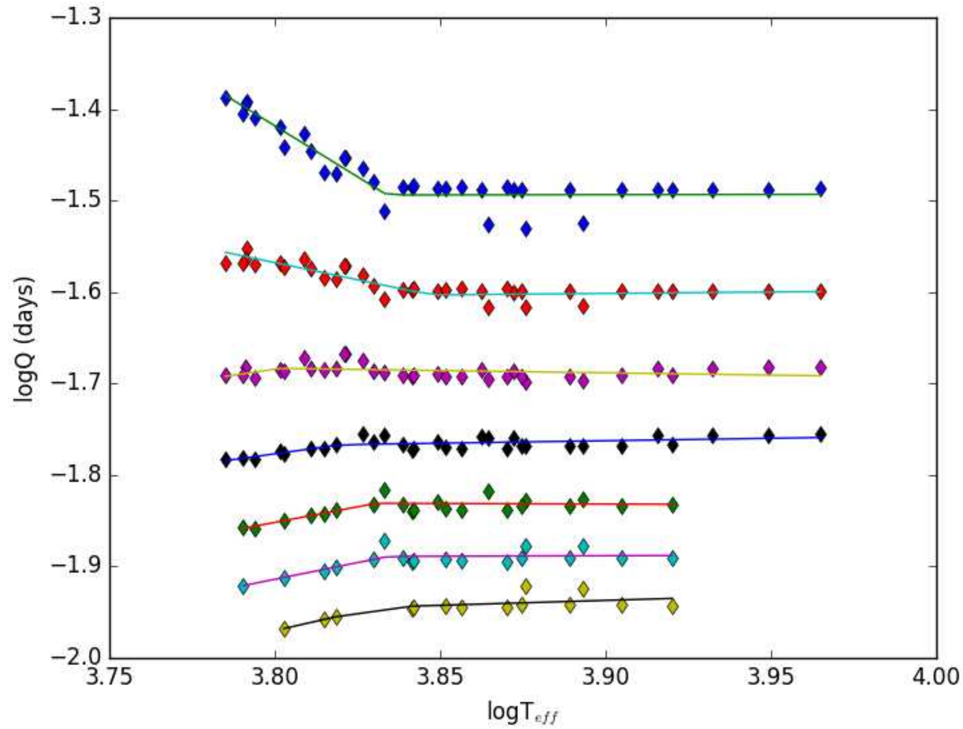

Figure 1.5: $\log(Q)$ plotted against $\log(T_{eff})$ for zero overshoot, zero rotation, and $Z = 0.02$ with the first seven radial orders in ascending order [3].

the model. The relationship between the behaviour of m_{low} and convective overshoot (f_{ov}) can be described with a linear fit:

$$m_{low} = (12 \pm 2) f_{ov} - (2.8 \pm 0.1) \tag{1.6}$$

The cause of this variation in the pulsation constants was determined by plotting the fractional depth of the convective envelope of the models against the log of their effective temperature. It was found that the convective envelope remained shallow at a relatively constant depth for models of effective temperatures higher than $\log(T_{eff}) = 3.84$, which corresponds to the break-point of the $Q - T_{eff}$ behaviour. Models of effective temperature lower than this point had convective envelopes whose depth corresponded to their temperature. This behaviour can be seen in Figure 1.6, showing that the change in m_{low} is caused by convective zone depth, which is dependent on convective overshooting.

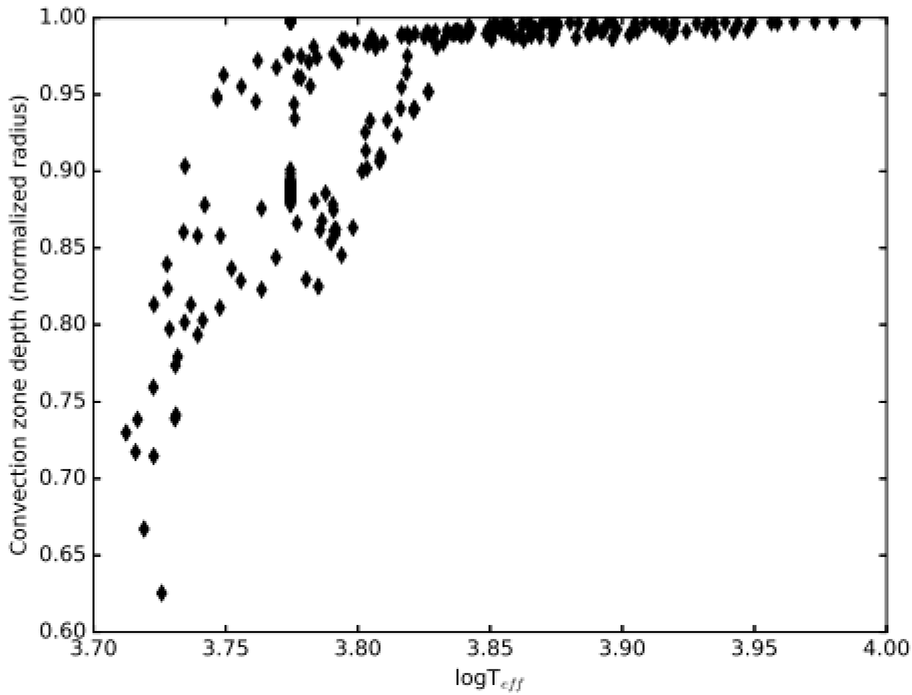

Figure 1.6: Normalized convective zone depth plotted against $\log(T_{eff})$ for all original models [3].

However, in this original research only models of solar-like composition, that is to say $Z = 0.02$, were used. Composition is defined in terms of a star's metallicity content, Z, which refers to the mass fraction of the star that is not Hydrogen or Helium. This raises the question of whether or not this $m_{low} - f_{ov}$ relationship is affected by metallicity, and if that effect can be quantified. In this work, I investiagte this question in order to understand how the metal composition of δ Scuti stars can affect their overshoot behaviour and pulsations as they evolve along the main sequence. This will give important insight into the interior structure of these types of variable stars.

MODELS

2.1 MESA

The models analyzed in this research were created using the same method as those in the original work [3]. They were made using an open-source program called `MESA`, which stands for Modules for Experiments in Stellar Astrophysics [4] [7] [5] [6]. `MESA` is a suite of open-source libraries to be used for modelling a variety of stellar structures. The specific module used for this research is `MESA star`, a 1-D stellar evolution module which runs on FORTRAN 95. `MESA` utilizes the standard Mixing Length Theory (MLT) of convection to model energy transport within the models' interiors [1].

MLT is a 1-D model which replaces the complicated and chaotic behaviour within a convective zone with a group of average convective elements. These elements all share the same physical properties at a given radial distance from the stellar centre, r, thus making convection much easier to model. The "mixing length" part of Mixing Length Theory refers to the assumption that each of these elements travels, on average, a distance of Λ, the mean free path, before mixing with the surrounding fluid. The mean free path used in these models is shown in equation 2.1.

$$\Lambda = \alpha\left(\frac{P}{G\rho^2}\right)^{\frac{1}{2}} \tag{2.1}$$

Where α is the mixing length parameter, P is the pressure, G is the gravitational constant, and ρ is the density. All the elements are also assumed to have the same speed, $\overline{v}$, the approximate average speed of all the elements at a distance r. With all of these assumptions put together, the mean life of a convective element is equal to $t = \frac{\Lambda}{\overline{v}}$, which refers to the amount of time between when the element is created and when it loses its identity to mixing with the other fluids in the convective zone. Although MLT utilizes many assumptions and simplifications in order to model convection, it is the most popular theory among astronomers to model stellar convection. This is because of the difficulty associated with modelling convection in stellar evolution. Convection is a 3D process occurring over short timescales, but which effects matter over large timescales, such as stellar evolution, and is only more easily modelled in 1D. Despite this, MLT yields excellent large scale results, which are sufficient for this research.

`MESA` treats the convective mixing within the model as a diffusive process and calculates diffusion coefficients, D, for the `MESA star` code at various points within the star. To account for convective overshoot mixing, as discussed in section 1.1, `MESA star` determines an overshoot mixing diffusion coefficient, equation 2.2, after the MLT calculations are complete.

$$D_{ov} = D_0 exp\Big(\frac{-2r}{f_{ov} H_P}\Big) \tag{2.2}$$

Here D_0 is the diffusion coefficient, as calculated by `MESA` using MLT, at the convective boundary, r is the radial distance within the star from the core boundary, f_{ov} is the convective overshoot parameter, and H_P is the pressure scale height at the core boundary, defined as:

$$H_P = \frac{kT}{mg} \tag{2.3}$$

Where k is Boltzmann's constant, T is the mean atmospheric temperature in Kelvins, m is the mean mass of a molecule in the atmosphere, and g is the acceleration due to gravity at the point H_P is being calculated at. The convective overshoot parameter in equation 2.2 is allowed to vary between 0 and 0.1 for these models in steps of 0.02. This parameter determines the amount of convective overshooting present in the stellar model both above convective cores and below convective envelopes. Although these can be set separately, the same value was used for all convective regions in this work.

`MESA star` also allows for variation in the rotation rate of the models. The rotation of the models in this research was achieved by imposing a surface rotation rate and forcing the zero-age main sequence model to be uniformly rotating. `MESA` uses a shellular approximation for rotation, which allows it to include rotation in a pseudo 1D fashion. This is necessary since `MESA` must model a 2D effect in a 1D model. The rotation rate of the models is parameterized as Ω/Ω_c, where Ω is the surface angular velocity and Ω_c is the critical rotation. The critical rotation can be defined as the rotation rate at which the gravitation force of the model is balanced by the centripetal acceleration at its equator. In this research we allow the rotation parameter to vary between 0 and 0.4, however it should be of note that this differs from the range used in the original research. The rotation parameter was allowed to vary between 0 and 0.6 originally, however, it was found that above 0.4 Ω/Ω_c the frequencies of the p-modes of the models can vary by more than 10% from each other, and as such are not considered to be reliable.

`MESA star` runs by reading input from two FORTRAN namelist files. The first file has information outlining the evolutionary calculations to be preformed by `MESA star`, the opacity data, the chemical composition and nuclear network of the model, among other properties of the input model before it is evolved by the program. The second file simply specifies the controls to be applied to the model as it is evolved. Once `MESA star` finishes evolving a model, it produces two kinds of output files. The first output file is a log, which records the properties of the model as it evolved over time, including age and mass. The second type of output file is a profile, which records properties of the model on a timestep, which can be varied by the user, at each zone within the model from the surface to the centre. In this research the timestep was set to 100 as the model began evolving, and was switched to 30 after the model reached the main sequence. Since the main sequence evolution of the models were what I was interested in in this research, the models stopped evolving once they left the main sequence. The girds of models created in this research have varying solar masses, rotation rates, and amounts of convective overshoot, which are the same as the models in the original research, but with three new grids of models with varying metallicity compositions. These parameters used can be seen below.

Masses ($M_\odot$)	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0	2.1	2.2

Convective Overshoot (f_{ov})	0.0	0.02	0.04	0.06	0.08	0.1

Rotation Rates (Ω/Ω_c)	0.0	0.1	0.2	0.3	0.4

Metallicities (Z)	0.04	0.01	0.001

In order to understand how these parameters can affect how a star evolves along its main sequence, Figures 2.1 - 2.4 have been provided. Each figure is an HR diagram in which four of the five parameters are kept constant, and the evolution tracks illustrate the effects of the varying parameter.

2.2 GYRE

Once the models were evolved along their main sequence by `MESA star`, their profiles where converted from `.data` files to `.data.GYRE` files in order to be read by the complementary open-source stellar oscillation code, `GYRE` [10]. `GYRE` is able to solve both adiabatic and non-adiabatic pulsation equations for stellar models using a Magnus Multiple Shooting (MMS) numerical scheme. Adiabatic refers to a process in which

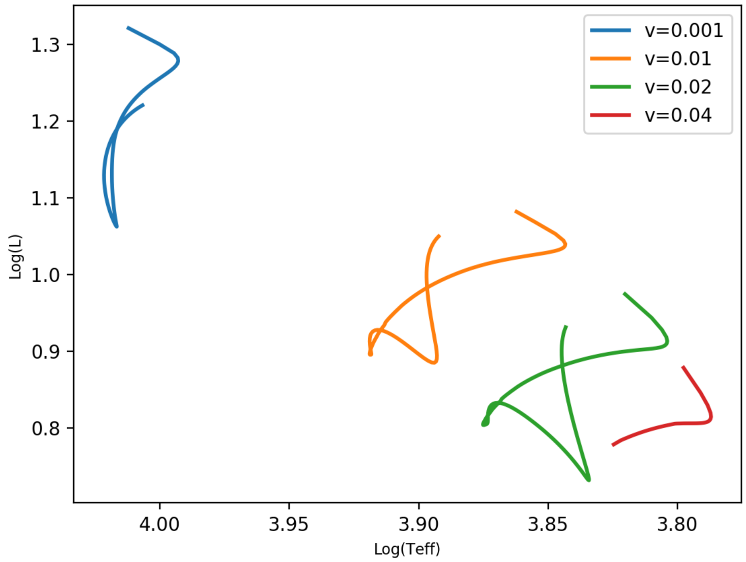

Figure 2.1: An HR diagram for varying metallicities with mass= 1.6$\odot$, $f_{ov} = 0.0$, and $\Omega/\Omega_c = 0.0$.

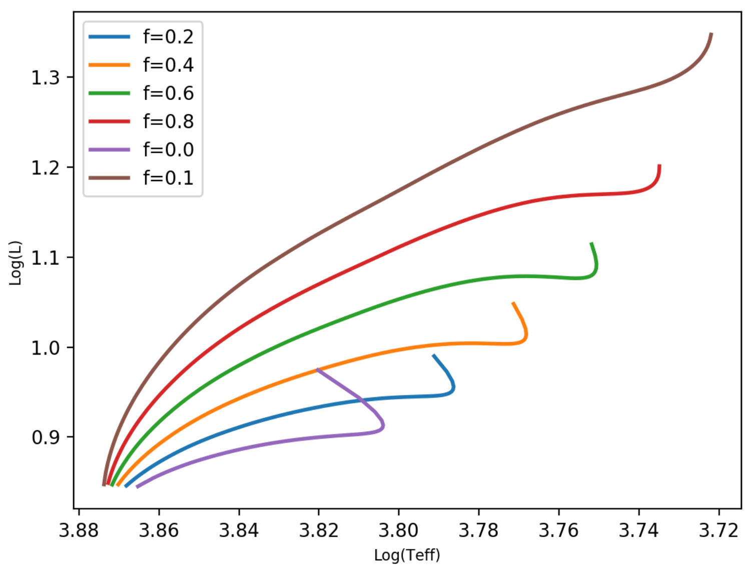

Figure 2.2: An HR diagram for varying convective overshoots with mass =1.6$\odot$, $Z = 0.02$, and $\Omega/\Omega_c = 0.0$.

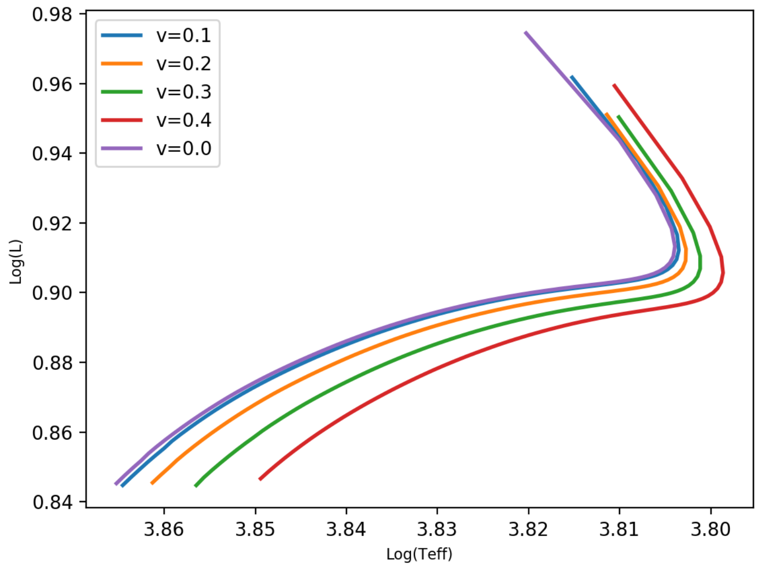

Figure 2.3: An HR diagram for varying rotation rates with mass=1.6$\odot$, $Z = 0.02$, and $f_{ov} = 0.0$.

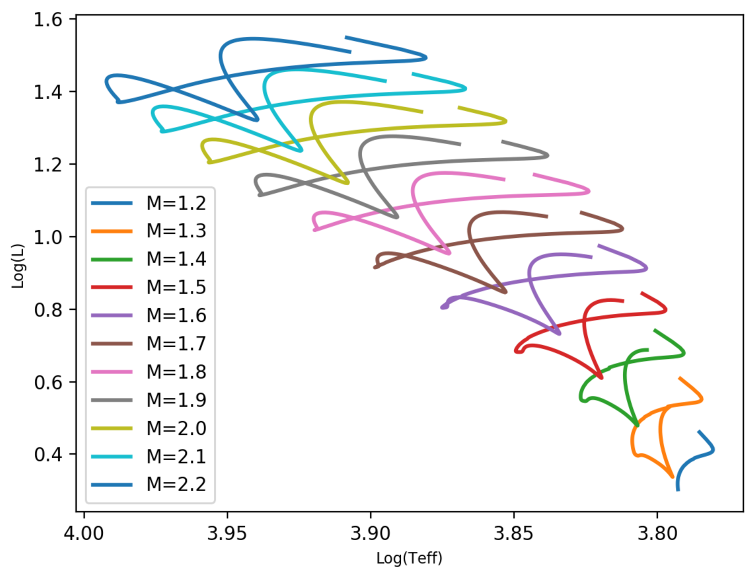

Figure 2.4: An HR diagram for varying masses with $f_{ov} = 0.0$, $Z = 0.02$, and $\Omega/\Omega_c = 0.0$.

there is no transfer of heat or mass between the system and the surroundings, and non-adiabatic refers to when a process where this transfer does occur. These pulsation equations are a type of boundary-value problem (BVP), in which the oscillation frequency, ω, acts as an eigenvalue. The MMS scheme then solves this BVP for the eigenfrequencies and eigenfunctions of the input stellar model. Once this is completed `GYRE` outputs a text file with data on the model's effective temperature, mass, radius, and oscillation frequencies for the time in the model's evolution corresponding to the input file data. For the purposes of this research the adiabatic data from the output file was used for analysis since it is much simpler to compute and because for the purposes of this research we are only interested in the adiabatic results.

RESULTS AND DISCUSSION

Once data was obtained by the creation of the `GYRE` output files, this data was analyzed by a `Python` script (see Appendix). When evolving the models along the main sequence, five evenly spaced profiles were selected for each to be used for data analysis. These were chosen because the lifetimes of the models varied, thus producing a varying number of profiles between models for the timestep used. It would have been too time consuming to use all of the profiles created for each model, so it was determined that using five evenly spaced profiles for each would be the most efficient way to analyze the data. The first function of the script read these five `GYRE` output files for each model and calculated the pulsation constant, Q, using Equation 1.5 with the mass, radius, effective temperature, and pulsation data. All of these pulsation constants were then written onto a summary file with their corresponding mass, overshoot parameter, velocity parameter, metallicity, effective temperature, radius, pulsation frequency, and profile number, which corresponded to the model's age. Next, a new function read the data from the summary file and used it to plot the log of the pulsation constants of all profiles of all models with the same overshoot parameter, velocity parameter, and metallicity against the log of their effective temperatures. The data created a piece-wise function, and a corresponding trend-line was added to the plot. An example of one of these plots can be seen in Figure 3.1.

3.1 m_{low} Results

It was then necessary to find the dependency of the first slope in the piece-wise function, m_{low}, on convective overshoot for each metallicty, as was done in the original research. In order to achieve this, the average slope for m_{low} across all masses and

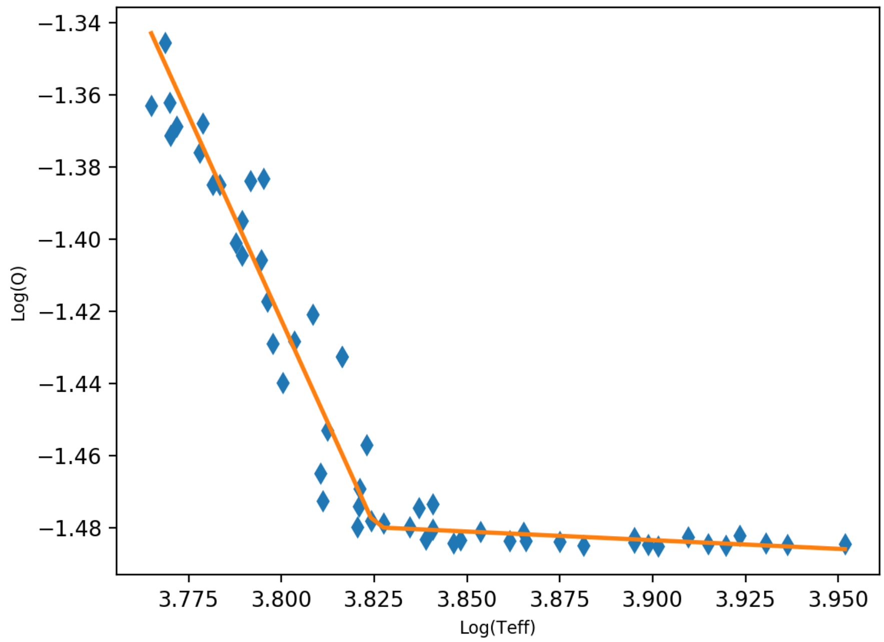

Figure 3.1: Calculated $\log(Q)$ values for zero overshoot, zero rotation models at $Z = 0.04$ plotted against $\log(T_{eff})$.

rotation parameters was calculated for each overshoot parameter and metallicity, and was plotted against overshoot parameter. This yielded two plots, one for $Z = 0.01$, shown in figure 3.2, and one for $Z = 0.04$, shown in figure 3.3. Results for the $Z = 0.001$ data will be discussed in section 3.2.

By taking the equations for the linear fits of the plots, a convective overshoot dependency, similar to that found in the original data, was quantified for $Z = 0.01$, seen in Equation 3.1, and for $Z = 0.04$, seen in Equation 3.2. To more clearly illustrate the effect of metallicity, the convective overshoot dependency for $Z = 0.02$, Equation 1.6, is shown here again.

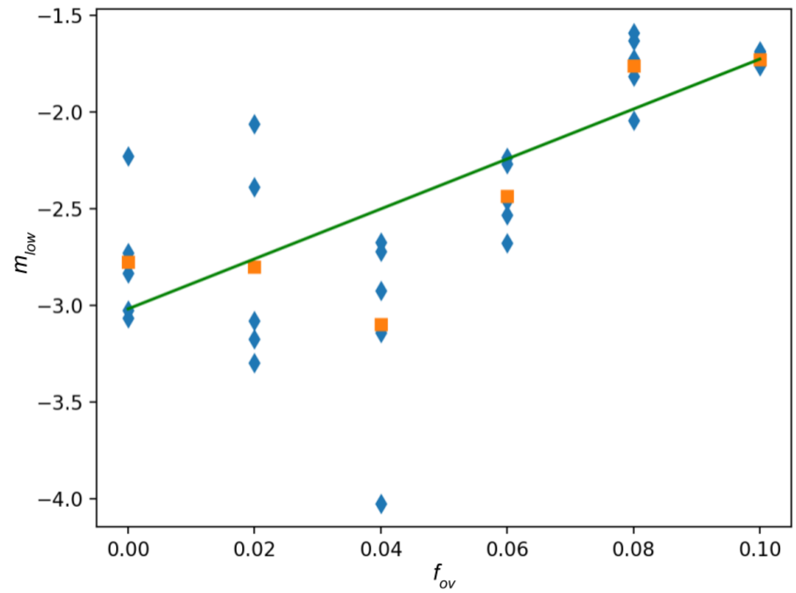

Figure 3.2: Average m_{low} plotted against f_{ov} for $Z = 0.01$ where the diamonds are the slopes for individual plots and the squares are the averages.

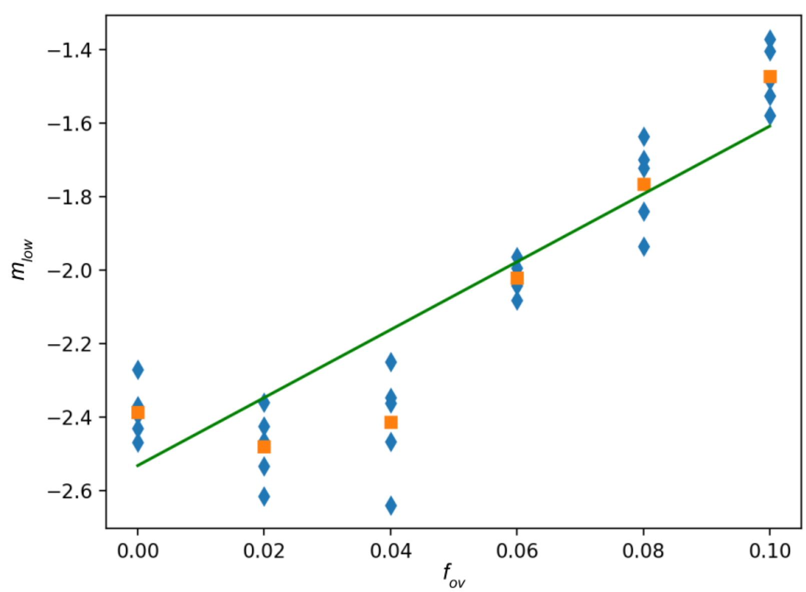

Figure 3.3: Average m_{low} plotted against f_{ov} for $Z = 0.04$ where the diamonds are the slopes for inidvidual plots and the squares are the averages.

$$m_{low} = (12.92 \pm 2.54) f_{ov} - (3.02 \pm 0.25) \tag{3.1}$$

$$m_{low} = (12 \pm 2) f_{ov} - (2.8 \pm 0.1) \tag{1.6}$$

$$m_{low} = (9.23 \pm 1.73) f_{ov} - (2.53 \pm 0.10) \tag{3.2}$$

For $Z = 0.04$, $Z = 0.02$, and $Z = 0.01$ respectively, Equations 3.1, 1.6, and 3.2 together show that there is indeed a metallicity dependency on the equations' first and second term, which will be here referred to as $term_1$ and $term_2$ respectively. The errors in these equations were determined by the standard deviation in the fit of Figures 3.2 and 3.3, and the same method was used in the original work to determine the error in Equation 1.6. It should be of note that $term_1$ and $term_2$ agree within the error bars for these three equations, so although there is a metallicity dependency, it may perhaps not be a statistically significant one. In order to quantify the metallicity dependence of the relations, the same method was used as determining the convective overshoot dependency. The values of $term_1$ and $term_2$ were both separately plotted against the metallicity for their relations. This can be seen in figures 3.4 and 3.5.

Equations 3.3 and 3.4 are the weighted least square fitting functions for figures 3.4 and 3.5 respectively, where $term_3$ and $term_4$ refer to the slopes of the plots.

$$term_3 = (-127.57 \pm 93.85) Z + (14.38 \pm 2.78) \tag{3.3}$$

$$term_4 = (14.40 \pm 6.31) Z - (3.10 \pm 0.19) \tag{3.4}$$

By taking the equations of the fits of each of the plots and putting them back in for $term_1$ and $term_2$ in the m_{low}-f_{ov} relationship, equation 3.5 was obtained.

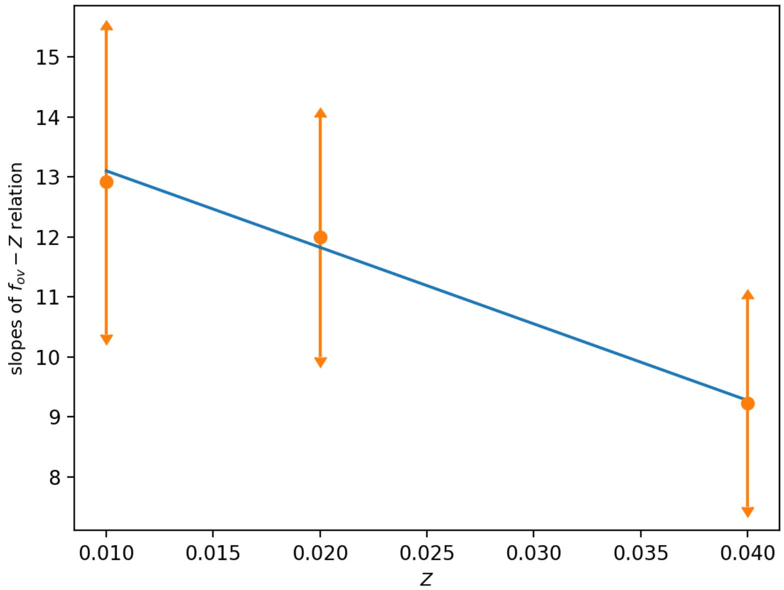

Figure 3.4: The slopes of the $f_{ov} - Z$ relation plotted against all convective overshoots.

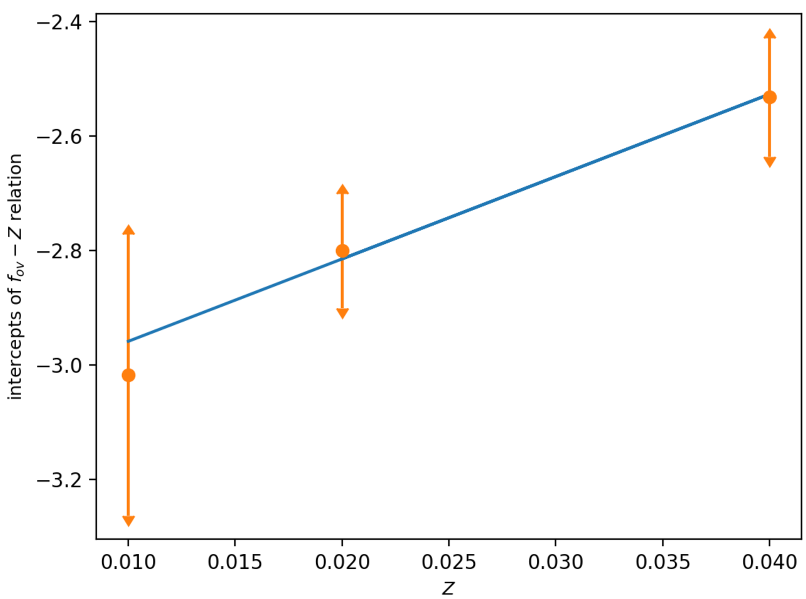

Figure 3.5: The intercepts of the $f_{ov} - Z$ relation plotted against all convective overshoots.

$$
\begin{aligned}
m_{low} = & [(-127.57 \pm 93.85)Z + (14.38 \pm 2.78)]f_{ov} \\
& + [(14.40 \pm 6.31)Z - (3.10 \pm 0.19)]
\end{aligned}
\tag{3.5}
$$

Equation 3.5 shows a clear, quantified relation between metallicity and the correlation between δ Scuti pulsation constants as a function of effective temperature and their convective overshoot.

3.2 m_{high} Results

When plotting the pulsation constants of the $Z = 0.001$ models against effective temperature, due to the significantly lower metal composition of these models compared to the other discussed in this paper there were either very few or no models with $\log(T_{eff})$ values lower than 3.8, which was approximately the break point for each of the piece-wise functions that best fit the higher metallicity models data. This increase in temperatures along the main sequence evolution with lower metallicity was to be expected, and can be seen in Figure 2.1. Because of this there was not enough data to draw any relationships between pulsation constants as a function of effective temperature, or m_{low}, and convective overshoot at low metallicity. At this metallicity however, as can be seen in figure 3.6, with this shift in effective temperature range of the main sequence evolution of the models, an uptick in the data was found at temperatures higher then $\log(T_{eff}) = 4.05$, which was not seen in the plots for the higher metallicity models.

The behaviour of this uptick is similar to that of m_{low}, in that its slope, which will be referred to as m_{high}, varies with convective overshoot although it is much smaller in amplitude. It should be noted that the scatter in Figure 3.6 is not significantly

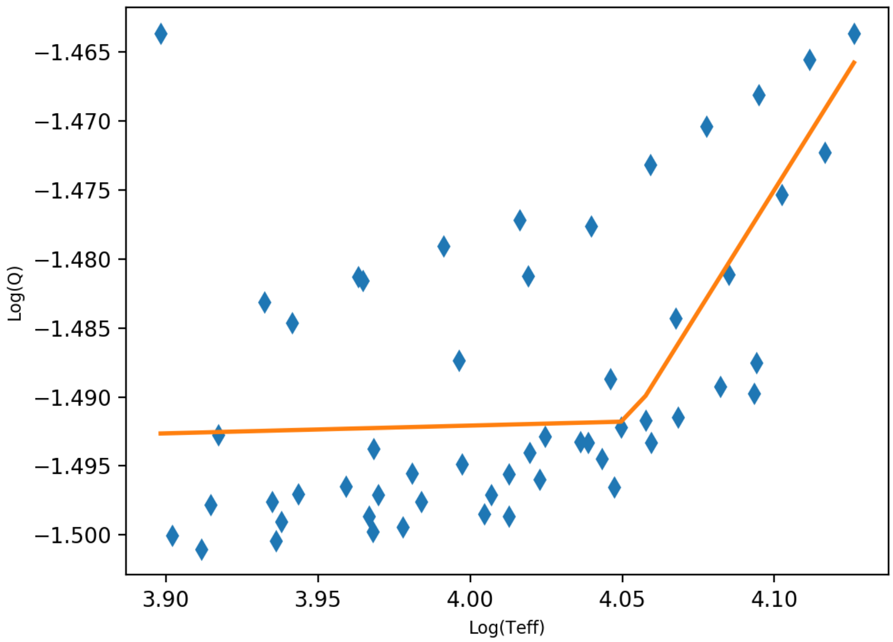

Figure 3.6: $\log(Q)$ against $\log(T_{eff})$ for zero rotation, 0.02 overshoot, and $Z = 0.001$.

greater than in Figure 3.1, it is just more obvious due to the smaller scale of the plot. To find the relationship between m_{high} and convective overshoot the same methods were used as with the higher metallicity models. This is shown in Figure 3.7. The average values of m_{high} were plotted against their corresponding convective overshoot parameters and the linear equation of the fit was used to determine the convective overshoot dependency of the slope. Equation 3.6 below shows that relation.

$$m_{high} = (-1.16 \pm 0.64) f_{ov} + (0.37 \pm 0.03) \tag{3.6}$$

Since only the $Z = 0.001$ metallicity content models had data at $log(T_{eff})$ values higher than 4.05, the behaviour of m_{high} could not be investigated with stars of other metallicities. As such, a metallicity dependence could not be determined for m_{high} in this research.

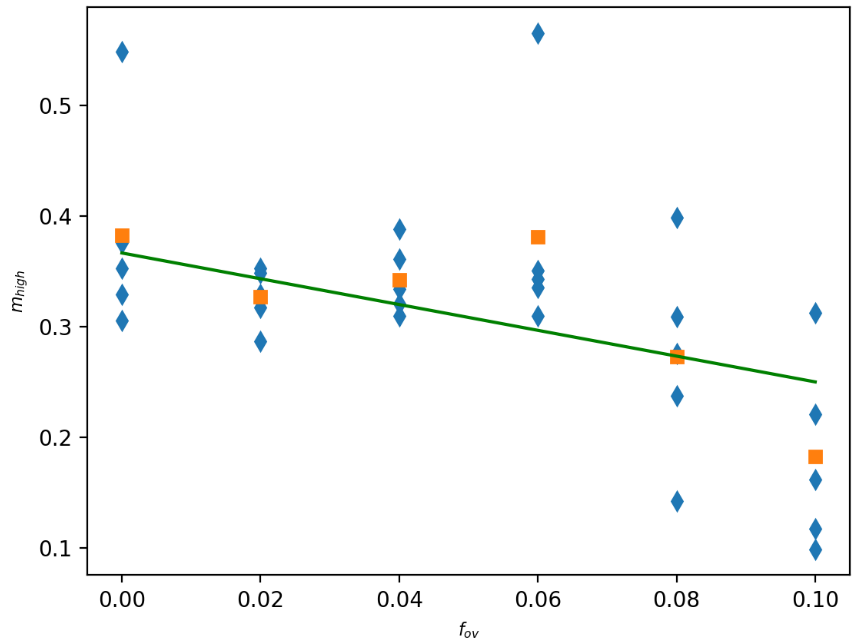

Figure 3.7: m_{high} plotted against f_{ov} for $Z = 0.001$.

3.2.1 Branching Effect

Looking at Figure 3.6 again, one can notice two sets of data points branching off from one another. This behaviour is seen for all plots of $Z = 0.001$ models. It can be seen more clearly at a higher convective overshoot and when the data points in each set are distinguished from one another, as in Figure 3.8. It was determined that for all of the plots the cause of this branching effect is the average density of the profile, which the data point represents. An estimate of the value of this threshold average density was determined for each convective overshoot, since it appeared to vary slightly with this parameter. Rotation rate did not seem to play any significant role in affecting the value. Models with densities higher than the first threshold densities and lower than the second threshold densities are part of the red data set in Figure 3.8.

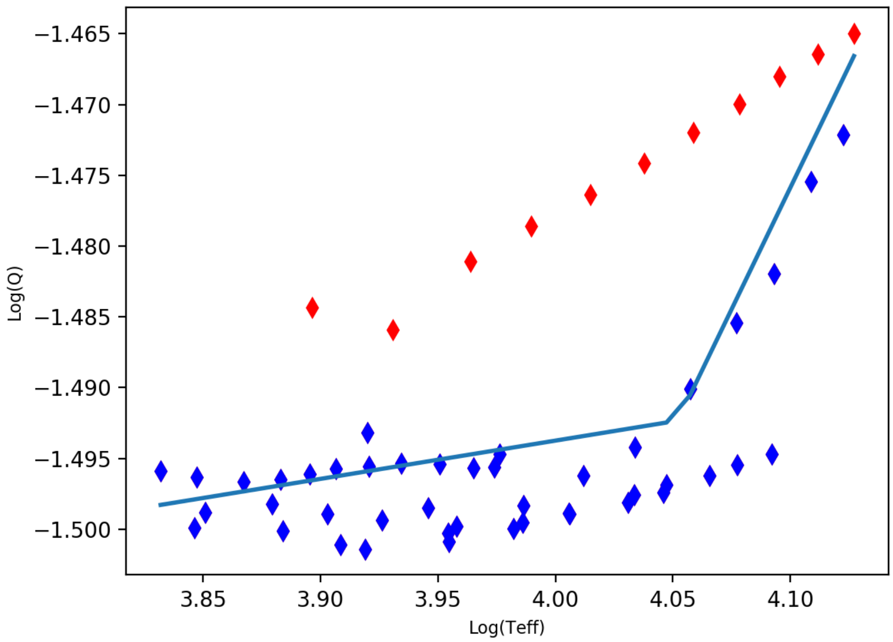

Figure 3.8: $Log(Q)$ against $Log(T_{eff})$ for zero rotation, 0.06 overshoot, and $Z = 0.001$. The red represents high densities and the blue represents lower densities. The piece-wise fitting function is for both sets of points.

Convective Overshoot	Threshold Density ($\frac{\bar{\rho}}{\rho_{\odot}}$)
0.0	1.1564 ± 0.0794
0.02	1.2179 ± 0.0165
0.04	1.1504 ± 0.0799
0.06	1.1164 ± 0.0277
0.08	0.8584 ± 0.2327 and 0.0303 ± 0.0029

It is unknown whether or not there exists a second, lower threshold density for the other convective overshoots as there were no profiles with ratios that low in the models in this research. One should also notice that there is no threshold density for $f_{ov} = 0.1$. This is because since utilizing these $f_{ov} = 0.1$ models in determining convective overshoot and metallicity dependence in this research, it has since been noticed that this level of overshooting it extremely high, and is unlikely to be seen in real stars. As such, determining the threshold density for this overshoot is not really necessary, but these models are still used in the work done earlier in this thesis. The cause of this branching effect could not be determined in this work due to time constraints, but it is likely there is a more fundamental characteristic of the models that is affecting the density, and thus causing the branching.

3.2.2 m_{high} Cause

In the original research done by Lovekin and Guzik, after discovering and quantifying the $m_{low} - f_{ov}$ relation they were able to identify the cause of this behaviour. In this research, an attempt was made to identify the cause of m_{high} in a similar way. First, the uppermost surface convective zone depth of all of the $Z = 0.001$ models was plotted against their $log(T_{eff})$ values, the same method that was used for m_{low}. This plot is shown in Figure 3.9. From this plot it is clear that there is some sort of interesting change in surface convection zone depth occurring around the break-point ($Log(T_{eff}) \approx 4.05$) associated with the m_{high} relation. It is not clear however, if this change in behaviour is what is causing m_{high}, so the convection zone depths of models from just before and after the break-point were investigated in order to provide some insight. This was achieved by plotting the Log of the models' convection zone depth against their normalized radius to determine how many and where their convection zones were within their interiors, as shown in Figure 3.10. This figure shows that there are several separate convection zones located at the surface of the models, and for models hotter than the break-point the deepest of these convective zones vanishes. In order to confirm this behaviour for all models used in this research, the normalized deepest surface convection zone depth was plotted against $Log(T_{eff})$ for all $Z = 0.001$ models, and is shown in Figure 3.11. This confirms that at the break-

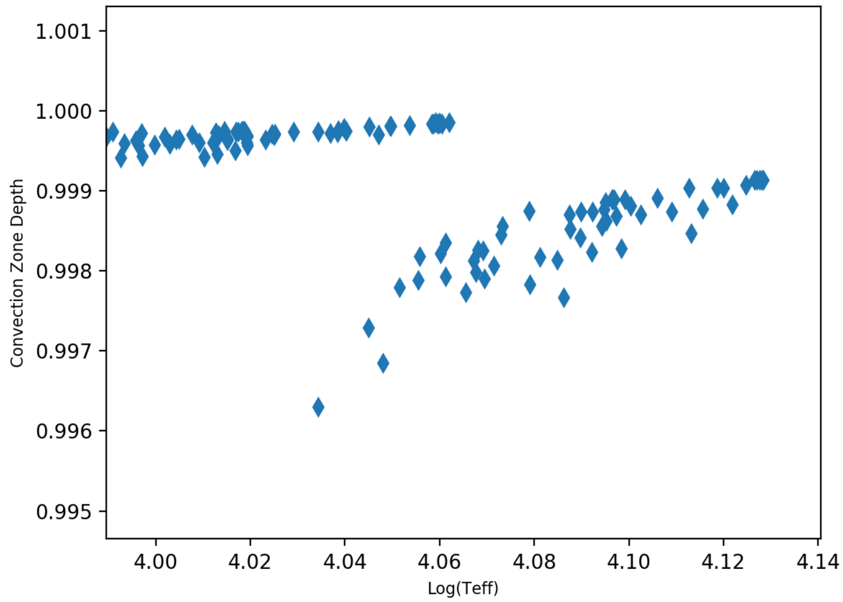

Figure 3.9: Normalized uppermost surface convection zone depth plotted against $Log(T_{eff})$ for all $Z = 0.001$ models.

point the deepest surface convection zone moves closer to the surface of the star. Since the pulsation modes only propagate in the radiative zones, a change in the depth of the convective zones will alter the size of the available pulsation area within the star. This change will affect the frequency, and hence the pulsation constant. There can be more to understand about this effect than this research is able to determine, and more investigation is needed in future work.

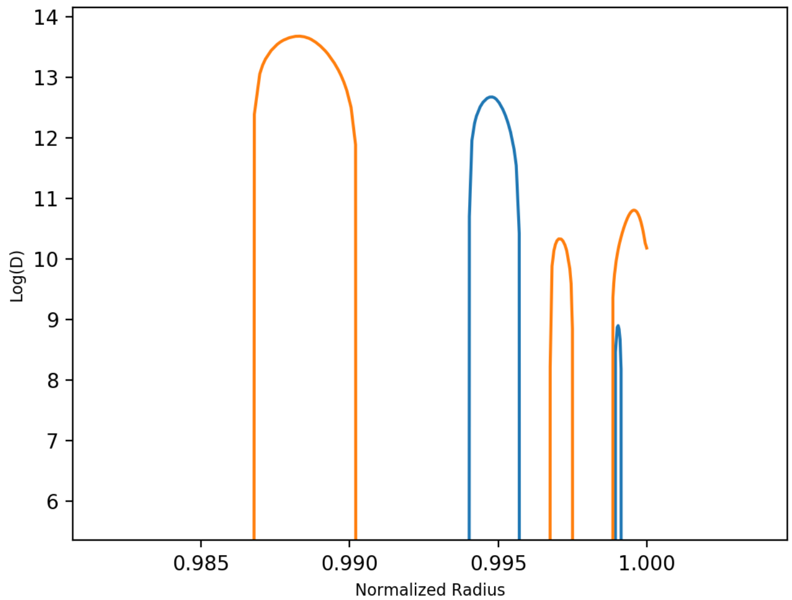

Figure 3.10: log of the convective diffusion constant, D_{ov}, against normalized radius. The orange line represents a model cooler than the break-point, and the blue line represents one hotter than the break-point.

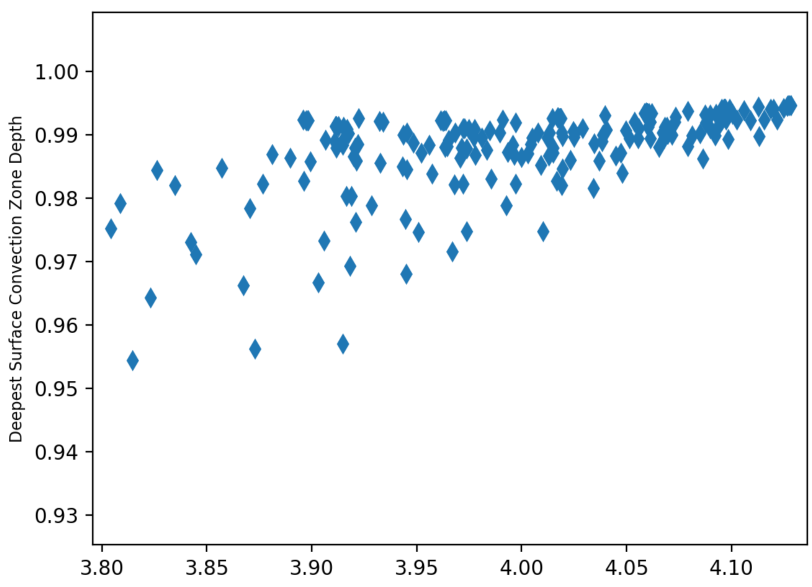

Figure 3.11: Normalized deepest convection zone depth plotted against $Log(T_{eff})$ for all $Z = 0.001$ models.

CONCLUSIONS

The goal of this research was to expand upon the work of Lovekin and Guzik in 2017. A grid of stellar models of δ Scuti variable stars with varying masses, rotation rates, amounts of convective overshoot, and chemical compositions was created to investigate the behaviour of the pulsation constants of the models as a function of their effective temperatures in relation to the amount of convective overhooting. Although a relation between m_{low} and f_{ov} was found in the original research, it was undetermined if this relation held for stars of other metal compositions, and if the relation had a metallicity dependency. It was found in this research, by varying the metal compositions of the models in ways that were not done before, that the relation did was still present for models of other metallicities, and that there was a possible metallicity dependency, although not particularly statistically significant, which was quantified as shown in Equation 3.5.

$$\begin{aligned} m_{low} =& [(-127.57 \pm 93.85)Z + (14.38 \pm 2.78)]f_{ov} \\ &+ [(14.40 \pm 6.31)Z - (3.10 \pm 0.19)] \end{aligned} \tag{3.5}$$

In addition to this result, a second relation, m_{high}, between models' pulsation constants as a function of effective temperature and their convective overshoot was found, this time at higher effective temperature values and lower metal compositions. This occurred specifically above $\log(T_{eff}) \geq 4.05$, and for models with $Z = 0.001$, since these were the only models to reach high enough temperatures to show this effect. This relation was quantified and is shown in Equation 3.6.

$$m_{high} = (-1.16 \pm 0.64) f_{ov} + (0.37 \pm 0.03) \tag{3.6}$$

Due to the fact that this behaviour was only seen at the lowest metallicity models in this research, it could not be determined if this relation has a metallicity dependence at this time. The cause of this m_{high} was determined to be due to the location of the deepest surface convection zone of the star, however further investigation into this effect will be necessary.

It was also found that for the plots of the $Z = 0.001$ models the data points branched off from one another, created two separate data sets. This branching effect was found to be caused by the density of the star, and threshold densities were determined for each convective overshoot (excluding $f_{ov} = 0.1$, due to it being unrealistically high). Due to a lack of time in this research the fundamental cause in this change of density was unable to be determined.

Future work in the research would include creating more grids of models at lower metallcities in order to determine the dependence of m_{high}. It would also be necessary to investigate the cause of the branching effect at the lowest metallicity. The final step in the future of this research would be to attempt to apply the full results of this research to observed High Amplitude δ Scuti stars (HADS). HADS are a class of δ Scuti variables that are thought to be radial pulsators, and thus would be most likely to show the effects found in this thesis, as it focused on radially-pulsating models. As shown by Lovekin and Guzik [3], there is no Q-dependence for $l = 1$ modes, and δ Scuti stars that are not HADS are more likely to be non-radial pulsators.

References

[1] E. Böhm-Vitense, Zeitschrift fur Astrophysik **46**, 108 (Jan 1958).

[2] J. P. Cox, *Theory of stellar pulsation*, 1980.

[3] C. C. Lovekin and J. A. Guzik, **849**, 38 (Nov 2017).

[4] B. Paxton et al., The Astrophysical Journal Supplement Series **192**, 3 (Jan 2011).

[5] B. Paxton et al., The Astrophysical Journal Supplement Series **208**, 4 (Sep 2013).

[6] B. Paxton et al., The Astrophysical Journal Supplement Series **220**, 15 (Sep 2015).

[7] B. Paxton et al., The Astrophysical Journal Supplement Series **234**, 34 (Feb 2018).

[8] B. M. Peterson and B. Ryden, *Foundations of Astrophysics*, Pearson Addison-Wesley, San Francisco, 2010.

[9] M. Templeton, Delta Scuti and Delta Scuti Variables, https://www.aavso.org/vsots_delsct, 2010.

[10] R. H. D. Townsend and S. A. Teitler, **435**, 3406 (Nov 2013).

[11] M. Vogel, Contemporary Physics **53**, 443 (Sep 2012).

[12] M. von Steinkirch, *The Ledoux Criterion for Convection in a Star*, State University of New York, Stony Brook, 2012.

APPENDIX

A.1 Sample MESA Inlist File

```
&star_job

      create_pre_main_sequence_model = .true.

      save_model_when_terminate = .false.

      change_v_flag = .true.
      new_v_flag = .true.

      new_rotation_flag = .true.

      ! rotation off until near zams
      change_rotation_flag = .false.

      ! this is the rotation parameter
      new_omega_div_omega_crit = 0.0
      near_zams_relax_omega_div_omega_crit = .true.
      num_steps_to_relax_rotation = 10

      pgstar_flag = .true.

      set_initial_model_number = .true.
      initial_model_number = 0

      !pgstar_flag = .true.

/ ! end of star_job namelist



&controls

      initial_mass = 1.6 ! mass of the model

      ! these two lines stop the evolution when
      ! the model leaves the main sequence
      xa_central_lower_limit_species(1) = 'h1'
```

```
        xa_central_lower_limit(1) = 0.01

        initial_z = 0.001 ! metallicity of the model

        ! These two lines stop the evolution at zams
        ! They are commented out after reaching zams
        Lnuc_div_L_zams_limit = 0.99d0
        stop_near_zams = .true.

              write_pulse_data_with_profile = .true.
              pulse_data_format = 'GYRE'

        photo_interval = 50
        profile_interval = 100  ! the timestep for the profiles
                                ! 100 before zams, 30 after
        history_interval = 1
        terminal_interval = 10
        write_header_frequency = 10

        ! Below are the convective overshoot parameters
        !(all are the same value)

        overshoot_f0_above_nonburn_core = 0.0
        overshoot_f0_below_nonburn_shell = 0.0

        overshoot_f0_above_burn_h_core = 0.0
        overshoot_f0_above_burn_h_shell = 0.0
        overshoot_f0_below_burn_h_shell = 0.0

        overshoot_f_above_nonburn_core = 0.0
        overshoot_f_below_nonburn_shell = 0.0

        overshoot_f_above_burn_h_core = 0.0
        overshoot_f_above_burn_h_shell = 0.0
        overshoot_f_below_burn_h_shell = 0.0


/ ! end of controls namelist

&pgstar

/ ! end of pgstar namelist
```

A.2 Sample gyre.in File

```
&model
        model_type = 'EVOL'
        file = 'profile11.data.GYRE'  ! the profile being
           analyzed
        file_format = 'MESA'
/

&mode
        l = 0            ! Harmonic degree
       m = 0
/

&constants

/

&osc
                nonadiabatic = .FALSE.
                outer_bound = 'VACUUM'

/

&num
                diff_scheme = 'COLLOC_GL4'

/

&scan
    ! Scan for modes using a uniform-in-frequency grid;
    ! best for p modes
    grid_type = 'LINEAR'
    freq_min = 2.0       ! Minimum frequency to scan from
    freq_max = 30.0      ! Maximum frequency to scan to
    n_freq = 350         ! Number of frequency points in scan
    freq_frame = 'COROT_O'
/

&grid
                alpha_osc = 10
                alpha_exp = 2
/
```

```
&ad_output
    ! the name of the output file
    summary_file = 'test1a_0001_rot0f002.txt'
    summary_file_format = 'TXT'
    summary_item_list = 'T,M_star,R_star,l,m,n_pg,freq'
    ! the information in the output file
    /

&nad_output

    /
```

A.3 Python Function for Calculating Q

```python
import numpy as np
import scipy as sp
import pylab as plt
import glob
import astropy
from astropy.table import Table
import astropy.constants as const
import astropy.units as u
import itertools
import statistics

def findQ():

    """ A script to calculate the Q value of the first 5
       orders of l = 0 p - modes in the MESA/GYRE models.
       Results are tabulated as a function of mass,
       metallicity, effective temperature, convective
       overshoot, and rotation rate
    """

    alpha = [0,0.02,0.04,0.06,0.08,0.1]
    velocities = [0,0.1,0.2,0.3,0.4]
    metallicities = [0.04,0.01,0.001]
    # we are adding data to the summary file here
    Qdata = open("Q_data_summary.txt", "w+")
    for a in alpha:
        #convert to strings to create file names in directory
        overshoot = str(a).split('.')
```

```
# this if/else statement is to account for the
# 0 without a decimal in alpha
if len(overshoot) == 1:
    overshoot = overshoot[0]
else:
    overshoot = overshoot[0] + overshoot[1]
for v in velocities:
    rotation = str(v).split('.')
     # also for the 0 without a decimal in velocities
    if len(rotation) == 1:
        rotation = rotation[0]
    else:
        rotation = rotation[0] + rotation[1]
    for met in metallicities:
        metal = str(met).split('.')
        metal = metal[0] + metal[1]

        directory = "LAB/mesa-r10398/gyre/gyre/work/
           test*_"+str(metal)+"_rot"+str(rotation)+"f
           "+str(overshoot)+".txt"
        flist = glob.glob(directory)
        for f in flist:

            # we find mass, radius, and effective
               temperature in these files

            datafile = open(f)
            stardata = datafile.readlines()
            Teff = stardata[0]
            Teff = np.float(Teff)
            mass,r = stardata[3].split()
            mass = np.float(mass)*u.g
            r = np.float(r)*u.cm
            meanrho = mass/(4./3.*np.pi*r**3)
            rhobar_solar = 1.41*u.g/u.cm**3
            ratio = meanrho/rhobar_solar
            scalefactor = np.sqrt(const.G.cgs*mass/r
               **3)/(2*np.pi)

            pulsations = np.loadtxt(f,skiprows = 6)
            # put this in an astropy table - it will
               be easier in the long run!
            l = pulsations[:,0]
```

```
                    n_p = pulsations[:,2]   # actually reading n_p
                    m = pulsations[:,1]
                    frequencies = pulsations[:,3]
                    frequencies = frequencies[np.logical_and(n_p < 6, n_p > 0)]
                    l = l[np.logical_and(n_p < 6, n_p > 0)]
                    m = m[np.logical_and(n_p < 6, n_p > 0)]
                    n_p = n_p[np.logical_and(n_p < 6, n_p > 0)]

                    frequencies = (frequencies*scalefactor).to(1/u.d) #scaling frequencies
                    Q = np.sqrt(meanrho/rhobar_solar)/frequencies

                    for i in range(len(Q)):   # every value in Qdata needs to be unitless
                        Qdata.write('%e %f %f %f %e %i %i %i %f %f %f %f \n' %(mass/u.g,a,v,met,Teff,l[i],m[i],n_p[i],Q[i]/u.d,frequencies[i]*u.d,r/u.cm,ratio))
    Qdata.close()
```

A.4 Python Function for Plotting Q vs T_{eff}

```
import numpy as np
import scipy as sp
import pylab as plt
import glob
import astropy
from astropy.table import Table
import astropy.constants as const
import astropy.units as u
import itertools
import statistics

def plotQ(vplot,alphaplot,thisn,metplot):
    from scipy import odr

    """ a script to plot Q as a function of log(Q) vs log(Teff). value returned also needed to run all following functions.
```

```
"""

Qdata = np.loadtxt("Q_data_summary.txt")
mass = Qdata[:,0]
alpha = Qdata[:,1]
v = Qdata[:,2]
metal = Qdata[:,3]
Teff = Qdata[:,4]
l = Qdata[:,5]
m = Qdata[:,6]
n = Qdata[:,7]
Q = Qdata[:,8]
periods = 1./Qdata[:,9]
r = Qdata[:,10]
ratio = Qdata[:,11]

Q = Q[np.logical_and(np.logical_and(np.logical_and(alpha
    ==alphaplot,v == vplot),metal == metplot),n == thisn)]
l = l[np.logical_and(np.logical_and(np.logical_and(alpha
    ==alphaplot,v == vplot),metal == metplot),n == thisn)
    ].astype(int)
m = m[np.logical_and(np.logical_and(np.logical_and(alpha
    ==alphaplot,v == vplot),metal == metplot),n == thisn)
    ].astype(int)
r = r[np.logical_and(np.logical_and(np.logical_and(alpha
    ==alphaplot,v==vplot),metal == metplot),n == thisn)]
mass = mass[np.logical_and(np.logical_and(np.logical_and(
    alpha==alphaplot,v==vplot),metal == metplot),n ==
    thisn)]
Teff = Teff[np.logical_and(np.logical_and(np.logical_and(
    alpha==alphaplot,v==vplot),metal == metplot),n ==
    thisn)]
periods = periods[np.logical_and(np.logical_and(np.
    logical_and(alpha == alphaplot,v == vplot),metal ==
    metplot),n == thisn)]
ratio = ratio[np.logical_and(np.logical_and(np.
    logical_and(alpha == alphaplot,v == vplot),metal ==
    metplot),n == thisn)]

ells = [0]
ens = [thisn]
alphas = np.unique(alpha)
```

```
def parabola(x,a,b):
    x = np.array(x)
    return a*x + b

for i,k in itertools.product(ells,ens):

    if len(periods) > 0:
        meanP = np.log10(np.mean(periods))
    else:
        meanP = 0
    # this is just a check that periods has a value

    xfit = np.log10(Teff)
    xfit = xfit.reshape(len(xfit))
    y1 = np.log10(Q)
    y1 = y1.reshape(len(y1))

    p, V,a,b,c = np.polyfit(xfit,y1,2,cov=True, full =
        True)

    brokenfit = odr.Model(fitfunc)
    mydata = odr.RealData(xfit,y1)

    from scipy import optimize
    popt, pconv = sp.optimize.curve_fit(parabola,xfit,y1)

    myodr = odr.ODR(mydata, brokenfit, beta0=[1.5, 0,
        0.5, -1.5, 3.82, 4.05])
    # the values in beta0 are for fitting the data -->
        beta[3],[4], and [5] are the respective slopes of
        the piecewise function
    myoutput = myodr.run()

    plt.figure()
    plt.plot(xfit,y1,'d')
    plt.plot(sorted(xfit), fitfunc(myoutput.beta, sorted(
        xfit)), linewidth=2.0)
    plt.xlabel('Log(Teff)', fontsize=8)
    plt.ylabel('Log(Q)', fontsize=8)
```