Stellar Spectra: AST4310 Assignment 1

Evan Markel

Abstract. This assignment traces and reproduces the historical analytical procedures developed by pioneers in stellar spectroscopy. This exercises and IDL based analyses describe the processes of early spectral classification, the discovery of stellar atoms and ions according to thermodynamical physical theories-especially Saha-Boltzmann distributions, and spectral radiation analysis according to Planck's law and relevant corrective terms.

1. Introduction

In this analysis, I will present my findings in the project in roughly the same order as the exercises as the assignment's narrative structure follows the historical analytic processes of understanding spectral phenomena.

2. Saha-Boltzmann Calibration of Harvard Sequence

The Boltzmann distribution relates the distribution of an atom/ion's particles in a particular energy state **s** compared to particles in all energy states in in the same ionization stage **r**. across its relevant discrete energy states. There are 3 hydrogen lines visible in the

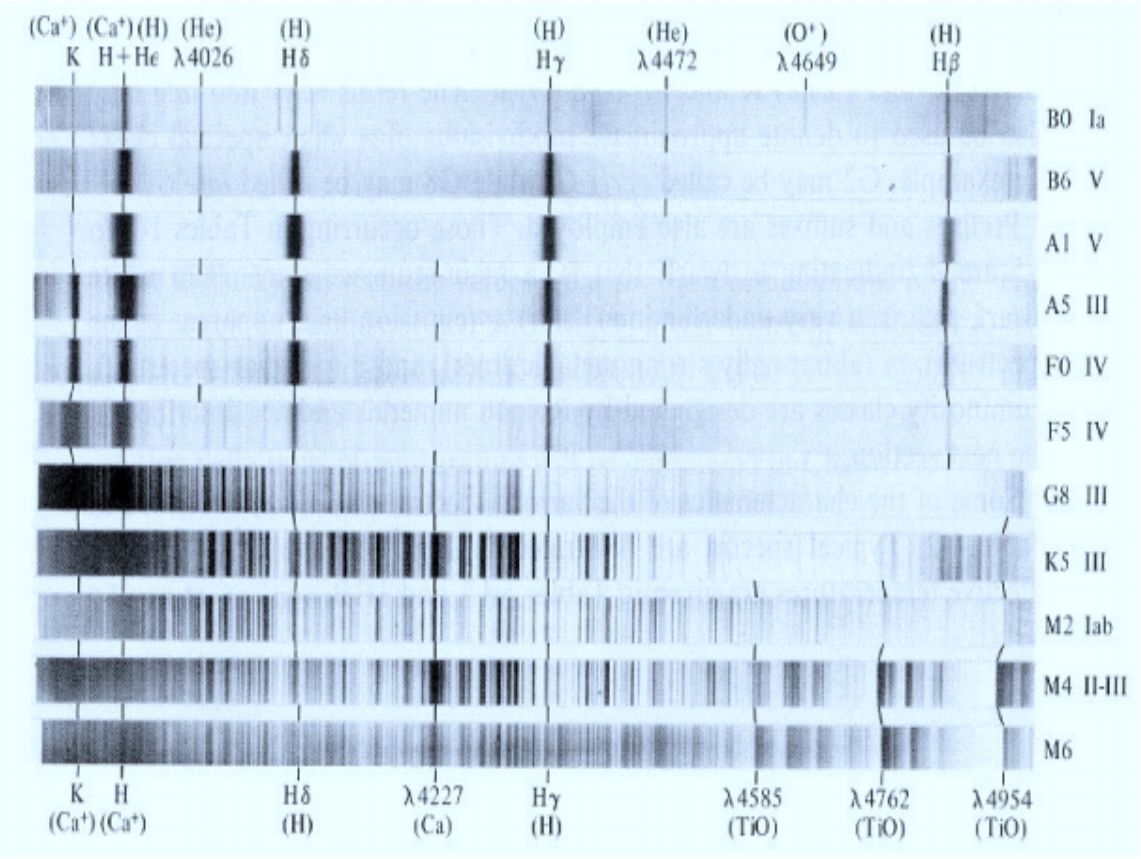

Fig. 1. Figure 5 (from project description) shows 3 Hydrogen I lines that are shown in the energy level diagram (figure 7) in project file.

above figure. Starting from the top right, we see an **H**β, (Fraunhofer *F* line, which has the longest λ in the figure. This corresponds to the Balmer β bound-bound transitional line at 4,861Å in figure 7. In the middle we see the hydrogen line **H**γ which correspondingly has a λ of 4,3400 Å. The final visible hydrogen line is **H**δ at a λ of 4,102 Å. The furthest line in the figure above (Ca^+ (K)) has a λ of 3,934 Å. No other transitions for the hydrogen atom in figure 7 exist in the range [3,934, 4,861]. It makes sense that only Balmer series lines are visible since their wavelengths lie in the visible spectrum, while the Lyman belong to the ultraviolet, and the Paschen and Brackett to the infrared. Inspecting figure 7, we see that the similarly named series all share lower level

transitions. The transition for a β $[(1, n \rightarrow n+2)]$ for a given series will share an upper level transition with the next higher series' alpha $[(1, n+1 \rightarrow n+2)]$. Likewise, a γ transition will share an upper level with the next series' β and also the α of the series above that.

Payne's assumption that the strength of the absorption lines depended on the population density of the lower level of a transition seems intuitive based on the strength of the spectra in figure 5 and the idea that more excitations should happen if you have a higher number of particles in the initial energy state. For the purposes of this exercise, we will assume linear scaling and use the Boltzmann law to estimate the strength ratios of the α lines for the Lyman, Balmer, Paschen, and Brackett series. Assuming these are the only 4 series for neutral hydrogen, the sum of all four strength ratios should be equal to 1. Since the ionization level $r = 1$ we know that $g_{1,s} = 2s^2$ and $\chi_{1,s} = 13.6(1 - \frac{1}{s^2})eV$. In this case the Boltzmann equation is given by

$$\frac{n_{1,s}}{N_1} = \frac{g_{1,s}}{U_1} exp\left(-\frac{\chi_{1,s}}{k*T}\right)$$

Since kT is the same for all 4 transitions when calculating relative strength ratios, we can cancel it out of our calculation. I then calculated the Boltzmann equation for the alpha lines of the four series. Here we use the Boltzmann constant $k = 8.61734x10^{-}5evK-1$ and $T_{sun} \sim 6000K$. Then I computed the relative strengths $\frac{n_{1,1}}{n_{1,s>1}}$. The results are included in the table below:

Series	s	Boltzmann calc	Boltzmann-final value	$n_{(1,1)}/n_{(1,s>1)}$
Lyman	s=1	2*exp(-13.6*0/kT)	2.0000	1.0
Balmer	s=2	8*exp(-10.2/kT)	2.162e-8	9.2507e+7
Paschen	s=3	18*exp(-12.09/kT)	1.257e-9	1.5911e+9
Brackett	s=4	32*exp(-12.75/kT)	6.234e-10	3.2051e+9

We see that the ground state (Lyman) dominates the strength ratios by orders of magnitude compared with the higher level α series owing to the $\chi_{1,1}$ term causes the exponent to equal 1. Also, the strength ratio of Balmer α to Paschen α = 17.2, and the strength ratio of Balmer α to Brackett α = 34.6.

The Saha distribution compares the particle densities of an atom or ion across its different ionization stages. There is a fundamental difference in the Saha distribution compared to Boltzmann's due to the equations' differing dependence on temperature and the Boltzmann dependence on $exp(-\frac{\chi_{r,s}}{kT})$, which causes the higher s terms to vanish very quickly. This dependence on $chi_{r,s}$ outweighs the influence of temperature for the Boltzmann distribution. However, the Saha distribution has a factor of $T^{3/2}$ due to the kinetic energy of the free electron associated with the ionization transition. This terms lets the Saha distribution become much more sensitive to temperature relative to the Boltzmann distribution. There are also only 2 given ionization levels for a particular temperature.

High-level values of the Boltzmann distribution can become roughly equal to the Saha distribution's values for the next-ion population density based on specific values of the electron density N_e. We know the partition function U_{r+1} is roughly equal to U_r for the Saha distribution so we can choose a value of N_e that factors in the kinetic energy of the free electron in order to equate Saha and Boltzmann.

$$\frac{g_{r,s}}{U_r} exp\left(\frac{-\chi_{r,s}}{k*T}\right) = \frac{2U_{r+1}}{N_e*U_r} * (\frac{2\pi*m_e kT}{h^2})^{3/2} * exp(\frac{-\chi_r}{kT})$$

Solving for N_e allows us to cancel some terms out. We can see from table 1 in the assignment that U_r is roughly the same so the terms when divided equal about 1.

$$N_e\ (2\frac{2\pi*m_e kT}{h^2})^{3/2} * \frac{exp(\frac{-\chi_{r,s}}{\chi_r})}{g_{r,s}}$$

The density of electrons can make ionization or recombination much easier and this can account for the depletion or population of an ionization stage.

3. Numerical Analysis of Schadeenium and Hydrogen

The next few sections contain IDL code and psuedocode that helps us numerically illustrate the adherence of solar spectra to the Boltzmann and Saha relations originally derived from terrestrial physics.

My IDL program SSA2.pro contains functions that calculate the partition functions, the Saha, and Boltzmann distributions of the fictional iron-like element E (Schadeenium), as well as hydrogen and $\mathbf{Ca}^+$ (Fraunhofer K line).

First, we compared U_r of element E for r = 1,2,3,4. The results matched the partition function values given in Table 1 for temperature = 5000K, 10000K, and 20000K. $U_{r=1,2,3,4} \approx [1.1, 1.46, 2.23]$ for the 3 respective temperatures. Similarly, the Saha and Boltzmann distribution functions in SSA2.pro were identical to those listed in Table 1. We notice that the Boltzmann distribution causes the high excitation states to be several orders of magnitude lower than the lower s values. Also, the Saha population distributions across the different ionization levels $r = 1, 2, 3, 4, 5$ are much more influenced by the temperature varying from 5000K to 10000K to 20000K.

The $SAHABOLT_E$ function in SSA2.pro combines the Saha and Boltzmann distributions and allows us to plot stellar strength according to Payne's analysis, which is as a phenomenon of the physical laws. The following plot represents the IDL numerical calculation of the Saha-Boltzmann distributions for 'Schadeenium'.

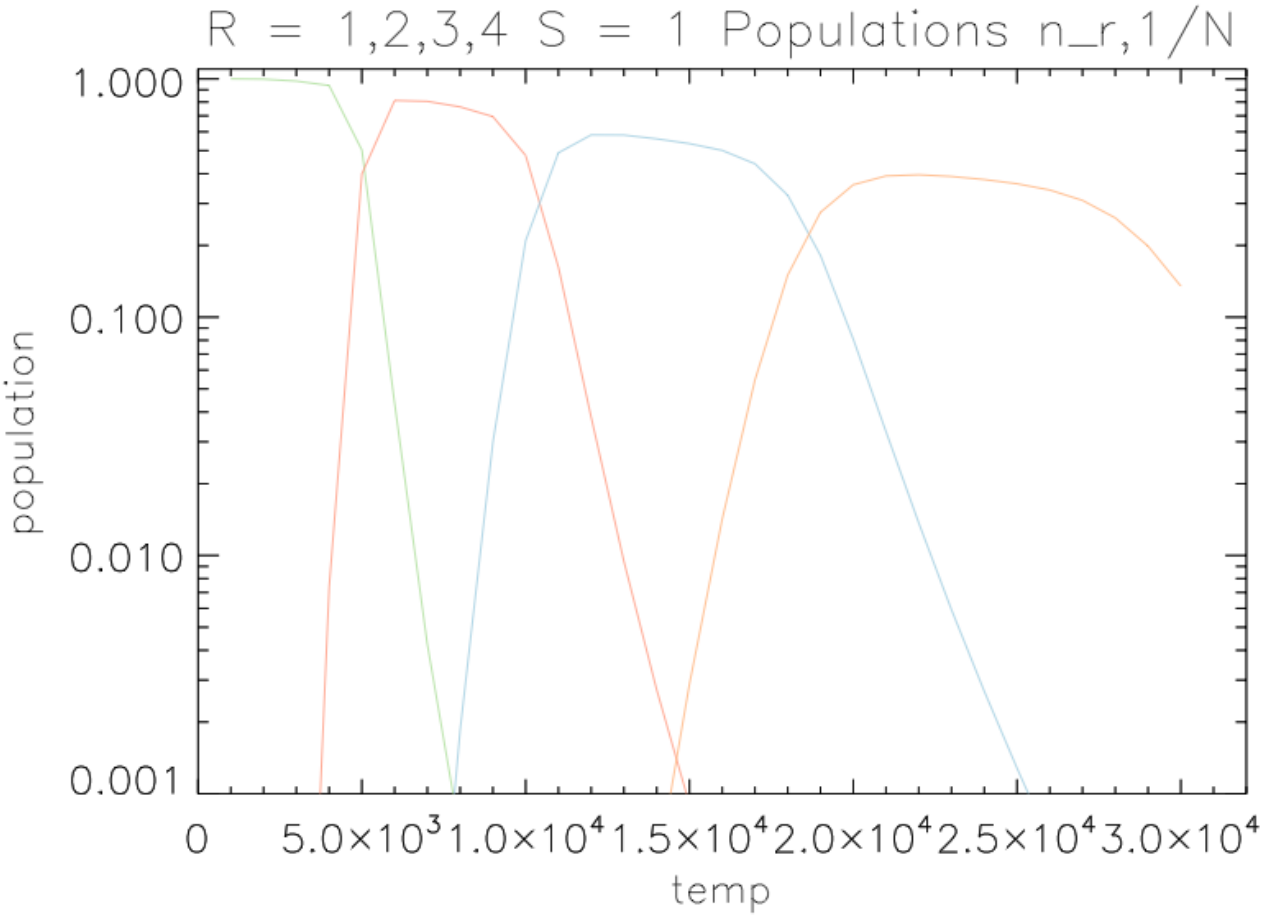

Fig. 2. This plot represents the first four ionization states for 'Schadeenium' while in the ground state (s=1). They are differently colored with the lowest (r=1) peaking at the coolest temperature and the highest (r=4) peaking at the warmest temperature.

We see the population densities are all of the same order of magnitude for the first 4 ionization levels. We also see the strict temperature dependence that we know is a central aspect of the Saha distribution, which causes the steep flanks on the left and right side of each peak. This was apparent from the third table in Table 1 in the assignment. Higher temperatures will see elemental peaks at higher ionization states. As T goes to zero, we won't see any spectral populations as there would be little or no radiation! As T goes to infinity we would see the elements completely ionize and result in something like a superheated plasma.

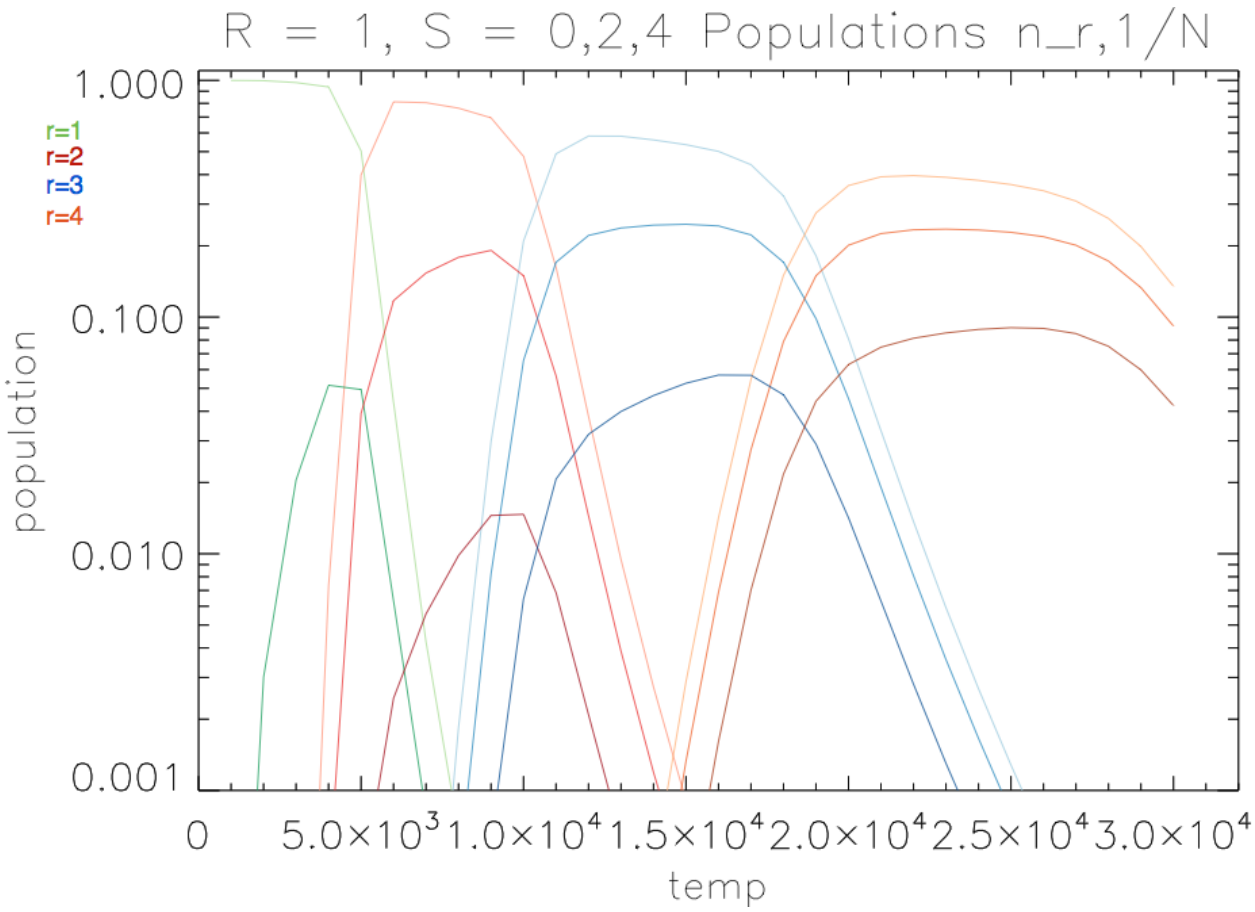

Fig. 3. The 4 different colors represent the first 4 ionization states for 'Schadeenium' and the hues of the colors get progressively darker as s goes from 0 to 2 to 4. The figure closely matches the patterns of real elements in figure 6 of the assignment.

Higher excitation states have a much smaller population density than the ground state curves. This follows according to the Boltzmann distribution. The higher level excitation states sit almost directly below (well, *slightly* to the right of) the peak lower level population densities in this plot which demonstrates the relative temperature independence of the partition functions and the Boltzmann distribution. Elements with lower ionization energies may see a more balanced population density for increasing r, while elements with higher ionization energies may see higher population densities in the lower ionization states. We would also see the elements become less T dependent with higher ionization energies and more T dependent with lower ionization energies.

Now we begin our analysis of hydrogen. The functions written in IDL can be adapted for real elements. One observation to note here is that the population densities are even more skewed towards the ground state for hydrogen than for the iron-like 'Schadeenium.'

Section 2.8 looks at the line strength of $Ca^{+}K$ versus $\mathbf{H}\alpha$. The hydrogen Balmer line is much weaker than the ionized Calcium line, which is counter-intuitive based on our previous analysis of high relative population density of the two elements in solar composition. However, the Saha distributions have also shown us that there is a strong temperature dependence that can influence spectral intensity as the Balmer line is not a ground state transition but a higher order (s=2 $\longrightarrow$ 3*transition*). From our analysis of the relative strengths of the Lyman vs. Balmer series, there are around 8 orders of magnitude difference in populations!. It's true that hydrogen is much more abundant than calcium in the sun around $2x10^{-6}$. But numerical comparison between the two Fraunhofer lines shows that at the solar temperature range, $Ca^{+}K$ is 7,643 times more abundant than $\mathbf{H}\alpha$. At 4000K the ratio is around 10^{6} and around 150 at 6000K. The strength ratio as a function of temperature is plotted below.

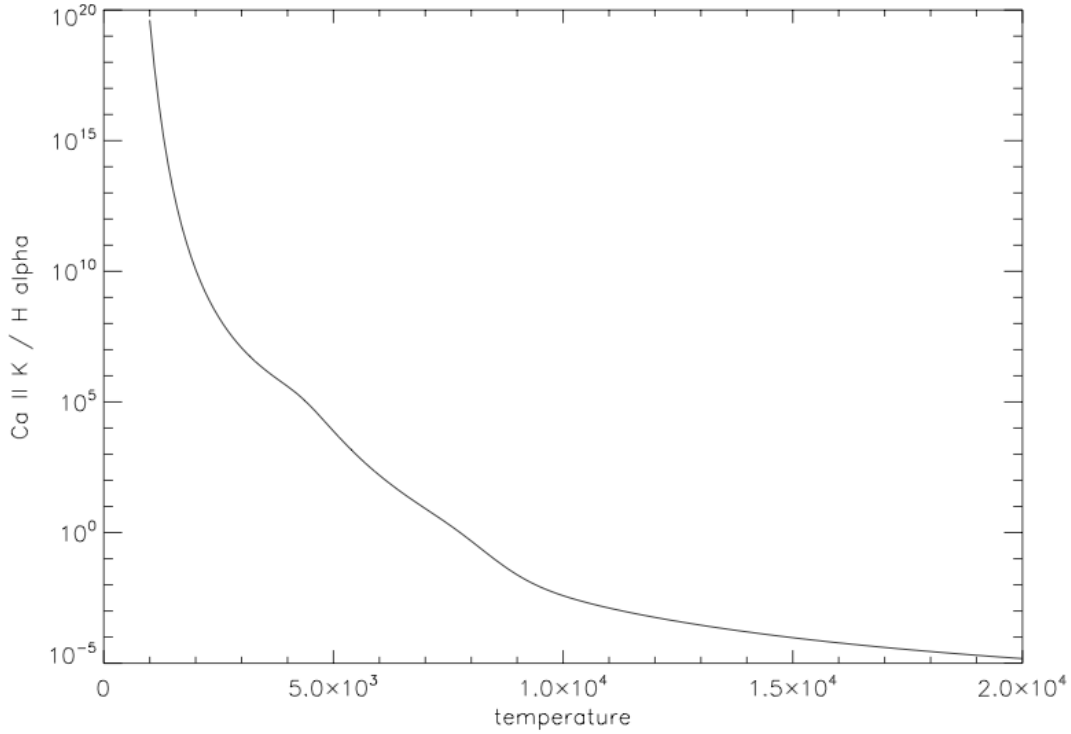

Fig. 4. We see in this plot that the abundance of Calcium is much greater than Hydrogen up until temperatures increase to about 7700K. So Calcium is the stronger line in the range of the photosphere (4000-6000K).

We now look at the temperature sensitivity of the two elements. The differing temperature sensitivity ranges also contribute to the composition of spectral lines in the solar photosphere.

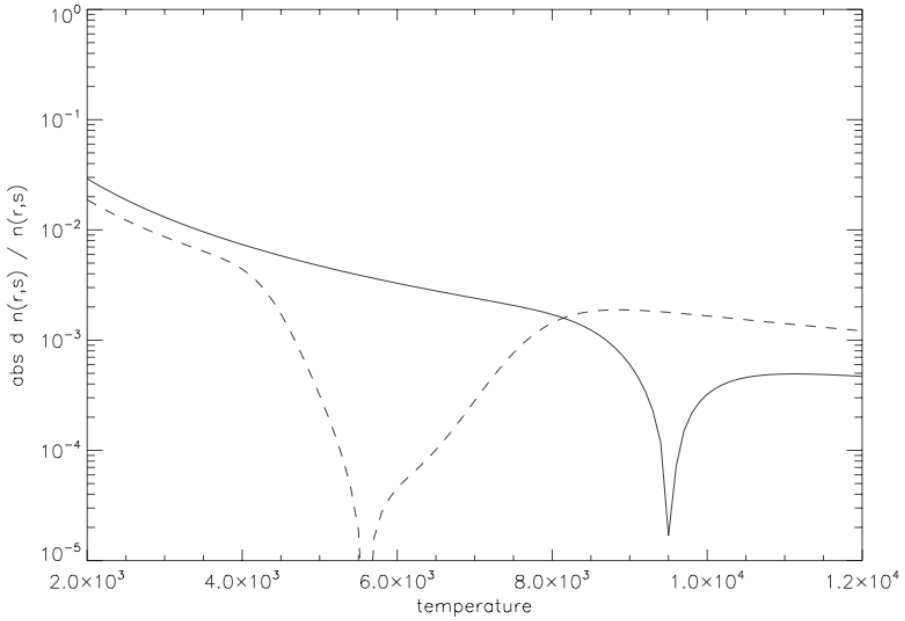

Fig. 5. The Ca^+K curve is much less sensitive to incremental temperature changes around 5600K while the **H**α curve has the same affect at around 9500K. We can see that the temperature insensitivity is for a wider range of T for a^+K than for **H**α.

In order to assess which flank of each population's dip is positively and negatively influenced by the temperature perturbations, we need to graph the temperature variation of the respective populations in relative units.

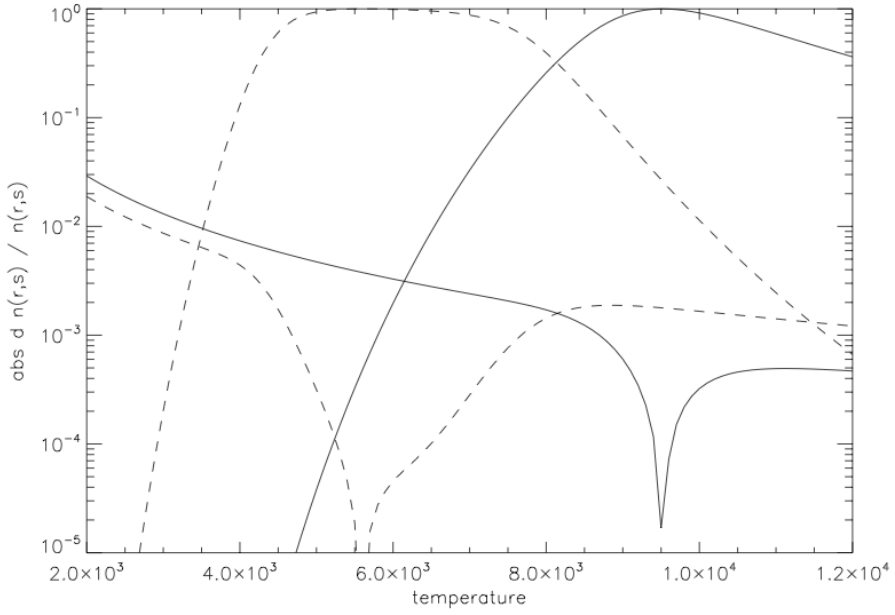

Fig. 6. Overlaid relative population density.

We see from the overlaid relative population/Δ temperature that the population becomes constant around the dips for the respective atoms and the left flanks describe population growth to their respective peaks and then a further increase in temperature causes a decrease in population. Hence the right flanks correspond to $\Delta n < 0$. The left flanks of the population curves are due to the Saha distributions and the right side is due to Boltzmann as we've seen in the previous sections. The overlaid population

densities demonstrate that the wide peaks for the respective atoms are relatively temperature independent over the Δ temperature range. Thus, given the small variations of temperature in the sun, we continuously see the dominance of the calcium band over the hydrogen band. The solar range is between [4000K, 6000K] which corresponds to the peak population (and most temperature independent) region of the Calcium curve.

Now, we'll take a specific look at hydrogen ionization. The difference in hydrogen's ionization temperature sensitivity allows us to determine the temperature of stars accurately.

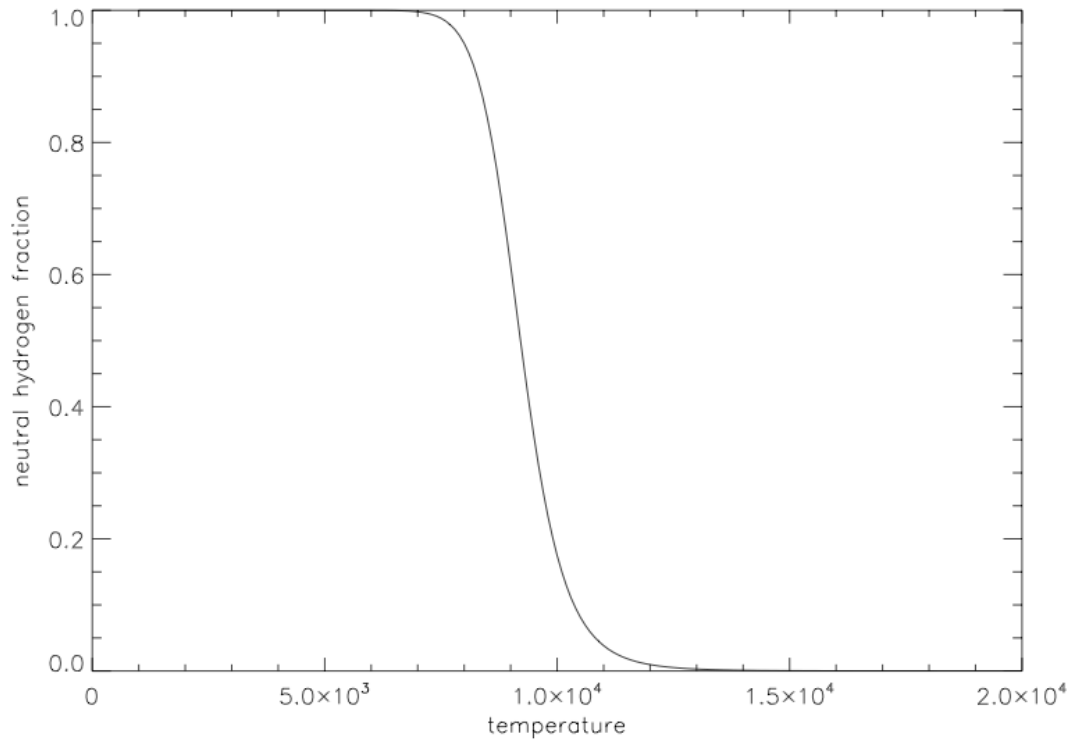

Fig. 7. The high temperature sensitivity of neutral hydrogen is apparent here so a spectral analysis of a distant star can accurately tell us its temperature.

Since the ionization happens during a very narrow bandwidth of temperatures, we have divided the stellar continuum into 'hot' and 'cool' stars based on the 50% ionization level. We can see from the graph that it is around 9000K. From coding the following into IDL, we can get a closer numerical solution. The divergent sources of ionization make the difference between hot and cool stars more distinct. Hot stars acquire electrons from ionized hydrogen while cool stars acquire electrons from metals.

$$IDL > forT = 2000, 20000, 5doprint, T, sahabolt_H(T, 1e2, 1)$$

The solution is approximately 9212K as indicated in the table of results:

Temperature	ionization ratio of H
9185	0.51442833
9190	0.51175719
9195	0.50908843
9200	0.50642192
9205	0.50375790
9210	**0.50109600**
9215	**0.49843689**
9220	0.49578065
9225	0.49312720
9230	0.49047696
9235	0.48782975
9240	0.48518602

4. Formation of Spectral Lines, Radiative Processes, and Growth Curves

Section 3 of the assignment deals with the formation of spectral lines and the behavior of radiation subjected to Planck's function, which describes the black body radiative intensity as a function of wavelength and temperature. As the development of thermodynamics during the fin de siècle period was a bit frantic, there are other related laws and regimes that characterize the spectrum and will be mentioned in this report. The Wien displacement law states that the emission curves look roughly the same for different temperatures as a function of wavelength. The law also divides the spectrum into two main regimes: the exponential Wien regime for shorter wavelengths, and the linear Rayleigh-Jeans regime for longer wavelengths.

The first IDL code for section 3 writes the Planck function. Following the code are 3 graphs of the planck function: one with normal scales, the next with a logarithmic y axis, and finally with a logarithmic x and y axis.

$functionplanck_1, temp, wav; planckfunctionintermsoftempandwavelength$
$h = 6.62607D - 27; planckconstantinerg * s$
$c = 2.99792D10; speedoflightincm * s^-1$
$k = 1.38065D - 16; BoltzmannconstantinergK^-1$
$const = 2 * h * c^2$
$econst = (h * c)/k$
$evar = temp * wav$
$pfunc = (const/(wav^5)) * 1/(exp(econst/(evar)) - 1)$
$return, pfunc$

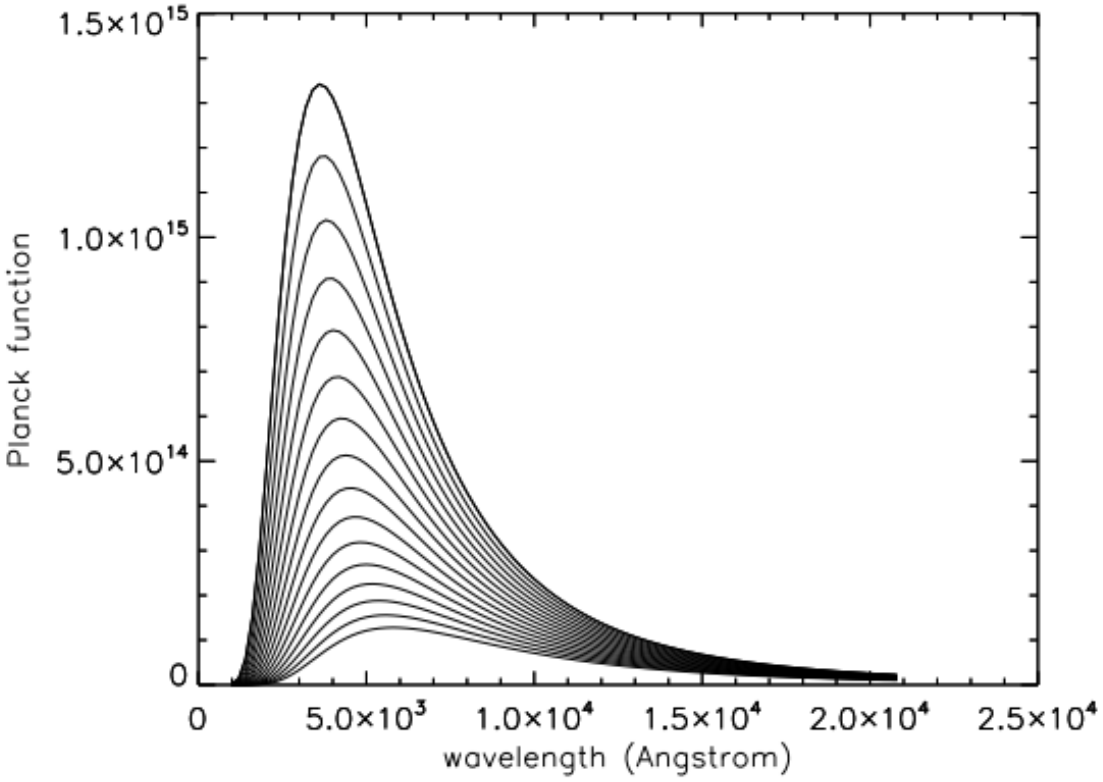

Fig. 8. The first Planck plot shows the Wien displacement law: same curve shape for varying wavelengths (from 1000 to 20800 Angstroms). The different curves correspond to different values of temperature (from 5000K to 8000K increments of 200K) as functions of wavelength.

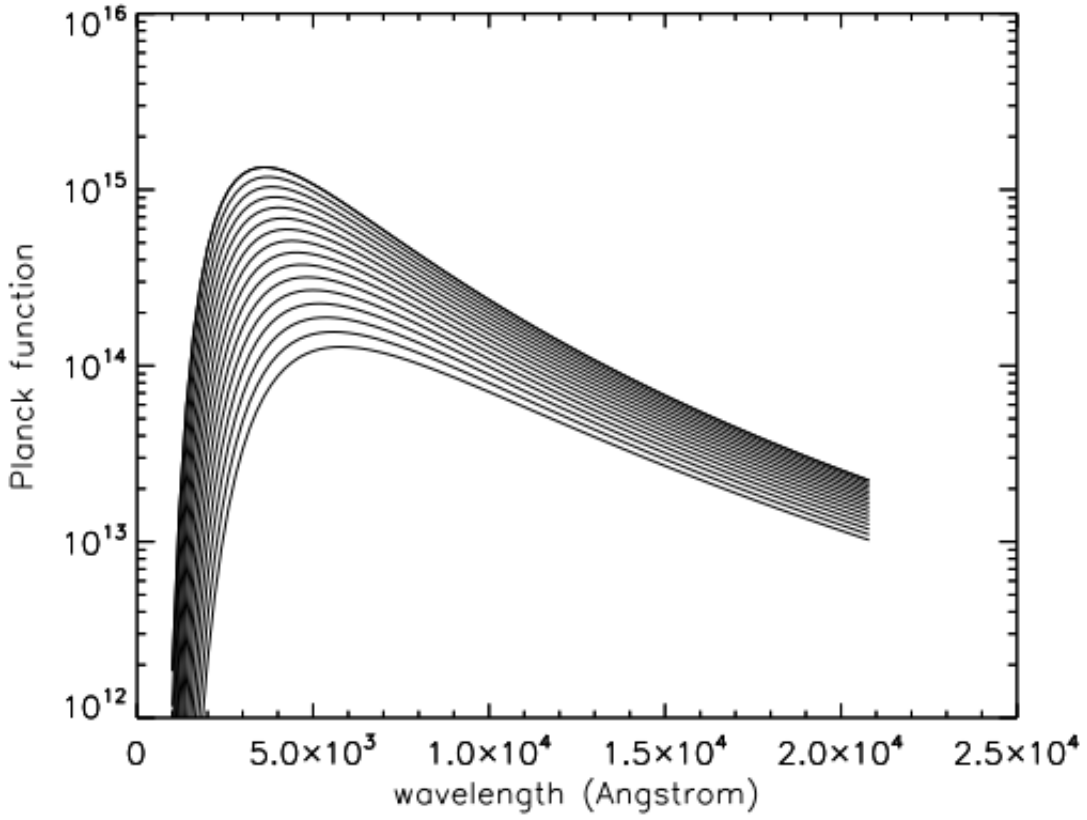

Fig. 9. This plot has a logarithmic y axis which demonstrates the difference in the Rayleigh Jean's regime (right hand side of peak) and the Wien regime (left hand side of peak).

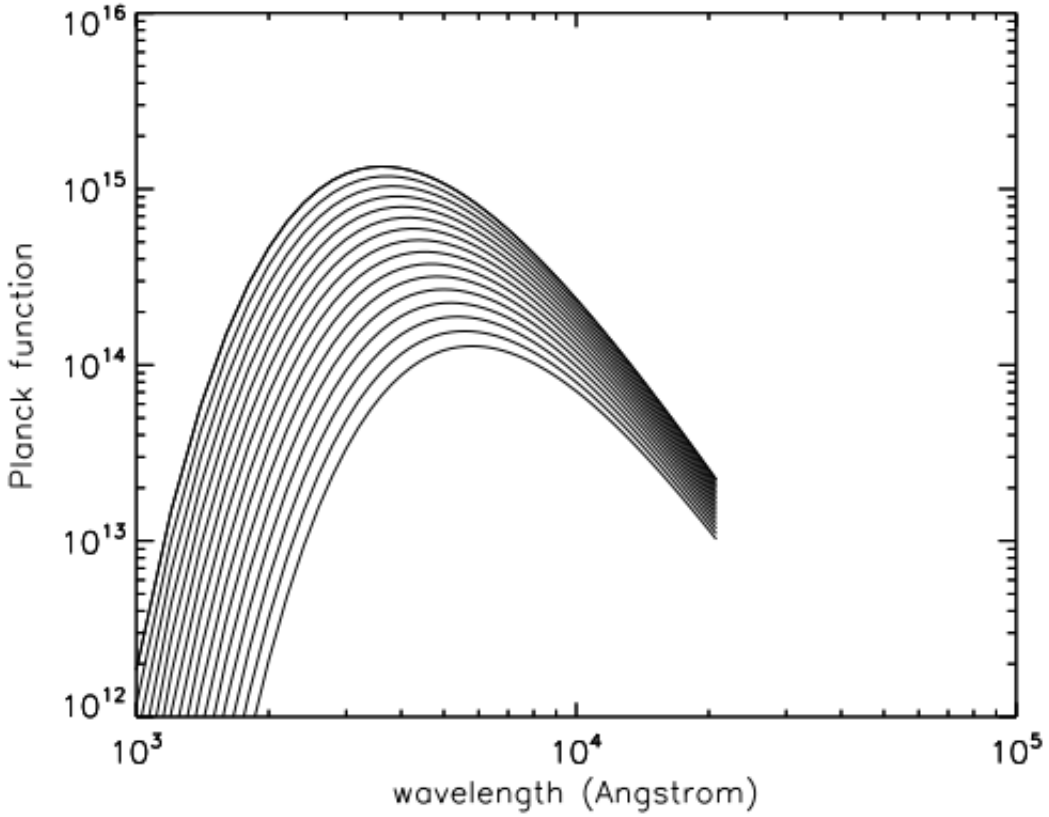

Fig. 10. This plot of the Planck function has a logarithmic x and y axis.

Now we look at radiation passing through an atomic shell surrounding the sun, which causes the absorption lines that we see in the spectrum. The solar radiation beam passes through an isothermal layer and the initial intensity is reduced by a factor of $I(0)e^{-\tau}$. We then add in the shell layer's radiation contribution and can derive the total emergent intensity I_λ.

$$I_\lambda = I_\lambda(0)e^{-\tau} + \int_0^\tau B_\lambda[T(x)]e^{-(\tau-\tau(x))}\,\mathrm{d}\tau(x)$$

$$\int_0^\tau B_\lambda[T(x)]e^{-(\tau-\tau(x))}\,\mathrm{d}\tau(x) = B_\lambda[T(x)]e^{\tau(x)-\tau}|_0^\tau$$

$$I_\lambda = I_\lambda(0)e^{-\tau} + B_\lambda[T(x)]e^{-(\tau-\tau)} - B_\lambda e^{0-\tau}$$

$$I_\lambda = I_\lambda(0)e^{-\tau} + B_\lambda\,(1 - e^{-\tau})$$

We see that the emergent intensity has an exponential dependence on τ and that very small τ will cause the emergent intensity to be dominated by the incident intensity I(0). Very large τ will cause the emergent intensity to be dominated by the Planck function of the isothermal layer. Now we'll look at the emergent intensity as a function of τ.

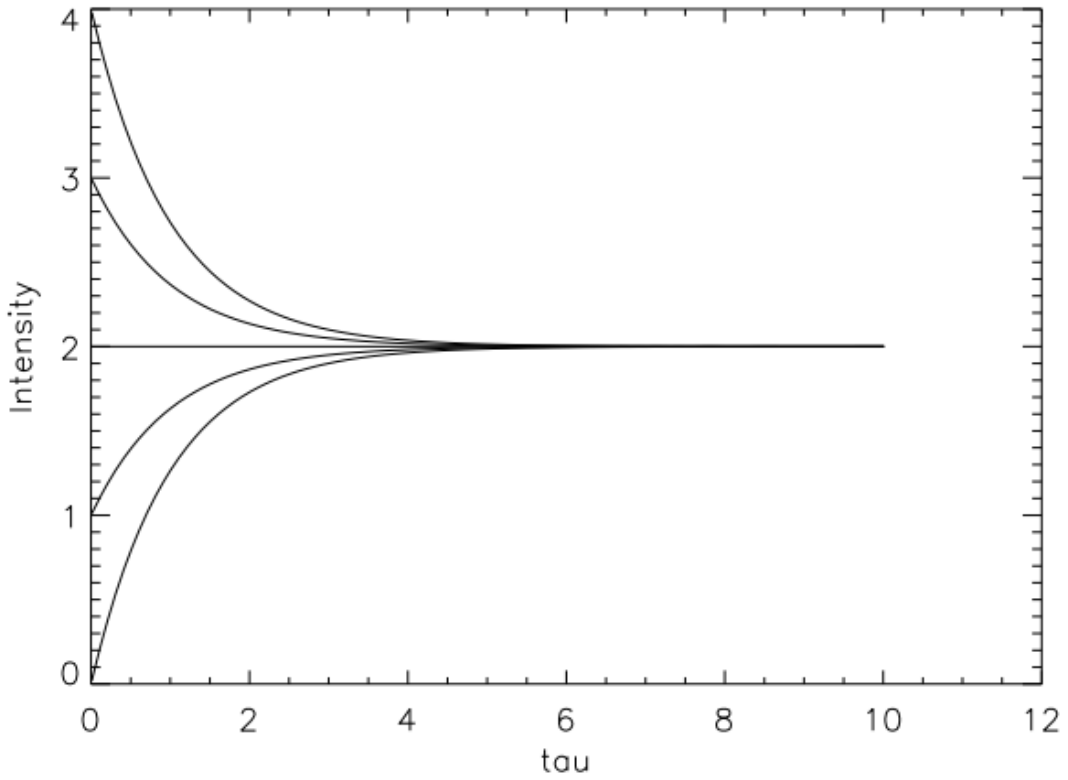

Fig. 11. We see that as increases from 0.01 to 10, the emergent intensity drops exponentially as shown in the equations. For $\tau << 1$, we see that there is very little influence and the emergent intensity is very close to I(0) = 4. As τ grows large (optically thick) the results converge on our value for the Planck function (2). This is apparent when we study our emergent intensity equation and see that B_{lambda} is the convergent result for $\tau >> 0$.

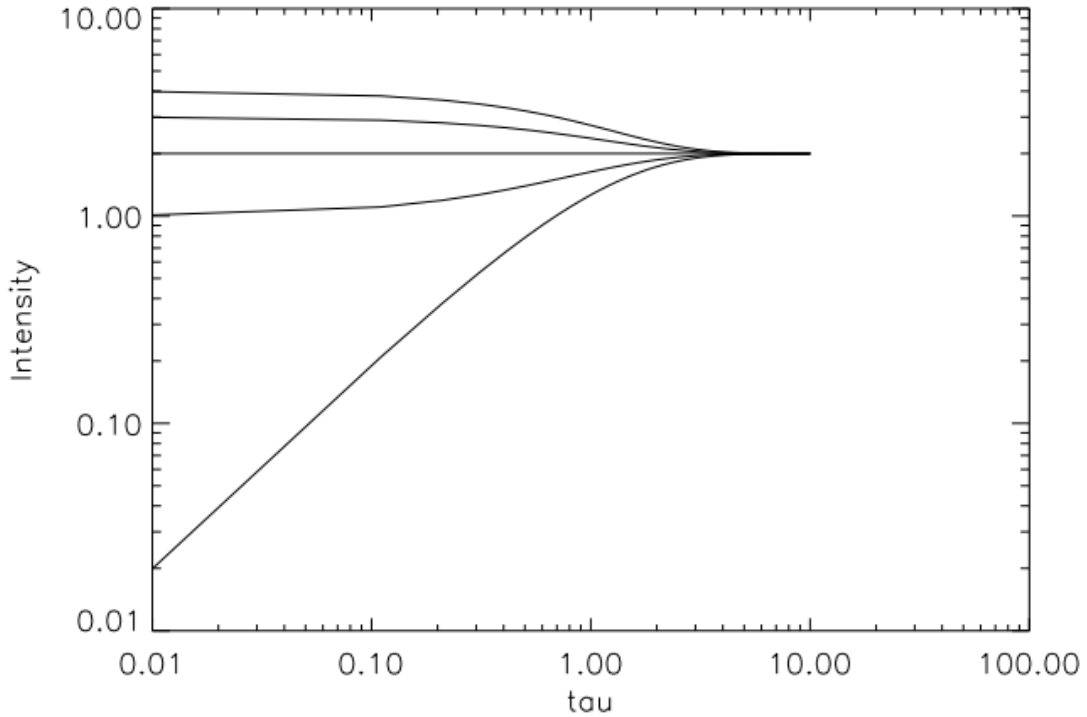

Fig. 12. On this logarithmic plot of the same function, we can see the behavior of intensity for small values of τ. On the logarithmic scale we see that for initial intensities less than B_λ there is a nearly linear dependence on τ for small values, which is similar to the linear Taylor approximation for exponential functions. For initial intensities greater than $B_\lambda = 2$, there is almost no effect from the optically thin layer.

So we see that optically thin layers produce black body radiation where there isn't a higher intensity radiative field propagating through it. Optically thin layers are nearly invisible to high intensity passing radiation. Optically thick layers do not depend on τ as τ can be seen as going nearly to inf. The perceived radiation is that of the black body radiation of the layer with no trace of the initial I(0). This can be seen in the function as τ grows large, $e^{-\tau}$ vanishes and the emergent intensity equation no longer depends on I(0) or on τ itself. It only depends on the black-body radiation of the layer.

Next we apply this idea of radiative interaction to a solar model that involves a 'reversing layer' of atoms that produce the absorption lines in the spectrum. This model has a similar structure to the emergent intensity model previously, except now the I(0) term becomes dependent on τ and acquires a Planck function based on the the surface temperature $T_{surface}$ and also wavelength λ. τ is also dependent on λ. We also introduce real world 'fuzziness' to spectral lines. Doppler shifts between individual atoms moving at fast relative speeds as well as Coulomb disturbances ("damping") causes the spectral lines to be broader than the theoretical δ function. The convolution associated with these two effects is given by the Voigt function, which is dependent upon a, a measure of the damping, and u, a Gaussian distribution centered on the very center of the absorption line that is due to the Doppler shift. I've plotted the Voigt function for varying values of u and a. The results are below.

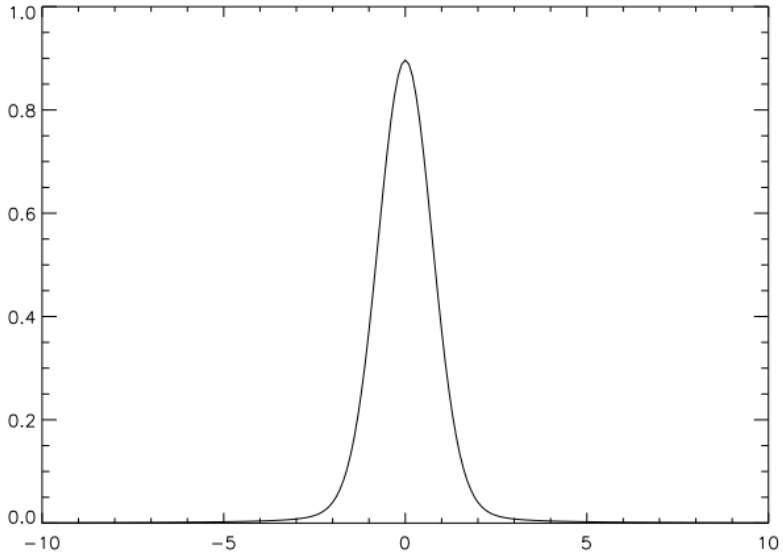

Fig. 13. This first graph of the Voigt function plots values of u on the x axis ranging from -10 to 10 and for constant a = 0.1. Both are input into the IDL Voigt function which is portrayed on the y-axis. We see the width and finite height of the function as an indication of the real-world convolution of the absorption line. We can see the effect of both u-due to the Doppler effect, and the Lorentzian wings due to a-the damping effect.

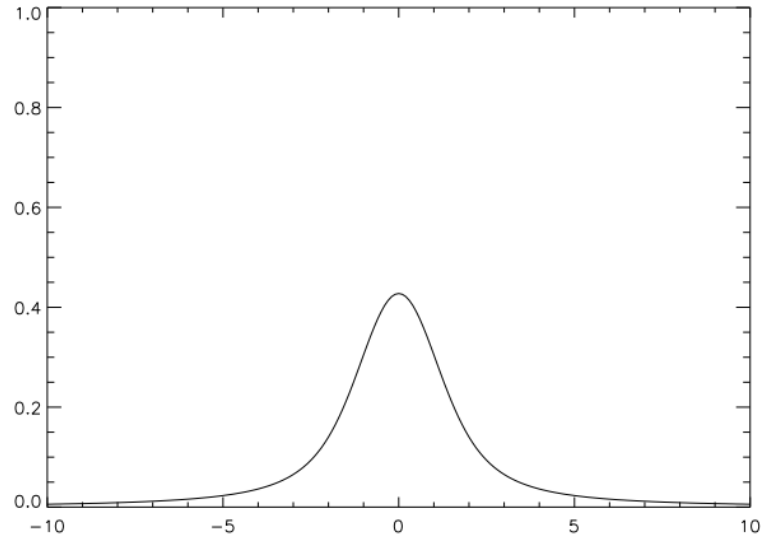

Fig. 14. This next graph is identical to the first except that a has now been increased from 0.1 to 1.0. The effect is a broadening along the x-axis and a more shallow peak at u=0. The Lorentzian wings have a larger presence and have a more dominant role in this broadening.

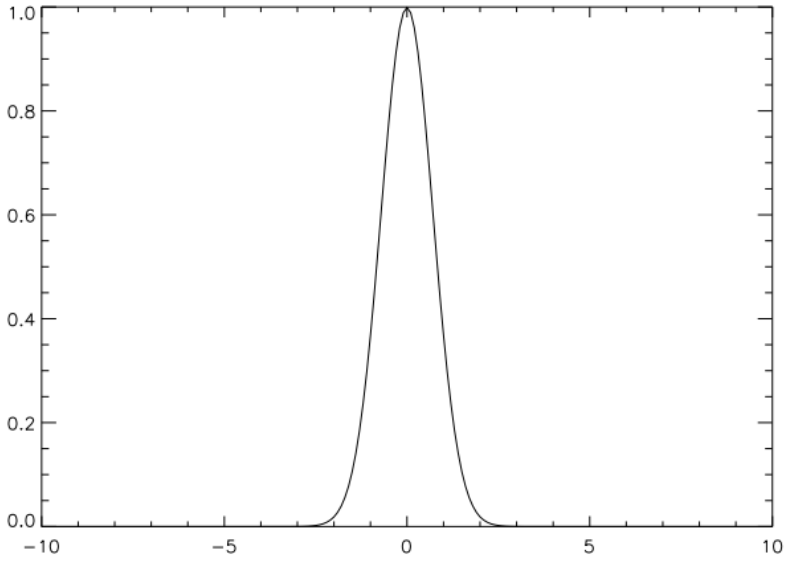

Fig. 15. This graph now decreased a to 0.001. We see a much narrow and taller peak. The Lorentzian wings contribute little and most of the width is due to the Doppler effect.

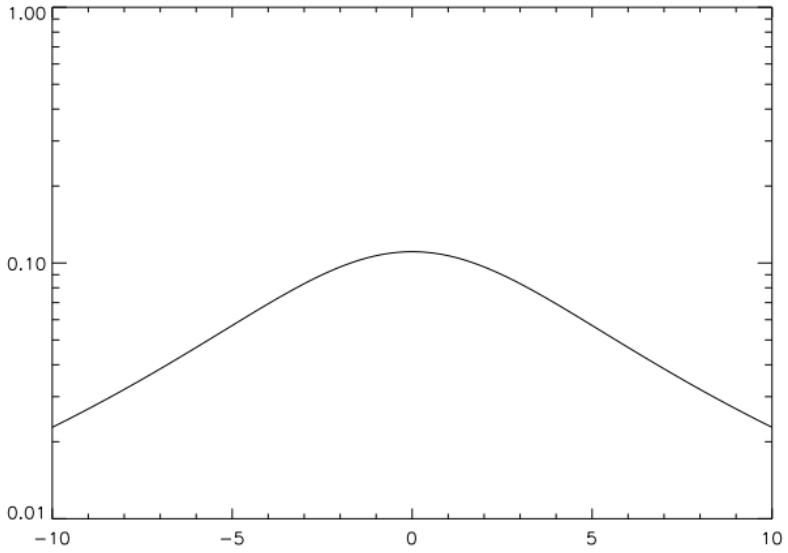

Fig. 16. We now graph a version with a=0.1 and with the y-axis as logarithmic so we can inspect the wings of the function. The slope of the wings grows flatter for larger values of $abs(u)$.

We can see a few patterns from these four graphs. Firstly, higher values of a indicate more damping due to greater Coulomb interactions. A stellar atmosphere with higher a values will see a wider range of λ with absorption lines for a given compound. Also, the wings get very small as abs(u) moves away from the theoretical absorption line, but it never quite reaches zero. The log plot also shows us that the area under the curve due to the Lorentz wings can be more significant than one may think after inspecting the first 3 plots. It's important to not underestimate the Lorentzian damping in contributions to spectral broadening.

We now can plot accurate spectral line profiles building on our knowledge of the Planck distribution, emergent intensity given a solar reversing layer and the solar surface, and convolutions experienced in real world analysis due to the Doppler effect and Coulomb disturbances. The following plots are derived from Schuster-Schwarzschild line profiles.

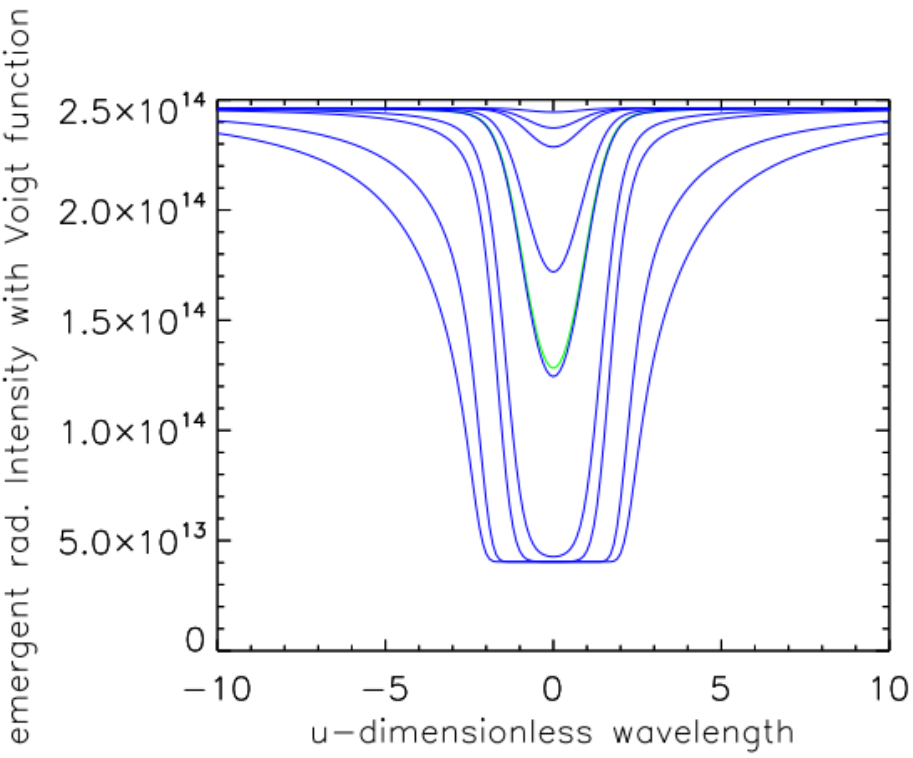

Fig. 17. This graph plots the Schuster-Schwarzschild line profile for $T_{surface} = 5700K$, $T_{layer} = 4200K$, $a = 0.1$, $\lambda = 5000e-8cm$, u ranging from -10 to 10. $\tau(0) = 1$ (in green) and then τ ranging from [0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100] (in blue).

One of the most striking features of this plot is the strong dependence on τ. We see for small values of τ the emergent intensity is roughly unchanged near the absorption wavelength (u=0). By analyzing these plots and IDL functions, values for $\tau(0)$ on the order of .025 or less tend to be optically thin. Here the Doppler effect dominates the spectral profile. τ displays traditionally optically thick behavior beginning around $\tau(0) = 5$. The emergent intensity is based solely on the Planck function of the surface temperature as the layer contributions vanish as $e^{-(\tau<<1)} \rightarrow 1$ eliminates the second half of the emergent radiation equation (equation 13 in assignment).

We also see that there is a lower limit on intensity for the optically thick values of τ. This is evident from the equation as large values of τ render the equation independent of τ as $e^{-\tau>>1} \rightarrow 0$. In this case the emergent intensity $I_\lambda = B_\lambda(T_{layer})$. So larger values of τ converge on $B_\lambda(T_{layer})$, which in the case of the plot above is about $4.0x10^{13}$.

The Lorentzian wings only are visible for larger $\tau(0)$ as I_λ is nearly independent of τ for small values. However, the wings are still there, they are just too small to be illustrated by this plot. The wings are asymptotic as we can see from the convolution Voigt function that u values range from $-\infty$ to ∞.

We'll now introduce λ as a variable and alter the wavelength values from the ultraviolet to the infrared spectrum and plot the curves.

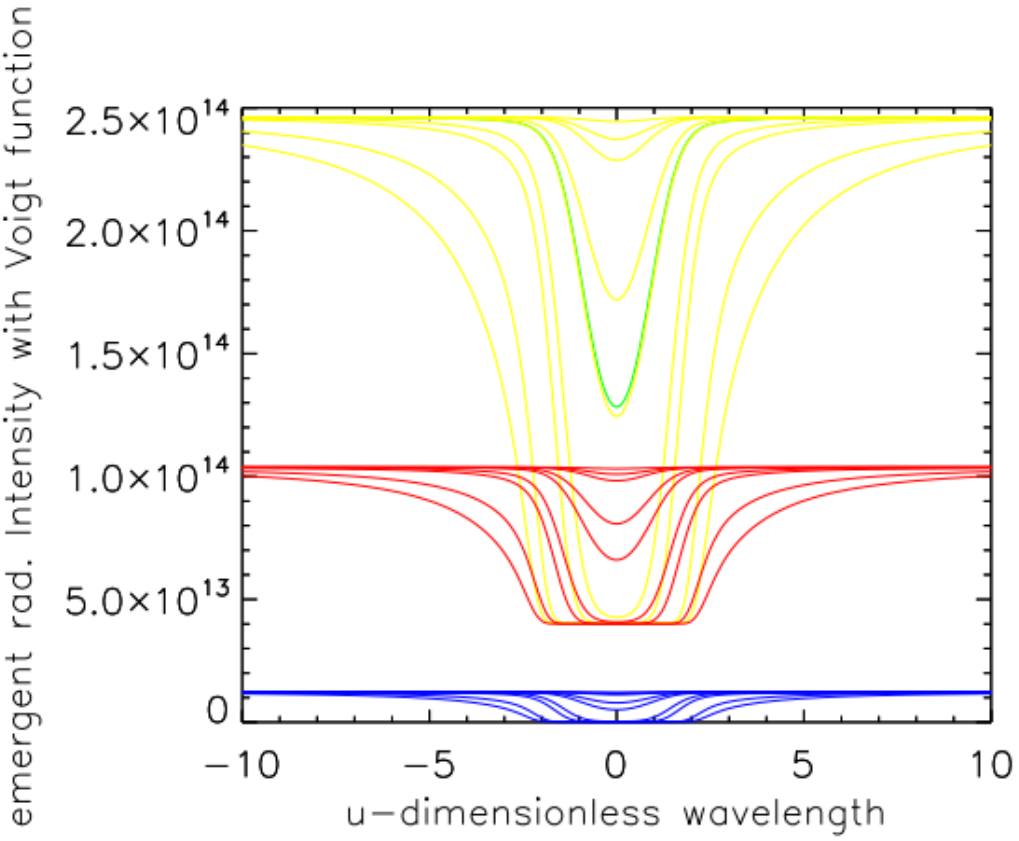

Fig. 18. This graph plots the Schuster-Schwarzschild line profile for 3 different wavelengths: 2000 Å in blue (ultraviolet), 5000Å in yellow (visible), and 10000 Å in red (near infrared). $T_{surface}$ = 5700K, T_{layer} = 4200K, a = 0.1, u ranging from -10 to 10. $\tau(0)$ = 1 (in green) and then τ again ranging from [0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100].

The limit value reached by I_{cont} is determined by the wavelength with surface temperature held constant at 5700K in Planck function. We see this plot as a different illustration of Wien's displacement law than the usual formulation: Planck functions for the same temperature have roughly the same shape but displaced for different values of λ. We see there is a large difference in the emergent intensities given the 3 wavelengths: visible light is the strongest, followed by near infrared, and then ultraviolet. I've calculated the Planck function for the pictured wavelengths and surface/layer temperatures. The numerical results are below and highlight the different intensities based on differing wavelengths.

Temperature(K)	**wavelength Å**	**Planck function**
5700	2000	$1.2292x10^{13}$
5700	5000	$2.4627x10^{14}$
5700	10000	$1.037x10^{14}$
4200	2000	$1.35x10^{11}$
4200	5000	$4.03x10^{13}$
4200	10000	$4.004x10^{13}$

We now graph a plot of the relative intensity which normalizes the I_λ to the $I_{continuous}$.

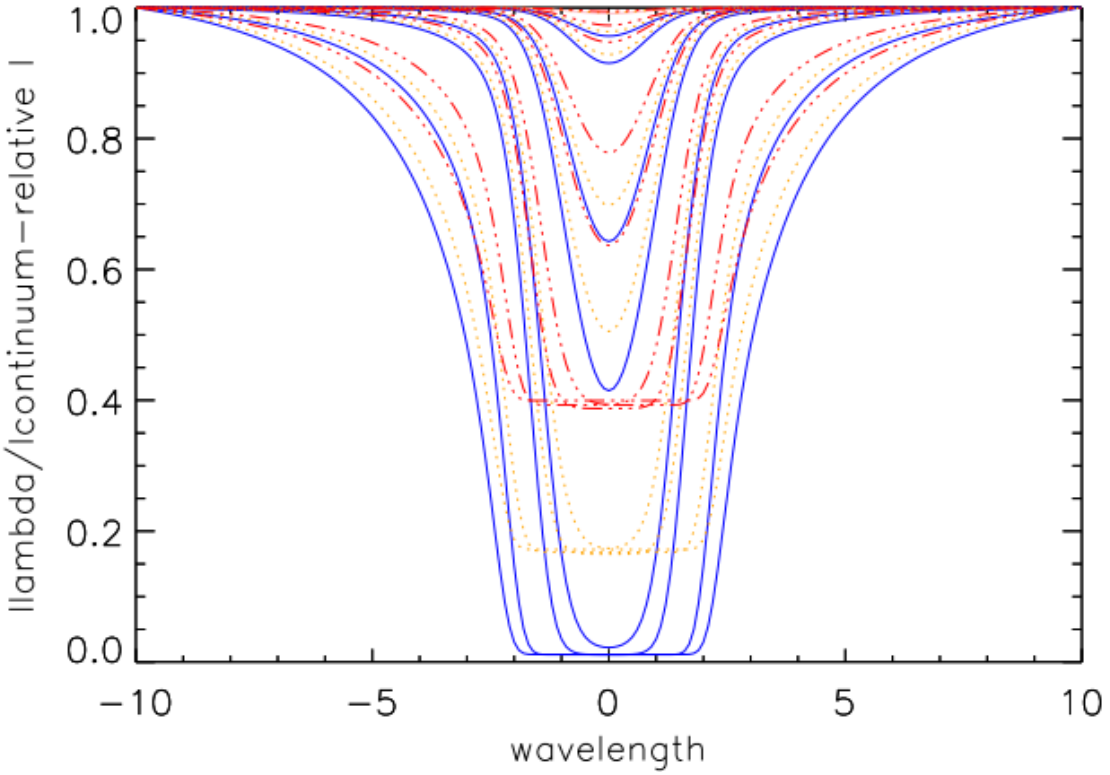

Fig. 19. This graph plots the normalized Schuster-Schwarzschild line profile for 3 different wavelengths: 2000 Å in blue (ultraviolet), 5000Å in yellow (visible), and 10000 Å in red (near infrared). $T_{surface} = 5700K$, $T_{layer} = 4200K$, $a = 0.1$, u ranging from -10 to 10. $\tau(0) = 1$ (in green) and then τ again ranging from [0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100].

This plot illustrates the relative absorption of radiation for the 3 different wavelengths, at least for optically thick τ. We see that the layer is most absorptive for the ultraviolet wavelength radiation, which absorbs around 99% of relative intensity near u=0. The layer is the next most absorptive is for the visible wavelength, which maximally absorbs 80% of relative intensity. The least absorptive wavelength is the near infrared, of which 40% is absorbed at its peak absorption point.

We now look at the equivalent width of the spectral lines, which is a mathematical method that allows for easier to recognize spectral signatures and relative strengths. As we saw from the logarithmic plot of the Voigt function, the contribution from the Lorentzian wings can be important and hard to visually tally. This simplified geometric view of the area under the curve allows us to account for the long wings that occur when $\tau(0)$ is large and make comparisons between spectral bands much easier.

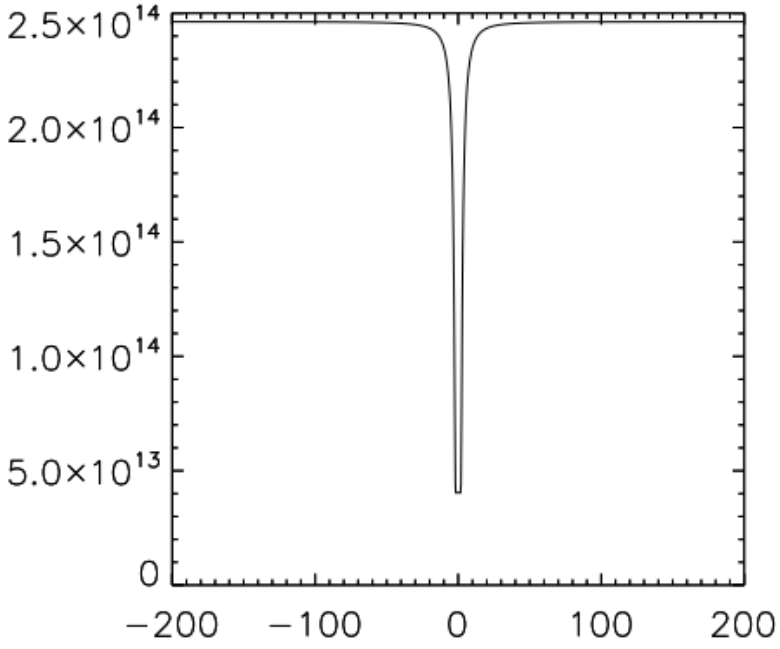

Fig. 20. Here we graph the profile function as usual with the parameters a, u, and $\tau(0)$

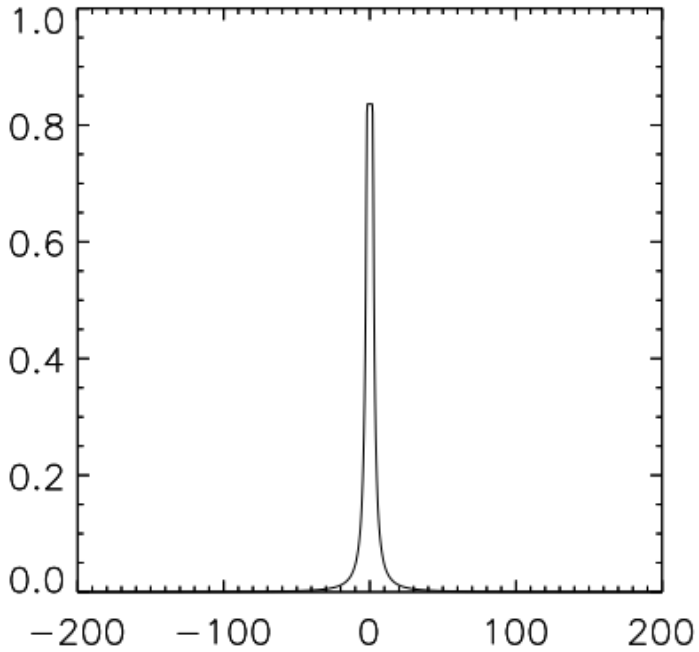

Fig. 21. This plot uses the same profile function as above but computes the equivalent width using the line depth relative to the $I(0)$ continuum. The equivalent width (area under the curve) is given by the total of the sum of the Y-axis elements multiplied by 0.4 which is the interval we have graphed u by.This gives us the area under the curve. The result is **7.53681**.

There should be a direct correlation between the equivalent width of a spectral line caused by a population of atoms in the reversing layer and the entity $\tau(0)$ of the reversing layer. Here we write an IDL code to plot the growth curve of the equivalent width of a population of atoms equivalent width as a function of $\tau(0)$.

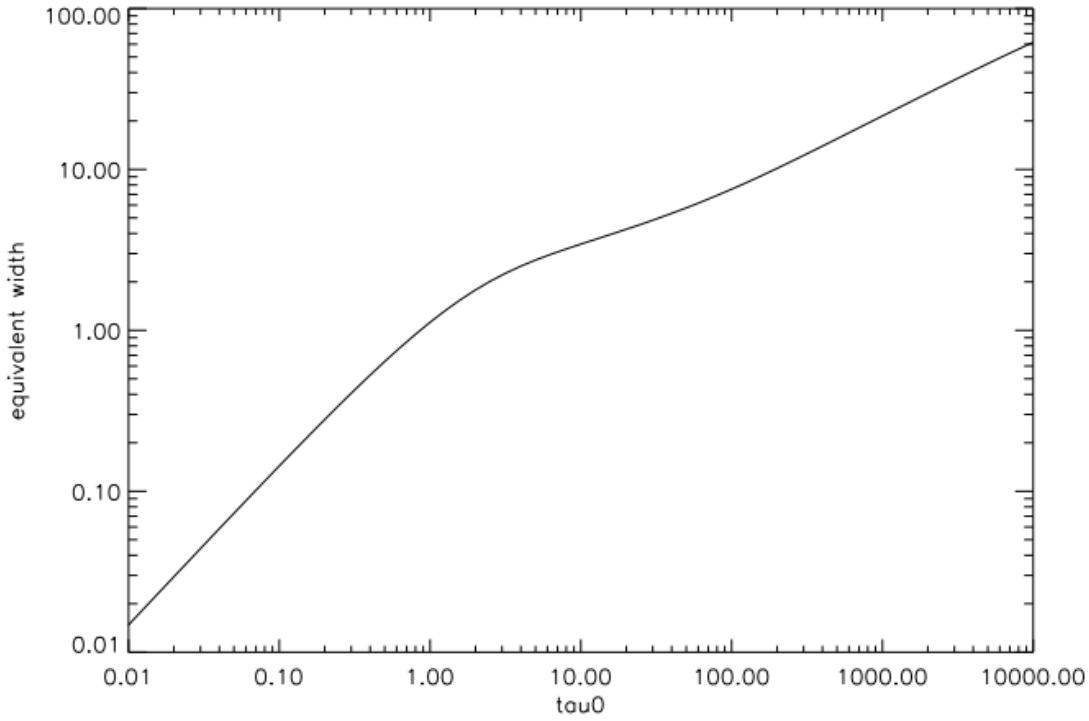

Fig. 22. This plot shows us the dependence of $\tau(0)$ on the different regions of absorption strength. The y-axis plots the equivalent width which accounts for the Planck function, the Voigt function, and wavelength.

There are three distinct regions in the plot that we can analyze based on previous behavior characteristics of τ. We see for optically thin regions (for $\tau(0) < 1$) that the growth is logarithmically 1:1. The Planck function for $T_{surface}$ dominates the spectrum, so growth of $\tau(0)$ has an exponential growth in absorption equivalent width. The Doppler effect is also much more dominant than the damping effect in this region. The middle region where $\tau(0)$ is close to 1 demonstrates a mixture of contributions from the surface and reversing layer Planck functions when $e^{-\tau(0)}$ is in the order of magnitude of 1. This is where the population begins to saturate and reaches peak density. The third section of the plot enters the optically thick region of $\tau(0)$, which we suggested earlier was for values over 5.0. Here the absorption width for the population of atoms is approaching saturation as we saw in section 3.3. Also, the Planck function becomes much more dependent on the temperature of the reversing layer and greater values of $\tau(0)$ contribute less and less to the absorption intensity. The Lorentzian wings also dominate the Voigt function over the Doppler effects. This occurs as the population density reaches a certain point where additional absorption occurs less quickly. From analyzing figure 14 in the assignment, we see that solar iron lines are around $log0.8$. The units here scale with the transition probability and the population density of the the lower lever of each transition line, as computed from the Saha-Boltzmann laws.

We have been studying solar absorption lines. In order to switch from absorption to emission, we need to invert the temperature differential in the Schuster-Schwarzschild model. By allowing T_{layer} to be greater than $T_{surface}$, we will then see emission lines coming from the emergent radiation instead of absorption lines. This is equivalent to experiments on earth where a gas is heated and the resulting spectrum is analyzed for emission characteristics. I have included several plots from 3-3 to 3-5 that reflect the emission. The change here has been to keep $T_{surface} = 5700K$ and increase T_{layer} to 7400K.

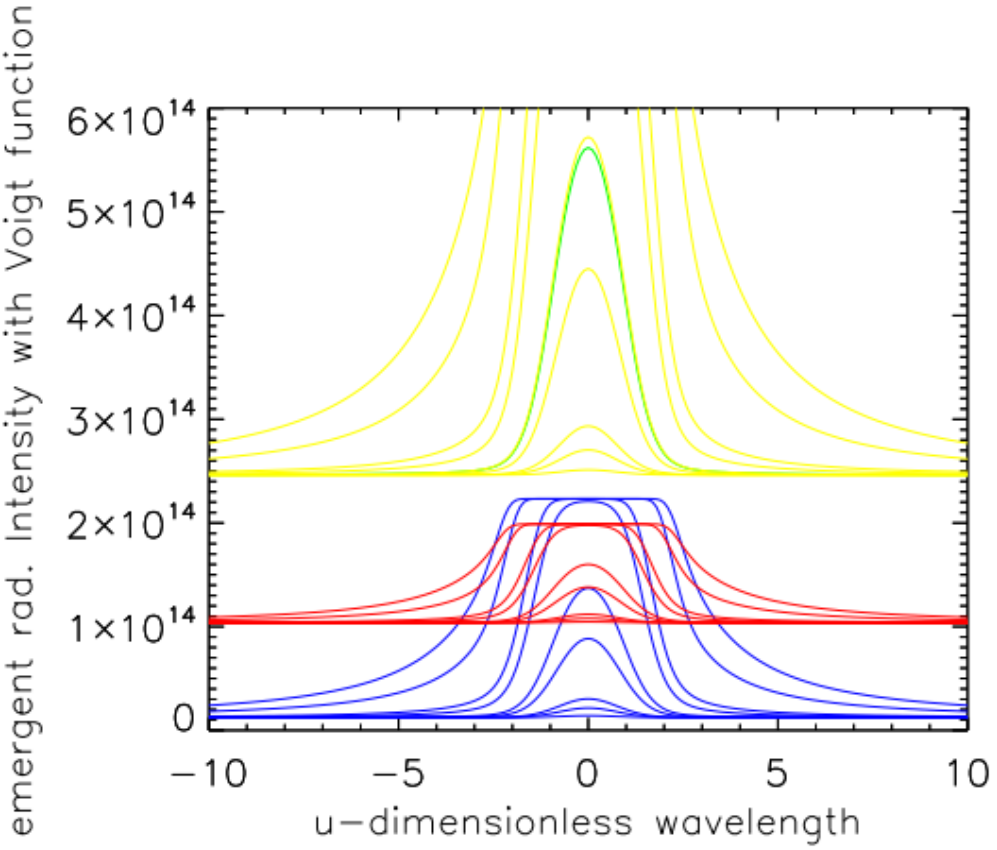

Fig. 23. This plot shows the Schuster-Schwarzschild emission spectrum for varying $\tau(0)$. The plot includes the following characteristics for the 3 different wavelengths: 2000 Å in blue (ultraviolet), 5000Å in yellow (visible), and 10000 Å in red (near infrared). $T_{surface} = 5700K$, $T_{layer} = 7400K$, $a = 0.1$, u ranging from -10 to 10. $\tau(0) = 1$ (in green) and then τ again ranging from [0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100].

We can see there is an inversion in the y-axis. From the growth curve we can also see that there is an inversion on the $\tau(0)$ dependence. We see that the 1:1 regime is now for the optically thick τ»1 region and the slow growth to emission occurs for the optically thin τ«1 region. The inversion of the temperature differential makes this reversal of τ dependence intuitive as optically thick regions will emit more than optically thin. The intensity of the reversing layer is more dominant than the surface in this case.

5. Conclusions

In this exercise we have explored the the nature of solar spectral lines and redeveloped many of the early theories of spectral analysis. We focused first on the terrestrially derived physical laws that govern the atomic energy level transitions and interactions that cause the absorption and emission phenomena. Applying these to the solar spectra was a powerful tool in determining the composition and processes of the sun. In the second part of the assignment, we focused on the laws and regimes of thermodynamics that govern the behavior of electromagnetic radiation. Through IDL functions, numerical analyses, and the extrapolation of information from plots, we were able to derive methods of identifying spectral phenomena and the ways in which these distinct atomic signatures can tell us a great deal about objects' radiation signatures very far away from us.

6. Appendix-IDL Code

Here are my IDL codes that generated my plots and tables and was the basis for much of the analysis as in the report.

SSA2.pro

```
function partfunc_E, temp
; partition functions Schadee element
; input: temp (K)
; output: fltarr(4) = partition functions U1,..,U4
u=fltarr(4)
chiion=[7,16,31,51]
k=8.61734D-5
for r = 0,3 do $
for s = 0, chiion[r]-1 do u[r]=u[r] + exp(-s/(k*temp))
return,u
end

function boltz_E,temp,r,s
  ; compute Boltzmann population for level r,s of Schadee element E
  ; input: temp (temperature, K)
  ;r (ionization stage nr, 1 - 4 where 1 = neutral E)
  ;s (level nr, starting at s=1)
  ; output: relative level population n_(r,s)/N_r
  u=partfunc_E(temp)
  keV=8.61734D-5
  ; Boltzmann constant in ev/deg
  relnrs = 1./u[r-1]*exp(-(s-1)/(keV*temp))
  return, relnrs
end

function saha_E,temp,elpress,ionstage
  ; compute Saha population fraction N_r/N for Schadee element E
  ; input: temperature, electron pressure, ion stage
  ; physics constants
  kerg=1.380658D-16     ; Boltzmann constant (erg K; double precision)
  kev=8.61734D-5        ; Boltzmann constant (eV/deg)
  h=6.62607D-27         ; Planck constant (erg s)
  elmass=9.109390D-28 ; electron mass (g)
  ; kT and electron density
  kevT=kev*temp
  kergT=kerg*temp
  eldens=elpress/kergT
  chiion=[7,16,31,51]     ; ionization energies for element E
  u=partfunc_E(temp)      ; get partition functions U[0]...u[3]
  u=[u,2]                 ; add estimated fifth value to get N_4 too
  sahaconst=(2*!pi*elmass*kergT/(h*h))^1.5 * 2./eldens
  nstage=dblarr(5)        ; double-precision float array
  nstage[0]=1.            ; relative fractions only (no abundance)
  for r=0,3 do $
    nstage[r+1] = nstage[r]*sahaconst*u[r+1]/u[r]*exp(-chiion(r)/kevT)
  ntotal=total(nstage)    ; sum all stages = element density
  nstagerel=nstage/ntotal ; fractions of element density
  return,nstagerel[ionstage-1] ; ion stages start at 1, IDL at 0
end

function sahabolt_E,temp,elpress,ion,level
  ; compute Saha-Boltzmann populaton n_(r,s)/N for level r,s of E
  ; input: temperature, electron pressure, ionization stage, level nr
  return, saha_E(temp,elpress,ion) * boltz_E(temp,ion,level)
end

function sahabolt_H,temp,elpress,level
  ; compute Saha-Boltzmann population n_(1,s)/N_H for hydrogen level
  ; input: temperature, electron pressure, level number
  ; physics constants
```

```
  kerg=1.380658D-16   ; Boltzmann constant (erg K; double precision)
  kev=8.61734D-5      ; Boltzmann constant (eV/deg)
  h=6.62607D-27       ; Planck constant (erg s)
  elmass=9.109390D-28 ; electron mass (g)
  ; kT and electron density
  kevT=kev*temp
  kergT=kerg*temp
  eldens=elpress/kergT        ; energy levels and weights for hydrogen
  nrlevels=100                ; reasonable partition function cut-off value
  g=intarr(2,nrlevels)        ; declaration weights (too many for proton)
  chiexc=fltarr(2,nrlevels)   ; declaration excitation energies (idem)
  for s=0,nrlevels-1 do begin ; enclose multiple lines with begin...end
    g[0,s]=2*(s+1)^2                  ; statistical weights
    chiexc[0,s]=13.598*(1-1./(s+1)^2) ; excitation energies
  endfor                      ; begin...end cannot go on command line!
  g[1,0]=1                    ; statistical weight free proton
  chiexc[1,0]=0.              ; excitation energy proton ground state
  ; partition functions
  u=fltarr(2)
  u[0]=0
  for s=0,nrlevels-1 do u[0]=u[0]+ g[0,s]*exp(-chiexc[0,s]/kevT)
  u[1]=g[1,0]
  ; Saha
  sahaconst=(2*!pi*elmass*kergT/(h*h))^1.5 * 2./eldens
  nstage=dblarr(2)        ; double-precision float array
  nstage[0]=1.            ; relative fractions only
  nstage[1] = nstage[0] * sahaconst * u[1]/u[0] * exp(-13.598/kevT)
  ntotal=total(nstage)    ; sum both stages = total hydrogen density
  ; Boltzmann
  nlevel = nstage[0]*g[0,level-1]/u[0]*exp(-chiexc[0,level-1]/kevT)
  nlevelrel=nlevel/ntotal ; fraction of total hydrogen density
  ;stop                   ; in for parameter inspection
  return,nlevelrel
end

function partfunc_Ca, temp
  ; partition functions Ca+
  ; input: temp (K)
  ; output: fltarr(4) = partition functions U1,..,U4
  u=fltarr(4)
  chiion=[6.113,11.871,50.91,67.15]
  k=8.61734D-5
  for r = 0,3 do $
    for s = 0, chiion[r]-1 do u[r]=u[r] + exp(-s/(k*temp))
  return,u
end

function boltz_Ca,temp,r,s
  ; compute Boltzmann population for level r,s of Schadee element E
  ; input: temp (temperature, K)
  ;r (ionization stage nr, 1 - 4 where 1 = neutral E)
  ;s (level nr, starting at s=1)
  ; output: relative level population n_(r,s)/N_r
  u=partfunc_E(temp)
  keV=8.61734D-5
  ; Boltzmann constant in ev/deg
  relnrs = 1./u[r-1]*exp(-(s-1)/(keV*temp))
  return, relnrs
end

function saha_Ca,temp,elpress,ionstage
  ; compute Saha population fraction N_r/N for Schadee element E
  ; input: temperature, electron pressure, ion stage
  ; physics constants
```

```
  kerg=1.380658D-16     ; Boltzmann constant (erg K; double precision)
  kev=8.61734D-5        ; Boltzmann constant (eV/deg)
  h=6.62607D-27         ; Planck constant (erg s)
  elmass=9.109390D-28 ; electron mass (g)
  ; kT and electron density
  kevT=kev*temp
  kergT=kerg*temp
  eldens=elpress/kergT
  chiion=[6.113,11.871,50.91,67.15]     ; ionization energies for element E
  u=partfunc_Ca(temp)      ; get partition functions U[0]...u[3]
  u=[u,2]                  ; add estimated fifth value to get N_4 too
  sahaconst=(2*!pi*elmass*kergT/(h*h))^1.5 * 2./eldens
  nstage=dblarr(5)        ; double-precision float array
  nstage[0]=1.            ; relative fractions only (no abundance)
  for r=0,3 do $
    nstage[r+1] = nstage[r]*sahaconst*u[r+1]/u[r]*exp(-chiion(r)/kevT)
  ntotal=total(nstage)    ; sum all stages = element density
  nstagerel=nstage/ntotal ; fractions of element density
  return,nstagerel[ionstage-1] ; ion stages start at 1, IDL at 0
end

function sahabolt_Ca,temp,elpress,ion,level
  ; compute Saha-Boltzmann populaton n_(r,s)/N for level r,s of E
  ; input: temperature, electron pressure, ionization stage, level nr
  return, saha_Ca(temp,elpress,ion) * boltz_Ca(temp,ion,level)
end


resultsplot.pro
set_plot, 'ps'
device, filename='resultsplotsep.ps'
temp=1000*indgen(31) ; make array 0,...,30000 in steps of 1000 K
print,temp ; check
pop=fltarr(5,31) ; declare float array for n(r,T)

for T=1,30 do $ ; $ continues statement to next line
for r=1,4 do pop[r,T]=sahabolt_E(temp[T],131.,r,1)

cgPlot, temp, pop[1,*], /ylog,yrange=[1E-3,1.1], xrange=[0,32000], /window,$
BACKGROUND=white, TITLE='R = 1,2,3,4 S = 1 Populations n_r,1/N',$
COLOR='Grn3', XTITLE='temp',YTITLE='population', LEGENDS='r1=grn3'
cgPlot, temp, pop[2,*], /WINDOW, /OVERPLOT,$
COLOR='Red3', LEGENDS= 'r2=red3'
cgPlot, temp, pop[3,*], /WINDOW, /OVERPLOT,$
COLOR='Blu3', LEGENDS= 'r3=blu3'
cgPlot, temp, pop[4,*], /WINDOW, /OVERPLOT,$
COLOR='ORG3', LEGENDS= 'r4=org3'

for T=1,30 do $; repeat for s=2 (excitation energy = 1 eV)
  for r=1,4 do pop[r,T]=sahabolt_E(temp[T],131.,r,2)
  cgPlot, temp, pop[1,*], /window, TITLE='R = 1,2,3,4 S = 2 Populations n_r,2/N' $
    COLOR='Grn5', XTITLE='temp',YTITLE='population', LEGENDS='r1=grn5''
  cgPlot, temp, pop[2,*], /WINDOW, /OVERPLOT,$
    COLOR='Red5', LEGENDS= 'r2=red5'
  cgPlot, temp, pop[3,*], /WINDOW, /OVERPLOT,$
    COLOR='Blu5', LEGENDS= 'r3=blu5'
  cgPlot, temp, pop[4,*], /WINDOW, /OVERPLOT,$
    COLOR='org5', LEGENDS= 'r4=org5'

for T=1,30 do $
  ; repeat for s=4 (excitation energy = 3 eV)
  for r=1,4 do pop[r,T]=sahabolt_E(temp[T],131.,r,4)
  cgPlot, temp, pop[1,*], /window,TITLE='R = 1,2,3,4 S =4 Populations n_r,4/N' $
    COLOR='Grn7', XTITLE='temp',YTITLE='population', LEGENDS='r1=grn7'
```

```
  cgPlot, temp, pop[2,*], /WINDOW, /OVERPLOT,$
    COLOR='Red7', LEGENDS= 'r2=red7'
  cgPlot, temp, pop[3,*], /WINDOW, /OVERPLOT,$
    COLOR='Blu7', LEGENDS= 'r3=blu7'
  cgPlot, temp, pop[4,*], /WINDOW, /OVERPLOT,$
    COLOR='org7', LEGENDS= 'r4=org7'
device,/close
filenmame='resultsplot2.ps'
end

2-9CavsHtemp.pro
; temperature sensitivity CaIIK and Halpha
set_plot, 'ps'
device, filename='2-9CavsHtempcombo.ps'
temp=indgen(101)*100.+2000.   ; T = 2000-12000, delta T = 100
dNCadT = temp                 ; declare array
dNHdT = temp                   ; declare array
dT=1.
for i=0,100 do begin
  NCa = sahabolt_ca(temp[i],1e2,2,1)      ; Ca ion ground state
  NCa2 = sahabolt_ca(temp[i]-dT,1e2,2,1)  ; idem dT cooler
  dNCadT[i]= (NCa - NCa2)/dT/NCa          ; fractional diff quotient
  NH = sahabolt_H(temp[i],1e2,2)          ; H atom 2nd level
  NH2 = sahabolt_H(temp[i]-dT,1e2,2)      ; idem dT cooler
  dNHdT[i] = (NH-NH2)/dT/NH               ; fractional diff quotient
endfor
plot,temp,abs(dNHdT),/ylog,yrange=[1E-5,1],$
  xtitle='temperature',ytitle='abs d n(r,s) / n(r,s)'
oplot,temp,abs(dNCadT),linestyle=2        ; Ca curve dashed
NCa=temp      ; declare array
NH=temp       ; declare array
for i=0,100 do begin
  NCa[i] = sahabolt_ca(temp[i],1e2,2,1) ; Ca ion ground state
  NH[i] = sahabolt_H(temp[i],1e2,2)     ; H atom 2nd level
endfor
oplot,temp,NH/max(NH)
oplot,temp,NCa/max(NCa),linestyle=2     ; Ca curve again dashed
device,/close
filenmame='2-9CavsHtempcombo.ps'
end

CavsHplot.pro
set_plot, 'ps'
device, filename='CavsHplot.ps'
temp=indgen(191)*100.+1000  ; T = 1000-20000 in delta T = 100
CaH = temp    ; declare ratio array
Caabund=2.E-6 ; A_Ca = N_Ca / N_H
for i=0,190 do begin
  NCa = sahabolt_Ca(temp[i],1e2,2,1)
  NH = sahabolt_H(temp[i],1e2,2)
  CaH[i]=NCa*Caabund/NH
endfor
plot,temp,CaH,/ylog,$
  xtitle='temperature',ytitle='Ca II K / H alpha'
  device,/close
  filenmame='CavsHplot.ps'
  end

2-9dipplot.pro
; recompute as arrays and overplot relative
set_plot, 'ps'
device, filename='2-9dipplot.ps'
NCa=temp      ; declare array
NH=temp       ; declare array
```

```
for i=0,100 do begin
  NCa[i] = sahabolt_ca(temp[i],1e2,2,1) ; Ca ion ground state
  NH[i] = sahabolt_H(temp[i],1e2,2)     ; H atom 2nd level
endfor
oplot,temp,NH/max(NH)
oplot,temp,NCa/max(NCa),linestyle=2     ; Ca curve again dashed
device,/close
filenmame='2-9dipplot.ps'
end

2-10.plot
set_plot, 'ps'
device, filename='2-10plot.ps'
temp=indgen(191)*100.+1000. ; array 1000 - 20 000 in steps 1000
nH=temp                     ; declare same size array
for i=0,190 do nH[i]=sahabolt_H(temp[i],1e2,1)
plot,temp,nH,$
  xtitle='temperature',ytitle='neutral hydrogen fraction'
  device,/close
  filenmame='2-10plot.ps'
end

ssa3.pro
set_plot, 'ps'
device, filename='plancktempwav.ps'
wav=indgen(100)*200.+1000.  ; produces wav[0,...99] = 1000 - 20800
print,wav                   ; check that
b=wav                       ; declare float array of the same size
c=wav
for i=0,99 do b[i]=planck_1(8000.,wav[i]*1.E-8)
cgplot,wav,b,/WINDOW, ASPECT =0.8,xtitle='wavelength (Angstrom)',ytitle='Planck function', $
  xmargin = [45,5],$
  charsize=1.1              ; bigger characters
for T=8000.,5000.,-200. do begin ; step from 8000 K down to 5000 K
  for i=0,99. do b[i]=planck_1(T,wav[i]*1.E-8)
 cgplot,wav,b,/window, /OVERPLOT              ; overplots extra curves in existing graph
endfor                      ; begin...end sequences can't go on command line
device,/close
filenmame='plancktempwav.ps'
end

addup.pro
function addup,arr
  ;+
  ; sums 1D array ARR (but IDL's total is faster and more general)
  ;-
  arraysize=SIZE(arr)
  if (arraysize[0] ne 1) then print,'addup input is not a 1D array'
  sumarr=0
  for i=0,arraysize[1]-1 do sumarr=sumarr+arr[i]
  return,sumarr
end

3-3voigt.pro
set_plot, 'ps'
device, filename='3-3voigtalog.ps'
u=indgen(201)/10.-10.                    ;u = -10 to 10 in 0.1 steps
vau=u                                    ;declare same-size array
a=5                                     ;damping parameter
for i=0,200 do vau[i]=voigt(a,abs(u[i]))  ;taking abs corrects IDL errors
plot,u,vau,/ylog                  ;yrange fixed to compare plots
print, vau
device,/close
filenmame='3-3voigtalog.ps'
```

```
end

planck_1.pro
function planck_1, temp, wav    ;planck function in terms of temp and wavelength
h= 6.62607D-27 ;planck constant in erg*s
c=2.99792D10  ;speed of light in cm*s^-1
k= 1.38065D-16 ; Boltzmann constant in ergK^-1
const= 2*h*c^2
econst= (h*c)/k
evar= temp*wav
pfunc= (const/(wav^5))*1/(exp(econst/(evar))-1)
return, pfunc
end

planckplot.pro
set_plot, 'ps'
device, filename='plancktempwavLOGY.ps'
wav=indgen(100)*200.+1000.  ; produces wav[0,...99] = 1000 - 20800
print,wav                   ; check that
b=wav                       ; declare float array of the same size
c=wav
for i=0,99 do b[i]=planck_1(8000.,wav[i]*1.E-8)
cgplot,wav,b,/WINDOW, ASPECT =0.8,/ylog,xtitle='wavelength (Angstrom)',
ytitle='Planck function', $
  xmargin = [45,5],$
  charsize=1.1              ; bigger characters
for T=8000.,5000.,-200. do begin ; step from 8000 K down to 5000 K
  for i=0,99. do b[i]=planck_1(T,wav[i]*1.E-8)
 cgplot,wav,b,/window, /OVERPLOT              ; overplots extra curves in existing graph
endfor                      ; begin...end sequences can't go on command line
device,/close
filenmame='plancktempwavLOGY.ps'
end

planckplotlogxy.pro
set_plot, 'ps'
device, filename='plancktempwavLOGXY.ps'
wav=indgen(100)*200.+1000.  ; produces wav[0,...99] = 1000 - 20800
print,wav                   ; check that
b=wav                       ; declare float array of the same size
c=wav
for i=0,99 do b[i]=planck_1(8000.,wav[i]*1.E-8)
cgplot,wav,b,/WINDOW, ASPECT =0.8,/ylog,/xlog,xtitle='wavelength (Angstrom)',
ytitle='Planck function', $
  xmargin = [45,5],$
  charsize=1.1              ; bigger characters
for T=8000.,5000.,-200. do begin ; step from 8000 K down to 5000 K
  for i=0,99. do b[i]=planck_1(T,wav[i]*1.E-8)
 cgplot,wav,b,/window, /OVERPLOT              ; overplots extra curves in existing graph
endfor                      ; begin...end sequences can't go on command line
device,/close
filenmame='plancktempwavLOGXY.ps'
end

schusterdifferentlambda.pro
set_plot, 'ps'
device, filename='3-3schusterprofiledifferentlambda.ps'
Ts=5700                 ; solar surface temperature
Tl=4200                 ; solar T-min temperature = 'reversing layer'
a=0.1                   ; damping parameter
wav=5000.D-8            ; wavelength in cm
tau0=.95                  ; reversing layer thickness at line center
u=indgen(201)/10.-10.   ; u = -10 to 10 in 0.1 steps
int=u                   ; declare array
```

```
for i=0,200 do begin
  tau=tau0 * voigt(a,abs(u[i]))
  int[i]=planck_1(Ts,wav) * exp(-tau) + planck_1(Tl,wav)*(1.-exp(-tau))
endfor
cgplot,u,int, aspect=.82, xtitle='u-dimensionless wavelength', ytitle='emergent rad.
Intensity with Voigt function', color = 'green'

btau0=[0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100]
for itau=0,8 do begin
  for i=0,200 do begin
    tau=btau0[itau] * voigt(a,abs(u[i]))
    int[i]=planck_1(Ts,wav) * exp(-tau) + planck_1(Tl,wav)*(1.-exp(-tau))
  endfor
  cgplot,u,int, color='yellow', /overplot, yrange = [0,1e15]
endfor

wav=2000.D-8            ; ultraviolet wavelength in cm
rtau0=[0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100]
for itau=0,8 do begin
  for i=0,200 do begin
    tau=rtau0[itau] * voigt(a,abs(u[i]))
    int[i]=planck_1(Ts,wav) * exp(-tau) + planck_1(Tl,wav)*(1.-exp(-tau))
  endfor
  cgplot,u,int, color='blue', /overplot
endfor

wav=10000.D-8            ; infrared wavelength in cm
tau0=[0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100]
for itau=0,8 do begin
  for i=0,200 do begin
    tau=tau0[itau] * voigt(a,abs(u[i]))
    int[i]=planck_1(Ts,wav) * exp(-tau) + planck_1(Tl,wav)*(1.-exp(-tau))
  endfor
  cgplot,u,int, color='red', /overplot
endfor

device,/close
filenmame='3-3schusterprofiledifferentlambda.ps'
end

schusterprofile.pro
set_plot, 'ps'
device, filename='3-3schusterprofile2.ps'
Ts=5700                 ; solar surface temperature
Tl=4200                 ; solar T-min temperature = 'reversing layer'
a=0.1                   ; damping parameter
wav=15000.D-8            ; wavelength in cm
tau0=.95                  ; reversing layer thickness at line center
u=indgen(201)/10.-10.   ; u = -10 to 10 in 0.1 steps
int=u                   ; declare array
for i=0,200 do begin
  tau=tau0 * voigt(a,abs(u[i]))
  int[i]=planck_1(Ts,wav) * exp(-tau) + planck_1(Tl,wav)*(1.-exp(-tau))
endfor
cgplot,u,int, aspect=.82, xtitle='u-dimensionless wavelength', ytitle='emergent rad.
Intensity with Voigt function', color = 'green'

tau0=[0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100]
for itau=0,8 do begin
  for i=0,200 do begin
    tau=tau0[itau] * voigt(a,abs(u[i]))
    int[i]=planck_1(Ts,wav) * exp(-tau) + planck_1(Tl,wav)*(1.-exp(-tau))
  endfor
  cgplot,u,int, color='blue', /overplot
```

```
endfor
device,/close
filenmame='3-3schusterprofile2.ps'
end

scaledintensity.pro
function addup,arr
  ;+
  ; sums 1D array ARR (but IDL's total is faster and more general)
  ;-
  arraysize=SIZE(arr)
  if (arraysize[0] ne 1) then print,'addup input is not a 1D array'
  sumarr=0
  for i=0,arraysize[1]-1 do sumarr=sumarr+arr[i]
  return,sumarr
end

equivalentwidth.pro
function profile,a,tau0,u
  ; return a Schuster-Schwarzschild profile
  ; input: a = damping parameter
  ; tau0 = SS layer thickness at line center
  ; u = wavelength array in Doppler units
  ; output: int = intensity array
  Ts=5700
  Tl=4200
  wav=5000.E-8
  int=u
  usize=SIZE(u)   ; IDL SIZE returns array type and dimensions
  for i=0,usize[1]-1 do begin
    tau=tau0 * voigt(a,abs(u[i]))
    int[i]=planck_1(Ts,wav)*exp(-tau) + planck_1(Tl,wav)*(1.-exp(-tau))
  endfor
  return,int
end

3-4eqivalentwidthplot.pro
function addup,arr
  ;+
  ; sums 1D array ARR (but IDL's total is faster and more general)
  ;-
  arraysize=SIZE(arr)
  if (arraysize[0] ne 1) then print,'addup input is not a 1D array'
  sumarr=0
  for i=0,arraysize[1]-1 do sumarr=sumarr+arr[i]
  return,sumarr
end

3-5.pro
set_plot, 'ps'
device, filename='3-5absorption.ps'
tau0=10^(indgen(61)/10.-2.) ; 10^-2 to 10^4, 0.1 steps in the log
eqw=tau0
; same size array
for i=0,60 do begin
  int=profile(a,tau0[i],u)
  reldepth=(int[0]-int)/int[0]
  eqw[i]=total(reldepth)*0.4
endfor
cgplot,tau0,eqw,xtitle='tau0',ytitle='equivalent width',/xlog,/ylog

device,/close
filenmame='3-5absorption.ps'
```