Nikhil Patten
23 September 2022
Dr. Moe
ASTR5420

95%

Please show all work. If you collaborate with other students, write their names at the top of your homework. Please also write the total number of hours you spent on this. If you use any code or plotting routines, print those out and attach them to your hand-written solutions.

1. **(20%) Molecular and Dust Extinction and Reddening.** +16/20

   (a) **The intrinsic color of our Sun is $(B-V)_0 = 0.65$ mag, where the B and V central wavelengths are 4450 and 5510 Å, respectively. Compute the reddening $E(B-V)$ and apparent color $(B-V) = (B-V)_0 + E(B-V)$ of the sun near sunset assuming $A_V = 3$ mag of visual extinction and that molecules in our atmosphere attenuate light via Rayleigh scattering. Keep in mind that orange K5V stars have $(B-V)_0 = 1.1$ mag and red M5V stars have $(B-V)_0 = 1.8$ mag. Compute the relative visibility $R_V = A_V/E(B-V)$ of Rayleigh scattering.**

$$A_B = A_V \left(\frac{\lambda_B}{\lambda_V}\right)^{-\alpha}$$
$$\alpha = 1.8$$

where did this come from?

$$\lambda_B = 4450$$
$$\lambda_V = 5510$$
$$A_B = 3\left(\frac{4450}{5510}\right)^{-1.8} \text{ mag}$$
$$A_B = 4.41 \text{ mag}$$
$$E(B-V) = A_B - A_V$$
$$E(B-V) = 4.41 \text{ mag} - 3 \text{ mag}$$

Rayleigh has cross-section $\sigma \propto \lambda^{-4}$

$$\boxed{E(B-V) = 1.41 \text{ mag}}$$

$$(B-V) = (B-V)_0 + E(B-V)$$
$$(B-V)_0 = 0.65 \text{ mag}$$
$$(B-V) = 0.65 \text{ mag} + 1.41 \text{ mag}$$

$$\boxed{(B-V) = 2.06 \text{ mag}}$$

$$R_V = \frac{A_V}{E(B-V)}$$
$$R_V = \frac{3}{1.41}$$

$$\boxed{R_V = 2.13}$$

Code:

Code:

```
In [28]: R_VMW = 3.1
         # NP Milky Way relative visibility
         alphamw = -1*np.log((1 +R_VMW)/R_VMW)/np.log(l_B/l_V)
         # NP Calculating wavelength dependence for Milky Way relative
         # NP visibility
         print('alpha_MW: ' +format(alphamw, '.2E'))

alpha_MW: 1.31E+00
```

$$\boxed{\alpha_{MW} = 1.31}$$

correct, but explain significance (Mie scattering)

25/25

2. **(25%) Estimate how long it takes photons produced in the core of the Sun to radiate beyond the photosphere. To compute this, separate the Sun's internal structure into three parts:**

   (a) **The Sun's radiative core within $R < 0.25R_\odot$ is completely ionized ($T = 10^7$ K) and has a mean density of $\rho = 100\text{g cm}^{-3}$. Calculate the mean free path l_{ph} of a photon, the number $N = (d/l_{ph})^2$ of steps the photons take to travel a linear distance d according to a random walk, and then the timescale $\tau_{rad} = Nl_{ph}/c$ it takes the photons to radiate through that distance.**

$$l_{ph} = \frac{1}{\kappa\rho}$$

At large T, ρ:

$$\kappa_T = 0.2\,(1+X)\ \text{cm}^2\ \text{g}^{-1}$$

Calculate mean free path.

$$X \sim 0.5$$
$$\kappa_T = 0.2\,(1+0.5)\ \text{cm}^2\ \text{g}^{-1}$$
$$\kappa_T = 0.3\ \text{cm}^2\ \text{g}^{-1}$$
$$l_{ph} = \frac{1}{100\ \text{g cm}^{-3}\,(0.3\ \text{cm}^2\ \text{g}^{-1})}$$

$$\boxed{l_{ph} = 3.33 \times 10^{-2}\ \text{cm}}$$

Calculate number of steps.

$$N = \left(\frac{d}{l_{ph}}\right)^2$$
$$N = \left(\frac{R_f - R_i}{l_{ph}}\right)^2$$
$$N = \left(\frac{0.25\,(6.957\times 10^{10}) - 0}{3.33\times 10^{-2}}\right)^2$$

$$\boxed{N = 2.72 \times 10^{23}}$$

Calculate time scale.

$$\tau_{core} = N\frac{l_{ph}}{c}$$
$$\tau_{core} = (2.72\times 10^{23})\,\frac{3.33\times 10^{-2}}{3\times 10^{10}}$$

Code:

Calculate time scale.

$$\tau_{mid} = N \frac{l_{ph}}{c}$$
$$\tau_{mid} = \left(9.80 \times 10^{20}\right) \frac{1}{3 \times 10^{10}}$$

Code:

```
In [5]: print('Middle:')
        # NP Printing label for this section
        K_Tm = 1
        # NP Opacity in Sun, middle
        rho_m = 1
        # NP Density in Sun, middle
        l_m = 1 /(K_Tm *rho_m)
        # NP Mean free path in Sun, middle
        print('l_ph: ' +format(l_m, '.2E') +' cm')
        # NP Printing result
        d_m = (0.7 -0.25) *(R_o *1e2)
        # NP Calculating distance traveled by photon in cm
        N_m = (d_m/l_m) **2
        # Calculating number of steps taken by photon
        print('Steps taken: ' +format(N_m, '.2E'))
        # NP Displaying result
        t_m = N_m *(l_m /(c *10**2))
        # NP Calculating timescale
        print('Timescale: ' +format(t_m/(np.pi *10**7), '.2E') \
            +' years')
        # NP Printing result

        Middle:
        l_ph: 1.00E+00 cm
        Steps taken: 9.80E+20
        Timescale: 1.04E+03 years
```

$$\boxed{1040 \text{ years}}$$

(c) **The outer layer of the Sun across $R = 0.70 - 1.00R_\odot$ is fully convective. In class, we showed that the timescale for eddies to cross a fully convective star is $\tau_{conv} = (M_* R_*^2 / L_*)^{1/3}$. For an outer convective envelope, $\tau_{conv} = (M_{env} R_* R_{env} / L_*)^{1/3}$ gives a better approximation, where $M_{env} = 0.02M_\odot$ and $R_{env} = 0.3R_\odot$ is the mass and thickness, respectively, of the Sun's convective envelope. Compute τ_{conv}.**

$$\tau_{conv} = \left(\frac{M_{env} R_* R_{env}}{L_*}\right)^{1/3}$$
$$\tau_{conv} = \left(\frac{(0.02M_\odot)\, R_* L_* \,(0.3R_\odot)}{L_\odot}\right)^{1/3}$$

Code:

Plug in to previous equation for radiative timescale.

$$\tau_{rad} = (R_f - R_i)^2 \frac{\kappa}{c}\left(\frac{dM}{4\pi r^2 dr}\right)$$
$$\tau_{rad} = (R_f - R_i)^2 \frac{dM\kappa}{4\pi r^2 c dr}$$
$$\tau_{rad} = (1R_\odot - 0.7R_\odot)\frac{0.02M_\odot\left(103\text{ cm}^2\text{ g}^{-1}\right)}{4\pi\left(0.85R_\odot\right)^2\left(3\times10^{10}\text{ cm s}^{-1}\right)}$$

Code:

```
In [427]: t_rad = (1-0.7) *R_o *1e2 *0.02 *M_o *1e3 *103 /(4 \
              *np.pi *(0.85 *R_o *1e2)**2 *c *1e2)
          # NP Calculating radiative timescale
          print('Timescale: ' +format(t_rad/(np.pi*10**7), '.2E')\
              +' years')
          # NP Printing result

          Timescale: 2.08E+03 years
```

$$\boxed{\tau_{rad} = 2080\text{ years}}$$

As expected, $\tau_{rad} >> \tau_{conv}$. In fact, τ_{rad} is roughly 10^5 times larger than τ_{conv}.

+20/20

3. **(20%) One application of the linear Eddington approximation is the Eddington-Barbier relation where $F \propto T^4 \propto \tau$. Specifically, $F(\tau) \propto \left[T(\tau)/T_{eff}\right]^4 = 3/4(\tau + 2/3)$, which is valid only below the photosphere ($\tau > 2/3$) where we can assume LTE. For example, at a layer corresponding to an optical depth $\tau = 10$, the flux is 8 times higher and the temperature is $8^{1/4} \approx 1.7$ times hotter than at the photosphere. However, the probability that the photons from that depth escape from the Sun unimpeded is only $e^{-10} \approx 5\times10^{-5}$.**

   (a) **What fraction of the escaping stellar flux is emitted from below the photosphere? (Hint: integrate $F(\tau)e^{-\tau}$ and think about your limits of integration.)**

$$\frac{F_{\tau>2/3}}{F_{tot}} = \int_{\tau_1}^{\tau_2} F(\tau)\, e^{-\tau} d\tau$$
$$\frac{F_{\tau>2/3}}{F_{tot}} = \int_{2/3}^{\infty} \frac{3}{4}\left(\tau + \frac{2}{3}\right) e^{-\tau} d\tau$$
$$\frac{F_{\tau>2/3}}{F_{tot}} = \frac{3}{4}\int_{2/3}^{\infty} \tau e^{-\tau} + \frac{2}{3}e^{-\tau} d\tau$$

the escaping stellar flux from this optically thin layer above the photosphere.

$$F(\tau) = C\tau^3$$
$$1 = C(2/3)^3$$
$$1 = C\frac{8}{27}$$
$$C = \frac{27}{8}$$
$$\frac{F_{\tau<2/3}}{F_{tot}} = \int_{\tau_1}^{\tau_2} F(\tau) e^{-\tau} d\tau$$
$$\frac{F_{\tau<2/3}}{F_{tot}} = \int_0^{2/3} \frac{27}{8}\tau^3 e^{-\tau} d\tau$$

From integral calculator:

$$\int \tau^3 e^{-\tau} d\tau = \left(-\tau^3 - 3\tau^2 - 6\tau - 6\right) e^{-\tau}$$

Plug back in.

$$\frac{F_{\tau<2/3}}{F_{tot}} = \frac{27}{8}\left[\left(-\tau^3 - 3\tau^2 - 6\tau - 6\right) e^{-\tau}\Big|_0^{2/3}\right]$$
$$\frac{F_{\tau<2/3}}{F_{tot}} = \frac{27}{8}\left[(6)e^0 - \left(\left(\frac{2}{3}\right)^3 + 3\left(\frac{2}{3}\right)^2 + 6\left(\frac{2}{3}\right) + 6\right) e^{-2/3}\right]$$
$$\frac{F_{\tau<2/3}}{F_{tot}} = \frac{27}{8}\left[6 - \left(\frac{8}{27} + \frac{4}{3} + 4 + 6\right) e^{-2/3}\right]$$
$$\frac{F_{\tau<2/3}}{F_{tot}} = \frac{27}{8}\left[6 - \left(\frac{8}{27} + \frac{36}{27} + \frac{108}{27} + \frac{162}{27}\right) e^{-2/3}\right]$$
$$\frac{F_{\tau<2/3}}{F_{tot}} = \frac{27}{8}\left[6 - \frac{314}{27}e^{-2/3}\right]$$

Code:

```
In [447]: f2 = (27/8) *(6 -(314 *np.exp(-2/3)/27))
          # NP Calculating ratio of flux above photosphere
          print('Ratio of flux emitted below photosphere: '\
              +format(f2, '.2E'))
          # NP Printing result

          Ratio of flux emitted below photosphere: 9.84E-02
```

$$\boxed{\frac{F_{\tau<2/3}}{F_{tot}} = 0.0984}$$

(c) **Do your answers from (a) and (b) sum to approximately the total stellar flux?**

$$F_{total} = F_{\tau>2/3} + F_{\tau<2/3}$$
$$F_{total} = 0.898F_{tot} + 0.0984F_{tot}$$

Code:

5. **(25%) Download the OPAL opacities table for solar abundances from WyoCourses. The 2D text table provides Rosseland mean opacities $\log \kappa_R$ (cm^2 g^{-1}) in rows of $\log T$ (K) and columns of $\log R = \log(\rho/T_6^3)$, where ρ is in units of g cm^{-3} and T_6 is in units of 10^6 K.**

   (a) **For each grid point, determine the corresponding ρ given T and R.**
   Reading in table.

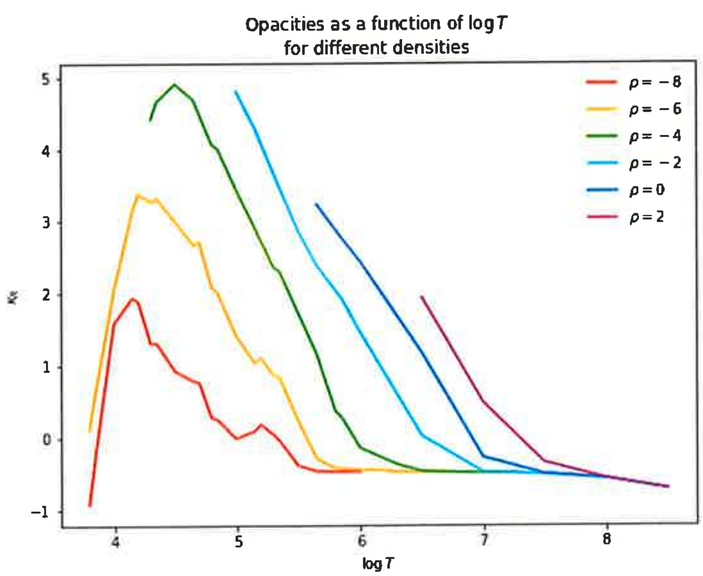
```
In [449]: opa = pd.read_csv('/d/users/nikhil/Downloads/OPAL_SolarComposition.txt')
          # NP Reading in raw text file
          goodopa = opa.to_numpy()
          # NP Converting to numpy array
          betteropa = [np.fromstring(goodopa[i][0], dtype = float, sep = ' ').tolist()\
              for i in range(len(goodopa))]
          # NP Creating a table for opacities and temps., skipping first row
          bestopa = [betteropa[i+1][1:] for i in range(len(betteropa) -1)]
          # NP Creating a list of only opacities
          row_lengths = []
          for row in bestopa:
              row_lengths.append(len(row))
          max_length = max(row_lengths)
          for row in bestopa:
              while len(row) < max_length:
                  row.append(None)
          pristineopa = np.array(bestopa)
          # NP Creating numpy array of opacities to better parse through array
          logTs = np.array([betteropa[i+1][0] for i in range(len(betteropa)-1)])
          # NP Creating array of temperatures
          Tgrid = np.array([[t for r in range(len(logRs))] for t in logTs])
          # NP Creating 2D numpy array of temperatures
          logRs = np.array(np.fromstring(goodopa[0][0][6:], dtype = float, sep = ' '))
          # NP Creating array of R's
          Rgrid = np.array([[r for r in logRs] for t in range(len(logTs))])
          # NP Creating 2D numpy array of temperatures

          /tmp/ipykernel_315015/361661287.py:5: DeprecationWarning: string or file could
          ed data; this will raise a ValueError in the future.
            betteropa = [np.fromstring(goodopa[i][0], dtype = float, sep = ' ').tolist()
```

   Creating densities

```
In [451]: rhos = np.array([np.array([np.round(r +3 *(t -6) ,3) for r in logRs])\
              for t in logTs])
```

   (b) **Plot $\log \kappa_R$ as a function of $\log T$ for $\log \rho = -8, -6, -4, -2, 0,$ and 2.**

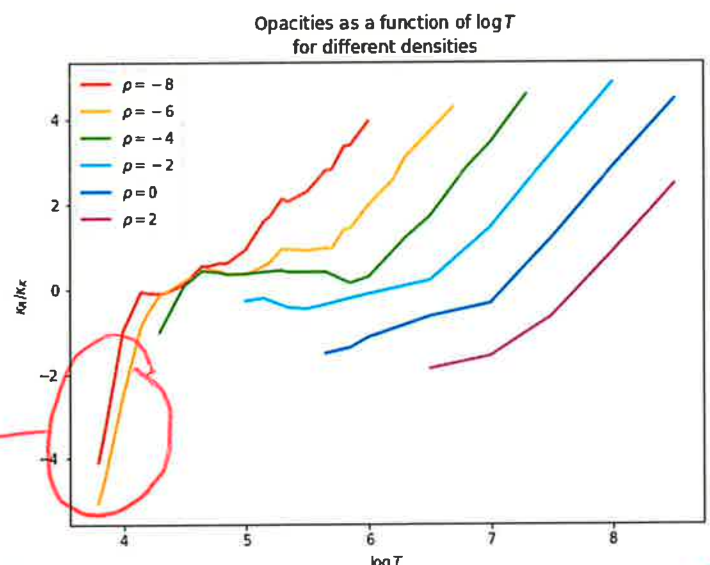


✓ looks good!

   Code:

```
In [458]: plt.figure(figsize = [8, 6])
          # NP Making figure larger
          plt.plot(Tgrid[ineight], pristineopa[ineight]-kgrid[ineight],\
              'r', label = r'$\rho=-8$')
          plt.plot(Tgrid[insix], pristineopa[insix]-kgrid[insix],\
              color = 'orange', label = r'$\rho=-6$')
          plt.plot(Tgrid[infour], pristineopa[infour]-kgrid[infour],\
              'g', label = r'$\rho=-4$')
          plt.plot(Tgrid[intwo], pristineopa[intwo]-kgrid[intwo],\
              color = 'cyan', label = r'$\rho=-2$')
          plt.plot(Tgrid[izero], pristineopa[izero]-kgrid[izero],\
              color = 'blue', label = r'$\rho=0$')
          plt.plot(Tgrid[itwo], pristineopa[itwo]-kgrid[itwo], color\
              = 'purple', label = r'$\rho=2$')
          # NP Plotting difference in log of opacities for each density
          plt.xlabel(r'$\logT$')
          plt.ylabel(r'$\kappa_R/\kappa_K$')
          # NP Labeling axes
          plt.legend()
          # NP Creating legend
          plt.title(r'Opacities as a function of $\logT$' +'\n'
              'for different densities')
          # NP Creating title for plot
          plt.savefig('/d/www/nikhil/public_html/ASTR5420/images/opacity'
              'densitydeviation.png')
          # NP Saving figure
```

Plot:



near photosphere

(small T & ρ) → bound-bound dominate (e.g. Balmer was strongest at T = 10,000 K)

Generally, extreme deviations from Kramer's opacity law occurs at larger temperatures and densities. This can be explained because Kramer's Law assumes the dominant sources of opacity are bound-free and free-free radiation. At larger temperatures and densities, like in the center of a star, opacity is due to Thompson scattering and not these processes which are assumed to be dominant in Kramer's law. Therefore, at higher temperatures and densities, there will be more of a deviation between opacities calculated by Kramer's law and actual opacities.