Due Monday, January 18 24:00

There are three options for the final project

1. You solve the numerical problems in this document, which is similar to an extended homework with a unifying theme. You will use all the tools we have learnt in class, both in terms of numerical methods and coding principles.

2. You propose your own project. I actually encourage this as long as you use a variety of methods from the class and the difficulty level is similar to this one. It is better that you advance your own research (especially for graduate students) than complete a task which is mostly pedagogical.[1] **You need my approval for this option, send me a paragraph summarizing your plan before December 21st**.

3. You re-solve problem sets 7 and 8 (all questions, including the bonus ones) using a programming language other than Python. I would advice you to choose this path only if you already know the language to some level, especially for languages quite different from Python such as C/C++. Two possibilities that come to my mind are C/C++ and Julia. C or C++ is typically the fastest language for intensive numerical computation. Julia is much younger than Python, but it is similar, hence easy to learn. It claims to be even more versatile and faster (sometimes on par with C). Julia has very nice differential equations suites, but in general number of available modules is nowhere near Python. You can propose another language, but you need my prior approval. If you choose this option, you should discuss the advantages/disadvantages of the language to Python in terms of ease of coding and testing, and speed.

   Do not underestimate how much trouble you might have in a language you are just learning. Plotting in C/C++ is a pain, so you will be allowed to save data, and then read and plot it using Python. Whatever language you choose, i should be able to get all your results with a few lines of terminal commands, e.g. you should write a Makefile if you choose C/C++, not ask me to install some IDE.

- Your project report should be a cohesive document where you discuss the actual physical and mathematical results and also the details of the coding such as unit testing, convergence etc. One way to do this is giving the main results in a Physical Review-style manuscript, and leaving thedetails of the computation and coding aspects in *supplemental materials*. This particular style is encouraged, but is not mandatory. You can find a LaTeX template for this style in the project folder, LaTeX is encouraged whatever structure you want to have with your project. All this said, it is better to write a report in a way you are comfortable with than trying a new format which would be hard to follow for the reader.

- In the following, you are usually given numerical problems without specific methods to follow, in a similar fashion to the ungraded problem sets. I encourage you to use builtin functions in the SciPy modules whenever appropriate, unless there is an instruction not to. If you feel you need to use a package not available through `conda`, ask for prior permission from me.

- There are no specific instructions for convergence tests, unit tests, or how to write your code, but you are supposed to do these for all your numerical computations. In essence, do the tests that would convince a knowledgable person who is suspicious of your results. You should document all your tests as well. You are allowed to submit your code as a single Jupyter notebook instead of .py files if you choose to. The notebook should follow the basic coding principles and be well-documented as usual.

- Use version control and remote repositories for code safety and accessibility. You will open a github account, and start a repository specifically for this project. You will invite me as a collaborator. I will check whether you actually used version control throughout or just pushed all the documents to your

[1] Though, I think this project is pretty rich and fun as well.

repository at once in the end. Do use your repository, and make a habit of it for all your projects. You will most likely thank me one day.[2]

Stars from Newton to Einstein (and maybe beyond)

Your mission, should you choose to accept it, is calculating the structures of various types of stars in Newtonian gravity, general relativity (GR) and alternative theories of gravity which try to surpass GR. You do not need any familiarity with GR to complete your task as we will give you the equations to work with.

Newton

Let us start with the concept of *hydrostatic equilibrium* of stars in Newtonian gravity. If you consider a layer of thickness dr at radius r in a spherically symmetric star, the weight of the material in the shell from r to $r + dr$ would give the change in pressure between these radii, and you would obtain the following system of ODEs

$$\begin{aligned} \frac{dm(r)}{dr} &= 4\pi r^2 \rho(r) \\ \frac{dp(r)}{dr} &= -\frac{Gm(r)\rho(r)}{r^2} \ . \end{aligned} \tag{1}$$

where $p(r)$ is the pressure, $m(r)$ is the mass of the star within r, $\rho(r)$ is the density, and the minus sign signifies the fact that the pressure increases at smaller radii.

Note that this ODE system cannot be integrated in this form since we have no equation for the density. This is resolved by remembering that we can relate P and ρ by using the equation of state (EOS) for the stellar matter. For example, for an ideal gas

$$PV = Nk_BT \implies P = \frac{k_B}{\mu m_{\rm H}} T\rho \tag{2}$$

where T is the temperature, $m_{\rm H}$ is the mass of the hydrogen atom and μ is the average molecular weight. It is clear that the relationship between pressure and density depends on other quantities which should have their own differential and algebraic equations, but we will ignore these and assume a *polytropic* EOS

$$P = K\rho^{\gamma} = K\rho^{1+\frac{1}{n}} \tag{3}$$

where n is known as the *polytropic index*. This is a simplification, but not an oversimplification, since many stars can be described this way to reveal nontrivial physical phenomena.

(a) Show that the two equations of stellar structure together with Eq. 3 can be recast as the famous *Lane-Emden equation*[3]

$$\frac{1}{\xi^2}\frac{d}{d\xi}\left(\xi^2 \frac{d\theta}{d\xi}\right) + \theta^n = 0 \tag{4}$$

with $\theta(0) = 1$, where ξ is a scaled radius and θ is a simple function of ρ, or equivalently p. Using Methematica, show that the regular solutions at the center behave as

$$\theta(\xi) = 1 - \frac{1}{6}\xi^2 + \frac{n}{120}\xi^4 + \dots \ , \tag{5}$$

[2] We have not covered these topics as of the preparation time of this document, but we will.

[3] This document is self-contained, i.e. you can complete the project by simply following the information within. However, if you want to read further or get some tips, *Stellar Structure and Evolution* of Kippenhahn, Weigert and Weiss (KWW) is a commonly used textbook, albeit being quite advanced compared to our level of discussion in many respects.

hence, the other initial condition is $\theta'(0) = 0$. Solve this equation for $n = 1$ analytically using Mathematica.

If you did the change of variables correctly, you should see that $\theta = 0 \iff \rho = 0$, hence the surface of the star is at ξ_n such that $\theta(\xi_n) = 0$. Show that the total mass of the star is given by

$$M = 4\pi\rho_c R^3 \left(\frac{-\theta'(\xi_n)}{\xi_n}\right) \ . \tag{6}$$

Lastly, show that if a group of stars share the same polytropic EOS, then their mass is a function of their radius

$$M \propto R^{\frac{3-n}{1-n}} \ , \tag{7}$$

and find the constant of proportionality.

White dwarfs (WDs) are the end stages of relatively low-mass stars when thermonuclear energy production stops. For example our Sun will run out of fuel and become a WD in ~ 6 billion years. WDs are extremely dense, about a solar mass squeezed into the volume of the Earth. Hence, their pressure is dominated by a quantum mechanical effect named *electron degeneracy*. This means the EOS for cold WDs is given by

$$P = C\left[x(2x^2-3)(x^2+1)^{1/2} + 3\sinh^{-1} x\right] \quad , \quad x = \left(\frac{\rho}{D}\right)^{1/q} \ . \tag{8}$$

You cannot use the Lane-Emden equation any more, but you can still go back to the original hydrostatic equilibrium equations, Eq. 1, to obtain the star structure. The logic is the same, you start with the initial values $\rho(0) = \rho_c$ and $m(0) = 0$, and integrate out using Eq. 8. Integration stops when $\rho = 0$, at which point you have the radius and the mass of the star, so you know the mass of the star with this particular radius. You can reverse this procedure of calculating the mass as a function of the radius ($M(R)$) for a given EOS, and obtain the unknown parameters C, D and q using nonlinear curve fitting to $M - R$ data.

The above task is conceptually easy, but computationally very demanding. Obtaining only one pair of (M, R) for given values of (C, D, q) is already very costly, since it requires solving an IVP. If you are trying to find (C, D, q) by nonlinear fitting to $M - R$ data in the most straightforward manner, things get even worse. You need to obtain mass as a function of the radius, $M(R)$, which is only *implicitly* known. This is because you cannot control R directly, but rather obtain both R and M at the end of the integration of the IVP for a choice of ρ_c, which is the only freely specifiable value. This means that both radius and mass can be considered functions of ρ_c: $R(\rho_c)$ and $M(\rho_c)$. For any data point $(\tilde{M}, \tilde{R})$, you need to solve a root finding problem for ρ_c with $R(\tilde{\rho}_c) = \tilde{R}$, then calculate the error $M(\tilde{\rho}_c) - \tilde{M}$ for your trial values of (C, D, q). Overall, for a single data point $(\tilde{M}, \tilde{R})$, you need to solve a root finding problem, $R(\rho_c) = \tilde{R}$, for a function whose single evaluation requires the solution of an IVP. Then, you repeat this for all the data points, possibly thousands of them, to obtain the total error. This is only the beginning point of your fit, since you need to repeat the whole procedure for various values of (C, D, q), and determine which one minimizes the total error! As you might guess, we will not take this path, but I want you to understand the steps above so that you can appreciate how hard it can be to interpret raw data, which often requires many more than three parameters.

(b) You can find observational data from many low-temperature WDs in the file `white_dwarf_data.csv`. [4] Two quantities are listed for each WD: mass (in solar masses $M_\odot$), and base-10 logarithm of the surface gravity in CGS units ($\log(g)$). The latter can be easily converted to radius using basic

[4] This data is taken from Montreal White Dwarf Database where I chose relatively cold stars, and eliminated some outliers due to their spectroscopic properties. MWDD contains many more WDs for which many more measurements are listed. Such databases are increasingly indispensable in astrophysics where categorization of large numbers of objects is a major undertaking. If you are interested, feel free to obtain other star samples from MWDD depending on your own criteria, and try to see if you can replicate the theoretical predictions about WDs. For example, you can look up the effect of temperature on the EOS, which we will not study, and see if you can quantify it using the data.

Newtonian gravity. Write a function to read this data, and show all the points in an M vs R plot using solar masses and average Earth radii as units. Note that this is a `.csv` file, so reading it with Python might be a bit different from reading a plain ASCII file which was the case for the Hubble data in the problem sets.

(c) To make curve fitting easier, we will use the fact that for low mass WDs $x \ll 1$. Then, we can retain only the leading term in the series expansion of Eq. 8, which gives

$$P = K_* \rho^{1+\frac{1}{n_*}} \ . \tag{9}$$

Use Mathematica to obtain the series expansion, and show that

$$n_* = q/(5-q) \ \ , \ \ K_* = \frac{8C}{5D^{5/q}} \ . \tag{10}$$

This is wonderful news, we can now use Eq. 7, which requires a much easier fit. Find n_* and K_* using an appropriate fit to the WD data in `white_dwarf_data.csv`. Using n_* and the Lane-Emden equation, find the central density ρ_c of the WDs you used in the fit, and plot them with respect to M.

- Remember that Eq. 9 only holds for low mass stars. It is up to you to decide how low is low. Note that lower the mass more accurate Eq. 9 is, but too few points would also make a bad fit. It should be relatively easy to see when the $M(R)$ curve is deviating from the low-mass behavior if you plot the correct quantities, hence the upper limit to the mass for this fit. *Hint: What is a good plot to reveal power-law dependence?*
- **Assume that q is an integer**, which comes from theory. If you use a builtin fitting function (which is our suggestion), it is likely that you cannot constrain a fitting parameter to be an integer. In that case, you can first fit for a real q, and then set your q to be the nearest integer (should be quite near in this case). Then, you do a second fit for the other parameters, in our case the single parameter K_*, by using the integer value you already determined. *Hint:* $q = 3$.
- You should be careful about numerically solving the Lane-Emden equation, or similarly the hydrostatic equilibrium equations near the origin. For example in Eq. 1 you have $m(r)/r^2$, which gives $0/0^2$ for your first time step at the origin when you integrate out. However, note that this is not a physical singularity. Near the origin, the mass will grow with the cube of the radius as long as the central density is not 0, so $m(\epsilon)/\epsilon^2 \sim \epsilon$ which actually converges to 0 rather than diverge. The simplest solution to this is having a conditional to check if $r = 0$, and setting the RHS directly to 0 instead of using the formula. Similar situations occur in many ODEs, and you usually hande them likewise.

(d) The low mass fit gave us q directly, and we know $C = 5K_* D^{5/q}/8$. This means we now have a single unknown parameter, D, for the general EOS in Eq.8. This is a considerable simplification over the naive three-parameter fit, but remembering how much computation we need in order to calculate the total error for a single value of the fitting parameter, this might still be overwhelming. Let us simplify the problem further.

First, realize that you do not really need to apply a root finding procedure on $R(\rho_c)$ to obtain the exact R value for each data point. WD masses and radii in the data are distributed continuously, so we will need pretty much all radius values between the minimum and maximum radius.

Let's concentrate on what we will do for a single guess value of the fitting parameter D. Instead of wasting computation time for root finding for each exact R value in the data, we will find the WD masses for a randomly chosen sample of radius values that cover the whole radius range in the data. More concretely, you start with an initial educated guess value of central WD density ρ_c, and find the R and M values for this particular star. Then you slightly change ρ_c to bigger or smaller values to obtain other stars with different R and M values, so that the radii of the stars you computed cover all the radius range in the data more or less evenly, let us say you have 20 sample (M, R) points for this particular D. How do you calculate what your current choice of D implies for the mass of a star with radius $\tilde{R}$ from your data set? You interpolate from the 20 sample points that cover the whole radius range of the data. You can actually use the same sample points to calculate $\tilde{M}(\tilde{R})$ for *all* WDs in your data set!

Note that the root finding problem is gone, and even further, we can calculate the total error for a given D using only 20 IVP solutions, instead of doing the same for each WD in our data set. The gain is potentially limitless, we could use the same 20 points to interpolate for as many WD measurements as we want, size of the data set does not matter. Of course this assumes that interpolation is computationally a lot cheaper than solving an IVP, which is obvious in our case. To complete the procedure, we repeat it for different values of D, and try to find D that minimizes the error.

- I gave the number 20 as an example, it is up to you how many sample points to use as long as they seem to represent the interpolated function accurately. Because the data is smooth without sudden changes, as few as 5 might be fine, but better be safe than sorry. In general, if the function to be fitted has more features, you would need more points for an accurate interpolation.
- Note that you should be careful about how you do the nonlinear fitting, so that for each D value you calculate one sample set of (M, R), and solely use interpolation afterwards. If you are not careful, you may end up recalculating the same sample points (solving IVPs) for each data point! If you choose to use builtin functions, be careful about how the they handle the fitting function you provide, e.g. `f` in `scipy.optimize.curve_fit(f, ...)`. For example, if `curve_fit` gives all data points as a vector to `f`, then you can easily implement it so that a single set of sample points is used to interpolate all elements of the vector. If you are not careful and `f` maps a double to a double, then depending on the inner mechanism of `curve_fit`, you may end up applying `f` separately to each element in the data vector, ending up recalculating the same sample points for each data point, slowing down the process by a factor of the number of data points.
- What kind of interpolation should you choose? It is up to you, but we learnt about a go-to interpolation method ideal for interpolating from **many** known points, and provides a relatively smooth interpolation for the whole range. I would personally use that one. *Hint: It starts with an "s."*
- What values of D should you search around? You have the clues that for low mass stars ρ/D is small, but it rises up to unity for heavier ones so that you have to use Eq. 8. This, and the previous part about low-mass stars should give you a rough idea. Similarly, you can get an idea from the low-mass case to choose a ρ_c value from which you start to build your sample to be used in interpolation.

Compare your results to the theoretical values (see Ch. 15 and 19 of KWW)

$$C = \frac{m_e^4 c^5}{24\pi^2\hbar^3} \quad , \quad D = \frac{m_u m_e^3 c^3 \mu_e}{3\pi^2\hbar^3} \tag{11}$$

where $\mu_e = 2$ is the number of nucleons per electron, and m_u is the atomic mass unit.

(e) Now that you have C, D and $q = 3$, plot the full M vs R curve using numerical solutions. You just have to sweep over various ρ_c values to obtain (M, R) by solving IVPs. You should see that there is a maximum possible mass M_{Ch} above which there is no WD solution. This is the famous *Chandrasekhar mass.*

Another way to see the Chandrasekhar mass is looking at the $x \gg 1$ limit of Eq. 8. Obtain this expansion using Mathematica, and show that the EOS becomes a polytrope with $n = 3$. Using the mass formula, Eq. 7, for polytropes and the exact theoretical values in Eq. 11, express M_{Ch} in terms of known constants. Compare this to the result from your $M(R)$ curve.

Existence of an upper mass limit begs a simple question: what happens if we add more mass to a WD at this limit? The solutions we computed are those in equilibrium, so since no more mass is supported, the WD simply goes out of equilibrium and starts to decrease its radius, i.e. collapse. Electron degeneracy cannot stop such a collapse, so this instability would continue until some new physical mechanism kicks in. This collapse sometimes can trigger a chain of other events which lead to the violent explosion and destruction of WDs known as *Type Ia supernova* explosions. Because such explosions happen at theoretically known systems (maximally massive WDs), one can actually calculate the energy output with the help of some observations, and then use this to measure the distance to such events. Because of the huge amount of energy output, such events can be observed billions of light years away. Combining all these ideas, some smart cosmologists can calculate the expansion rate of our universe, and ultimately conclude that it is accelerating.

Einstein

We mentioned that relatively low-mass main sequence stars end up as WDs. Heavier stars explode at the end of their lifetime in supernova explosions of various kinds (different from Type Ia), but the mass of their remaining cores can still exceed $M_{\rm Ch}$. This means such a core continues to collapse, and becomes so much squeezed that electrons and protons are forced to merge to form neutrons. If the mass is not too high, the degeneracy pressure of neutrons can overcome the collapse, and we obtain a stable star formed of mostly neutrons: a *neutron star (NS)*. These objects also contain a few solar masses, but within a radius of only $10-20$ km. If you check your table of physical constants, you can see that such an object has the density of the atomic nucleus. The gravity of these objects are so immense that Newton's theory breaks down, and we need to use Einstein's general relativity.

One can naively expect that a NS behaves just like a WD, we simply replace electrons with neutrons. This is known not to be the case, the chief reason being that even though a non-interacting electron model works very well for WDs, neutrons in NSs are strongly interacting. Actually, the interactions are so complicated that we do not currently know the NS equation of state, and hope to learn more about it using astrophysical observations such as gravitational wave detections. We will use a polytope with $n=1$ for simplicity

$$P = K_{\rm NS}\rho^2 \ , \tag{12}$$

which is "realistic" in the sense that it reproduces the qualitative bahavior of NSs.

As usual, we will scale our lengths, times and masses in the computations so that the quantities we calculate have order-of unity magnitudes. NS masses are similar to solar masses, so we will use $M_\odot$ as our mass unit, which is equivalent to scaling all our masses by solar mass: $m \to m/M_\odot$, $M_\odot = 1.989\times10^{30}$ kg.[5] We will use $\frac{GM_\odot}{c^2} \approx 1477$ m as our length unit, and $\frac{GM_\odot}{c^3} = 4.927\times10^{-6}$ s as our time unit. These are also equivalent to scaling length and time with the respective quantities. You might wonder why we chose such strange length and time units. What we described here is tantamount to setting the fundamental constants in GR to unity, $c=G=1$, which is known as using the geometric units. Go ahead and calculate the values of these constants in the given units if you do not believe me. This information is adequate for understanding which units to use in your code, but you can find more details in the appendix if you want to.

In the above units, $K_{\rm NS}\approx 100$. Remember that these units are designed such that the values of physical quantities in this problem will be around unity. For example, in a NS, a solar mass is confined into several Schwarzschild radii, $R \sim 10\times\frac{GM}{c^2}$, so the average density is around $\rho \sim M/R^3 \sim 10^{-3}M^{-2} \sim 10^{-3}$, since $M\sim1$ in units of solar masses. In SI units, the nuclear density is $\rho\sim10^{17}$ kg/m^3. Even though we use the above units in the numerical computations for simplicity, use the following units when plotting your results:

- km for length
- ms for time

[5] We changed the names of our variables when we scaled them before, e.g. $r\to\rho$ in the Lgrange point problem, but here we will use the same symbol, e.g. the scaled mass is still m. This should not be confusing since we will always use the same units, equivalently the same scaled variables, in our calculations.

- $M_\odot$ for mass
- kg/m^3 for density

(a) Hydrostatic equilibrium equations are modified due to general relativistic effects, and are called Tolman-Oppenheimer-Volkoff (TOV) equations:

$$m' = 4\pi r^2 \rho \tag{13}$$

$$\nu' = 2\frac{m + 4\pi r^3 p}{r(r-2m)} \tag{14}$$

$$p' = -\frac{m + 4\pi r^3 p}{r(r-2m)}(\rho + p) = -\frac{1}{2}(\rho + p)\nu' \tag{15}$$

where $'$ is the derivative w.r.t r. The first equation is already what you had in the Newtonian case.[6] The third equation is a modified version of the pressure equation in Newtonian gravity, and reduces to it in the limit $\frac{p}{\rho} \to 0$ and $\frac{m}{r} \to 0$.[7] The first of these limits is natural since the energy in GR contains the rest mass energy $E = mc^2$, which is orders of magnitude greater than any classical pressure. The second limit roughly says that the escape velocity from the surface of the star is much smaller than the speed of light, which should also hold for Newtonian gravity to be valid.

What is the middle equation, and the quantity ν? $e^{\nu(r)/2}$ is basically the time dilation factor of relativity. Just as inertial frames with relative velocity to us have a different flow of time as measured by us in special relativity by a factor of $\gamma = (1 - v^2)^{-1/2}$, positions with different strengths of gravity also have different flows of time in each other's view. $e^{\nu(r)/2}$ gives how slowly time passes at a point as observed by someone infinitely far away from the NS. Once again, you do not need these details to solve the problem, but perhaps you will be better motivated.

Obtain and plot the M vs R curve for NSs by integrating the TOV equations from the center out. As before, you should stop once you hit $\rho = 0 \iff p = 0$. The initial conditions are $m(0) = 0$, and $p(0) = p_c$. By our definition above, $e^{\nu(0)}$ is the time dilation factor due to gravity at the center of the star, compared to an observer at $r \to \infty$. So we should start with $\nu(0)$ that gives us $\nu(\infty) = 0$, so that there is no time dilation at infinity. This cannot be known directly in advance, but it does not matter for now either. Note that adding a constant to ν does not effect the solution of $m(r), p(r)$ in Eq. 13, i.e. the solutions are invariant for $\nu \to \nu + \text{const}$. Hence, we will simply use $\nu(0) = 0$ without loss of generality. We will later discuss how to find the correct constant term.

At the end of the integration you will obtain $M(p_c)$ and $R(p_c)$, and you can obtain the whole $M - R$ curve by changing p_c (you do not need to target a specific M or R value to obtain the curve).

(b) The $m(r)$ function you calculated above is not actually the sum of the rest masses of all the particles in the NS. Gravitational potential energy is negative in this system (since it is bound), and remember that energy is mass in relativity. So $m(r)$ in fact contains the rest mass of the neutrons and the negative contribution from the binding energy. The rest mass by itself, called the baryonic mass, is given by the equation

$$m_P' = 4\pi \left(1 - \frac{2m}{r}\right)^{-1/2} r^2 \rho \ . \tag{16}$$

So we can define the fractional binding energy as

$$\Delta \equiv \frac{M_P - M}{M} \tag{17}$$

[6] The meaning of $m(r)$ in TOV equations is actually different from what we have in Newtonian physics as we will discuss shortly.

[7] Note that these ratios are dimensionless in our choice of units/scaling.

Plot Δ vs R. You do not need a separate integration for this, you can simply add this equation to your TOV equations and obtain all quantitites at the same time.

(c) Plot the M vs ρ_c curve. A simple criterion for NS stability is

$$\frac{dM}{d\rho_c} > 0 \ \rightarrow \ \text{stable} \tag{18}$$

$$\frac{dM}{d\rho_c} < 0 \ \rightarrow \ \text{unstable} \ . \tag{19}$$

The logic for this is as follows. If we try to squeeze the star by a small amount δR, a stable star resists this process, meaning we have to do positive work. Since mass is energy, this work increases the mass of the star M. At the same time, a squeezed star also has higher density throughout, so ρ_c also increases. Overall, $\frac{dM}{d\rho_c} > 0$. The converse is disastrous: if we do negative work when we squeeze the star a bit, this means the star already wants to squeeze itself since this makes it go to a lower energy configuration. Such a star just continues squeezing until it reaches a stable configuration where it does not want to squeeze itself, or it continues to collapse indefinitely, forming a *black hole*. This is an instability. From this criterion, you should see that not all stars in your $M - R$ curve are stable. Plot the stable and unstable arms differently (e.g. one is dashed).

There is a maximal NS mass allowed by this EOS. Report this value as well.

(d) The maximal NS mass depends on the EOS, so keeping the polytropic index the same, calculate the maximally allowed masses for EOS with different values of K. Plot $M_{\text{max}}(K)$ vs K.

The most massive neutron star to be observed so far has a mass of $2.14M_\odot$. Based on this observation alone, what values of K are allowed?

(e) There is no matter outside the star, so we can easily generalize the pressure and mass values to this region as $p(r > R) = 0$ and $m(r > R) = m(R) = M$. The remaining equation becomes

$$\nu' = \frac{2M}{r(r-2M)} \quad (r > R) \ . \tag{20}$$

Using Mathematica, show that

$$\nu(r > R) = \ln\left(1 - \frac{2M}{r}\right) - \ln\left(1 - \frac{2M}{R}\right) + \nu(R) \ . \tag{21}$$

Beyond Einstein (Bonus +5 points)

First, let me give some short background about the equations we are going to solve. In principle, you can skip to Eq. 26 where the actual task starts, but yo would not get much of the physics.

Recall that we can shift ν by a constant without affecting the physical quantities. The usual procedure is that, we use the time as it is observed by an observer infinitely far from the NS to be the time coordinate we use. So, we want $e^{\bar{\nu}(\infty)} = 1$, i.e. there is no time dilation for someone at infinity. This can be easily done by defining

$$\bar{\nu}(r) = \nu(r) + \ln\left(1 - \frac{2M}{R}\right) - \nu(R) \tag{22}$$

which has the behavior

$$\bar{\nu}(r > R) = \left(1 - \frac{2M}{r}\right) \ . \tag{23}$$

satisfying $e^{\tilde{\nu}(\infty)} = 1$. We denoted this newly shifted function with a tilde to avoid confusion, and will use it from this point on.[8]

The curvature of spacetime is governed by an object called the *metric*: g_{ab}. For our purposes, it is a 4×4 matrix whose components depend on r, as

$$g_{ab} = \begin{pmatrix} -e^{\tilde{\nu}(r)} & 0 & 0 & 0 \\ 0 & \left(1 - \frac{2m(r)}{r}\right)^{-1} & 0 & 0 \\ 0 & 0 & r^2 & 0 \\ 0 & 0 & 0 & r^2 \sin^2\theta \end{pmatrix} . \tag{24}$$

The convention in gravitational physics is $a, b = 0, 1, 2, 3$, which agrees with Python array conventions.

This metric should not be very alien, the last three diagonal terms are the coefficients we have when calculating the Euclidean length in spherical polar coordinates in the $\frac{m(r)}{r} \to 0$ limit. The first term is similar to the $-c^2t^2$ in the *invariant interval* of special relativity ($c = 1$ here). Indeed, g_{ab} is the generalization of the spacetime metric that calculates the invariant interval in special relativity.

Gravitational physicists make an abuse of notation, and use the same symbol with upper indices as the inverse of the above matrix, called the *inverse metric*:

$$g^{ab} = \begin{pmatrix} -e^{-\tilde{\nu}(r)} & 0 & 0 & 0 \\ 0 & 1 - \frac{2m(r)}{r} & 0 & 0 \\ 0 & 0 & r^{-2} & 0 \\ 0 & 0 & 0 & (r\sin\theta)^{-2} \end{pmatrix} , \tag{25}$$

We will use these in the coming part of the project.

A scalar field, more properly a massless real Klein-Gordon field, obeys the wave equation in flat spacetime

$$\Box\phi = 0 \implies \left(-\frac{\partial^2}{\partial t^2} + \nabla^2\right)\phi = 0 \tag{26}$$

(remember that we use $c = 1$). The wave operator $\Box$ is modified in curved spacetime, and becomes

$$\Box\phi = 0 \implies \sum_{j,k=0}^{3} \frac{1}{\sqrt{|g|}} \frac{\partial}{\partial x^j}\left(g^{jk}\sqrt{|g|}\frac{\partial\phi}{\partial x^k}\right) = 0 . \tag{27}$$

In spherical coordinates we will use, $x^0 = t$, $x^1 = r$, $x^2 = \theta$, $x^3 = \phi$, and g is the determinant of the *metric*. If we are interested in the spacetime of a NS, our metric is the one in Eq. 24, and

$$g = -\left(1 - \frac{2m(r)}{r}\right)^{-1} e^{\tilde{\nu}} r^4 \sin^2\theta . \tag{28}$$

We will assume spherical symmetry for the scalar field as well, which means ϕ is only a function of t and r. After all dust settles

$$\Box\phi = -\frac{1}{\sqrt{|g(r)|}}\frac{\partial}{\partial t}\left[e^{-\tilde{\nu}(r)}\sqrt{|g(r)|}\,\frac{\partial\phi}{\partial t}\right] + \frac{1}{\sqrt{|g(r)|}}\frac{\partial}{\partial r}\left[\left(1 - \frac{2m(r)}{r}\right)\sqrt{|g(r)|}\,\frac{\partial\phi}{\partial r}\right] = 0 \tag{29}$$

The common approach is reducing this equation to first order as

$$\partial_t\Phi = \quad \partial_r\left(f(r)\Pi\right) \tag{30}$$

$$\partial_t\Pi = \frac{1}{r^2}\partial_r\left(r^2 f(r)\Phi\right) \tag{31}$$

[8] Note that time is relative, but we usually observe NSs from a far distance, and use our time as the standard to define time at the center of the star. Setting $\nu(0) = 0$ was the converse, we used the measurements of an observer at the center of the NS as our time coordinate, according to whom our clocks at infinity tick differently, i.e. $\nu(\infty) \neq 0$. We now switch to using time as it is observed far away by a simple scaling of the time, which is equivalent to a shift in the exponential factor.

where we defined

$$\Phi \equiv \partial_r \phi \tag{32}$$

$$\Pi \equiv \frac{1}{f(r)} \partial_t \phi \tag{33}$$

$$f(r) \equiv e^{\tilde{\nu}/2} \left(1 - \frac{2m(r)}{r}\right)^{1/2} \tag{34}$$

We will first see how scalar waves behave under GR, i.e. solve Eqs. 30, 31, and then compare this to the predictions of an alternative theory of gravity.

(a) Argue that the boundary conditions are

$$\Phi(0) = \partial_r \Pi(0) = 0 \tag{35}$$

$$\lim_{r\to\infty} \Phi \sim \frac{1}{r} \tag{36}$$

$$\lim_{r\to\infty} \Pi \sim \frac{1}{r} \ . \tag{37}$$

We have an infinite spatial domain. It is up to you how to handle it, but if you choose to impose boundary conditions at a large but finite $r = r_{\text{out}}$, argue that

$$\Phi(r_{\text{out}}) + \Pi(r_{\text{out}}) = 0 \ . \tag{38}$$

(b) Let us see how scalar waves propagate in spacetimes of NSs governed by the EOS in Eq. 12. Start with the initial data

$$\phi(0, r) = 10^{-3} e^{-r^2/(2a^2)} \tag{39}$$

$$\dot{\phi}(0, r) = 0, \tag{40}$$

where $a = 1$ in our code units (i.e. $a = \frac{GM_\odot}{c^2}$ in SI units). Evolve $\phi(t, r)$ for 8 different NS masses varying from $M = M_\odot$ to the maximal NS mass. This means you solve for, and store $m(r), \nu(r), p(r)$ for each of these stars (including the region outside the star), and then use them in Eq. 29.

Evolve the scalar for a duration you deem to be long enough to see its qualitative behavior. Plot and save the time evolution on surfaces.

(c) In a certain class of alternative theories of gravity named scalar-tensor theories, the equation of motion for the scalar is modified as

$$\Box\phi = 4\pi\beta e^{2\beta\phi^2} (\rho - 3p)\phi \ . \tag{41}$$

Add the new term to Eq. 29, and repeat your time evolutions from the last part for $\beta = -6$. Describe how the bahavior changes compared to GR. I will spoil the surprise: for some of the cases, and not necessarily all, you should see $\phi(t, r)$ grow and reach a more or less stable configuration not changing in time, in stark difference to what you had before.

This phenomenon is called *spontaneous scalarization*, and basically says that even though $\phi = 0$ is a stable solution in GR, it can become unstable around NSs in other gravity theories. This instability causes the scalar to grow to large values, but other mechanisms stop this growth eventually so that we have a stable solution with a large-amplitude ϕ field. Such scalar fields are relatively easy observables, and they are interesting targets for NS observations to see if there are signs of modifications to GR.

(d) The results of the previous problem suggest that there is a time independent solution to the modified wave equation in the scalar-tensor theory we investigated. Such solutions can be found as the result of a Boundary value problem. This is easy to see: if there is no time dependence, we are only left with the r derivatives in Eq. 41, hence, we have a second order ODE where the function is specified at $r = 0$ and $r = \infty$: a boundary value problem.

Solve this BVP using any method you like for cases where you observed spontaneous scalarization in the previous part. It is sufficient to find the "lowest" mode. Do your solutions agree with the final stage of the growing scalar field in time evolution?

(e) A boundary value problem has infinitely many solutions, so find a second solution for each of the BVPs you solved. Take them as your initial value, and observe their time evolution. Do these solutions stay the same like the stable "lowest" mode solutions you had? Half your step size for the BVP and the evolution (both in space and time), and repeat your numerical calculation for one of the cases. Repeat once more by having the step size again. What do you see in the time evolution of your BVP solutions? How do you interpret these results?

Appendix: Units

The SI units commonly used in daily life and engineering are designed for historic reasons or human practical concerns. Because of this, you can see that fundamentally important constants such as c or $\hbar$ have extremely large or small values in these units. When one is dealing with fundamental phenomena it is convenient to use units where these constants have order-of-unity values, even better if they are exactly 1. There are three base quantities: length, time and mass. There are also three fundamental constants: c, G and $\hbar$. One can find measurement units for the base quantities so that in those units $c = G = \hbar = 1$. These are called Planck units. You can show that they are

$$\ell_{\mathrm{P}} = \left(\frac{\hbar G}{c^3}\right)^{1/2} \tag{42}$$

$$t_{\mathrm{P}} = \left(\frac{\hbar G}{c^5}\right)^{1/2} = \frac{\ell_{\mathrm{P}}}{c} \tag{43}$$

$$m_{\mathrm{P}} = \left(\frac{\hbar c}{G}\right)^{1/2} = \frac{\hbar}{c\ell_{\mathrm{P}}} \ . \tag{44}$$

From a physical perspective, these are expected to be the relevant scales of length, time and mass when relativity, gravity and quantum mechanics come together, e.g. in quantum gravity.

From a practical perspective, we do not need to mention units explicitly any more, setting $c = G = \hbar = 1$ already fixes the three units we use. Moreover, we can get rid of all factors of $c, G, \hbar$ in our formulas, since they are 1. In an alternative view, mass in these units is the count of how many Planck masses we have in an object, a dimensionless number, and the case is similar for length and time. This is similar to how we obtained dimensionless quantities in the problem sets by scaling them by the natural unit scales. Omitting $c, G, \hbar$ in formulas because they are simply 1 in Planck units is equivalent to scaling all the masses with the Planck mass, lengths with Planck length and times with Planck time. One difference from our scaled quantities in the problem sets is that we still use the same symbol for a quantity rather than define a new one, i.e. when we have a mass m, what we mean is actually $\frac{m}{m_{\mathrm{P}}}$. For example Einstein's famous relationship for the equivalence of mass and energy becomes

$$E = mc^2 \ \rightarrow \ E = m \ , \tag{45}$$

or the energy of a photon with frequency ω becomes

$$E = \hbar\omega \quad \rightarrow \ E = \omega \ , \tag{46}$$

or the Compton wavelength of a particle of mass m is given by

$$\lambda_{\mathrm{C}} = \frac{2\pi\hbar}{mc} \quad \rightarrow \ \lambda_{\mathrm{C}} = \frac{2\pi}{m} \ . \tag{47}$$

Going back to the SI units is equivalent to setting $c, G, \hbar$ to their values in the SI units. For this, we should recover the $c, G, \hbar$ factors we omitted in the formulas when we assumed they were all unity. This is easily done by multiplying any formula by correct powers of $c, G, \hbar$. Take the Compton wavelength formula without $c, G, \hbar$ as an example, which seems to suggest that a wavelength, something of length, is proportional to something of inverse mass. Note that this is normal in Planck units, since we have already fixed all base units, meaning mass is nothing but how many Planck masses we have in a particle. To make the units meet again, we should multiply the right hand side with something with units $\frac{\text{mass}}{\text{time}}$. Than what exponents do we need so that $c^\alpha G^\beta \hbar^\gamma$ has units of $\frac{\text{mass}}{\text{time}}$? This problem always has a unique solution of α, β, γ, which is $\alpha = -1$, $\beta = 0$, $\gamma = 1$ for our case.

In many cases, we do not want to set all three fundamental units to 1, but only some of them. If we are dealing with special relativity, it is very convenient to set $c = 1$ to avoid factors of c everywhere, but $\hbar, G$ do not play any role in this theory, so they are left as usual. This is equivalent to measuring length and time with the same units, since $c = 1$ is $\frac{\text{length}}{\text{time}}$. In particle physics, quantum mechanics also plays a role so $c = \hbar = 1$, but there is no decision about G. This is the reason you hear statements like "mass of the Higgs boson is 125 GeV'.' Remember that GeV is a measurement unit for energy, not mass. What is meant here is that you have already set $c = \hbar = 1$, and the mass is 125 GeV/c^2.

Similarly, we deal with relativity and gravity in GR, so we set $c = G = 1$, but do not make any claims about $\hbar$. This is the geometric unit system. Because we did not fix all three constants, we still have units unlike the Planck system, i.e. there is still one scale we can freely choose for one of the units. We can pick any of them, but because GR is intimately related to length, expressing all quantities in powers of length is common. For example, since $c = 1$, time has the same units of length, $[T] = [L]$. To convert mass into length, you need to multiply by $G/c^2 = 1$, so mass has the same units as length as well $[M] = [L]$, and the same is true for the energy since $E = mc^2$ and c is unitless since $[c] = [L]/[T] = [L]/[L] = 1$. Density is $\rho = M/L^3$ so $[\rho] = [L]/[L]^3 = [L]^{-2}$, so not everything is length, but some quantities are other powers of length. You might see expressions like the Schwarzschild radius being $R = 2M$, or the density of a neutron star behaving as $\rho \sim M^{-2}$. These might look alien to you if you have not seen this approach before, but you get used to them pretty quickly (and develop a distaste for SI units). Such unit systems are the norm in theoretical studies of fundamental physics, and they are very commonly used in publications as well.