

Introduction

Throughout this project I code in python, making use of the SymPy package for symbolic mathematics. We also use the GraviPy package, built on top of SymPy, giving us data structures to store and manipulate the kinds of tensors that appear in general relativity. We'll only need it to compute the Christoffel symbols. SciPy will be used to perform numerical integration.

Question 1

We use the equivalent action

$$\mathcal{S} = \int \mathcal{L} d\tau$$

$$\mathcal{L} = g_{ij} \dot{x}^i \dot{x}^j = g_{tt} \dot{t}^2 + 2g_{t\phi} \dot{t} \dot{\phi} + g_{\phi\phi} \dot{\phi}^2 + g_{rr} \dot{r}^2 + g_{\theta\theta} \dot{\theta}^2$$

where the dot denotes differentiation with respect to τ . The conserved quantities come from the lack of dependence of \mathcal{L} on t, ϕ .

No t dependence gives

$$\frac{\partial \mathcal{L}}{\partial t} = 2g_{tt} \dot{t} + 2g_{t\phi} \dot{\phi} = 2E \quad (1)$$

for E constant, and similarly no ϕ dependence gives

$$\frac{\partial \mathcal{L}}{\partial \phi} = 2g_{t\phi} \dot{t} + 2g_{\phi\phi} \dot{\phi} = -2L_z \quad (2)$$

for L_z constant. We get a further conserved quantity from no τ dependence

$$\mathcal{L} - \dot{x}^i \frac{\partial \mathcal{L}}{\partial \dot{x}^i} = -\mathcal{L} = -1 \quad (3)$$

where $\mathcal{L} = 1$ is by timelikeness ¹

To determine the effective potential we sub (1) and (2) into (3). (1) and (2) give a system of 2 equations for $\dot{t}, \dot{\phi}$ which can be solved to give

$$\dot{t} = \frac{Eg_{\phi\phi} + L_z g_{t\phi}}{g_{tt}g_{\phi\phi} - g_{t\phi}^2} \quad (4)$$

$$\dot{\phi} = -\frac{Eg_{t\phi} + L_z g_{tt}}{g_{tt}g_{\phi\phi} - g_{t\phi}^2} \quad (5)$$

and thus (3) becomes (after some algebra)

$$g_{rr} \dot{r}^2 + g_{\theta\theta} \dot{\theta}^2 = -V_{eff}(r, \theta, E, L_z) \quad (6)$$

$$V_{eff}(r, \theta, E, L_z) = -1 + \frac{E^2 g_{\phi\phi} + L_z^2 g_{tt} + 2EL_z g_{t\phi}}{g_{tt}g_{\phi\phi} - g_{t\phi}^2}$$

Question 2

Computing the Christoffel symbols and returning latex is pretty straightforward using GraviPy, and was done using *compute_kerr_christoffel.py*. The non zero Christoffel symbols (up to symmetry) in the lower indices are displayed below. We make some attempt to simplify the expressions through substituting in Σ and Δ , but there may exist simpler forms our program had missed. This won't matter for later computation. Here's what we get:

¹this is because of our signature, in the course this would be -1

$$\begin{aligned}
\Gamma_{tr}^t &= -\frac{m(a^2 + r^2)(a^2 \cos^2(\theta) - r^2)}{\Delta \Sigma^2} \\
\Gamma_{t\theta}^t &= -\frac{a^2 m r \sin(2\theta)}{\Sigma^2} \\
\Gamma_{r\phi}^t &= -\frac{am(2\Sigma r^2 + a^4 \sin^2(\theta) - a^4 + a^2 r^2 \sin^2(\theta) + r^4) \sin^2(\theta)}{\Delta \Sigma^2} \\
\Gamma_{\theta\phi}^t &= -\frac{2amr(\Sigma a^2 + \Sigma r^2 - a^4 + 2a^2 m r \sin^2(\theta) - 2a^2 r^2 - r^4) \sin(\theta) \cos(\theta)}{\Delta \Sigma^2} \\
\Gamma_{tt}^r &= \frac{\Delta m(-a^2 \cos^2(\theta) + r^2)}{\Sigma^3} \\
\Gamma_{t\phi}^r &= \frac{\Delta am(a^2 \cos^2(\theta) - r^2) \sin^2(\theta)}{\Sigma^3} \\
\Gamma_{rr}^r &= \frac{r}{\Sigma} + \frac{m}{\Delta} - \frac{r}{\Delta} \\
\Gamma_{r\theta}^r &= -\frac{a^2 \sin(2\theta)}{2\Sigma} \\
\Gamma_{\theta\theta}^r &= -\frac{\Delta r}{\Sigma} \\
\Gamma_{\phi\phi}^r &= \frac{\Delta(-8\Sigma^2 r + a^4 m(\cos(4\theta) - 1) + 8a^2 m r^2 \sin^2(\theta)) \sin^2(\theta)}{8\Sigma^3} \\
\Gamma_{tt}^\theta &= -\frac{a^2 m r \sin(2\theta)}{\Sigma^3} \\
\Gamma_{t\phi}^\theta &= \frac{amr(a^2 + r^2) \sin(2\theta)}{\Sigma^3} \\
\Gamma_{rr}^\theta &= \frac{a^2 \sin(2\theta)}{2\Delta \Sigma} \\
\Gamma_{r\theta}^\theta &= \frac{r}{\Sigma} \\
\Gamma_{\theta\theta}^\theta &= -\frac{a^2 \sin(2\theta)}{2\Sigma} \\
\Gamma_{\phi\phi}^\theta &= -\frac{(\Sigma(\Sigma a^2 + \Sigma r^2 + 2a^2 m r \sin^2(\theta)) + 2a^2 m r(a^2 + r^2) \sin^2(\theta)) \sin(\theta) \cos(\theta)}{\Sigma^3} \\
\Gamma_{tr}^\phi &= -\frac{am(a^2 \cos^2(\theta) - r^2)}{\Delta \Sigma^2} \\
\Gamma_{t\theta}^\phi &= -\frac{2amr}{\Sigma^2 \tan(\theta)} \\
\Gamma_{r\phi}^\phi &= \frac{\Sigma^2 r - 2\Sigma m r^2 - \frac{a^4 m(\cos(4\theta) - 1)}{8} - a^2 m r^2 \sin^2(\theta)}{\Delta \Sigma^2} \\
\Gamma_{\theta\phi}^\phi &= \frac{2a^2 m^2 r^2(a^2 + r^2) \sin(\theta) \sin(2\theta) + (\Sigma - 2mr)(\Sigma(\Sigma a^2 + \Sigma r^2 + 2a^2 m r \sin^2(\theta)) + 2a^2 m r(a^2 + r^2) \sin^2(\theta)) \cos(\theta)}{\Delta \Sigma^3 \sin(\theta)}
\end{aligned}$$

Question 3

Using a section of *q3.schwarzschild.py*, just as above, we calculate the non zero Christoffel symbols, finding

$$\begin{aligned}
\Gamma_{tr}^t &= \frac{m}{r(-2m+r)} \\
\Gamma_{tt}^r &= \frac{m(-2m+r)}{r^3} \\
\Gamma_{rr}^r &= \frac{m}{r(2m-r)} \\
\Gamma_{\theta\theta}^r &= 2m-r \\
\Gamma_{\phi\phi}^r &= (2m-r)\sin^2(\theta) \\
\Gamma_{r\theta}^\theta &= \frac{1}{r} \\
\Gamma_{\phi\phi}^\theta &= -\frac{\sin(2\theta)}{2} \\
\Gamma_{r\phi}^\phi &= \frac{1}{r} \\
\Gamma_{\theta\phi}^\phi &= \frac{1}{\tan(\theta)}
\end{aligned}$$

Now substituting $\theta = \pi/2$, $g_{\phi\phi} = -r^2$, $g_{tt} = (1 - 2m/r)$ and $g_{t\phi} = 0$ into (6), we get, making sure to use the correct signature:

$$V_{eff}(r, E, L_z) = -1 + \frac{E^2}{1 - 2m/r} - \frac{L_z^2}{r^2} = g_{rr}\dot{r}^2 + g_{\theta\theta}\dot{\theta}^2 \quad (7)$$

which has zeros as solutions of

$$r^3(E^2 - 1) + 2mr^2 - L_z^2 r + 2mL_z^2 = 0$$

which for $E = 0.97$, $L_z = 4$, $m = 1$ are $\approx 3.07, 7.61, 23.16$. By graphing, our potential is positive for $7.61 < r < 23.16$ or $2 < r < 3.07$ (ignoring $r < 2$ within schwarzschild radius, not physical for a timelike geodesic). Now a bound orbit in the equatorial plane must have $\dot{\theta} = 0$, and so (7) becomes

$$\dot{r}^2 = -(1 - 2/r)V_{eff}$$

Only some of these solutions will be bound... TODO

3a

Using (1), (2) and (7) we can calculate initial values for \dot{t} , $\dot{\phi}$ and $\dot{\theta}$ respectively. Calling the function `compute_one_geodesic()` in `q3_schwarzschild.py`, we produce the following plots using parameters $r = 15$, $\theta = \pi/2$, $\dot{r} = 0$, $E = 0.97$, $L = 4$. Some fiddling around with the integration parameters was required to get a smooth enough curve, over a few orbits. Looking at the ϕ plot, we have just over 3 orbits. Calculating E , L_z , and $g_{rr}\dot{r}^2 + g_{\theta\theta}\dot{\theta}^2 + V_{eff}(r, \theta, E, L_z)$ at each point we get our 3 conservation laws to high precision, assuring us the numerical integration is accurate enough for our purposes .

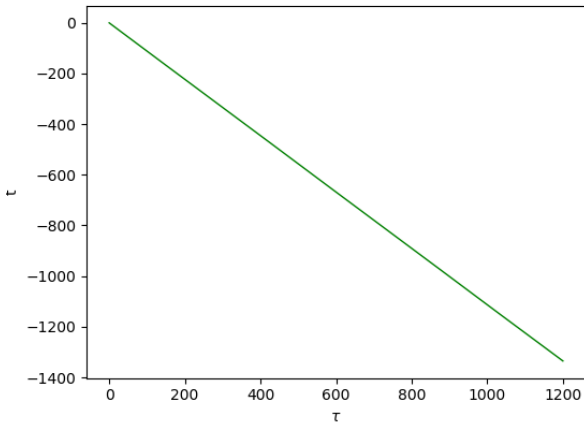


Figure 1: t

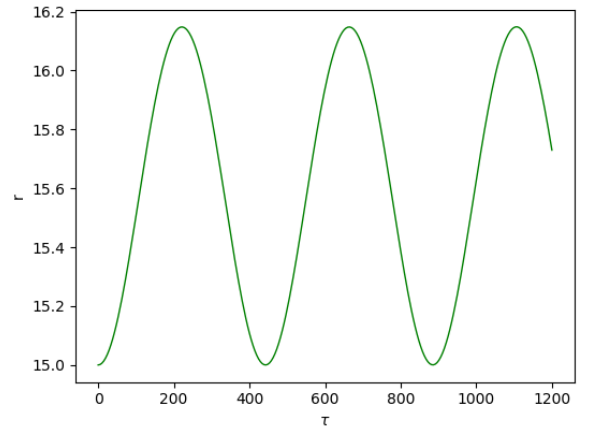


Figure 2: r

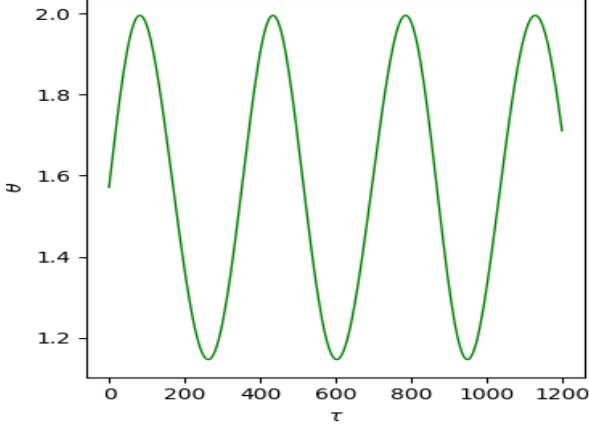


Figure 3: θ

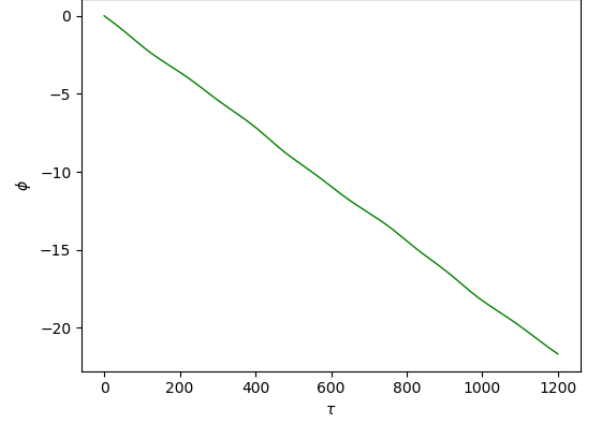


Figure 4: ϕ

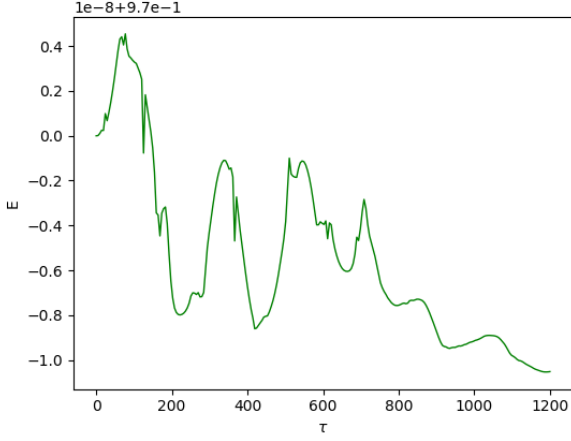


Figure 5: E

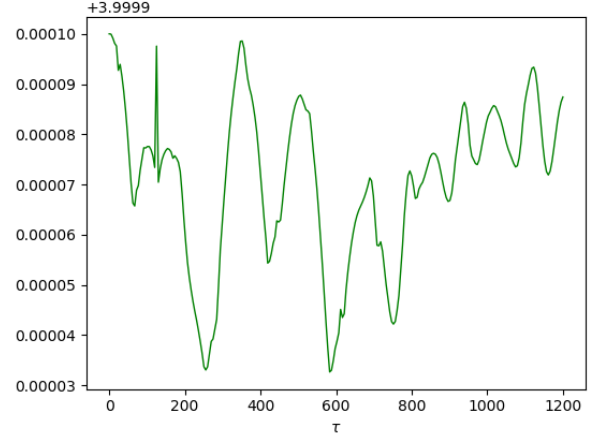


Figure 6: L

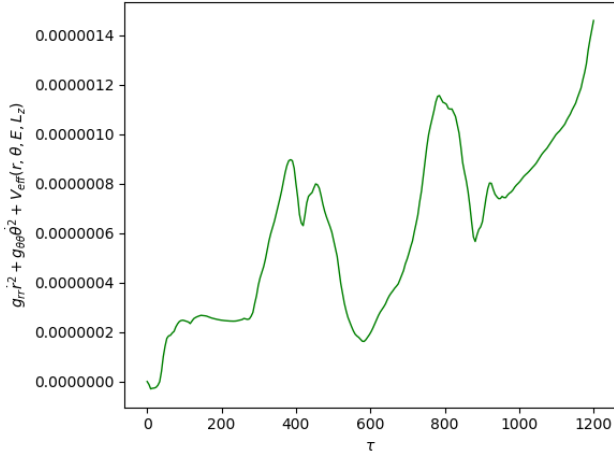


Figure 7: $g_{rr}\dot{r}^2 + g_{\theta\theta}\dot{\theta}^2 + V_{eff}(r, \theta, E, L_z)$

3b

We find as expected our code runs into issues if r is not in the range defined previously. Running the code listed under 3b in `q3.py` with several r_0 values we obtain for fixed $\theta = \pi/2, \dot{r} = 0, E = 0.97, L = 4$ the following plots. We see the values form a closed curve under these conditions, with the shape of curve depending on the initial parameters

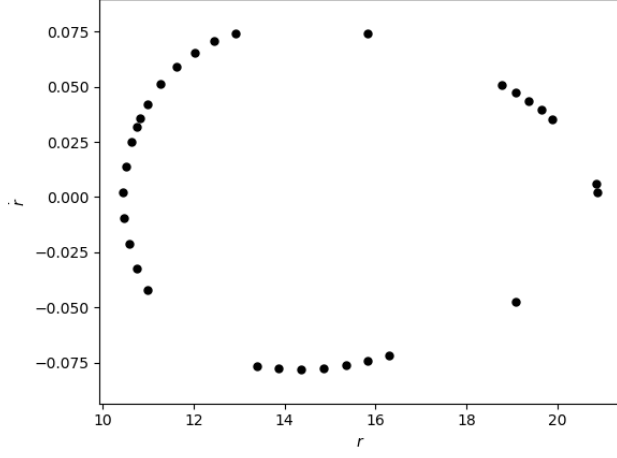


Figure 8: $r_0 = 10$

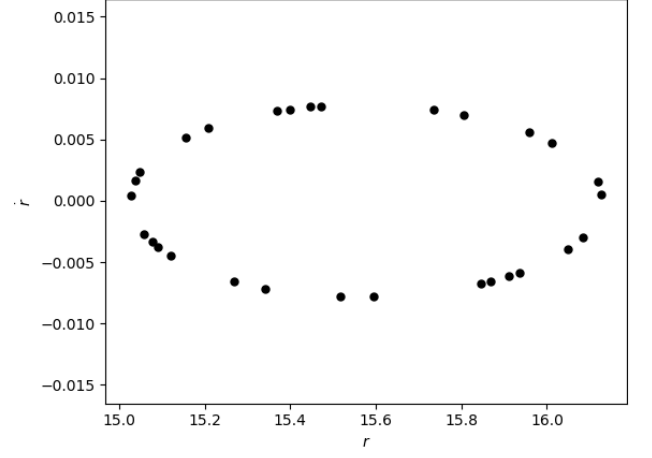


Figure 9: $r_0 = 15$

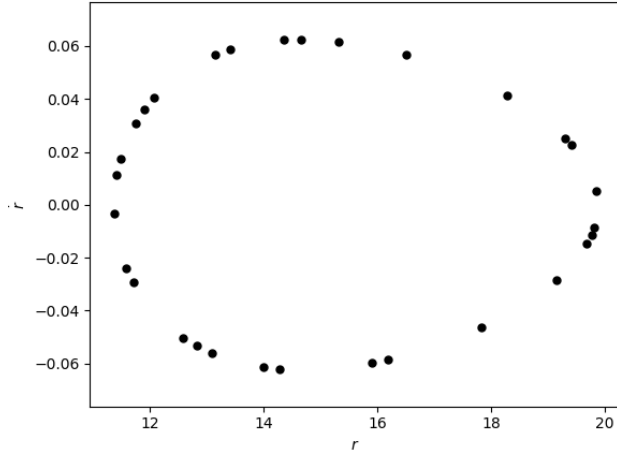


Figure 10: $r_0 = 20$

Question 4

Plotting the potential using *q4.py* and using a numerical root finder, we see positive potential for $4.513 < r < 14.564$. Now modifying *q3.py* to accomodate the Kerr metric, we produce the following Poincare maps, which all look very much like the Schwartzchild ones

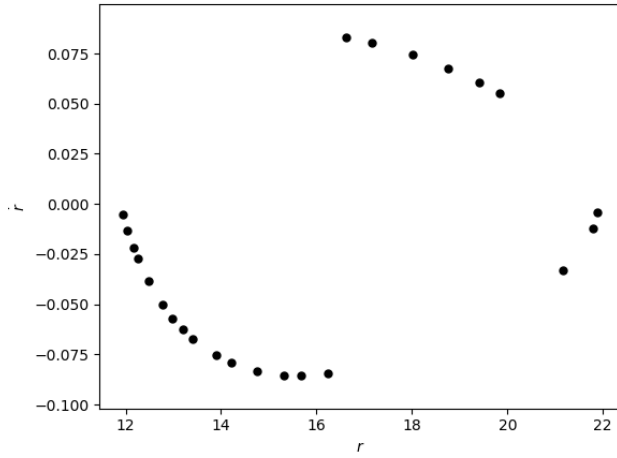


Figure 11: $r_0 = 10$

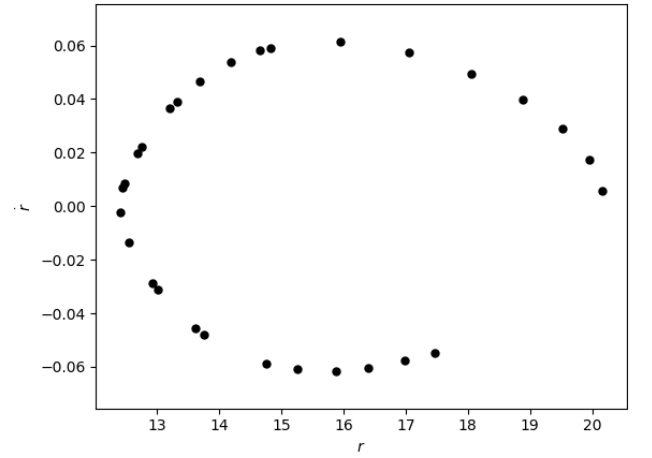


Figure 12: $r_0 = 15$

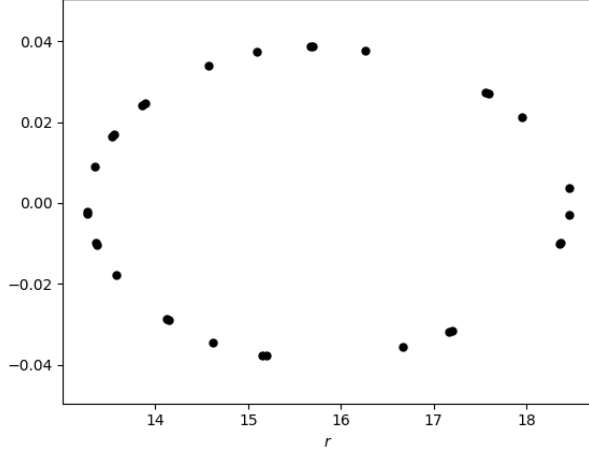


Figure 13: $r_0 = 20$

Question 5

The script *q5.py* attempts to symbolically simplify a hand computed expression for \dot{Q} by symbolically calculating and substituting in $\dot{\theta}$ in terms of $r, m, a, \theta, \dot{t}, \dot{r}$ and $\dot{\phi}$ from the geodesic equation and Christoffel symbols. I couldn't manage to get this to work.

In the $a = 0$ limit Q becomes

$$Q = L_z^2 \csc^2 \theta + r^2 \dot{\theta}^2$$

choosing $\theta(\tau) = \pi/2$, which we are always free to do, we get

$$Q = L_z^2$$

the square of the total orbital angular momentum.

$$- \frac{2 \left(-\Delta \Sigma^2 \dot{\phi}^2 a^2 - \Delta \Sigma^2 \dot{\phi}^2 r^2 + \Delta \Sigma^2 \dot{\theta}^2 a^2 + \Delta \Sigma \delta a^2 + \Delta \Sigma \dot{\phi}^2 a^4 - 2 \Delta \Sigma \dot{\phi}^2 a^2 m r \sin^2(\theta) + 2 \Delta \Sigma \dot{\phi}^2 a^2 r^2 + \Delta \Sigma \dot{\phi}^2 r^4 - \Delta \Sigma \dot{t}^2 a^2 + 2 \Delta \dot{\phi}^2 a^4 m r \sin^2(\theta) + 2 \Delta \dot{\phi}^2 a^2 m r^3 \sin^2(\theta) - 4 \Delta \dot{\phi} t a \right)}{\Delta \Sigma}$$