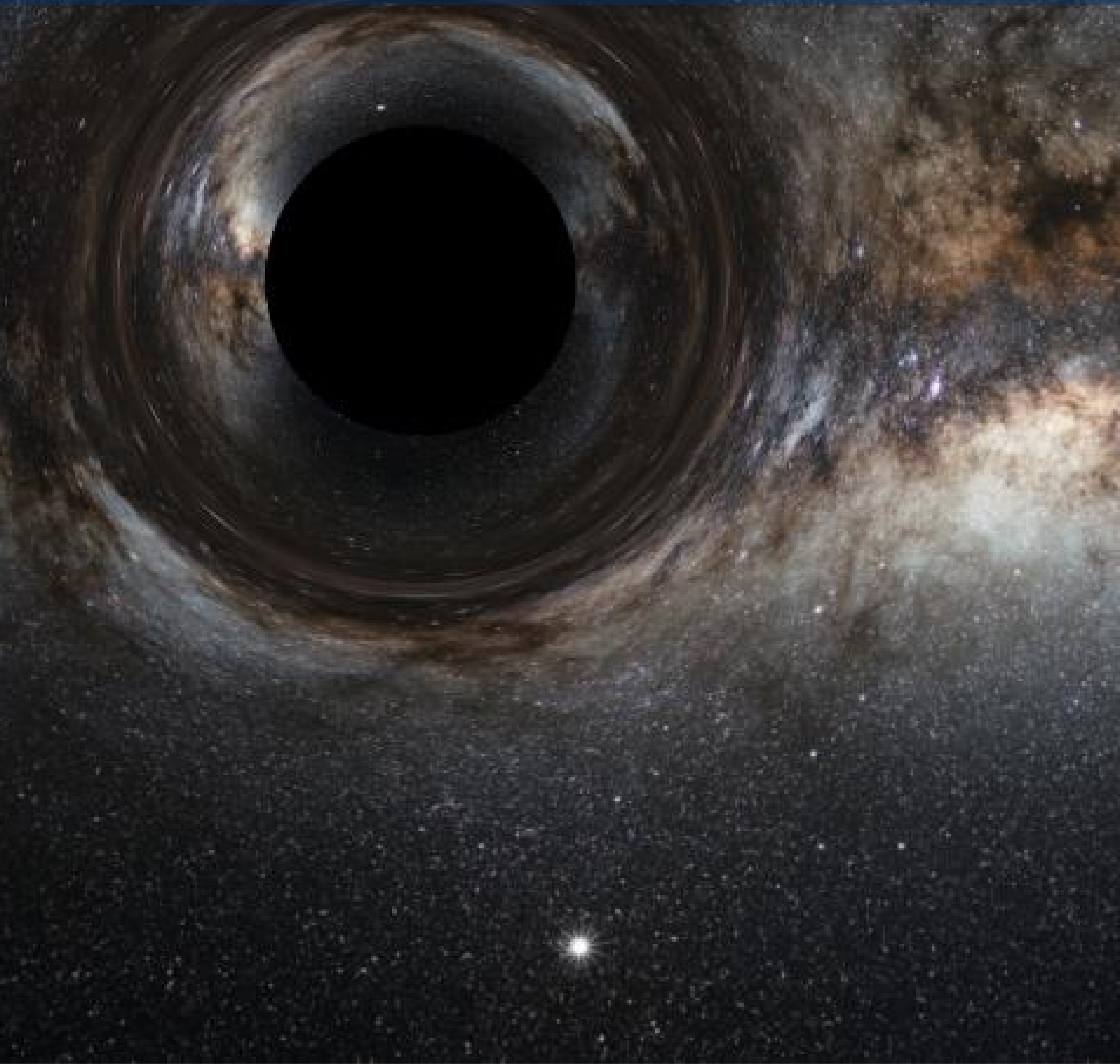
Visualising Geodesics in Schwarzschild Spacetime

Using Manim

Gayatri P



Visualising Geodesics in Schwarzschild Spacetime

Using Manim

Krittika Summer Project
6.0

Author:	Gayatri P
Supervisor:	Aditya Khambete
Second supervisor:	Anuj Nandekar
Facilitator:	Rudrya Arya

Project duration:	May 2025 – July 2025
Project Files Hosted at:	GitHub and GDrive

Cover image:	NASA
Template style:	Thesis style by Richelle F. van Capelleveen

Contents

1. Introduction

1.1 General Relativity and the Equivalence Principle

At the cornerstone of General Relativity lies the equivalence principle, which states that the effects of gravity are indistinguishable from the effects of acceleration. There's no way an observer can differentiate between being in a uniform gravitational field and being in an accelerating reference frame. In other words, with some particular coordinate transformation, the effect of free-fall vanishes.

Einstein even generalised this to include non-uniform gravitational fields, stating that there exists local inertial frames, in which the effects of any gravitational field will vanish.

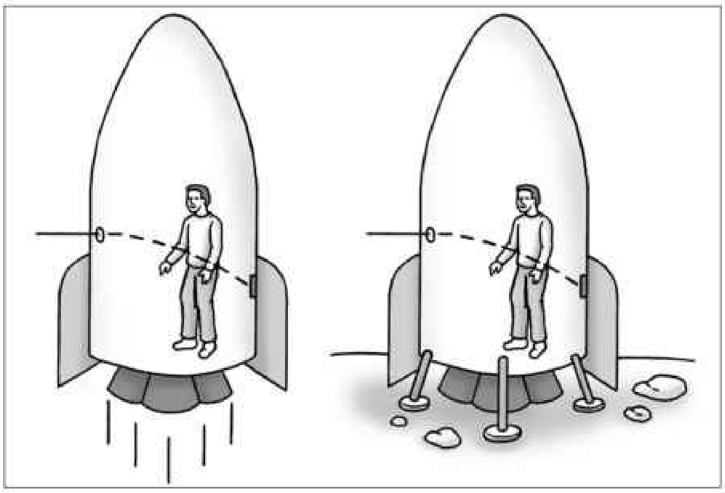


Figure 1.1: A figure depicting the path of light as observed by an outside oberver (left) and by a person on the spaceship (right).

Consider a simple scenario where a spaceship is accelerating in the z-direction with a beam of light travelling in the x-direction. While the beam of light is travelling in a stright line for an oberver outside, from inside the spaceship, the path of the light beam will look curved. This is indistinguishable from the effect of gravity on light. Then what exactly is the distinguishing factor between a clever coordinate transformation and a gravitational field? The answer lies in tidal forces. As we have discussed only the local effects till now, there is no coordinate transformation than can remove the effects of gravitational field globally.

In the following chapters, we will briefly discuss the mathematical framework of differential geometry that descibes gravitational fields. We how to describe spacetime geometry in tensor notation, having been introduced to it in Special Relativity, with metric tensors. We then discuss the Schwarzschild metric and black holes. We also attempt predict the trajectory of matter or light near the vicinity of the black hole, thereby dicovering the concept of event horizons and photospheres.

2. Mathematical Formulation

The core mathematical background of General Relativity lies in differential geometry. Here, any space is generally defined by an n-dimensional manifold that, upon close inspection of any small patch, looks indistinguishable from flat n-dimensional Euclidean space, $\mathbb{R}^n$.

In Euclidean space,

$$ds^2 = dx^{i2} \tag{2.1}$$

Einstein realized that the problem of deciding whether or not a gravitation field was real was similar to a problem that Riemann had explored, deciding whether or not a space was flat. The core premise of general relativity is that gravity is not a force in the conventional sense, but rather a manifestation of the curvature of spacetime.

For a general curved space however, the formula for the distance between two close points becomes,

$$ds^2 = g_{mn} x dx^m dx^n \tag{2.2}$$

where g_{mn} is the spacetime metric. Mathematically, it is a symmetric tensor defined at every point in spacetime.

We know that geometry associated with the Special Theory of Relativity is the Minkowskian geometry. It defines length, or proper distance (or proper time) as

$$\begin{aligned} ds^2 &= dx^2 - dt^2 \\ \text{or, } d\tau^2 &= dt^2 - dx^2 \end{aligned} \tag{2.3}$$

which is an invariant quantity across coordinate transformation. The Minkowskian metric is defined to be $g_{mn} = \text{diag}{-1, 1, 1, 1}$. The single negative sign is responsible for the unique causal structure of spacetime, dividing vectors into time-like ($ds^2 < 0$), space-like ($ds^2 > 0$), and light-like ($ds^2 = 0$).

2.1 Flatness and Curvature

A flat geometry is one where all Euclid's postulates are true. It cannot simply be characterized in terms of the metric $g_{i,j} = \delta_{i,j}$. The defining diagnostic quantity, which is zero everywhere the space is flat and non-zero otherwise, is called the Riemann curvature tensor.

For this, we will use the concept of a covariant derivative of a tensor. For a general curved spacetime, we need to take into account the change in the tensor, and the change in the coordinate system. This is given by,

$$\begin{aligned} D_r V_m =& \partial r V_m - \Gamma^t{}_{rm} V^t, \text{ covariant derivative.} \\ D_r T_{mn} =& \partial_r T_{mn} - \Gamma^t{}_{rm} T_{tn} - \Gamma^t{}_{rn} T_{mt} \end{aligned} \tag{2.4}$$

where Γ is known as the Christoffel symbol:

$$\Gamma^t{}_{mn} = \frac{1}{2}[\partial_n g_{sm} \ \ \partial_m g_{sn} - \partial_s g_{mn}] g^{st} \tag{2.5}$$

Consider a vector being transported along two paths. In a flat space two transported vectors will be identical, $D_r D_s V_m = D_s D_r V_m$, but this is not true in curved space. The Riemann Curvature Tensor (R), is hence defined using the difference in the two quantities,

$$\begin{aligned} D_s D_r V_m - D_r D_s V_m =& R_{sr}{}^t{}_m V_t \\ \text{where } R_{sr}{}^t{}_m \triangleq& \partial_r \Gamma^t{}_{sm} - \partial_s \Gamma^t{}_{rm} \ \ \Gamma^p{}_{sm} \Gamma^t{}_{pr} - \Gamma^p{}_{rm} \Gamma^t{}_{ps} \end{aligned} \tag{2.6}$$

2.1.1 Geodesics

As discussed before, a vector field V is said to be parallel transported along a curve with tangent vector if $D_p V^n = 0$. This means the vector is kept “pointing in the same direction" on the curved manifold. One can also prove that if $DV^n = 0$ along the curve, the length of V is preserved.

A geodesic is a curve that parallel transports its own tangent vector. If along the curve the derivative of the tangent vector is zero, that curve is

$$\begin{aligned} t^m \triangleq& \frac{dx^m}{ds} \\ dt^n \ \ \Gamma^n{}_{mr} t^r dx^m =& 0 \\ \frac{dt^n}{ds} =& -\Gamma^n{}_{mr} t^r \frac{dx^m}{ds} \\ =& -\Gamma^n{}_{mr} t^r t^m \end{aligned} \tag{2.7}$$

This describes a geodesic, or the “straightest possible line" on a curved manifold. In general relativity, particles in free fall travel along geodesics of spacetime.

2.2 Einstein Equations

The Riemann tensor contains all the local information about the curvature of the manifold. It possesses a number of important symmetries. By contracting its indices, we can form related tensors that will be essential to formulate Einstein’s equations.

Contracting the first and third indices of the Riemann tensor yields the Ricci tensor, $R_{\mu\nu} = R^{\rho}_{\mu\rho\nu}$ which is a symmetric $0, 2$ tensor. A further contraction gives the Ricci scalar, $R = g^{\mu\nu} R_{\mu\nu}$, a scalar function on the manifold.

A specific linear combination of these two quantities form the Einstein tensor,

$$R^{\mu\nu} - \frac{1}{2} g^{\mu\nu} R = G_{\mu\nu} \tag{2.8}$$

Following from the properties satisfied by the Riemann tensor, the covariant divergence is zero $DG_{\mu\nu} = 0$. This is important for consistency with the conservation of energy and momentum.

2.2.1 The Einstein-Hilbert Action

The action principle of General Relativity describes the action over a spacetime manifold, equipped with a metric of Lorentzian signature ($g = \det g_{\mu\nu}$) as,

$$S = \sqrt{-g} R d^4 x \tag{2.9}$$

This is called the Einstein-Hilbert action. The equations of motion can be now found by applying the principle of least action, which states that the variation of the action with respect to the field—in this case, the metric $g_{\mu\nu}$ must be zero. The variation δS is calculated as,

$$g_{\mu\nu} x \to g_{\mu\nu} x \;\; \delta g_{\mu\nu} x \tag{2.10}$$

Writing the Ricci scalar as $R = g^{\mu\nu} R_{\mu\nu}$, the Einstein-Hilbert action changes as

$$\delta S = d^4 x \left(\delta \sqrt{-g} g^{\mu\nu} R_{\mu\nu} \;\; \sqrt{-g} \delta g^{\mu\nu} R_{\mu\nu} \;\; \sqrt{-g} g^{\mu\nu} \delta R_{\mu\nu} \right) \tag{2.11}$$

In terms of the inverse metric $\delta g^{\mu\nu}$,

$$\begin{aligned} g_{\rho\mu} g^{\mu\nu} &= \delta^{\nu}_{\rho} \\ \Rightarrow \quad \delta g_{\rho\mu} g^{\mu\nu} \;\; g_{\rho\mu} \delta g^{\mu\nu} &= 0 \\ \Rightarrow \quad \delta g^{\mu\nu} &= -g^{\mu\rho} g^{\nu\sigma} \delta g_{\rho\sigma} \end{aligned} \tag{2.12}$$

$\therefore$ the middle term in Eq. 2.11 is proportional to $\delta g^{\mu\nu}$.

Now using the result that the variation of $\sqrt{-g}$ is given by

$$\delta \sqrt{-g} = -\frac{1}{2} \sqrt{-g} g_{\mu\nu} \delta g^{\mu\nu} \tag{2.13}$$

we can write

$$\delta S = d^4 x \sqrt{-g} \left(R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} \right) \delta g^{\mu\nu} \;\; \sqrt{-g} g^{\mu\nu} \delta R_{\mu\nu} \tag{2.14}$$

By calculating the variation of Christoffel symbols and the Reimann tensor, one can write the final term in the above equation as

$$g^{\mu\nu}\delta R_{\mu\nu} = \nabla_\mu X^\mu \quad \text{with} \quad X^\mu = g^{\rho\nu}\delta\Gamma^\mu_{\rho\nu} - g^{\mu\nu}\delta\Gamma^\rho_{\nu\rho} \tag{2.15}$$

Hence we can now ignore this final term using divergence theorem to get,

$$\delta S = \left(R_{\mu\nu} - \frac{1}{2}g_{\mu\nu}R\right)\delta g^{\mu\nu}\sqrt{-g}d^4x \tag{2.16}$$

Setting $\delta S = 0$, we get the vacuum Einstein field equations:

$$R^{\mu\nu} - \frac{1}{2}g^{\mu\nu}R = G_{\mu\nu} = 0 \tag{2.17}$$

These equations describe the geometry of empty space. Solutions to these equations include the trivial Minkowski spacetime, but also non-trivial solutions like the Schwarzschild black hole and gravitational waves.

The Cosmological Constant

By multiplying the volume form by a constant, the modified action becomes

$$S = \sqrt{-g}R - 2\Lambda d^4x \tag{2.18}$$

Where Λ is the cosmological constant. Its inclusion modifies the field equations to:

$$G_{\mu\nu} = \Lambda g_{\mu\nu} = 0 \tag{2.19}$$

This term can be interpreted as an intrinsic energy density and pressure of the vacuum itself. A positive Λ drives an accelerated expansion of the universe, consistent with modern cosmological observations. It is analogous to the potential energy V in the definition of the Lagrangian, $L = T - V$.

3. Black Holes

The Schwarzschild metric describes the unique, static, spherically symmetric solution to the vacuum Einstein equations ($R_{\mu\nu} = 0$). Birkhoff's Theorem states that any spherically symmetric vacuum solution must be the Schwarzschild solution, hence the uniqueness. It describes the spacetime outside any non-rotating, uncharged, spherical body. The metric is given by the line element,

$$ds^2 = -\left(1 - \frac{2MG}{c^2 r}\right)dt^2 \quad \frac{dr^2}{(1 - \frac{2MG}{c^2 r})c^2} \quad \frac{1}{c^2} r^2 d\Omega^2 \tag{3.1}$$

One has to note that the Schwarzschild solution is an idealization based on the assumption that the mass M is concentrated at a point. If the mass is spread out, this equation is not true in the interior, but it is true in the exterior.

At the Schwarzschild radius, $r = R_s = 2GMc^2$, there appears to be a singularity. However, by calculating coordinate-invariant quantities like the Kretschmann scalar, one finds that the curvature at R_s is finite. This indicates that this is just a coordinate singularity, an artifact of the chosen coordinate system, and not a physical one. This surface is called the event horizon.

At $r = 0$, the Kretschmann scalar diverges, indicating a region of infinite curvature. This is a genuine physical singularity where the theory of general relativity breaks down.

3.1 The Event Horizon & Photosphere

Let us know see how relativistic particles move in this metric. Consider a radial orbit where $c = 1$.

$$-ds^2 = d\tau^2 = \left(1 - \frac{2MG}{r}\right)dt^2 - \frac{dr^2}{(1 - \frac{2MG}{r})} - \underbrace{r^2 d\Omega^2}_{\text{assume spherical symmetry}}$$

$$S = -M\sqrt{\left(1 - \frac{2MG}{r}\right) - \frac{r^2}{(1 - \frac{2MG}{r})}}\,dt \tag{3.2}$$

$$L = -M\sqrt{\left(1 - \frac{2MG}{r}\right) - \frac{r^2}{(1 - \frac{2MG}{r})}} \tag{3.3}$$

Now consider Energy conservation using simple Hamiltonian mechanics.

$$P_r = \frac{\partial L}{\partial r}, \text{ radial momentum}$$
$$H = P_r r - L \tag{3.4}$$
$$= \frac{(1 - \frac{2MG}{r})m}{\sqrt{(1 - \frac{2MG}{r}) - \frac{r^2}{(1 - \frac{2MG}{r})}}} \tag{3.5}$$
$$= E, \text{ which needs to be constant}$$
$$\text{and, } r^2 = (1 - \frac{2MG}{r})^2 - \frac{(1 - \frac{2MG}{r})^3}{E^2}$$
$$r \approx \sqrt{\frac{r - 2MG}{MG}} \text{ near the event horizon.} \tag{3.6}$$

This shows us that the particles slow down asymptotically as it approaches the event horizon.

Similar to the above method, one can also predict the behaviour of a light ray near a black hole using classical mechanics.

The action is given by:

$$S = -M \; d\tau$$
$$= -M \sqrt{\underbrace{(1 - \frac{2MG}{r})}_{=F,\text{ say}} dt^2 - \underbrace{\frac{1}{1 - \frac{2MG}{r}}}_{=G,\text{ say}} dr^2 - r^2 d\Omega^2}$$
$$= -M \sqrt{Fr - Grr^2 - r^2\theta^2} dt \tag{3.7}$$
$$\text{where, } L = -M\sqrt{Fr - Grr^2 - r^2\theta^2} \tag{3.8}$$

Now we calculate the angular and radial momenta.

$$L = P_\theta$$
$$= \frac{\partial L}{\partial \theta}$$
$$= \frac{mr^2\theta}{\sqrt{Fr - Grr^2 - r^2\theta^2}} \tag{3.9}$$
$$= m\kappa \tag{3.10}$$
$$P_r = \frac{\partial L}{\partial r}$$
$$= \frac{mGr}{\sqrt{Fr - Grr^2 - r^2\theta^2}} \tag{3.11}$$

Here P_r is not conserved, but the energy is which is given by the classical Hamiltonian as follows.

$$
\begin{aligned}
H =& P_r r \;\; L\theta - L \\
=& \frac{Frm}{\sqrt{Fr - Grr^2 - r^2\theta^2}} = E \\
\text{For a circular orbit, } E =& \frac{Frm}{\sqrt{Fr - r^2\theta^2}} && (3.12) \\
\frac{L}{M} =& \frac{r^2\theta}{\sqrt{Fr - r^2\theta^2}} \\
=& \kappa, \text{ reduced angular momentum} && (3.13)
\end{aligned}
$$

Solving for θ:

$$
\begin{aligned}
\kappa^2 =& \frac{r^4\theta^2}{Fr - r^2\theta^2} \\
\kappa^2 Fr =& \kappa^2 r^2\theta^2 \;\; r^4\theta^2 \\
\theta^2 =& \frac{\kappa^2 Fr}{\kappa^2 r^2 \;\; r^4} && (3.14)
\end{aligned}
$$

Substituting,

$$
\begin{aligned}
E =& \frac{Frm}{\sqrt{Fr - r^2\theta^2}} \\
E =& \frac{\sqrt{F}\sqrt{r^2\kappa^2 \;\; r^4}}{r^2} m
\end{aligned}
$$

We want now to let $M \to 0$ for a photon. In that case,

$$
E = \frac{\sqrt{F}L}{r} \quad (3.15)
$$

$$
\begin{aligned}
E =& \frac{\sqrt{1 - \frac{2MG}{r}}L}{r} \\
\log E =& \frac{1}{2}\log 1 - \frac{2MG}{r} - \log r \\
\frac{1}{E}\frac{dE}{dr} =& \frac{MG}{rr - 2MG} - \frac{r - 2MG}{rr - 2MG} \\
=& \frac{3MG - r}{rr - 2MG}
\end{aligned}
\quad (3.16)
$$

We obtain an unstable equilibrium at $r = 3MG$ which is also known as the photosphere of a blackhole. Outside this radius, the photon will spiral out while inside the radius it will spiral in.

4. Simulation

We can numerically simulate the trajectory of a particle near the blackhole by solving the geodesic equation. In particular, for a non-rotating uncharged black hole, we apply the Schwarzschild Metric (Eq. 3.1) which yields the following non-zero components of the metric tensor,

$$g_{tt} = -\left(1 - \frac{2M}{r}\right) \tag{4.1}$$

$$g_{rr} = \frac{1}{1 - \frac{2M}{r}} \tag{4.2}$$

$$g_{\theta\theta} = r^2 \tag{4.3}$$

$$g_{\phi\phi} = r^2 \sin^2\theta \tag{4.4}$$

where M is the mass of the black hole and r is the distance from its center. Note that we use natural units with $G = c = 1$ for simplicity. The event horizon of the blackhole will be,

$$R_{\text{EH}} = 2M \tag{4.5}$$

The path of a free-falling particle (a geodesic) is described by the geodesic equation:

$$\frac{d^2x^\alpha}{d\lambda^2} \; \Gamma^\alpha_{\beta\gamma}\frac{dx^\beta}{d\lambda}\frac{dx^\gamma}{d\lambda} = 0 \tag{4.6}$$

Here, λ is the affine parameter along the path (proper time τ for massive particles), $x^\alpha = t, r, \theta, \phi$ are the coordinates, and $\Gamma^\alpha_{\beta\gamma}$ are the Christoffel symbols of the second kind. The Christoffel symbols are calculated from the metric tensor as follows.

$$\Gamma^\alpha_{\beta\gamma} = \frac{1}{2}g^{\alpha\delta}\left(\frac{\partial g_{\delta\beta}}{\partial x^\gamma}\;\frac{\partial g_{\delta\gamma}}{\partial x^\beta} - \frac{\partial g_{\beta\gamma}}{\partial x^\delta}\right) \tag{4.7}$$

By solving the above equation with the Schwarzschild metric, we obtain the following non-zero Christoffel symbols.

$$\Gamma^t_{rt} = \Gamma^t_{tr} = \frac{M}{rr - 2M} \tag{4.8}$$

$$\Gamma^r_{tt} = \frac{Mr - 2M}{r^3} \tag{4.9}$$

$$\Gamma^r_{rr} = -\frac{M}{rr - 2M} \tag{4.10}$$

$$\Gamma^r_{\theta\theta} = -r - 2M \tag{4.11}$$

$$\Gamma^r_{\phi\phi} = -r - 2M \sin^2\theta \tag{4.12}$$

$$\Gamma^\theta_{r\theta} = \Gamma^\theta_{\theta r} = \frac{1}{r} \tag{4.13}$$

$$\Gamma^\theta_{\phi\phi} = -\sin\theta\cos\theta \tag{4.14}$$

$$\Gamma^\phi_{r\phi} = \Gamma^\phi_{\phi r} = \frac{1}{r} \tag{4.15}$$

$$\Gamma^\phi_{\theta\phi} = \Gamma^\phi_{\phi\theta} = \frac{\cos\theta}{\sin\theta} = \cot\theta \tag{4.16}$$

4.1 Solving ODEs using numerical intergation

To solve the geodesic equation numerically, we convert the single second-order equation into a system of eight first-order Ordinary Differential Equations (ODEs) which is solved in parallel. Let us define the 4-velocity as $p^\alpha = \frac{dx^\alpha}{d\lambda}$, where λ is our affine parameter. The state vector is then

$$Y = (x^\alpha, p^\alpha) = \left(t, r, \theta, \phi, p^t, p^r, p^\theta, p^\phi\right)$$

We can obtain the first four ODEs using the definition of the 4-velocity components.

$$\frac{dt}{d\lambda} = p^t \tag{4.17}$$

$$\frac{dr}{d\lambda} = p^r \tag{4.18}$$

$$\frac{d\theta}{d\lambda} = p^\theta \tag{4.19}$$

$$\frac{d\phi}{d\lambda} = p^\phi \tag{4.20}$$

Now, by rearranging the geodesic equation, we get 4 more ODEs, as follows.

$$\frac{dp^\alpha}{d\lambda} = -\Gamma^\alpha_{\beta\gamma} p^\beta p^\gamma$$

.

- For $\alpha = t$,

$$\frac{dp^t}{d\lambda} = -\Gamma^t_{\beta\gamma}p^\beta p^\gamma = -\left(\Gamma^t_{rt}p^r p^t \;\; \Gamma^t_{tr}p^t p^r\right) = -2\Gamma^t_{rt}p^r p^t$$
$$= -\frac{2Mp^t p^r}{rr - 2M} \tag{4.21}$$

- For $\alpha = r$,

$$\frac{dp^r}{d\lambda} = -\Gamma^r_{\beta\gamma}p^\beta p^\gamma = -\left(\Gamma^r_{tt}\left(p^t\right)^2 \;\; \Gamma^r_{rr}\left(p^r\right)^2 \;\; \Gamma^r_{\theta\theta}\left(p^\theta\right)^2 \;\; \Gamma^r_{\phi\phi}\left(p^\phi\right)^2\right)$$
$$= -\left(\frac{Mr - 2M}{r^3}\left(p^t\right)^2 \; -\frac{M}{rr - 2M}\left(p^r\right)^2 - r - 2M\left(p^\theta\right)^2 \; -r - 2M\sin^2\theta\left(p^\phi\right)^2\right) \tag{4.22}$$

- For $\alpha = \theta$,

$$\frac{dp^\theta}{d\lambda} = -\Gamma^\theta_{\beta\gamma}p^\beta p^\gamma = -\left(\Gamma^\theta_{r\theta}p^r p^\theta \;\; \Gamma^\theta_{\theta r}p^\theta p^r \;\; \Gamma^\theta_{\phi\phi}\left(p^\phi\right)^2\right) = -\left(2\Gamma^\theta_{r\theta}p^r p^\theta \;\; \Gamma^\theta_{\phi\phi}\left(p^\phi\right)^2\right)$$
$$= -\left(\frac{2p^r p^\theta}{r} - \sin\theta\cos\theta\left(p^\phi\right)^2\right) \tag{4.23}$$

- For $\alpha = \phi$,

$$\frac{dp^\phi}{d\lambda} = -\Gamma^\phi_{\beta\gamma}p^\beta p^\gamma = -\left(\Gamma^\phi_{r\phi}p^r p^\phi \;\; \Gamma^\phi_{\phi r}p^\phi p^r \;\; \Gamma^\phi_{\theta\phi}p^\theta p^\phi \;\; \Gamma^\phi_{\phi\theta}p^\phi p^\theta\right)$$
$$= -\left(2\Gamma^\phi_{r\phi}p^r p^\phi \;\; 2\Gamma^\phi_{\theta\phi}p^\theta p^\phi\right)$$
$$= -2\left(\frac{p^r p^\phi}{r} - \cot\theta p^\theta p^\phi\right) \tag{4.24}$$

This system of 8 first-order ODEs, (4.17) through (4.24), have been implemented python and are solved using the standard SciPy libraries which uses the Runge-Kutta (RK45) algorithm.

4.2 Simulation Setup and Conserved Quantities

Time translation and rotation about the z-axis gives rise to two symmetries which correspond to the conserved quantities, specific energy, E, and the specific angular momentum, L_z.

The specific energy E and specific angular momentum L_z are defined from the p^α and $g_{\alpha\beta}$ using,

$$E = -g_{tt}p^t = \left(1 - \frac{2M}{r}\right)\frac{dt}{d\lambda} \tag{4.25}$$

$$L_z = g_{\phi\phi}p^\phi = r^2\sin^2\theta\frac{d\phi}{d\lambda} \tag{4.26}$$

To solve the initial value problem, we need to provide the initial position vector ($t_0 = 0$, r_0, θ_0, ϕ_0), along with ϵ, which defines the type of particle ($\epsilon = -1$ for massive particles with timelike geodesics and $\epsilon = 0$ for massless particles, with null geodesics).

$$g_{\mu\nu} p^\mu p^\nu = \epsilon \tag{4.27}$$

To constrain the particle to the equitorial plane (for simplicity), we set

$$\theta_0 = \frac{\pi}{2} \tag{4.28}$$

$$p_0^\theta = 0 \tag{4.29}$$

Since E and L_z are conserved throughout the entire trajectory, we can use them to determine the initial values for p_0^t and p_0^ϕ.

$$p_0^t = \frac{E}{1 - \frac{2M}{r_0}} \tag{4.30}$$

$$p_0^\phi = \frac{L_z}{r_0^2} \tag{4.31}$$

We can find the initial radial velocity, p_0^r, with our known values and $\theta_0 = \pi 2, p_0^\theta = 0$

$$\frac{1}{1 - \frac{2M}{r_0}} \left(p_0^r\right)^2 = \epsilon \left(1 - \frac{2M}{r_0}\right) \left(p_0^t\right)^2 - r_0^2 \left(p_0^\phi\right)^2 \tag{4.32}$$

$$\left(p_0^r\right)^2 = \left(1 - \frac{2M}{r_0}\right) \left[\epsilon \frac{E^2}{1 - \frac{2M}{r_0}} - \frac{L_z^2}{r_0^2}\right] \tag{4.33}$$

$$\left(p_0^r\right)^2 = E^2 - \left(1 - \frac{2M}{r_0}\right) \left(-\epsilon \frac{L_z^2}{r_0^2}\right) \tag{4.34}$$

The sign of $p_0^r = \pm\sqrt{\left(p_0^r\right)^2}$ is chosen depending on whether the particle is initially moving inwards (-) or outwards (+). If the term under the square root is negative, the chosen (E, L_z, r_0) combination is physically forbidden. The next section presents the results of some simulations with some physical parameters.

5. Results

As described in the previous chapter, we use the input parameters E, L_z, ϵ and an initial position of

$$r_0 = 10, \theta_0 = \frac{\pi}{2}, \phi_0 = 0$$

Since we stay in the x-y plane ($\theta_0 = \frac{\pi}{2}$) we can project the x and y coordinates of the particle to produce a 2D image of the trajectory path. Figure 5.1 and 5.2 show the simulation results after $N = 500$ iterations.

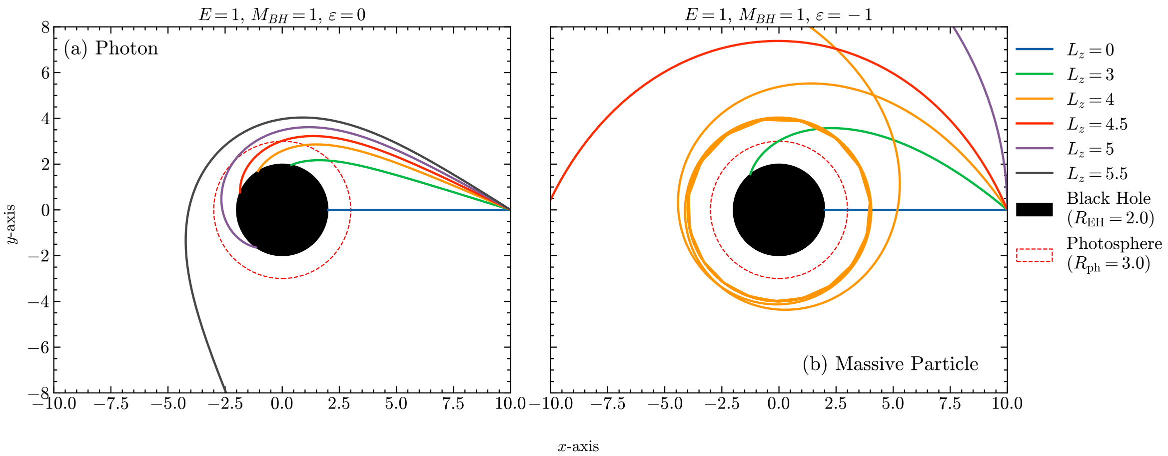


Figure 5.1: Trajectory of a (a) photon and a (b) massive particle for different values of angular momentum L_z and fixed energy $E = 1$. One can note that the photon escapes the orbit of the black hole only at $L_z = 5.5M$ while the massive particle is able to achieve that with a lower L_z.

In all these simulations, one can see that no particle that enters the photosphere can escape, as predicted. Note that in some cases (Figure 5.1.b for $E = 1, L_z = 4$), the particle seems to enter an orbit around the black hole. However, after some number of iterations, the numerical errors of the integrator pile up causing the trajectory to diverge. This is a numerical error and can be minimised by reducing the error tolerance at each step of the RK45 integrator. The current simulation uses 10^{-10} relative tolerance, which is good enough for normal use cases.

All the code is documented in this github repository. Furthermore, animation videos have been generated using the tool Manim and can be seen in this folder.

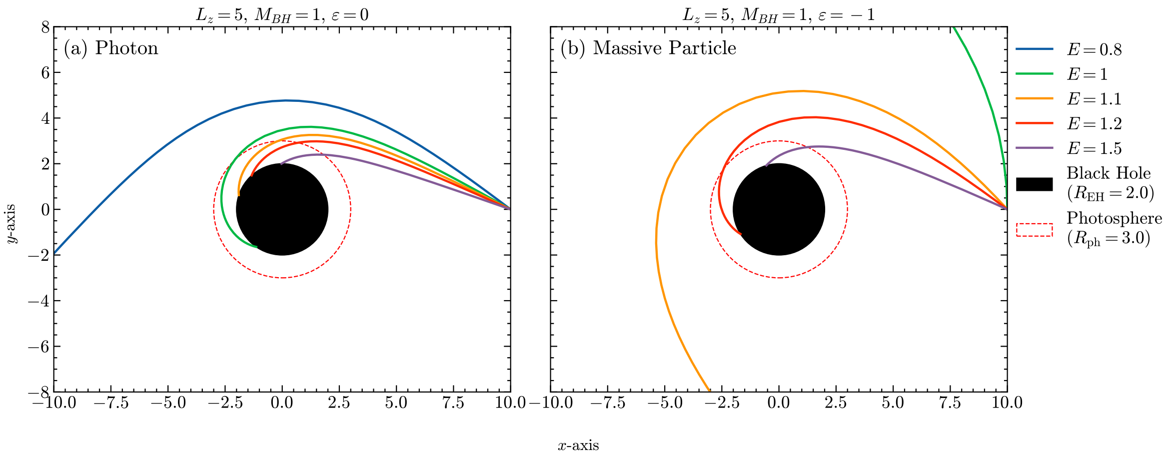


Figure 5.2: Trajectory of a (a) photon and a (b) massive particle for different values of energy E and fixed angular momentum $L_z = 5$. We see that the photon escapes the orbit of the black hole only at $E \sim 1.5M$ while the massive particle is able to achieve that with a lower $E \sim 1.1M$.

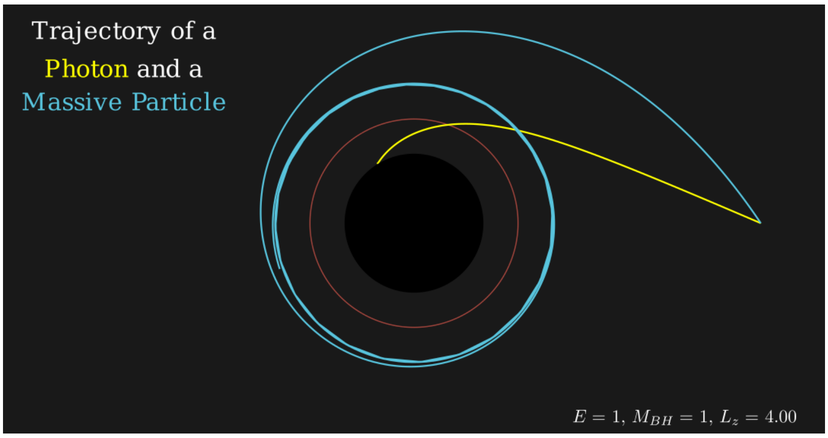


Figure 5.3: With increased error tolerance, we can see that a stable orbit is achieved by the massive particle at $E = 1, L_z = 4$.

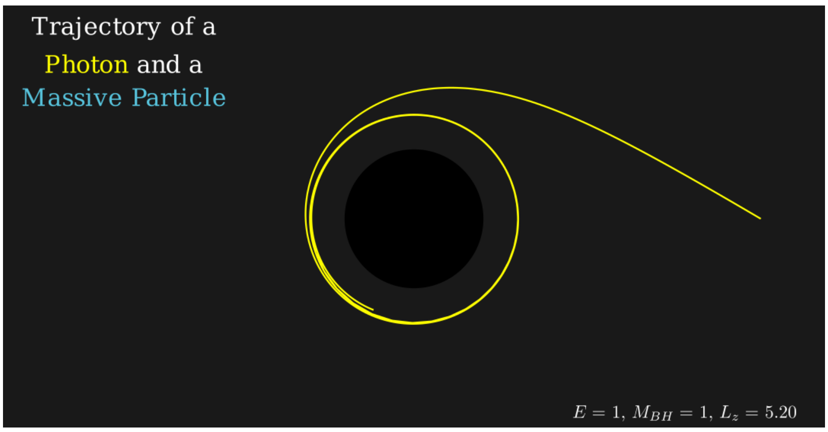


Figure 5.4: Similarly, we can see that a stable orbit is achieved by the photon particle at $E = 1, L_z \sim 5.2$.

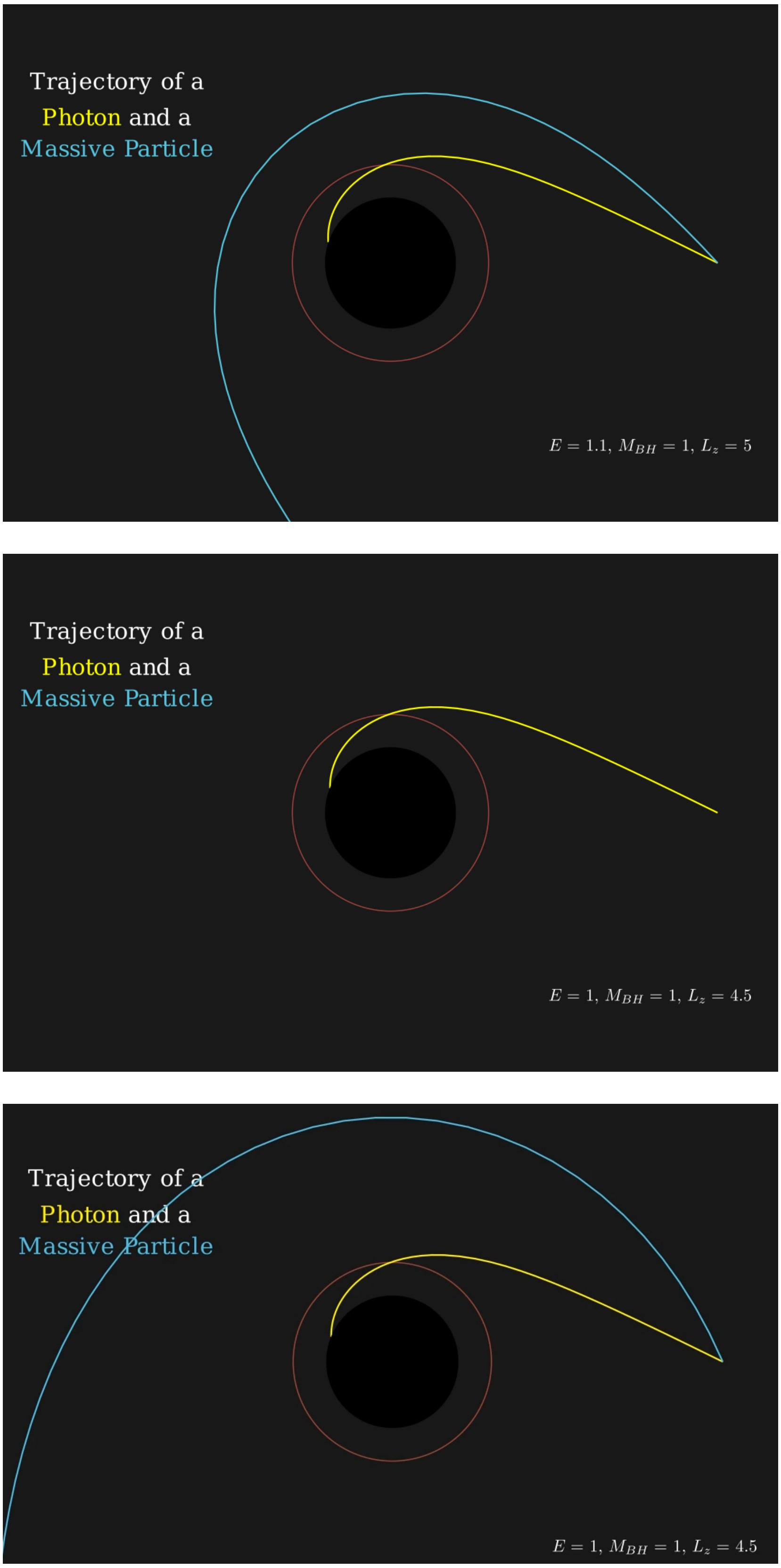


Figure 5.5: Some snapshots of the Manim animations. The red circle represents the photosphere of the black hole.

A. Appendix

Below are the Python codes for `solver.py`, which is responsible for numerically solving the Geodesic Equation, and `main.py`, which extracts solution from `solver.py` and generates the animation using Manim.

Listing A.1: solver.py

```
import numpy as np

class Schwarzschild:
    def __init__(self, M):
        self.M = M
        self.r_EH = 2*self.M

    def g_tt(self, r, theta):
        return -(1 - 2*self.M/r)

    def g_rr(self, r, theta):
        return 1/(1 - 2*self.M/r)

    def g_thth(self, r, theta):
        return r**2

    def g_phph(self, r, theta):
        return r**2*np.sin(theta)**2

    def geodesic_eq_t(self, y):
        t, r, theta, phi, p0, p1, p2, p3 = y

        return -2*self.M*p1*p0/(r*(r-2*self.M))

    def geodesic_eq_r(self, y):
        t, r, theta, phi, p0, p1, p2, p3 = y
        sin = np.sin(theta)
        M = self.M
        ch0 = M*(r-2*M)/r**3
        ch1 = -M/(r*(r-2*M))
        ch2 = -(r-2*M)
        ch3 = -(r-2*M)*sin**2
        return -(ch0*p0**2 + ch1*p1**2 + ch2*p2**2 + ch3*p3**2)

```

```
    def geodesic_eq_theta(self, y):
        t, r, theta, phi, p0, p1, p2, p3 = y
        sin = np.sin(theta)
        cos = np.cos(theta)
        M = self.M
        return -(2*p1*p2/r - sin*cos*p3**2)

    def geodesic_eq_phi(self, y):
        t, r, theta, phi, p0, p1, p2, p3 = y
        sin = np.sin(theta)
        cos = np.cos(theta)
        return -2*(p1*p3/r - cos*p2*p3/sin)

    def compute_4momentum(self, r0, E, Lz, epsilon = 0):
        p0 = E/(1 - 2*self.M/r0)
        p3 = Lz/r0**2
        try:
            # negative solution spirals inward
            p1 = -np.sqrt(E**2 - (1-2*self.M/r0)*(Lz**2/r0**2 - epsilon))
        except:
            print('This set of inital conditions is forbidden.')
            p1 = 0
        return p0, p1, p3

    def geodesic_eq(self, y):
        """
        Return 8 differential equations which solve for the geodesic.
        """
        t, r, theta, phi, p0, p1, p2, p3 = y

        # Initializing the derivatives of the coordinates and four-momentum of our particle
        derivatives = np.zeros_like(y)

        derivatives[0] = p0
        derivatives[1] = p1
        derivatives[2] = p2
        derivatives[3] = p3
        derivatives[4] = self.geodesic_eq_t(y)
        derivatives[5] = self.geodesic_eq_r(y)
        derivatives[6] = self.geodesic_eq_theta(y)
        derivatives[7] = self.geodesic_eq_phi(y)

        return derivatives

    def RKF45(self, y, h, tol = 1e-7):
        abs_error = np.inf

        while abs_error > tol:
            k1 = self.geodesic_eq(y)
            k2 = self.geodesic_eq(y +       1/4*k1*h)
            k3 = self.geodesic_eq(y +      3/32*k1*h +      9/32*k2*
```

```
h)
            k4 = self.geodesic_eq(y + 1932/2197*k1*h - 7200/2197*k2*h + 7296/2197*k3*h)
            k5 = self.geodesic_eq(y +   439/216*k1*h -          8*k2*h +  3680/513*k3*h -  845/4104*k4*h)
            k6 = self.geodesic_eq(y -      8/27*k1*h +          2*k2*h - 3544/2565*k3*h + 1859/4104*k4*h - 11/40*k5*h)

            new_y = y + 16/135*k1*h + 6656/12825*k3*h + 28561/56430*k4*h - 9/50*k5*h + 2/55*k6*h
            error = -1/360*k1*h + 128/4275*k3*h + 2197/75240*k4*h - 1/50*k5*h - 2/55*k6*h
            abs_error = np.sqrt(np.sum(error**2))
            new_h = 0.9*h*(tol/abs_error)**(1/5)
            h = new_h

        return new_y, h

    def solve(self, y0, n, h0):
        affine_parameter = np.zeros(n)

        # Solution Array
        y = np.zeros((n, len(y0)))
        y[0] = y0
        h = h0 # Initial step size

        # Performing numerical integration
        for i in range(n - 1):
            y_next, h = self.RKF45(y[i], h)

            y[i + 1] = y_next
            affine_parameter[i + 1] = affine_parameter[i] + h

        return affine_parameter, y
```

Listing A.2: main.py

```
from manim import *
from math import *
from solver import Schwarzschild

def trajectory(r0 = 3, E=1, Lz=5, M=1, epsilon=0):
    schwarz = Schwarzschild(M = M)

    # Initial Conditions
    t0 = 0
    r0 = r0
    theta0 = np.pi/2
    phi0 = 0

    p0 = 0 # time
    p1 = 0 # r
```

```
    p2 = 0 # theta
    p3 = 0 # phi

    n_rays = 1 # number of light rays
    N = 100 # number of simulation points

    lines = []
    for epsilon in [0, -1]:
        p0, p1, p3 = schwarz.compute_4momentum(r0, E, Lz, epsilon)
        y0 = [t0, r0, theta0, phi0, p0, p1, p2, p3]

        # Solve the equations of motion for a specified number of steps with a specified initial step size
        tau, sol = schwarz.solve(y0, N, 1e-5)

        t, r, theta, phi = sol[:, 0], sol[:, 1], sol[:, 2], sol[:, 3]

        x = r*np.cos(phi)*np.sin(theta)
        y = r*np.sin(phi)*np.sin(theta)
        lines.append(np.array([x, y, np.zeros(N)]).T)

    return lines, schwarz.r_EH

def circle(r):
    theta = np.linspace(-2*np.pi, 2*np.pi, 500)
    return np.array([r*np.cos(theta), r*np.sin(theta), np.zeros(500)]).T

class BlackHole(Scene):
    def construct(self):
        self.camera.background_color = ManimColor('#191919')
        axes = Axes(
            x_range=(-10, 10),
            y_range=(-10, 10),
            x_length=12, y_length=12
        )

        # Initial Conditions
        r0 = 10
        E = 1.01
        Lz = 4.5
        M = 1
        epsilon = 0
        all_photons, r_EH = trajectory(r0, E, Lz, M, epsilon)

        # Add Text
        self.add(Tex(r'$E =$' + f' {E},' +r' $M_{BH}=$'+f' {M},' +r' $L_z=$' + f' {Lz:.2f}', font_size=30).to_edge(DR).set_color(WHITE))
        self.add(MarkupText('Trajectory of a', font_size=30).to_edge(UL).set_color(WHITE))
        self.add(MarkupText(f'<span fgcolor="{YELLOW}">Photon</span> and a', font_size=30).next_to(self.mobjects[-1],DOWN))
```

```
        self.add(MarkupText(f'<span fgcolor="{BLUE}">Massive Particle</span>', font_size=30).next_to(self.mobjects[-1],DOWN))

        # Draw Trajectories
        curves = VGroup()
        colors = [YELLOW, BLUE]
        for i, points in enumerate(all_photons):
            curve = VMobject().set_points_as_corners(axes.c2p(points))
            curve.set_stroke(colors[i], 3)
            curves.add(curve)

        # Draw the Black Hole
        black_hole_core = VMobject().set_points_as_corners(axes.c2p(circle(r_EH)))
        black_hole_core.set_stroke(
                opacity=0
        )
        black_hole_core.set_fill(BLACK, opacity=1.0)
        self.add(black_hole_core)
        black_hole_core.z_index = 1

        # Draw the Photosphere
        photosphere = VMobject().set_points_as_corners(axes.c2p(circle(3*M)))
        photosphere.set_stroke(
                color=RED,
                width=2,
                opacity=0.6
        )
        dashed_photosphere = DashedVMobject(photosphere)
        self.add(dashed_photosphere)

        # Animate the light ray being traced
        self.play(
            *(
                Create(curve, rate_func=linear)
                for curve in curves
            ),
            run_time=2,
        )
        self.wait(1)
```

Bibliography

Carroll, S. M. 1997, Lecture Notes on General Relativity, arXiv, doi: 10.48550/arXiv.gr-qc/9712019

Schutz, B. 2009, A first course in general relativity (Cambridge University Press)