Physics 6G03:Computational Physics

Project: Numerical Gravity

Ashiful★Bhuiyan

Instructor: Prof. An-Chang Shi

Submission Date: December 19, 2025

Contents

1 Introduction

This project aimed at simulating and visualizing particle trajectories (geodesics) in the vicinity of a non-rotating massive object, described by General Relativity. While Newtonian gravity suffices for weak fields, relativistic effects such as the precession of perihelion and photon capture, require the full machinery of the geodesic equations in curved spacetime.

This project is based on the framework presented in the recent paper gr-orbit-toolkit [1]. We solve the geodesic equations of motion numerically for a test particle in a Schwarzschild metric.

The primary goals of this project is to develop and implement a python-based integrator to solve the coupled second order ODEs governing the particle's motion. Next, we compare the performance of a standard explicit solver such as the 4th order Runge-Kutta (RK4) method against a symplectic integrator such as the Velocity Verlet or Leapfrog, in order to analyze the long-term stability of orbital integration. Finally, we verify the simulation by reproducing an analytical result for the precession of Mercury's perihelion and checking for the conservation of constants of motion, that being the Energy, E, and Angular Momentum, $\vec{L}$.

2 Theory

2.1 The Schwarzschild Metric

We consider the spacetime geometry outside a spherical, non-rotating body of mass M, described by the Schwarzschild metric [2]. Using natural units where $G = c = 1$:

$$ds^2 = -\left(1 - \frac{2M}{r}\right) dt^2 + \left(1 - \frac{2M}{r}\right)^{-1} dr^2 + r^2 d\theta^2 + r^2 \sin^2\theta d\phi^2$$

If we restrict the motion to the equatorial plane ($\theta = \pi/2$), the metric simplifies and the squared line element per unit affine parameter becomes (Hartle, p. 193)

$$\mathcal{L} \equiv g_{\mu\nu}\dot{x}^\mu \dot{x}^\nu = -\left(1 - \frac{2M}{r}\right) \dot{t}^2 + \left(1 - \frac{2M}{r}\right)^{-1} \dot{r}^2 + r^2\dot{\phi}^2 = \varepsilon.$$

Here the dot denotes differentiation with respect to proper time τ for massive particles, or an affine parameter λ for photons. For massive particles, $\varepsilon = -1$, while for null geodesics $\varepsilon = 0$.

2.2 The Potential

Using the symmetries of the Schwarzschild metric and the normalization of the four velocity, we can derive the system of ODEs required for the numerical integration. (We can also have done this through Euler-Lagrange equations).

Due to time-translation and axial symmetry of the Schwarzschild metric, there exist two conserved quantities along the geodesic. We define the specific energy and angular momentum as

$$E \equiv \left(1 - \frac{2M}{r}\right) \dot{t}, \qquad L \equiv r^2\dot{\phi},$$

$$E = \left(1 - \frac{2M}{r}\right)\dot{t} \implies \dot{t} = \frac{E}{1 - \frac{2M}{r}}$$

which are constants of motion (Hartle, Eqs. 9.21–9.22).

Now we sub the expressions for $\dot{t}$ and $\dot{\phi}$ back into the normalization equation ($\mathcal{L} = -1$):

$$-\left(1 - \frac{2M}{r}\right)\left(\frac{E}{1 - \frac{2M}{r}}\right)^2 + \left(1 - \frac{2M}{r}\right)^{-1}\dot{r}^2 + r^2\left(\frac{L}{r^2}\right)^2 = -1$$

Simplifying we get:

$$-\frac{E^2}{1 - \frac{2M}{r}} + \left(1 - \frac{2M}{r}\right)^{-1}\dot{r}^2 + \frac{L^2}{r^2} = -1$$

Multiply by $(1 - \frac{2M}{r})$:

$$-E^2 + \dot{r}^2 + \frac{L^2}{r^2}\left(1 - \frac{2M}{r}\right) = -\left(1 - \frac{2M}{r}\right)$$

Isolate for $\dot{r}^2$:

$$\dot{r}^2 = E^2 - \frac{L^2}{r^2}\left(1 - \frac{2M}{r}\right) - \left(1 - \frac{2M}{r}\right)$$

$$\dot{r}^2 = E^2 - \left(1 - \frac{2M}{r}\right)\left(1 + \frac{L^2}{r^2}\right)$$

So we find our effective potential $V_{\text{eff}}(r)$ structurally similar to Hartle's eqn for the effective potential in page 194, eqn 9.28.

$$\begin{aligned} V_{\text{eff}}(r) &= \left(1 - \frac{2M}{r}\right)\left(1 + \frac{L^2}{r^2}\right) \\ &= 1 - \frac{2M}{r} + \frac{L^2}{r^2} - \frac{2ML^2}{r^3} \end{aligned}$$

Note that this definition of the effective potential differs from Hartle's by an overall factor and an additive constant, but leads to identical radial dynamics.

We can plot against the Newtonian Potential, given as:

$$V_{\text{Newton}}(r) = 1 - \frac{2M}{r} + \frac{L^2}{r^2}$$

The effective potential is defined so that the radial equation takes the form $\dot{r}^2 + V_{\text{eff}} = E^2$. This allows our Newtonian energy equation to differ by an overall normalization and does

not mean that a physical identification of V_{eff} with a Newtonian potential is implied. This differs from the standard energy per unit mass equation ($\frac{1}{2}\dot{r}^2 + V = \varepsilon$) by a factor of 2.

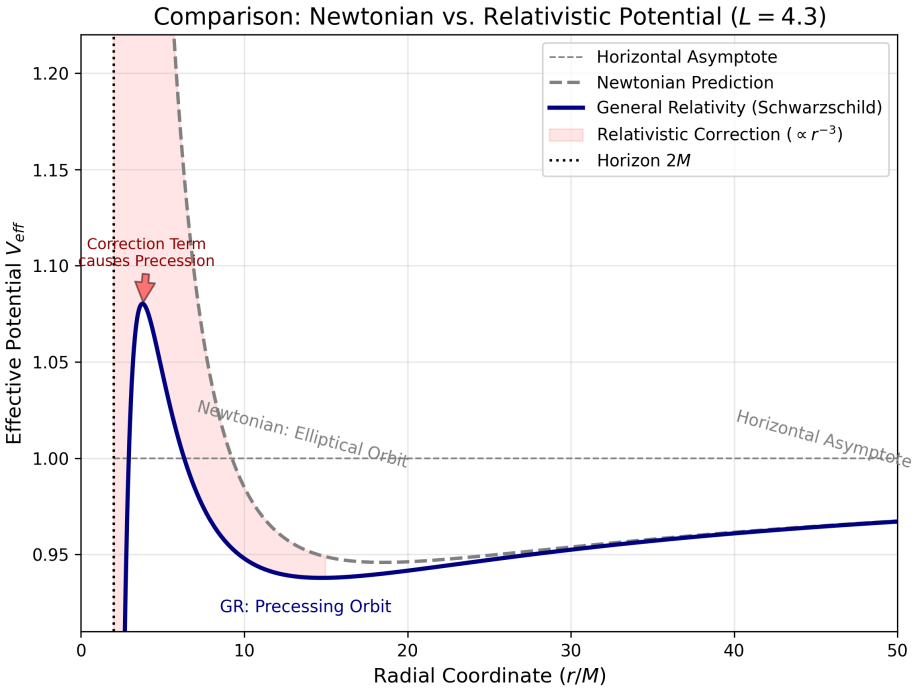


Figure 1: Comparison of the effective potential $V_{\text{eff}}(r)$ in Newtonian gravity (dashed gray) and Schwarzschild General Relativity (solid blue) for specific angular momentum $L = 4.3$. The shaded red region highlights the relativistic correction term proportional to r^{-3}. Two key relativistic effects are visible: (1) The potential well is deeper and shifted, which causes the anomalous precession of perihelia (e.g., Mercury), and (2) the centrifugal barrier is finite, allowing particles with sufficient energy to surmount the barrier and plunge into the black hole, a phenomenon impossible in Newtonian gravity for non-zero angular momentum.

2.3 Equation of Motion:

From $\dot{r}^2 + V_{\text{eff}} = E^2$, we want to get a form:

$$\ddot{r} = -\frac{1}{2}\frac{d}{dr}V_{eff}(r)$$

Rearrange to get $\dot{r}^2 = E^2 - V_{\text{eff}}$ and then by differentiating $\dot{r}^2$:

$$\frac{d}{d\tau}(\dot{r}^2) = 2\dot{r}\ddot{r}$$

The RHS:

$$
\begin{aligned}
&= \frac{d}{d\tau}\left[E^2 - \left(1 + \frac{L^2}{r^2} - \frac{2M}{r} - \frac{2ML^2}{r^3}\right)\right] \\
&= -\frac{d}{d\tau}\left(1 + L^2 r^{-2} - 2Mr^{-1} - 2ML^2 r^{-3}\right)
\end{aligned}
$$

Using chain rule ($\frac{d}{d\tau} = \dot{r}\frac{d}{dr}$):

$$
\begin{aligned}
&= -\dot{r}\left(-2L^2 r^{-3} + 2Mr^{-2} + 6ML^2 r^{-4}\right) \\
&= -\dot{r}\left(-\frac{2L^2}{r^3} + \frac{2M}{r^2} + \frac{6ML^2}{r^4}\right)
\end{aligned}
$$

Which gets us:

$$
2\dot{r}\ddot{r} = -2\dot{r}\left(-\frac{M}{r^2} + \frac{L^2}{r^3} - \frac{3ML^2}{r^4}\right)
$$

For $\dot{r} \neq 0$, we cancel $\dot{r}$ to obtain $\ddot{r} = -\frac{1}{2}\frac{d}{dr}V_{\text{eff}}$ which extends to turning points by continuity, we get that the radial motion is:

$$
\ddot{r} = -\frac{M}{r^2} + \frac{L^2}{r^3} - \frac{3ML^2}{r^4}
$$

This represents the Newtonian gravitation force, the centrifugal barrier ($\frac{L^2}{r^3}$) and the relativistic correction term ($-3ML^2/r^4$) which is responsible for the orbital precession.

For simulating photons (null geodesics), $\varepsilon = 0$. The radial equation changes to:

$$
\dot{r}^2 = E^2 - (1 - \frac{2M}{r})\frac{L^2}{r^2}
$$

In this case, only the ratio $b = L/E$ (impact parameter) is physically meaningful.

To avoid floating-point errors associated with astronomical numbers, we use dimensionless units where $GM = 1$. Results can be rescaled to SI units only for final validation against Mercury's orbital data.

The equations are integrated as a first-order system for radial and angular components with $\dot{t}, \dot{\phi}$ determined algebraically from the conserved quantities.

We model the motion of a test particle in a fixed Schwarzschild spacetime, neglecting any backreaction of the particle on the geometry. The integration is restricted to the exterior region $r > 2M$, where Schwarzschild coordinates remain well-defined. By spherical symmetry, the motion is confined to the equatorial plane ($\theta = \pi/2$). Timelike geodesics are parameterized by proper time τ, while null geodesics use an affine parameter.

2.3.1 Orbital Parameters and Initial Conditions

To simulate specific orbits, we relate the constants of motion E and L to the turning points of the orbit: the perihelion r_p (closest approach) and aphelion r_a (farthest distance). At these turning points, the radial velocity vanishes ($\dot{r} = 0$). Substituting this into the energy equation:

$$E^2 = \left(1 - \frac{2M}{r}\right)\left(1 + \frac{L^2}{r^2}\right)$$

Evaluating this expression at both r_p and r_a allows the constants of motion E and L to be determined. Circular orbits correspond to $r_p = r_a = r_c$, which occur at extrema of the effective potential. In particular, stable circular orbits exist only for $r_c \geq 6M$, defining the innermost stable circular orbit (ISCO).

A detailed classification of orbit types (bound, scattering, plunging), the existence of stable and unstable circular orbits, and the qualitative structure of the effective potential as a function of angular momentum are well established and discussed extensively in Hartle (Sec. 9.3). In this project we focus on numerically integrating representative bound and near-circular orbits relevant for perihelion precession, rather than reproducing the full analytical classification.

3 Numerical Implementation

We determine the initial conditions constants of motion, E and L, based on the desired geometric properties of the orbit. For a bound orbit defined by a perihelion distance r_p and an aphelion distance r_a, the radial velocity ($\dot{r}$) must go to 0 at both turning points. Consequently, the particle's specific energy must satisfy the effective potential equation at these radius:

$$E^2 = V_{\text{eff}}(r_p; L) \quad \text{and} \quad E^2 = V_{\text{eff}}(r_a; L)$$

Since the specific energy E is a constant of motion, the effective potential must be identical at both turning points. We find a non-linear constraint equation for the specific angular momentum L:

$$\left(1 - \frac{2M}{r_p}\right)\left(1 + \frac{L^2}{r_p^2}\right) - \left(1 - \frac{2M}{r_a}\right)\left(1 + \frac{L^2}{r_a^2}\right) = 0 \tag{1}$$

Instead of solving Eq. (1) analytically, we treat it as a root-finding problem of the form $f(L) = 0$. We implement the root-finding numerically using Brent's method (via `scipy.optimize.brenteq`), which combines the robustness of bisection with the speed of the secant method [4].

The algorithm proceeds as follows:

1. Define the objective function $\Delta V(L) = V_{\text{eff}}(r_p; L) - V_{\text{eff}}(r_a; L)$.
2. Identify the interval as $[L_{\min}, L_{\max}]$ where the sign of $\Delta V(L)$ flips.
3. Iteratively refine the estimate for L until convergence is achieved within a tolerance of 10^{-12}.
4. Once L is determined, compute the specific energy E via direct substitution into the potential equation.

With this numerical approach we can initialize for arbitrary eccentric orbits without deriving the analytical solutions for each metric configuration.

3.1 State Vector & Integration

For a particle moving in the Schwarzschild metric, the full position and velocity in 4D spacetime would be:

$$\vec{S}_{\text{full}} = [t, r, \theta, \phi, \dot{t}, \dot{r}, \dot{\theta}, \dot{\phi}]$$

Thanks to symmetry and conservation laws, we were able to restrict the motion to the equatorial plane $\theta = \pi/2, \dot{\theta} = 0$ and find the equations for $\dot{t}, \dot{\phi}$ using E and L. So the variables we need to actually update and solve are:

$$y = \begin{bmatrix} r \\ \dot{r} \end{bmatrix}$$

The solvers we will implement are going to be the Runge-Kutta order 4 (RK4), which is an explicit solver and the Velocity Verlet, which is a symplectic integrator chosen for the geometric preservation properies. After reduction using the constants of motion, the we arrive at a radial Hamiltonian, which is effectively a 1D Hamiltonian system that is separable, and we can use the sympletic integrators cleanly. The Hamiltonian looks like:

$$H = T(\dot{r}) + V_{\text{eff}}(r)$$

and so we expect the velocity verlet to exhibit superior long-term energy stability compared to the RK4.

3.1.1 Runge-Kutta Order 4

Let $\mathbf{y}_n$ represent the state vector at step n (containing position and velocity) and h represent the time step dt.

The system of differential equations is defined by

$$\mathbf{y}' = \mathbf{f}(\tau, \mathbf{y}),$$

where $\mathbf{f}$ corresponds to the `rhs_timelike` function. The step from $\mathbf{y}_n$ to $\mathbf{y}_{n+1}$ is given by:

$$\mathbf{k}_1 = \mathbf{f}(\mathbf{y}_n) \tag{2}$$

$$\mathbf{k}_2 = \mathbf{f}\left(\mathbf{y}_n + \frac{h}{2}\mathbf{k}_1\right) \tag{3}$$

$$\mathbf{k}_3 = \mathbf{f}\left(\mathbf{y}_n + \frac{h}{2}\mathbf{k}_2\right) \tag{4}$$

$$\mathbf{k}_4 = \mathbf{f}(\mathbf{y}_n + h\mathbf{k}_3) \tag{5}$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{h}{6}\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right) \tag{6}$$

The implemented code for the RK4:

```
def rk4_step(state, dt, L, M=1.0):
    k1 = rhs_timelike(state, L, M)
    k2 = rhs_timelike(state + 0.5*dt*k1, L, M)
    k3 = rhs_timelike(state + 0.5*dt*k2, L, M)
    k4 = rhs_timelike(state + dt*k3, L, M)
    return state + (dt/6.0)*(k1 + 2*k2 + 2*k3 + k4)
```

3.1.2 Velocity Verlet

The algorithm:

1. $v_{n+1/2} = v_n + \frac{1}{2}a(r_n)\Delta\tau$
2. $r_{n+1} = r_n + v_{n+1/2}\Delta\tau$
3. $v_{n+1} = v_{n+1/2} + \frac{1}{2}a(r_{n+1})\Delta\tau$
4. $\phi_{n+1} = \phi_n + \dot{\phi}(r_{mid})\Delta\tau$, with $r_{mid} = (r_n + r_{n+1})/2$

and the algorithm implemented in code:

```
# algorithm implementation
def verlet_step(state, dt, L, M=1.0, mode="gr"):
    r, v, phi = state

    # choose acceleration law
    a_fn = accel_gr if mode == "gr" else accel_newton

    a0 = a_fn(r, L, M)

    # half step velocity
    v_half = v + 0.5*a0*dt

    # full step position
    r_new = r + v_half*dt

    # update phi using midpoint radius for better phase accuracy
    r_mid = 0.5*(r + r_new)
    phi_new = phi + (L/(r_mid**2))*dt

    # new acceleration
    a1 = a_fn(r_new, L, M)

    # full step velocity
    v_new = v_half + 0.5*a1*dt

    return np.array([r_new, v_new, phi_new], dtype=float)
```

3.2 Validation and Metrics

To ensure the computation physics aspect is rigorous, we will quantify the error and do an error analysis.

We calculate the deviation of energy, E, and angular momentum, L, over time. For a perfect integrator we expect $\Delta E = 0$. We plot the simulation's $\Delta E(t)$ to demonstrate the expected drift and energy loss with the explicit Runge-Kutta methods compared to the symplectic methods.

An analytical check against the theoretical shift per orbit for the precession of the perihelion is done and is given by (eqn 8 from gr-toolkit [1]):

$$\Delta\phi \approx \frac{6\pi M}{a(1-e^2)}$$

Allowing us to compare the numerically integrated precession angle against this theoretical prediction.

4 Methodology: Simulation Scenarios

We now conduct simulation scenarios. In all cases, we utilize the dimensionless unit system where $GM = c = 1$. The simulations are implemented in Python using the `NumPy`, `SciPy` and `MatplotLib` libraries, utilizing the numerical framework described in Section 3. The full python `.ipynb` notebook is in github.com/AshifulBhuiyan/NumericalGravity

4.1 Scenario A: Trajectory Visualization

The first scenario is designed to qualitatively demonstrate the relativistic effects on orbital geometry. We simulate a test particle in a highly elliptical bound orbit, characterized by a perihelion $r_p = 10M$ and an aphelion $r_a = 50M$. These parameters are chosen to be deep within the gravitational well where the effective potential deviates significantly from the Newtonian $1/r$ form. We integrate the equations of motion for a duration of $\tau = 5000M$, sufficient to capture multiple radial periods. The resulting trajectory is plotted in the $x - y$ plane to visualize the characteristic "rosette" pattern caused by the precession of the perihelion, a phenomenon absent in the closed elliptical orbits of Newtonian gravity.

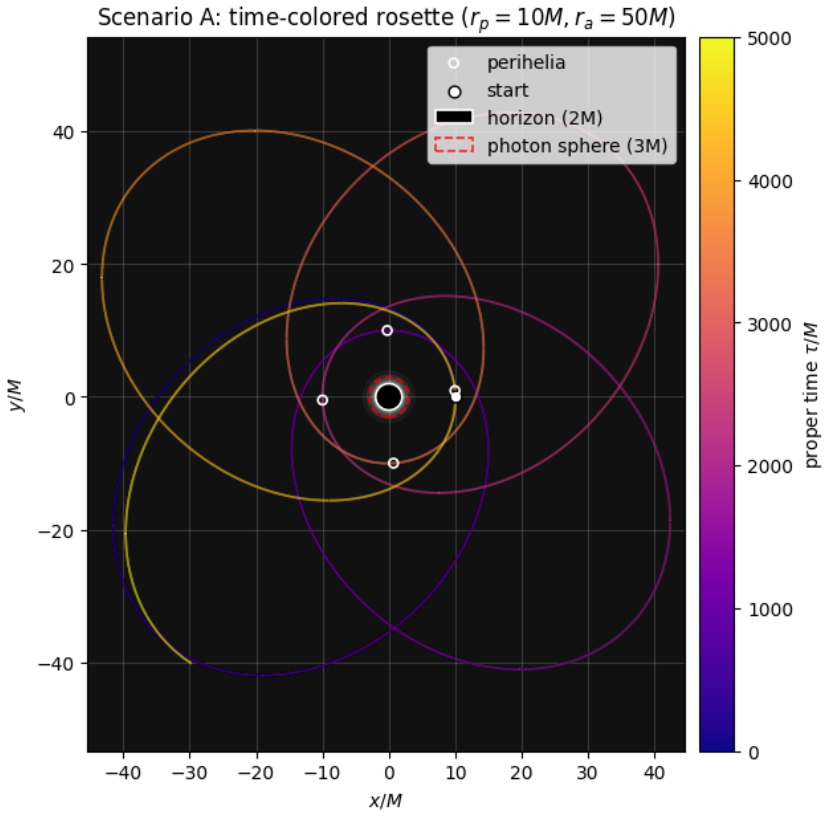
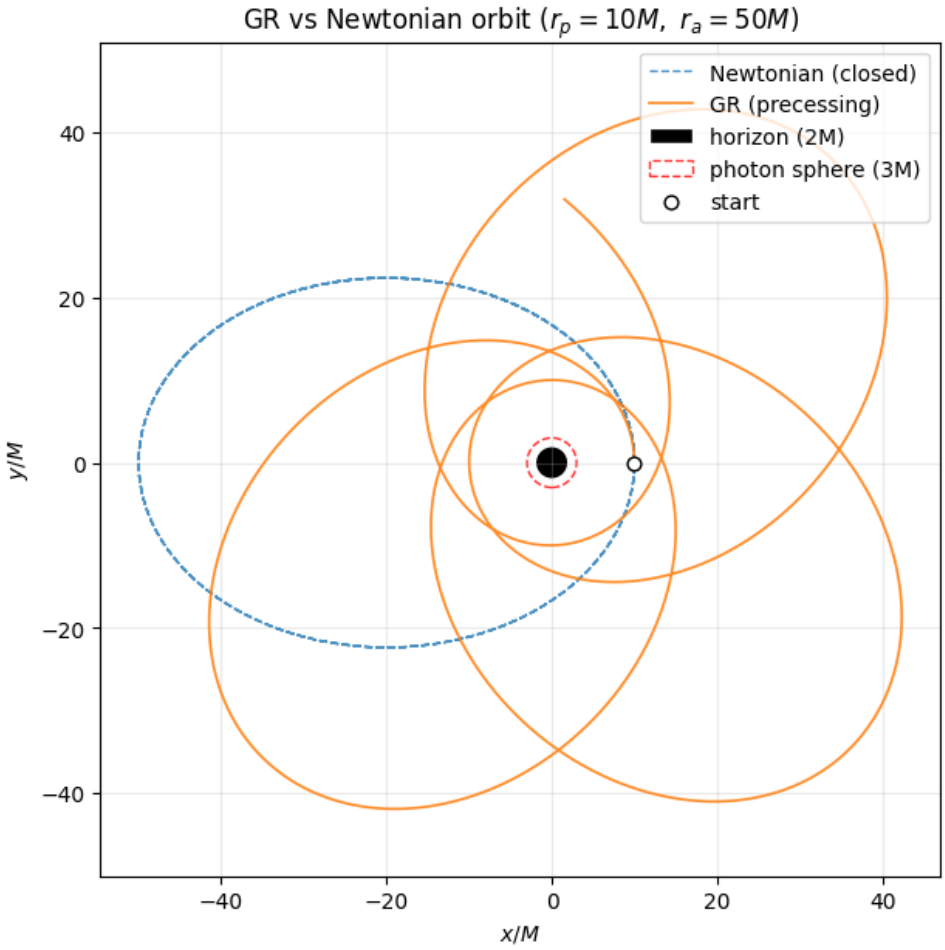


Figure 2: Visualization of a relativistic orbit around a Schwarzschild black hole. The trajectory exhibits perihelion precession, forming a characteristic rosette pattern. Parameters: $r_p = 10M$, $r_a = 50M$, integrated for $\tau = 5000M$.

To show the difference between CLassical Gravity and Gravity explained by GR, we consider a high eccentric orbit with perihelion $r_p = 10M$ and aphelion $r_a = 50M$. We computed both the relativistic and Newtonian case with the velocity Verlet scheme.



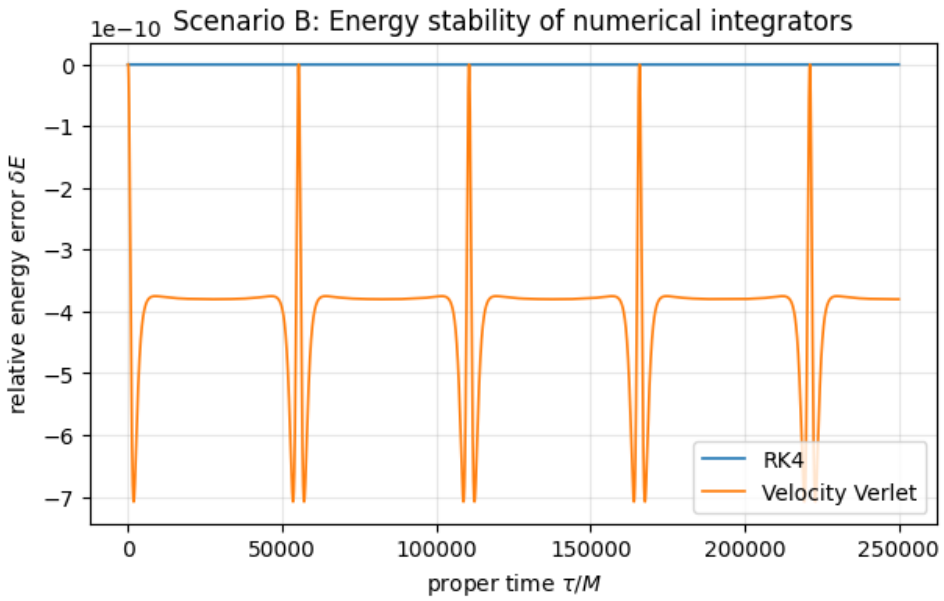
We clearly see that the Newtonian classical case is a closed ellipse whereas our GR case has precession and makes the rosette curves for this particular set of initial conditions.

4.2 Scenario B: Integrator Stability Analysis

To find the long-term stability of the numerical solvers, we simulate a stable circular orbit at radius $r_c = 20M$ for an extended duration of 10^5 time steps with a fixed step size $\Delta\tau = 0.5M$. We track the relative energys error, defined as:

$$\delta E(t) = \left| \frac{E(t) - E_0}{E_0} \right|$$

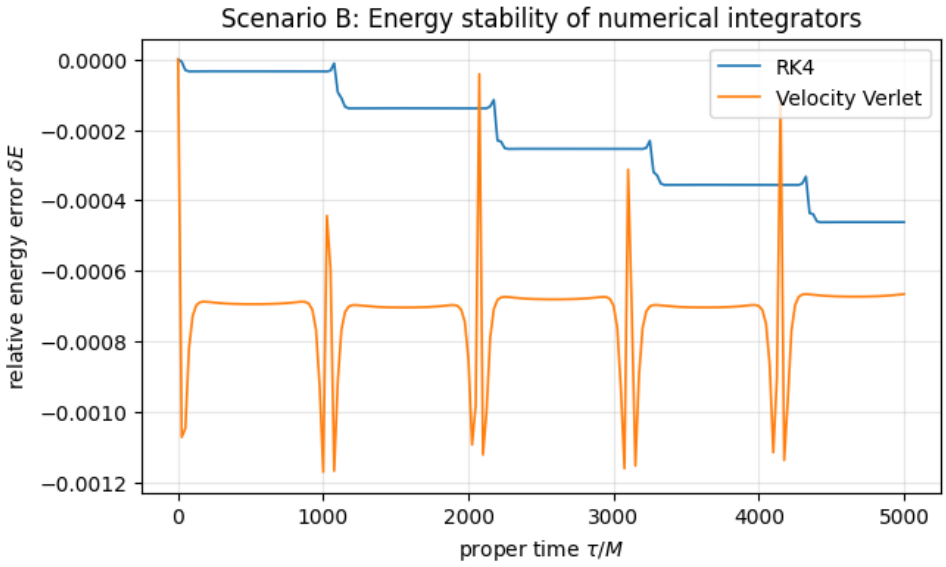
So we find the following:



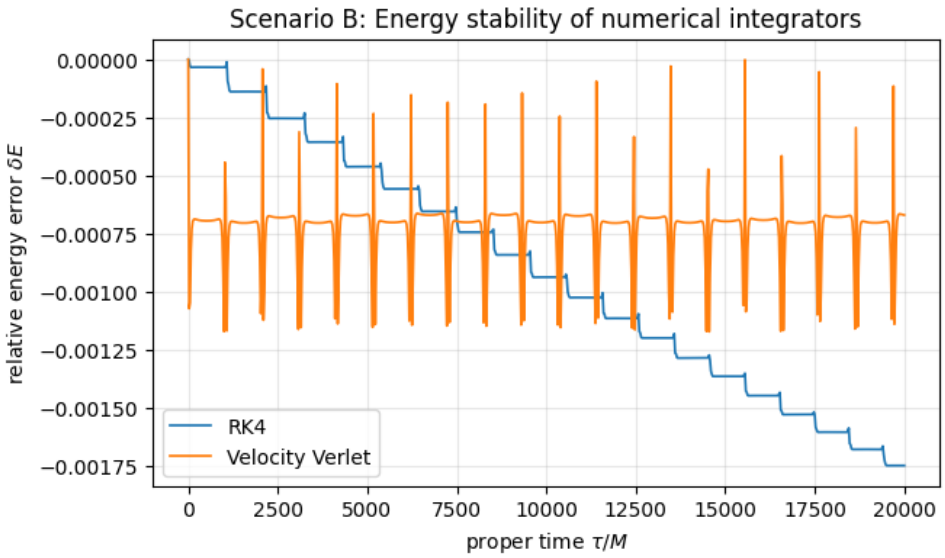
Although we predicted energy drift on the RK4 and for the VV to be conserved (not exactly, but bounded error), we find that the energy for RK4 method did not drift. For the VV method we see that relative error is periodic. This is most likely due to the orbit being at the perihelion where r is smallest, effective potential is therefore the steepest an so acceleration and curvature are the largest at the perihelion. We know that the VV conserves the symplectic structure, but does not perfectly conserve energy pointwise. What we see here are sharp and repeatable deviations at the turning points and no growth in error.

The RK4 method, we got a seemingly horizontal line at 10^{-10} over the entire integration time. This is mostly due to our timestep being small and due to the fact that RK4 is $O(h^4)$.

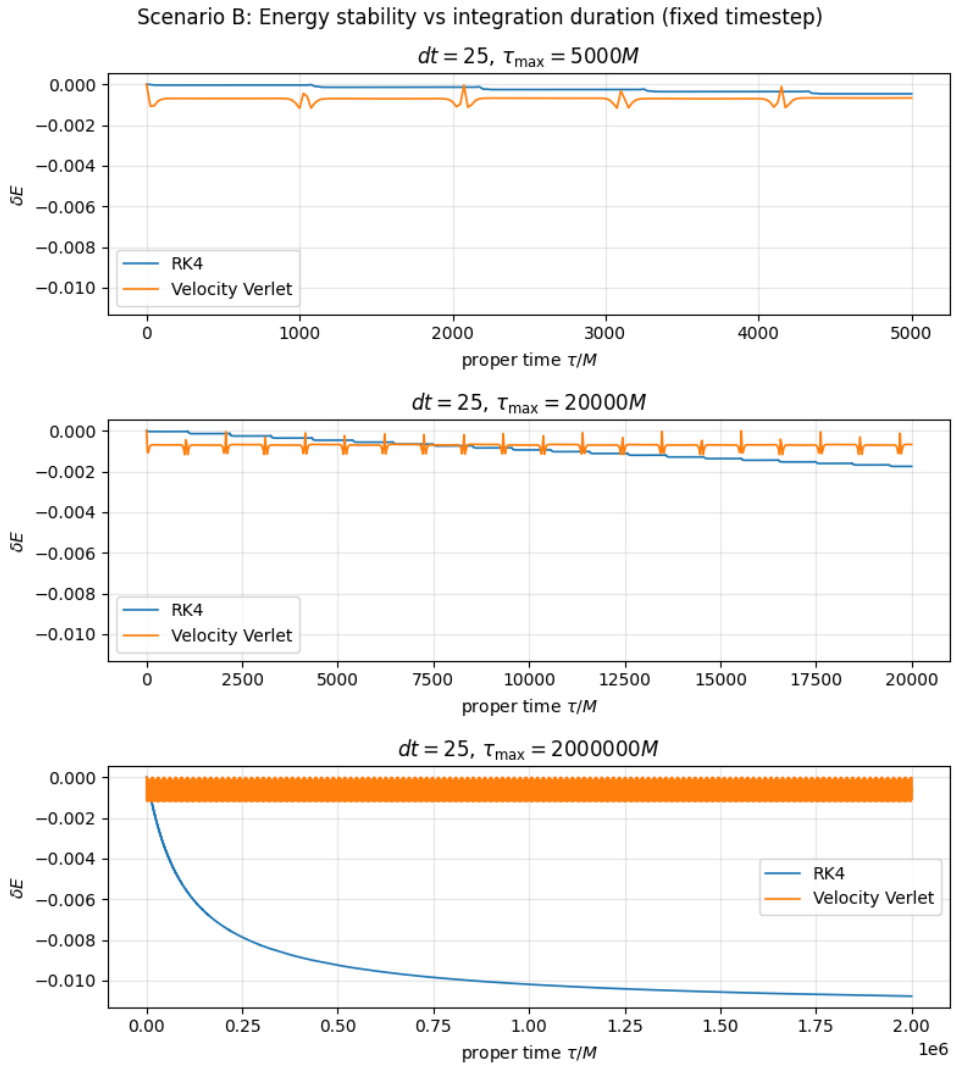
So if we increase the timestep dt to a bigger number like $dt = 25$, we find:



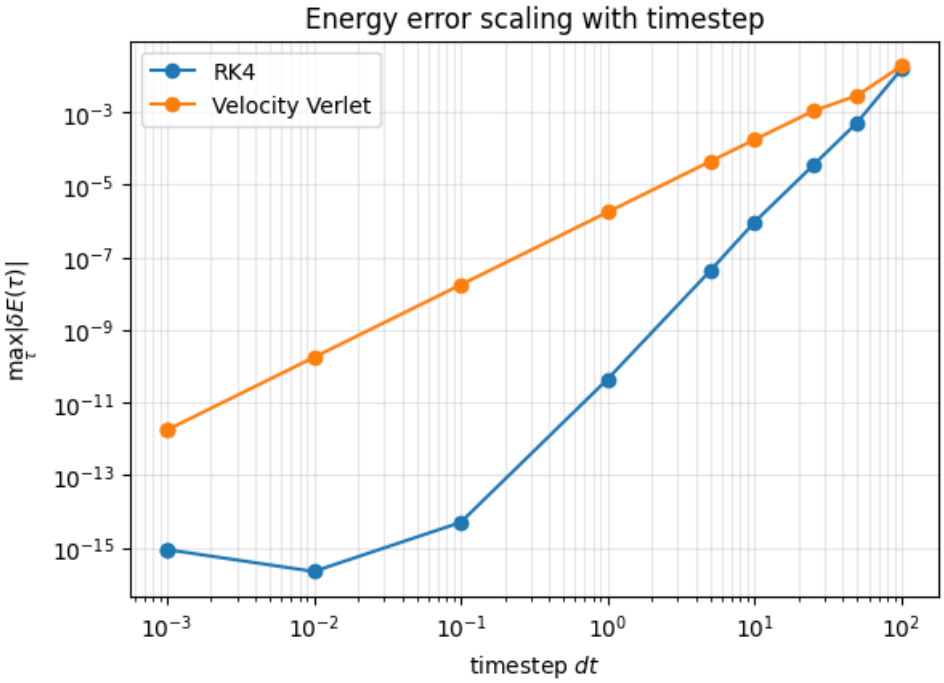
and voila, we find exactly what we predicted. Energy drift in the RK4 method and we see that RK4 does not preserve the symplectic structure, so the energy errors accumulate. If we increase the proper time τ to 20,000,



we see that the error in RK4 exceeds the VV. Going to the extreme limits for τ:



We begin to see that the error in RK4 begins to converge to a value of around $\delta E \approx -0.011$. The previous plots have been for a very large $dt = 25$. When we compare the timesteps dt compared to the error we get the following loglog plot:

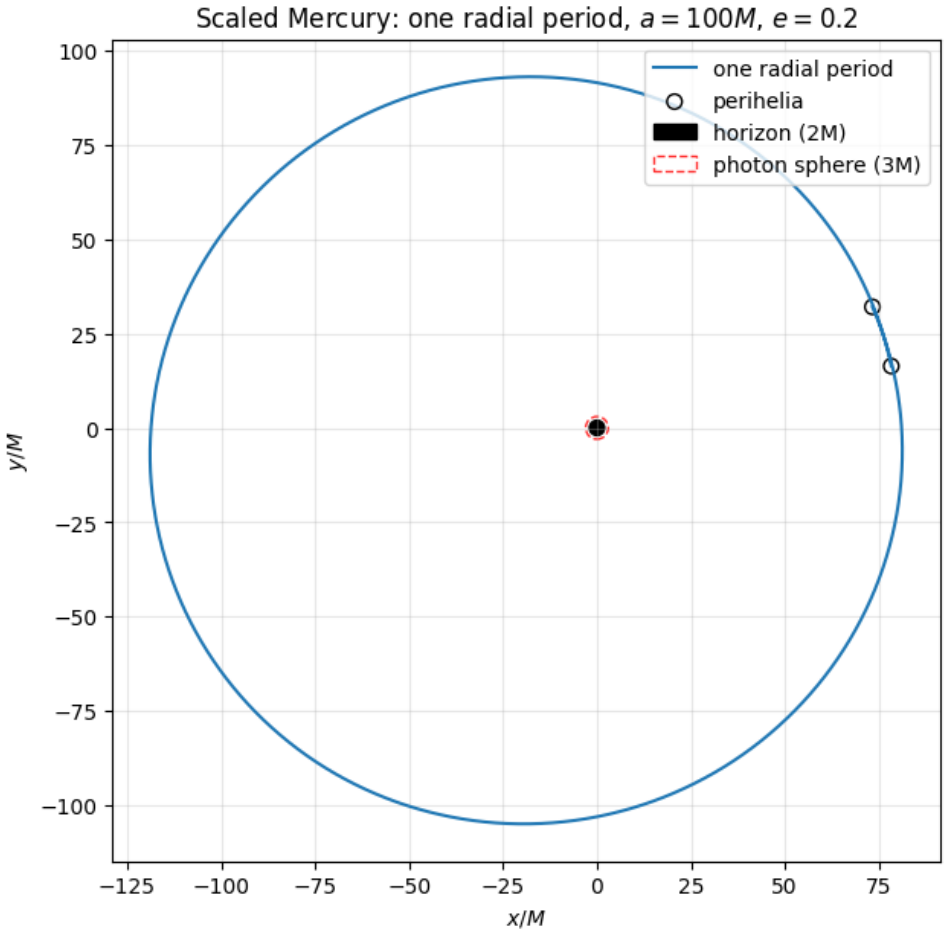


In the loglog scale error graph we compare the error with the timestep dt. We can see that RK4 hits machine precision quickly. Our VV shows a clean power-law scaling with dt. Considering that the VV is a second order method we expected the energy error to scale roughly as dt^2 which we find in our loglog scale error graph.

4.3 Scenario C: Quantitative Validation (Mercury)

Finally, we verify the physical accuracy of our numerical model by reproducing the anomalous precession of the perihelion of Mercury. Because the actual relativistic effect for Mercury is extremely small ($\sim$ 43 arcseconds per century) [2], simulating the real solar system scale is computationally inefficient. Instead, we simulate a "scaled Mercury" model where relativistic effects are measurable over a single orbit but still within the weak-field limit. So we define an orbit with semi-major axis $a = 100M$, which preserves the weak-field allowing us to produce a measurable precession over a single radial period and eccentricity $e = 0.2$ which is similar to Mercury's real precession [2]. We integrate the trajectory for exactly one radial period (from perihelion to perihelion) and calculate the angular shift $\Delta\phi$. This numerical result is compared against the standard first-order theoretical prediction derived from the weak-field approximation [2]:

$$\Delta\phi_{\text{theory}} = \frac{6\pi M}{a(1-e^2)}$$




```
Scaled Mercury parameters:
  M = 1.0
  a = 100.000000 M, e = 0.200000
  rp = 80.000000 M, ra = 120.000000 M
  L  = 9.956878
  E^2 = 0.990103270224

Perihelion precession (one radial period):
  Numerical  = 2.0608585080e-01 rad
  Theory     = 1.9634954085e-01 rad
  Relative error = 4.9587%
  Measured radial period (proper time) approx 6382.860 M
```

The simulation gives a relative error of approx 5% ! It is crucial to note that this deviation does not indicate numerical error, but rather the limitations of the theoretical formula. The expression $\Delta\phi_{\text{theory}}$ is a perturbative expansion only valid in the limit $M/r << 1$. In our scaled model, the graviational potential is strong enough that higher-order relativistic terms which are captured by our exact numerical integration become non-neglible. Consequently, the numerical result provides a more accurate description of the geodesic than the first-order approximation. The fact that $\Delta\phi_{\text{numerical}} > \Delta\phi_{\text{theory}}$ is consistent with the behavior of the Schwarzchild potential, which becomes steeper than the Newtonian $1/r$ approximation as r decreases, inducing the faster rate of precession.

Table 1: Convergence of Numerical Precession to Theoretical Prediction

Semi-major Axis (a)	**Field Strength** (M/a)	$\Delta\phi_{\text{num}}$ (rad)	$\Delta\phi_{\text{theo}}$ (rad)	**Relative Error**
$100M$	1.0×10^{-2}	2.061×10^{-1}	1.963×10^{-1}	**4.96%**
$1,000M$	1.0×10^{-3}	1.973×10^{-2}	1.963×10^{-2}	**0.46%**
$10,000M$	1.0×10^{-4}	1.964×10^{-3}	1.963×10^{-3}	**0.02%**

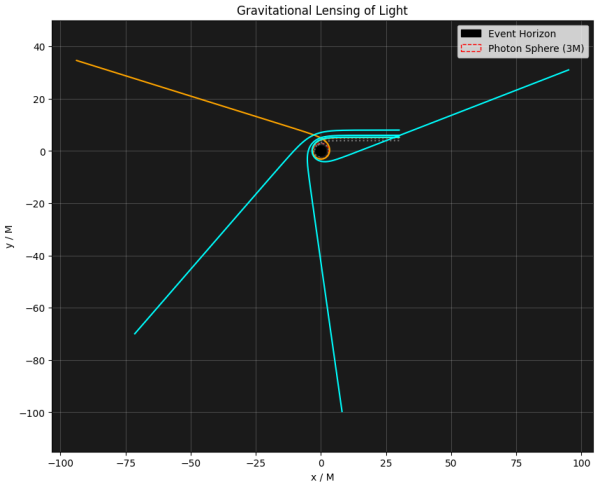
To prove that our model is better than theoretical formula, we set `a = 1000` and `a = 10000`, and we see that our relative error is going down by a factor of 10, as the semi-major axis a is increased by a factor of 10. This linear scaling of relative error with respect to the potential strength (M/a) confirms that the deviation is dominated by second-order terms in the post-Newtonian expansion which the standard theoretical formula ignores. We can see that the relative error scales as $O(M/p)$, exactly as expected for the difference between the full Schwarzchild solution and the first order approximation.

5 Conclusion

In this project, we successfully developed and validated a numerical model for simulating particle geodesics in the Schwarzschild spacetime. By comparing explicit and symplectic integration schemes, we demonstrated that high-order accuracy (as seen in RK4) is not a substitute for geometric conservation laws. The symplectic Velocity Verlet integrator proved superior for orbital dynamics, maintaining bounded energy error over long integration times where standard explicit methods failed. Our simulation accurately reproduced the anomalous precession of the perihelion, a cornerstone test of General Relativity. The 4.96% discrepancy observed in the "Scaled Mercury" scenario highlights the strength of the numerical approach: our simulation captures the full non-linear contributions of the Schwarzschild metric.

Future: Gravitational Lensing with the Null Geodesic Plot

As a final demonstration of the model's versatility, we extended the simulation to massless particles (null geodesics) by setting $\epsilon = 0$. Without going too deep into the physics or anything, as this is a teaser for my next project!



References

[1] gr-orbit-toolkit: A Python-Based Software for Simulating and Visualizing Relativistic Orbits https://www.arxiv.org/pdf/2511.19442

[2] Hartle, J.B.: The Geometry Outside a Spherical Star. In: Gravity: An Introduction to Einstein's General Relativity, pp. 186-218. Cambridge (2021) https://doi.org/10.1017/9781009042604.010

[3] Hobson MP, Efstathiou GP, Lasenby AN. Experimental tests of general relativity. In: General Relativity: An Introduction for Physicists. Cambridge: Cambridge University Press; 2006. p. 230-47 https://www.cambridge.org/core/books/abs/general-relativity/experimental-tests-of-general-relativity/1E1B14CB84B6F4B7042B7D9325C3A3AF

[4] R. L. Burden and J. D. Faires, *Numerical Analysis*, 9th ed. (Brooks/Cole, Boston, 2011).

[5] P. A. Tipler and R. A. Llewellyn, "Perihelion of Mercury's Orbit," Supplement to *Modern Physics*, 6th Ed. (W. H. Freeman, 2012). Available: https://www.macmillanlearning.com/studentresources/college/physics/tiplermodernphysics6e/more_sections/more_chapter_2_1-perihelion_of_mercurys_orbit.pdf