ASTR 78100 - Computational Astrophysics

Final Project Report

Binary Black Hole Inspiral Simulation

Post-Newtonian Dynamics with Gravitational Wave Emission

John Meftah

Professor: Dr. Ariyeh Maller

December 2025

I Introduction

This project simulates two black holes spiraling into each other until they merge. The code solves the relativistic two-body problem using Post-Newtonian (PN) dynamics, which is basically an expansion of general relativity in powers of v/c.

The simulation gives us the orbital trajectory, gravitational wave strain, and frequency evolution (the "chirp" that LIGO detects).

II Motivation

Back in September 2015, LIGO detected gravitational waves from two black holes merging. This event, called GW150914, was huge because the signal matched exactly what general relativity predicts: two massive objects spiraling together, speeding up as they get closer, and eventually colliding.

To model something like this, we need to solve the two-body problem in general relativity. The exact solution requires numerical relativity, which means expensive supercomputer simulations. But for the inspiral phase (before the final plunge), we can use Post-Newtonian dynamics as a good approximation.

Here's the key physics: gravitational waves carry away energy from the system, so the orbit gets smaller over time. This is what the 2.5PN radiation reaction term captures.

III The Setup

We work in the center of mass frame and track the relative separation vector $\vec{r} = \vec{r}_1 - \vec{r}_2$ between the two black holes.

The equations of motion are:

$$\frac{d\vec{r}}{dt} = \vec{v}$$
$$\frac{d\vec{v}}{dt} = \vec{a}_{\text{PN}}$$

In the code, this looks like:

```
r_vec = y[:3]
v_vec = y[3:]
return np.concatenate([v_vec, a_vec])
```

Once we have the relative vector, we can get the individual black hole positions using:

$$\vec{r}_1 = \frac{m_2}{M}\vec{r}$$
$$\vec{r}_2 = -\frac{m_1}{M}\vec{r}$$

```
r1 = (self.m2 / self.M) * r_vec
r2 = -(self.m1 / self.M) * r_vec
```

IV Numerical Methods

To actually solve these coupled differential equations, we use `solve_ivp` from SciPy with the **DOP853** integrator. This is an 8th-order Runge-Kutta method with adaptive step size control.

IV.A Why DOP853?

DOP853 (also called Dormand-Prince) is a high-order explicit Runge-Kutta method, and it works really well for this problem for a few reasons:

1. **High accuracy**: The 8th-order local truncation error is really important here because the inspiral dynamics are super sensitive to errors. The gravitational wave frequency increases rapidly near the merger, and small errors add up fast.
2. **Adaptive timestepping**: The orbital period changes drastically during the inspiral (going from milliseconds to microseconds). Adaptive steps let the integrator handle this automatically without us having to mess with the timestep manually.
3. **Smooth solutions**: Post-Newtonian dynamics produce smooth trajectories. High-order methods are really good at handling this kind of smooth behavior, much better than something like RK4.
4. **Dense output**: The method gives us interpolation between timesteps, so we can extract output at uniform time points without having to re-integrate everything.

IV.B Integration Details

The integrator solves this 6-dimensional system:

$$\frac{d}{dt}\begin{pmatrix} x \\ y \\ z \\ v_x \\ v_y \\ v_z \end{pmatrix} = \begin{pmatrix} v_x \\ v_y \\ v_z \\ a_x \\ a_y \\ a_z \end{pmatrix}$$

where $\vec{a}$ comes from the Post-Newtonian acceleration formulas.

Here's how it looks in code:

```
solution = solve_ivp(
    self._equations_of_motion,
    [0, t_max],
    y0,
    method='DOP853',
    events=self._merger_event,
    dense_output=True,
    rtol=1e-10,
    atol=1e-12
)
```

IV.C Tolerances

We use very tight tolerances to make sure the results are accurate:

- **Relative tolerance**: `rtol` $= 10^{-10}$
- **Absolute tolerance**: `atol` $= 10^{-12}$

These make sure the local error per step satisfies:

$$\text{error}_i \lesssim \texttt{atol} + \texttt{rtol} \times |y_i|$$

Why so tight? A few reasons:

1. Energy and angular momentum are only approximately conserved because the radiation reaction term is dissipative
2. The gravitational wave phase accumulates over thousands of orbits, so small phase errors compound over time
3. Near the merger, velocities get up to around 0.3c, where the Post-Newtonian approximation is already being pushed to its limits

IV.D Event Detection

The integrator uses event detection to automatically stop when we hit the ISCO (Innermost Stable Circular Orbit):

$$r_{\text{ISCO}} = \frac{6GM}{c^2}$$

```
def _merger_event(self, t, y):
    return np.linalg.norm(y[:3]) - self.r_isco

_merger_event.terminal = True
_merger_event.direction = -1
```

This stops the integration when the separation reaches ISCO from above (that's what direction $= -1$ means). This prevents the code from going into the regime where Post-Newtonian theory breaks down.

IV.E Output Sampling

After the integration is done, we use the dense output interpolant to sample the solution at $N = 10{,}000$ uniform time points:

```
t = np.linspace(0, t_merger * 0.9999, n_output_points)
y = solution.sol(t)
```

This gives us smooth, evenly-spaced data that's good for visualization and computing the waveform. Otherwise we would be stuck with the irregular timesteps that come out of the adaptive integrator.

V Mass Parameters

There are a few important mass combinations that show up everywhere in the equations:

$$\begin{aligned}
M &= m_1 + m_2 && \text{(total mass)} \\
\mu &= \frac{m_1 m_2}{M} && \text{(reduced mass)} \\
\eta &= \frac{\mu}{M} = \frac{m_1 m_2}{(m_1 + m_2)^2} && \text{(symmetric mass ratio, } 0 < \eta \leq 0.25\text{)} \\
\gamma &= \frac{GM}{rc^2} && \text{(PN expansion parameter)}
\end{aligned}$$

In code:

```
self.M = self.m1 + self.m2
self.mu = self.m1 * self.m2 / self.M
self.eta = self.mu / self.M
gamma = G_CONST * M / (r * C**2)
```

VI Post-Newtonian Acceleration

The total acceleration builds up order by order in the Post-Newtonian expansion:

$$\vec{a} = \vec{a}_{0\mathrm{PN}} + \vec{a}_{1\mathrm{PN}} + \vec{a}_{2\mathrm{PN}} + \vec{a}_{2.5\mathrm{PN}}$$

```
return a_newton + a_1pn + a_2pn + a_rad_reaction
```

VI.A 0PN: Newtonian

This is just regular Newtonian gravity:

$$\vec{a}_{\mathrm{N}} = -\frac{GM}{r^2}\hat{n}$$

```
a_newton = -G_CONST * M / r**2 * n_hat
```

If we only include this term, we get Kepler orbits. No inspiral at all.

VI.B 1PN: First Relativistic Correction

$$\vec{a}_{1\mathrm{PN}} = \frac{GM}{r^2}\left[A_r^{(1)}\hat{n} + A_v^{(1)}\frac{\vec{v}}{c}\right]$$

where:

$$A_r^{(1)} = (1+3\eta)\frac{v^2}{c^2} - 2(2+\eta)\gamma - \frac{3}{2}\eta\frac{(\hat{n}\cdot\vec{v})^2}{c^2}$$
$$A_v^{(1)} = 2(2-\eta)\frac{\hat{n}\cdot\vec{v}}{c}$$

```
v_c = v / C
n_dot_v = np.dot(n_hat, v_vec) / C

A1_r = (1 + 3*eta) * v_c**2 - 2*(2 + eta) * gamma - 1.5 * eta * n_dot_v
    **2
A1_v = 2 * (2 - eta) * n_dot_v

a_1pn = (G_CONST * M / r**2) * (A1_r * n_hat + A1_v * v_vec / C)
```

This term adds perihelion precession. It's the same kind of physics that explained why Mercury's orbit was weird.

Reference: Kidder, Phys. Rev. D 52, 821 (1995)

VI.C 2PN: Higher-Order GR

$$\vec{a}_{2\mathrm{PN}} = \frac{GM}{r^2}A_r^{(2)}\hat{n}$$

where:

$$A_r^{(2)} = \frac{3}{4}(12+29\eta)\gamma^2 + \eta(3-4\eta)\frac{v^4}{c^4} - \frac{1}{2}\eta(13-4\eta)\gamma\frac{v^2}{c^2}$$

```
A2_r = (
    0.75 * (12 + 29*eta) * gamma**2
    + eta * (3 - 4*eta) * v_c**4
    - 0.5 * eta * (13 - 4*eta) * gamma * v_c**2
)

a_2pn = (G_CONST * M / r**2) * A2_r * n_hat
```

This improves the accuracy when the velocities get really relativistic near the merger. But it's still a conservative force, so there's no energy loss yet.

Reference: Blanchet, Living Rev. Relativ. 17, 2 (2014)

VI.D 2.5PN: Radiation Reaction

This is the most important term:

$$\boxed{\vec{a}_{2.5\text{PN}} = -\frac{32}{5}\eta\frac{(GM)^3}{c^5r^4}\vec{v}}$$

```
a_rad_reaction = -(32/5) * eta * (G_CONST * M)**3 / (C**5 * r**4) *
    v_vec
```

This is called the Burke-Thorne radiation reaction force, and it's the first dissipative term in the expansion. Energy gets carried away by gravitational waves, which makes the orbit shrink.

This is why the black holes actually spiral in instead of just orbiting forever.

Reference: Burke & Thorne, in Relativity (1970)

VII Summary of PN Terms

Here's a quick summary of what each order does:

Order	Physics	Effect
0PN	Newtonian gravity	Stable orbits
1PN	First GR correction	Precession
2PN	Higher-order GR	Better accuracy
2.5PN	GW energy loss	**Inspiral**

VIII Gravitational Wave Strain

The strain amplitude (this is what LIGO actually measures):

$$h_0 = \frac{4G^2m_1m_2}{c^4Dr}$$

```
h0 = 4 * G_CONST**2 * m1 * m2 / (C**4 * distance * r)
```

There are two polarizations of gravitational waves:

$$h_+ = \frac{1+\cos^2\iota}{2}\,h_0\cos(2\Phi)$$
$$h_\times = \cos(\iota)\,h_0\sin(2\Phi)$$

```
cos_i = np.cos(inclination)
plus_factor = (1 + cos_i**2) / 2
cross_factor = cos_i

gw_phase = 2 * phase

h_plus = plus_factor * h0 * np.cos(gw_phase)
h_cross = cross_factor * h0 * np.sin(gw_phase)
```

Important note: the gravitational wave phase is twice the orbital phase. This is because gravitational waves are quadrupole radiation.

Reference: Maggiore, Gravitational Waves: Theory and Experiments (2008)

IX GW Frequency

The gravitational wave frequency is twice the orbital frequency:

$$f_{\text{GW}} = 2f_{\text{orbital}} = \frac{1}{\pi}\sqrt{\frac{GM}{r^3}}$$

```
f_orbital = np.sqrt(G_CONST * M / r**3) / (2 * np.pi)
f_gw = 2 * f_orbital
```

X Stopping Conditions

The simulation stops at the Innermost Stable Circular Orbit (ISCO):

$$r_{\text{ISCO}} = \frac{6GM}{c^2}$$

```
r_isco = 6 * G_CONST * M / C**2
```

Below this radius, the Post-Newtonian approximation breaks down completely, so we would need full numerical relativity to continue the sim.

XI Validation

To check that the code is working correctly, we compare it against the Peters (1964) formula. Peters derived an analytical expression for the inspiral rate:

$$\frac{dr}{dt} = -\frac{64}{5}\frac{G^3}{c^5}\frac{m_1 m_2 M}{r^3}$$

```
dr_dt_peters = -(64/5) * (G_CONST**3 / C**5) * (m1 * m2 * M) /
    separation**3
```

Our sim computes this numerically through the 2.5PN term. If the envelope of our numerical curve matches Peters' prediction, then we know the radiation reaction implementation is correct.

Reference: Peters, Phys. Rev. 136, B1224 (1964)

XII References

1. Peters, P.C. & Mathews, J., *Phys. Rev.* **131**, 435 (1963)
2. Peters, P.C., *Phys. Rev.* **136**, B1224 (1964)
3. Burke, W.L. & Thorne, K.S., in *Relativity* (1970)
4. Kidder, L.E., *Phys. Rev. D* **52**, 821 (1995)
5. Blanchet, L., *Living Rev. Relativ.* **17**, 2 (2014)
6. Maggiore, M., *Gravitational Waves: Theory and Experiments* (2008)
7. Hairer, E., Nørsett, S.P., Wanner, G., *Solving Ordinary Differential Equations I* (2008)