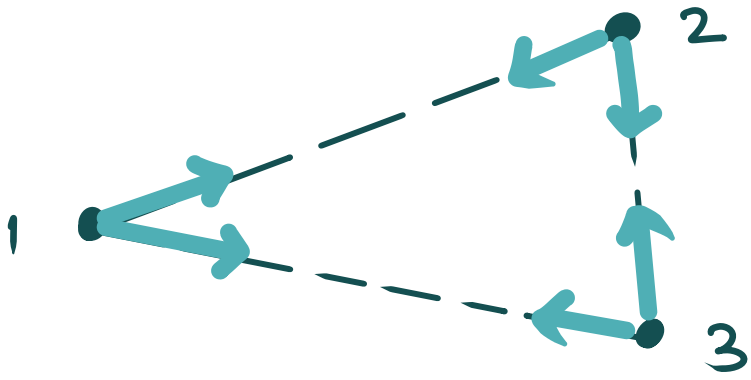
# PROJECT 1: Three-Body Orbits → xy-Plane → 3D-Plane

↳ Lyapunov 3B code.

$$F = \frac{G m_1 m_2}{r^2}$$




Initial Position & Velocity
↓
Forces
↓
Update velocities
↓
Update positions

## Input file: dat.pyt

| (TR) | (ss) | (e) | | | | | |
|---|---|---|---|---|---|---|---|
| 20.0 | 0.1 | 1.e-8 | | | | | |
| 1.0 | -0.5 | 0.0 | 0.0 | 0.0 | -0.8 | 0.0 | Bodies |
| 1.0 | 0.5 | 0.0 | 0.0 | 0.0 | 0.8 | 0.0 | |
| 0.001 | 0.0 | 1.0 | 0.0 | -1.0 | 0.0 | 0.0 | |

Run simulation time ($T_R$) = 20.0
Save a snapshot (ss) = 0.1
Errors (e) should be < $10^{-8}$

Body for each mass:

mass | x-position | y-position | z-position | x-velocity | y-velocity | z-velocity

Eg:

Body 1:

$m_1 = 1$

$x_1 = (-0.5, 0, 0)$

$v_1 = (0, -0.8, 0)$

# The Simulation Process?

1.) Centre of mass Frame: → The code shifts everything so the "center" stays at (0,0,0).

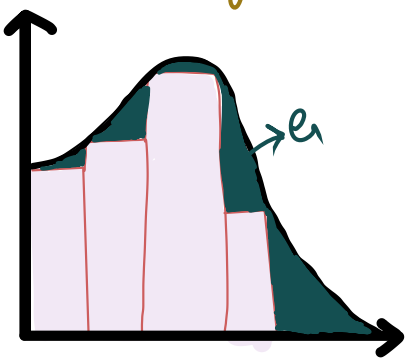
2.) Time Problem: → Uses "fake time" that automatically slows down when bodies get close.

3.) Integration Method (Bulirsch-Stoer): → calculate where bodies will be at 2-small steps

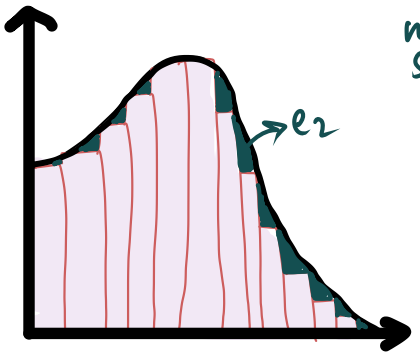
Again at 4 smaller steps → Again using 6-smaller steps → Compare all the answers

- → Agree closely → correct
- → Disagree → more smaller steps

4-steps:




12-steps




$\text{no. of steps} \propto \frac{1}{\text{error}}$

$e_1 > e_2$

($e_i$ : error)

4.) The Leapfrog Method (The Core):

Current state: Known position & velocity → STEP 1 → Move positions forward a little using current vel.

→ STEP 2 → Calculate new forces

→ STEP 3 → Update velocities

→ STEP 4 → Move positions forward again with new velocity

* This preserves energy better than updating both at once

## 5) Checking for Errors: → Code constantly monitors

Angular Momentum ⇓ Rotation should be conserved

Energy ⇓ Should stay constant

Output: (Err) file

T: 0.0383 2.71E-16 8.46E-19 -7.055E-02 0.2292

- At time
- Energy error
- Angular momentum error
- Other diagnostic numbers

## 6) Lyapunov Exponent (Measuring Chaos):

Track a "shadow orbit" that starts infinitesimally close to the real orbit

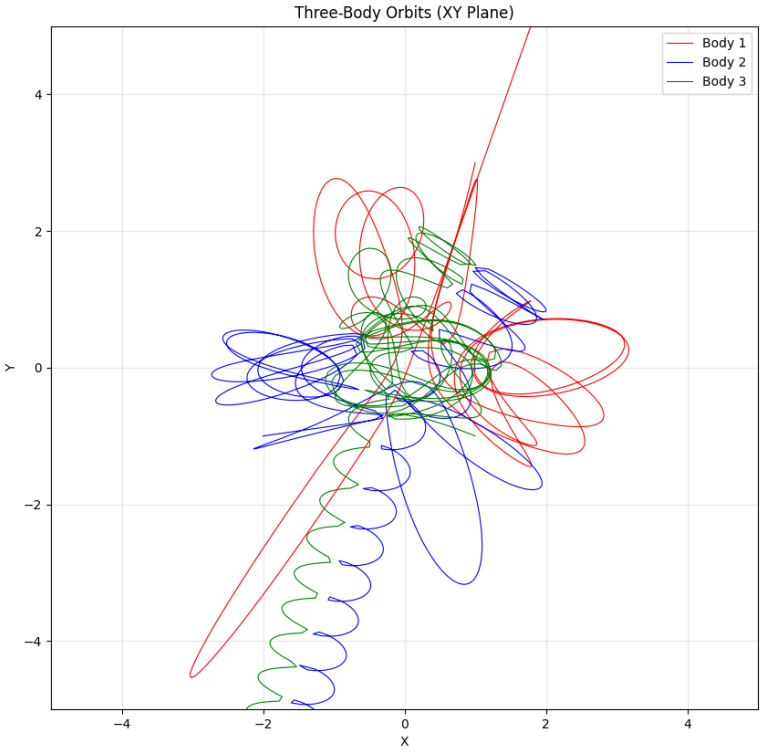
→ Measure how quickly shadow orbit diverges from real orbit

→ If it diverges exponentially fast ≡ CHAOTIC SYSTEM

## OUTPUT: → xyz.Ly File:

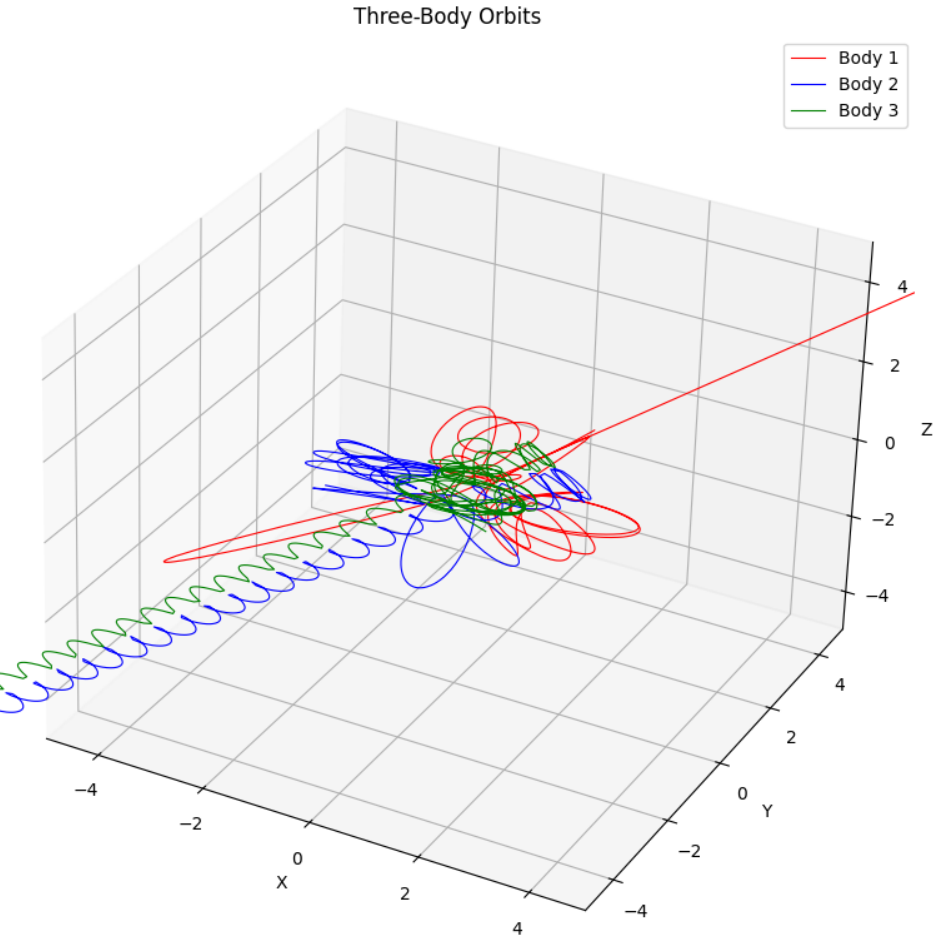
time x1 y1 z1 vx1 vy1 vz1 x2 y2 z2 vx2 vy2 vz2 x3 y3 v3 vx3 vy3 vz3

- Plots ⇓ 2D and 3D Plots




# 2D Plot of 3-Body Orbit

In X-Y Plane only




# 3D Plot of 3-Body Orbit

Plotted in XYZ