Planetary Systems

In this activity we will use the Verlet method, which we have seen already for solving the wave equation, to study the dynamics of systems of massive objects that interact with eachother gravitationally.

One new challenge that we will need to face, and overcome, is that we will work in space, with at least two spatial coordinates and velocities. Note, however, that the Verlet method only needs an initial velocity but does not solve esplicitly for the velocity of the objects as a function of time.

The Verlet method for Newtonian forces among point-like objects is quite simpler than in the case of the wave equation, which governs the motion of an extended continuous object (a string). Let us consider a system of equations:

$$m_\alpha \frac{dx_{\alpha,i}}{dt} = \sum_{\beta\neq\alpha} F_{\alpha\beta,i} \tag{1}$$

where $i = 1, 2, 3$ is one of the three Cartesian coordinates and α, β are indices that identify the varions massive objects. $F_{\alpha\beta,i}$ is the i-th component of the force that the boject β exerts on the object α.

If the initial position $x_{\alpha,i}(0)$ and velocity $\dot{x}_{\alpha,i}(0)$ are known, the Verlet method states that:

$$\begin{aligned} x_{\alpha,i}(\Delta t) &= x_{\alpha,i}(0) + \dot{x}_{\alpha,i}(0)\,\Delta t + \frac{1}{2} a_{\alpha,i}(0)\,\Delta t^2 \\ x_{\alpha,i}(t_{i+1}) &= 2x_{\alpha,i}(t_i) - x_{\alpha,i}(t_{i-1}) + a_{\alpha,i}(t_i)\,\Delta t^2 \end{aligned} \tag{2}$$

where $a_{\alpha,i}(t)$ is the i-th component of accelration of the object α at the previous time step. An important constraint of this method is that the force cannot depend on velocity, but only on the positions of all the objects.

We will start with a simplified case and increase complexity as we proceed.

1. Set up the code to work in 2-dimensional space with two massive objects only.
2. Write a python function that returns the acceleration vectors given all the objects' positions
3. Set the initial position and velocities of all the objects
4. Evolve the system using the Verlet method
5. Make a static plot of the orbits of all objects
6. Make an animation of the system dynamics

Extra fun. When your code is working properly, consider the following options to make it more complex:

- Change the value of the time step Δt. Does the dynamics of the system change significantly? Then you were sing too big a time step.
- Can you set up the system so that the center of mass is static?

- What happens if you change the initial conditions?
- What happens if you change the masses of the objects?
- Can you set up a circular orbit?
- Experiment with non-Newtonian gravity, e.g., by making the force proportional to a different power of the distance.
- Try re-writing the code to work in 3-dimensional space
- Try adding a third object to the system
- Try to set up a system with a central star, a planet, and a moon moving around the planet
- Try to obtain a slingshot effect
- Add as many objects as you like