PEP336 Project 1: Simulation of the orbital motion.

(Due Tuesday February 6.)

Part 1. [Euler method.]

In this project you will learn basic numerical schemes for integrating differential equations on the example of the orbital motion. We will start with the simplest method, which simply transcribes the second Newton's law into the equations without any numericam optimization - namely the Euler method. It is important that you move on with the project, and not get stuck for long time periods. If you are at any step unsure how to proceed, or if you are simply confused, stop by my office and I'll help you to get going.

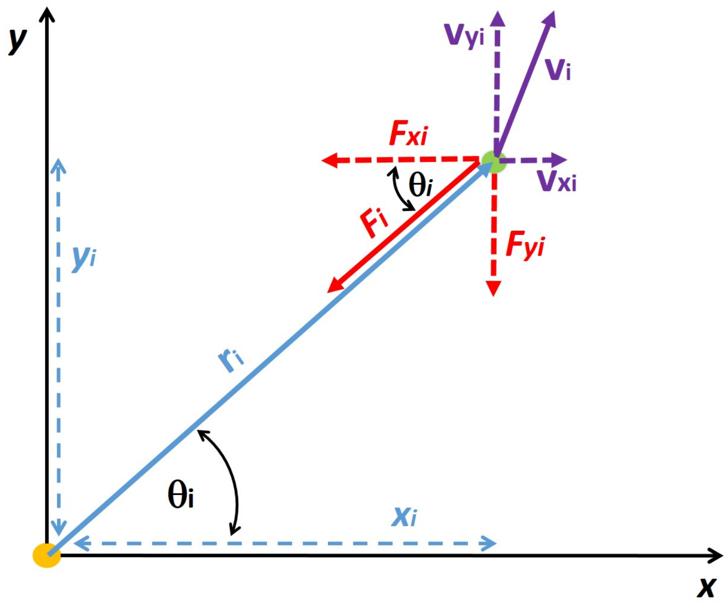
We will consider a planet of mass m orbiting the star of mass M (to be specific say it's the Earth and the Sun). For the time being we will work in the limit of small planetary mass $m \ll M$, and therefore we will consider the Sun to be immovable - the Sun attracts the planet with the gravitational force, but we will neglect the forces acting on the Sun. It is therefore convenient to place the origin of the reference frame at the position of the Sun. Also for the time being, we will consider a two-dimensional problem (we will easily generalize this to three-dimensional motion if needed). The plane of motion is determined by the position and velocity vectors of the planet $\vec{r} = (x, y)$ and $\vec{v} = (v_x, v_y)$, which are given as the initial conditions.

The essential simplification is that the time is discretized into small increments ('time steps') Δt, that have to be chosen so that they are much smaller then the period of the orbital motion $\Delta t \ll T$. Suppose that all time steps are equal.

At the beginning of the i-th step the position of the planet is $\vec{r}_i$ and its velocity is $\vec{v}_i$. At that time the force acting on the planet is $\vec{F}_i = -\left(GMm/r_i^2\right)\hat{r}_i$, where $\hat{r}_i$ is the unit vector in the radial direction. The acceleration is $\vec{a}_i = \vec{F}_i/m = -\left(GM/r_i^2\right)\hat{r}_i$. We assume that the time step Δt is so small that during that time any changes in force and acceleration are negligible, so we can apply the equations of motion with constant acceleration. Then we can easily find that at the end of step i, which is the beginning of time step $(i+1)$, the velocity and position are

$$\begin{aligned}\vec{r}_{i+1} &= \vec{r}_i + \vec{v}_i \Delta t + \frac{1}{2}\vec{a}_i \Delta t^2 \\ \vec{v}_{i+1} &= \vec{v}_i + \vec{a}_i \Delta t \ .\end{aligned}$$

Now that we have the velocity and position vectors at the beginning of the $(i+1)$-th time step, we calculate new force and repeat the above steps.



For the practical implementation we must work with x and y components of the above equations. Geometry of the problem is shown in the image above: angle $\theta_i$is measured from the x-axis to the position vector in i-th step; $\cos\theta_i = x_i/r_i$, $\sin\theta_i = y_i/r_i$ and $r_i = \sqrt{x_i^2 + y_i^2}$.

$$\vec{a}_i = -\frac{GM}{r_i^2}\hat{r}_i \ \Leftrightarrow \ \begin{cases} a_{x,i} = -\left(GM/r_i^2\right)\cos\theta_i = -GM\left(x_i/r_i^3\right) \\ a_{y,i} = -\left(GM/r_i^2\right)\sin\theta_i = -GM\left(y_i/r_i^3\right) \end{cases}$$

$$\vec{r}_{i+1} = \vec{r}_i + \vec{v}_i\Delta t + \frac{1}{2}\vec{a}_i\Delta t^2 \ \Leftrightarrow \ \begin{cases} x_{i+1} = x_i + v_{x,i}\Delta t + \frac{1}{2}a_{x,i}\Delta t^2 \\ y_{i+1} = y_i + v_{y,i}\Delta t + \frac{1}{2}a_{y,i}\Delta t^2 \end{cases}$$

$$\vec{v}_{i+1} = \vec{v}_i + \vec{a}_i\Delta t \ \Leftrightarrow \ \begin{cases} v_{x,i+1} = v_{x,i} + a_{x,i}\Delta t \\ v_{y,i+1} = v_{y,i} + a_{y,i}\Delta t \end{cases}$$

The best way to test the model is to initially set the Earth say along the x-axis at the initial distance of 1 AU, and the initial velocity along the y-axis set to be equal to the orbital speed of the Earth (which we found to be 29.8 km/s). For the initial conditions ('zeroth step') $x_0 = 1$ AU, $y_0 = 0$, $v_{x,0} = 0$ and $v_{y,0} = 29.8$ km/s we know that the orbit should be circular. We will not study the numerical analysis of the approximation schemes, so you may need to play a little with your code to get a reasonable time step Δt. Once you get the circular orbit, try increasing the initial speed to $v_o = 35$ km/s. In this case you should get an elliptic orbit. Plot the resulting orbits for this part of the project.

Note that Euler's method has a very poor accuracy, and you will typically not get closed orbits. Don't force the time steps to be super-small (like 1 second), it's OK to get the rosetta-like orbits at this point. In the next step we will increase accuracy by working with more sophisticated integration methods.

To ensure that your code works correctly, you can also calculate/plot the total energy of the planet. For example, you can calculate the energy per unit mass at i-th step as

$$\varepsilon = E/m = \frac{1}{2}v_i^2 + \left(-\frac{GM}{r_i}\right) .$$

Again, the energy may not be exactly constant because of the said issues with accuracy, but it should not vary wildly. We will address those issues in Part 2 of this project.

You are free to use any programming language/package you are comfortable with: Python, C++, Fortran, Mathematica, MatLab are all fine. You need not submit anything yet. Make sure that you have finished Step 1 by the due date, so you can continue working on the Runge-Kutta approximation scheme in Part 2 of the project.