Physics 305 – Computational Physics, Fall 2022
Term Project
Full project submission Due Date: Tuesday December 14, 5pm

The program in your term project can be either submitted as a python program or ipython notebook, where the latter is preferred. The program, an explanation of what the program does, along with answers to **all** questions asked should be uploaded to d2l.

You are expected to write a term paper (in word or Latex) on your project that discusses the problem you are trying to solve, the basic equations that govern the problem, includes plots that show the solutions, and describes the solution and the numerical method involved. In addition, you must demonstrate that your solution is correct by showing that the code converges at the expected order. If your code does not converge at the expected order you should try to identify potential reasons for why this is the case. You are expected to work on your term project by yourself..

Your term project will receive full credit **only** if: (a) the program runs successfully without errors using python 3, (b) the programs have explanatory comments and variable names that identify with the problem equations you are trying to solve, (c) give the correct output, and (d) demonstrate the validity of the solution through convergence plots. No credit will be given to late term projects.

The term paper is as important as the code (50% of the term project credit will go to the code and the other 50% to the paper). Answers to the questions and analysis requested below should be elaborated in the report. Plots should be clearly labeled and be properly described in the report, and not just shown. You will need to explain what each and every plot demonstrates. A polished paper written in word or LaTex (preferred, e.g. please try overleaf) is expected to get full credit.

Note: Before you present results from numerical integrations that answer the questions in the project, it is critical to *first* perform the convergence tests for one case, and to estimate errors. This will tell you how small a step size is necessary for accurate solutions. Only after errors are estimated, does it make sense to run your code for producing results that help you learn more about the system you study.

I. ORBITAL DYNAMICS IN THE SOLAR SYSTEM

Consider the Earth and another planet (which will be Jupiter) as they circle around the Sun. The goal of the project is to determine the orbit of a space ship from Earth to the Planet. In this problem we will neglect the gravity of the planets as the spaceship travels through the solar system. For simplicity, we will also consider the planetary orbits to be circular, co-planar, i.e., the motion will be 2D. The mass of the Sun is $M_\odot = 2 \times 10^{30}$ kg, whose gravity will dominate the motion of the spaceship. Thus, the equation of motion for the spaceship is

$$\frac{d^2\vec{r}}{dt^2} = -\frac{GM_\odot}{r^2}\hat{r} = -\frac{GM_\odot}{r^3}\vec{r}. \tag{1}$$

Note that if we dot the velocity $\vec{v} = d\vec{r}/dt$ to the previous equation we obtain

$$\frac{d\vec{r}}{dt}\frac{d^2\vec{r}}{dt^2} = -\frac{GM_\odot}{r^3}\vec{r}\frac{d\vec{r}}{dt} \tag{2}$$

but

$$\frac{d\vec{r}}{dt}\frac{d^2\vec{r}}{dt^2} = \frac{1}{2}\frac{dv^2}{dt} \tag{3}$$

and

$$-\frac{1}{r^3}\vec{r}\frac{d\vec{r}}{dt} = \frac{d}{dt}\left(\frac{1}{r}\right) \tag{4}$$

Thus, Eq. (2) becomes

$$\frac{d}{dt}\left(\frac{v^2}{2}\right) = \frac{d}{dt}\left(\frac{GM_\odot}{r}\right) \tag{5}$$

or the total energy per unit mass

$$E_{\rm tot} = \frac{v^2}{2} - \frac{GM_\odot}{r} \tag{6}$$

is conserved. The fact that $E_{\rm tot}$ is conserved will serve as a useful diagnostic to validate your numerical integration of the equations of motion (EOM).

The EOM(1) can be written as

$$\begin{aligned} \frac{d^2x}{dt^2} &= -\frac{GM_\odot}{r^3}x, \\ \frac{d^2y}{dt^2} &= -\frac{GM_\odot}{r^3}y, \end{aligned} \tag{7}$$

where $r = \sqrt{x^2+y^2}$. To solve this system of ODEs we need initial conditions for x, y and $\frac{dx}{dt}, \frac{dy}{dt}$. The distance of the Earth from the Sun is one astronomical unit $1{\rm AU} = 1.495978707 \times 10^{11}$ m. Assume that when the spaceship leaves Earth at $t = 0$, the Earth is at $x = 1{\rm AU}$ and $y = 0$.

A few considerations will help you experiment to solve the problem. The Earth's velocity around the Sun is determined by Kepler's third law

$$v_\oplus = \sqrt{\frac{GM_\odot}{1{\rm AU}}} = 29789.37{\rm m/s} \tag{8}$$

We will make the assumption that the Earth and all planets rotate counterclockwise. Thus, the spaceship has an initial boost from the Earth's orbit of 29789.37 m/s in the $+\,y$ direction ($dy_{\rm ini,\ boost}/dt = 29789.37{\rm m/s}, dx_{\rm ini,\ boost}/dt = 0$. The project's goal is to determine the initial velocity that the spaceship needs to correct for $\frac{dx_{\rm correction}}{dt}, \frac{dy_{\rm correction}}{dt}$ so that after solving Eqs. (7) with initial conditions for the velocity

$$\begin{aligned} \frac{dx}{dt} &= \frac{dx_{\rm ini,\ boost}}{dt} + \frac{dx_{\rm correction}}{dt} \\ \frac{dy}{dt} &= \frac{dy_{\rm ini,\ boost}}{dt} + \frac{dy_{\rm correction}}{dt}, \end{aligned} \tag{9}$$

you arrive at the target planet.

The escape velocity from Earth's orbit is $v_{\rm esc} = \sqrt{2}v_\oplus = 42128.53{\rm m/s}$. Thus, the magnitude of the initial velocity ($\sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}^2\right)}$) cannot be this large. A better value can be predicted by computing the precise initial velocity needed to reach the target planet with zero final velocity (to minimize fuel consumption). This can be achieved using energy conservation assuming. In particular, using

$$\frac{v_{\rm initial}^2}{2} - \frac{GM_\odot}{1{\rm AU}} = -\frac{GM_\odot}{r_{\rm planet}}. \tag{10}$$

where $r_{\rm planet}$ is the radius of the orbit of the target planet. You will be using Jupyter for your calculations for which $r_{\rm Jupyter} = 5.2{\rm AU}$.

1. Use RK4, to integrate numerically the non-dimensional system of ODEs (7) from $\tilde{t} = 0$ until you reach the target planet Jupyter. You will also need the target planet position at $t = 0$. Assume that at $t = 0$ Jupyter is located at 30 degrees below the y-axis, i.e., $x_{\rm Jupyter,0} = r_{\rm Jupyter}\cos(\pi/6), y_{\rm Jupyter,0} = -r_{\rm Jupyter}\sin(\pi/6)$. Kepler's third law states that Jupyter's angular velocity around the Sun is $\Omega = \sqrt{\frac{GM_\odot}{r^3}} = 1.6793 \times 10^{-8}{\rm rad/s}$. Thus, knowing Ω and $r_{Jupyter}$ you can know the position of the planet at any time.

$$\begin{aligned} x_{Jupyter}(t) &= r_{\rm Jupyter}\cos(\Omega * t - \pi/6) \\ y_{Jupyter}(t) &= r_{\rm Jupyter}\sin(\Omega * t - \pi/6) \end{aligned} \tag{11}$$

   You will need to use trial and error until you find the correct initial conditions for the velocity such that the solution of Eq. (7) yields a x, y pair at some time t that matches $x_{Jupyter}(t), y_{Jupyter}(t)$ to within 1%.

   Use your judgement as to how small a step size you need to solve this system accurately. If you cannot figure this out from pure thought, experiment with different step sizes. Use $\delta E = |(E_{\rm tot} - E_{\rm tot}(t=0))/E_{\rm tot}(t=0)|$ to determine the accuracy. If δE is smaller than 10^{-3} for all integration times, then you have decent accuracy.

2. **Convergence:** Demonstrate that as you decrease the step size h, δE at the final time converges to 0. Make a plot to determine the order of convergence of δE, and discuss why or why not this matches your expectation.

3. **Self-convergence**: Use a number of step sizes and make a plot to demonstrate that the code solution for x and y at the final time self-converges. Show your convergence plot. Does the solution converge at the order you expect? If not, try to explain why.

4. Using the order of convergence you determined, employ Richardson extrapolation to determine an error for the solution for $x(t)$, $y(t)$ at a time t of your choosing.

5. **Plot the orbits**: Use your most accurate simulation (based on the conservation of energy), and plot the trajectories of the Earth, Jupyter and the space ship.