PHYS 416/517 2019 – Orbital Mechanics

Garcia, Chapter 3 – ***Due 1 PM Thursday February 20, 2020***

The purpose of this unit is to introduce some more sophisticated techniques for solving ODEs, to do this we will look at some orbital mechanics problems and a simple problem of particle drifts in electromagnetic fields.

General suggestions:

1. Make sure you make the names of your programs the same as the exercise, for example the program for exercise 3 should be something like 'ex3.m'.
2. The programs should all be procedures; that way all the grader has to do is run the procedure to check if it is working.
3. Turn in your graphs and derivations electronically, so for example in the first problem that requires a derivation, you can either type it up or write it out and scan it.
4. It would be helpful to include all figures, comments and results for the problems into one file. I know this is tedious, but it makes the grader's job much easier and it should improve your grade.

Introduction

Newton's law of gravity gives the force on an object of mass m_i at location $\vec{r}_i$ due to a distribution of N masses m_j at location $\vec{r}_j$ as

$$\vec{F}_i = -\sum_{\substack{j \\ i \neq j}}^{N} G\frac{m_i m_j}{|\vec{r}_j - \vec{r}_i|^3}(\vec{r}_j - \vec{r}_i) \qquad (1)$$

The simplest example of orbital mechanics is one where you have an object of mass m orbiting a large central object at rest of mass M. In this case, the force on the object is simply,

$$\vec{F} = -G\frac{Mm}{|\vec{r}|^3}\,\vec{r} \qquad (2)$$

where $\vec{r}$ is the position of the object, and G is the gravitational constant (= 6.67×10^{11} $m^3/kg\ s^2$).

In this chapter, we will look at the solar system where the large central object is of course the Sun (mass=1.99×10^{30} kg). We will use astronomical units (au) as our measure of distance (=1.496×10^{11} m), which is the Earth-Sun distance and a year as the unit of time. In these units, it can be shown that $GM=4\pi^2$ au^3/yr^2.

According to Kepler's Laws, a 2 body Solar system with an object that is on a closed trajectory will orbit the Sun on an ellipse with the Sun at one of the foci. The figure below illustrates some of the salient features of an ellipse. The relationship between the semi-major axis a and the semi-minor axis b is given by

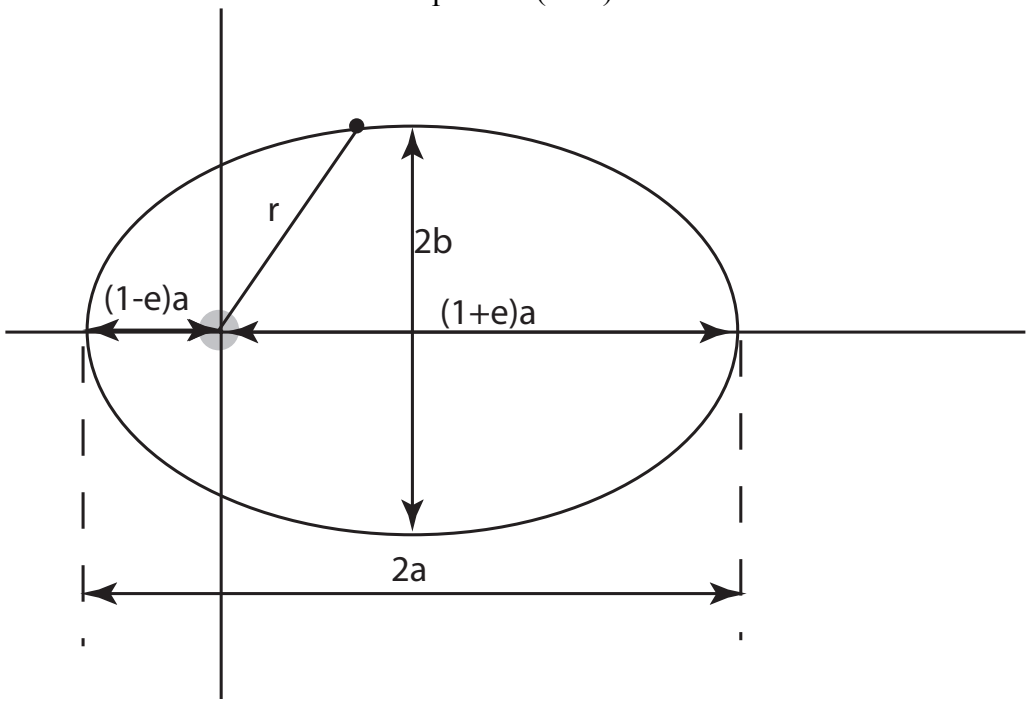
$$e = \sqrt{1 - \frac{b^2}{a^2}} \qquad (3)$$

In polar coordinates, the equation for an ellipse is given by

$$r = \frac{a(1-e^2)}{1 - e\cos\theta}$$

(4)

For the Earth $e = 0.017$ (nearly circular). Closest approach is defined as perihelion which is (1-*e*)*a* and the furthest distance is known as aphelion (1+ *e*)*a*.



There are 2 conserved quantities in this type of system, one is the energy *E* which is given by

$$E = \frac{1}{2}mv^2 - \frac{GMm}{r} \tag{5}$$

which can be shown to be equal to

$$E = -\frac{GMm}{2a} \tag{6}$$

Equations (5) and (6) can be combined to get the speed as a function of *r* as

$$v = \sqrt{GM\left(\frac{2}{r} - \frac{1}{a}\right)} \tag{7}$$

and the other is the angular momentum, which is defined as

$$\vec{L} = \vec{r} \times m\vec{v} \tag{8}$$

which for 2-dimensional (x-y) motion points in the z-direction. Using (7), one can show that

$$\left|\vec{L}\right|^2 = GMm^2a(1 - e^2) \tag{9}$$

This equation also known as Kepler's second law states that the orbiting satellite sweeps out an equal area(A)/time (i.e., $\frac{dA}{dt} = L/2m$). Since $T = \frac{A}{\frac{dA}{dt}} = \pi ab/(\frac{L}{2m})$ and equation (3) it can be used to derive Kepler's 3rd law

$$T^2 = \frac{4\pi^2}{GM}a^3 = a^3 \tag{10}$$

where *T* is the orbital period (In our units, $GM=4\pi^2$ au^3/yr^2).

Problems from Chapter 3

Part 1: Exercises using low order techniques

2. Prove that for the Kepler problem the Euler-Cromer method conserves angular momentum. [*Pencil*]

3. Modify the orbit program so that instead of running a fixed number of time steps, the program stops when the satellite completes one orbit.

(a) Have the program compute the period, eccentricity, semi-major axis, and perihelion distance of the orbit. Use the Euler-Cromer method and test the program with circular and slightly elliptical orbits. Compare the measured eccentricity with

$$e = \sqrt{1 + \frac{2EL^2}{G^2M^2m^3}}$$

(b) Show your program confirms Kepler's third Law. (i.e., $T^2 = \frac{4\pi^2}{GM} a^3$)

(c) Confirm that $\langle K \rangle = -\frac{1}{2}\langle V \rangle$, where $\langle K \rangle$ and $\langle V \rangle$ are the time-average kinetic and potential energy (virial theorem).

[Computer: Turn in the program and a couple of sample runs. A single program that does (a) – (c) is suggested.]

Comments: You could use the `interpf` function to get a better estimate of the orbital period, but I suspect you have had your fill of it by now. If you prefer, you can just run the code with a small enough time step to get a reasonable estimate of the orbital period. Make sure you do (a)-(c) for an elliptical orbit. Assume for example a starting location of 1 au on the x-axis and speed of π au/yr in the y direction.

Part 2: Exercise using the Runge-Kutta Method

12. Suppose that our comet is subjected to a constant force in one direction (e.g., gravitational attraction of a large but distant object). Modify the orbit program to simulate this system. Set the strength of the perturbing force to be 1% of the **initial gravitational force**. Using the fourth-order Runge-Kutta method, show that an initially circular orbit is transformed into an elliptical orbit with the semi-major axis perpendicular to the perturbing force. Produce a graph of the angular momentum as a function of time.

[Computer: Turn in the program and graphs of the orbit and the angular momentum.].

Important: **The key here is to NOT to modify the rk4 and rka functions for each problem, but to implement a new gravrk function (with new name) for each problem**.

Part 3: Exercise using adaptive Runge-Kutta

14. Consider the central force

$$\vec{F}(\vec{r}) = -\frac{GMm}{r^3}\left(1 - \frac{\alpha}{r}\right)\vec{r}$$

where α is a constant. Modify the orbit program to compute the motion of the object under this force law. Using the adaptive Runge-Kutta, show that the orbit precesses $360(1-a)/a$ degrees per revolution, where $a=\sqrt{1+GMm^2\alpha/L^2}$ and L is the angular momentum. [*Computer*] (Use *α=0.1 and make sure you check for energy conservation.).*

Part 4: Solar system simulation using adaptive Runge-Kutta

A. Using the orbit program as a starting point, create a model of the Sun-Earth-Jupiter system. Use this model to investigate the effect of Jupiter on the orbit of the Earth. How much more massive must Jupiter be before the orbit of the Earth becomes unstable? Make a few example plots for Jupiter with 1, 10, 100 and 1000 M_J
[Computer: Turn in the program and sample runs.]

B. Now allow the Sun to move and add Saturn to the simulation (so you now have a 4-body simulation) and see what effect increasing the mass of Jupiter has on its orbit.
[Computer: Turn in the program and sample runs.]
Hints:

1. Using the normalized units that the orbit program uses, where $GM_s=4\pi^2$, all the masses are in terms of solar masses, distances are in au and time is in years.
2. Use Kepler's 3rd law to estimate the initial speed of the planet.
3. With a little care, you can modify your program to compute the orbit for any number of orbiting bodies. One way is to pass the relevant constants (like mass) in the params variable as an array.

You will need to following table for model inputs.
The mass of the Sun is 2.0×10^{30} kg.

Planet	**Mass (kg)**	**Mean orbital radius (au)**
Earth	6.0×10^{24}	1.0
Jupiter	1.9×10^{27}	5.2
Saturn	5.7×10^{26}	9.54

Side note: MATLAB: Passing Functions to Functions

The Runge-Kutta function that will be used to solve some of the problems below introduces a new MATLAB feature: passing of functions. In MATLAB, a function can be used, as in the simple example below as

```
>> x=2
x =2
>> y=sin(x)
y =0.9093
```

However, another way is to use the feval (function evaluation) function

```
>> y=feval('sin',2)
y =0.9093
```

the quotes (‘ ‘) are used to pass the name of the function. In the later versions of MATLAB, this is being replaced by the ‘@’ symbol.

```
>> y=feval(@sin,2)
y =0.9093
```

In the version of MATLAB you are using both approaches work, although the documentation claims that the new approach is better. The advantage of using `feval` is that it allows you to pass function names to functions, so that you can reuse functions such as rk4 for any type of problem.

Side note: Python: Passing Functions to Functions

There are also python versions of the orbit program orbit.py. In this version, I have included the function definitions 'rk4', 'rka' and 'gravrk' at the top of the program. In your homework you can also elect to have them as separate routines, and use the import command or you can also include the solver routines from orbit by placing the command 'if __name__ == "__main__":' at the top of your main program with appropriate indentation. I found it simpler to just carry the functions definitions around at the top of each program, but that is my personal preference. Also, you will notice that the constant 'GM' is not passed in the function call (as in the MATLAB version) as it is known by all as a global variable, which I am not sure is the best approach, but it seems the simplest.

Optional Exercises –1 point extra credit

EC1. (Garcia ex 13). The Wilberforce pendulum, a popular demonstration device, is illustrated below (Figure 3.8 of Garcia). The pendulum as two modes of oscillation: vertical and torsional motion. The Lagrangian for this system is

$$L = \frac{1}{2}m\dot{z}^2 + \frac{1}{2}I\dot{\theta}^2 - \frac{1}{2}kz^2 - \frac{1}{2}\delta\theta^2 - \frac{1}{2}\epsilon z\theta$$

where m and I are the mass and the rotational inertia of the bob, k and δ are the longitudinal and torsional spring constants, and ϵ is the coupling constant between the modes. Some typical values are:
m = 0.5 kg, I = 10^{-4} kg m^2, k = 5 N/m. δ= 10^{-3} N m, and ϵ = 10^{-2} N.
(a) Find the equations of motion [Pencil]
(b) Write a program to compute z(t) and $\theta(t)$using fourth-order Runge-Kutta. Try the initial conditions z(0) = 10 cm, $\theta(0) = 0$ and z(0) = 0, and $\theta(t) = 2\pi$. Show that when the longitudinal frequency $f_c = \frac{1}{2\pi}\sqrt{\frac{k}{m}}$, equals the torsional frequency $f_\theta = \frac{1}{2\pi}\sqrt{\frac{\delta}{I}}$ the motion periodically alternates between being purely longitudinal and purely torsional.
[*Computer: Turn in the program and sample output.*]

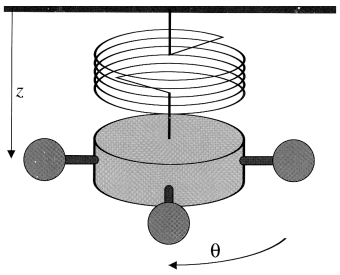


Figure 3.8: Wilberforce pendulum.

Wilberforce Pendulum for problem EC2.

EC2. [This is an extension of the Garcia problem 7 from the textbook.]
When a charged particle of mass m and charge q moves through a magnetic and electric field it feels the Lorentz force given by

$$\vec{F} = q(\vec{E} + \vec{v} \times \vec{B})$$

Where $\vec{v}$ is the velocity of the particle. If the magnetic field is constant, then it moves in a circle of radius

$$r_{gyro} = \frac{mv}{qB}$$

Where r_{gyro} is the gyroradius and has a period of

$$T = \frac{2\pi m}{qB}$$

Which is known as the gyroperiod. If the particle is in a static electric field E, then it can be shown that the center of gyration moves with a velocity

$$\vec{v}_{E\times B} = \frac{\vec{E}\times\vec{B}}{B^2}$$

Which is commonly referred to as $E \times B$ drift. If the magnetic field is not constant, but has a small gradient, then it can be shown that the center of gyration also moves with a velocity given by

$$\vec{v}_{\nabla B} = \frac{mv^2}{2qB}\frac{\vec{B}\times\nabla B}{B^2}$$

Which is known as ∇B drift. (A derivation of this equation can be found in any Plasma Physics Textbook.) By small, I mean that the gradient scale for the variation of the magnetic field is small compared to the particle's gyroradius.

Write a program using the Euler-Cromer method that solves for the trajectory of a particle in the following electric and magnetic fields. Assume that we have normalized quantities so that *m=1, q=1*. The initial condition for the particles is $\vec{r} = 0, \vec{v} = \hat{y}$

(a) $\vec{E} = 0, \vec{B} = \hat{z}$. (Zero electric field and unit magnetic field) Verify that for a reasonably timestep, that you get the correct gyroradius and period. Use this timestep for problems (b) and (c).

(b) $\vec{E} = \hat{x}$, and $\vec{B} = \hat{z}$. (Unit electric and magnetic field, orthogonal to each other). Verify that you get the correct $E \times B$ drift by plotting both the trajectory and the trajectory in the frame moving with the $E \times B$ velocity. In the moving frame the particle should just move in a circle.

(c) $\vec{E} = 0, \vec{B} = (1 + \alpha x)\hat{z}$, where $\alpha = 0.1$. (Zero electric field and magnetic field in the z-direction that varies with x). Verify that you get approximately the correct gradient B drift, by plotting both the trajectory and the trajectory in the frame moving with the ∇B velocity. As in (b) in the moving frame the particle should just move approximately in a circle.

Hints:

- It is probably useful to have the code tell you the estimated gyroperiod and gyroradius from the above formulas, so once you enter the timestep and the total simulation time, which should be greater than the gyroperiod, it will compute the total number of steps it needs to make. Note that the formulas will only work for the case of constant magnetic and zero electric field, but it can give you a rough estimate for the other parts of the problem also.
- The Euler-Cromer method is far from perfect for this problem, but you can minimize the error (to some tolerance, say ~1%) by an appropriately small timestep.