AST 2000 - Part 2
Planetary Orbits

Welcome to Part 2 of the AST2000 Spacecraft Project. There are 3 components in this part, one of which is optional for those working alone: To begin, you will need to simulate the orbits of your planets. Next (this is optional for those working alone), you will test your orbits using Kepler's laws. Finally, you will take a look at whether your planets are observable from other solar systems.

GOALS

∇ Plot the orbits of the planets in your solar system.

∇ Simulate the orbits of the planets in your solar system.

∇ Use Kepler's laws to verify the integrity of your planetary orbits (optional for those working alone).

∇ Analyse how your planets affect their star and evaluate whether the effect is noticeable from other solar systems.

SIMULATING THE ORBITS

To reduce our computational load we introduce the following simplifications:

⊗ Ignore planet-planet interactions and only consider the gravitational pull on the planets by the star.

⊗ Ignore relativistic effects.

⊗ Ignore the gravitational pull on the star by the planets. Assume the star is fixed at the origin throughout the simulation.

⊗ We are assuming that all the orbits exist in the same x-y plane. You can therefore safely ignore the z-axis.

⊗ All planets both orbit the star *and* rotate about the z-axis in a counterclockwise motion.

⊗ Ignore all other astronomical bodies in the solar system, including moons.

CHALLENGES

Relevant theory for Part 2 is described in Lecture Notes 1B and Lecture Notes 1C. Make sure you have read them *before* you continue.

In addition to the lecture notes you should also check the integration section in the Numerical Compendium. Note that the Leap Frog integration scheme is energy conservative.

A. Planetary orbits

We encourage you to use AU units to prevent floating point overflow. Also, the gravitational constant is very simple in AU units: [1]

$$G = 4\pi^2 \ \mathrm{AU}^3 \mathrm{yr}^{-2} \mathrm{M}_\odot^{-1}$$

1. Plotting the Orbits

Your first objective is to use the analytic solution of elliptical orbits found in Lecture Notes 1B in order to plot the orbits of all your planets. You can access all necessary data in the `SolarSystem` object held by your instance of the `SpaceMission` class.

2. Simulating the Orbits

Now that you have plotted the planetary orbits of your solar system, the next step is to simulate the orbits. "Wait, didn't we just find the analytic solution?" I hear you say. Well, while the analytic solution provides us with the shape of the orbits, it *does not* describe the position of the planets as a function of time t. We need to find this information by simulating their motion.

Simulate the planetary orbits for at least 20 revolutions of your home planet. Use a minimum of 10,000 time steps per year to ensure your results are accurate. Depending on your solar system you may even need more. Comment on your chosen number of time steps: why is this number sufficient? To verify your simulation has been completed successfully, plot your simulated orbits on top of the analytic orbits (the accuracy of your orbits will be checked later).

Running the simulation may take a considerable amount of time. **Remember to save your orbits** so you only have to perform this simulation once. You should also review your options and choose a programming style that is fast and that you understand well. If you have no preference, we invite you to take a look at the Numerical Compendium.

B. Kepler's laws

This is optional for those working alone.

1. Proving Kepler's 2nd and 3rd Laws

Kepler's 2nd law can be expressed as

$$\frac{dA}{dt} = constant \tag{1}$$

where A is the area swept out by $\mathbf{r}$ in figure 1. Let (r, θ) denote $\mathbf{r}$ in polar coordinates.
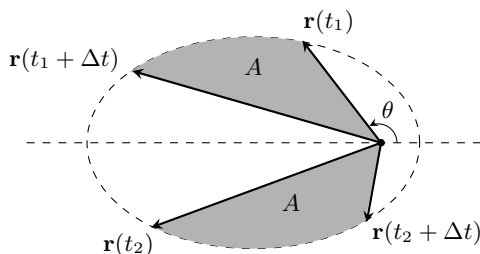

FIG. 1. An illustration of the area swept out by the position vector of a body in an elliptical orbit.

1. Consider the infinitesimal area dA swept out by $d\mathbf{r}$ during the time interval dt. Approximate this area using a right triangle and derive

$$dA = \frac{1}{2}r^2 d\theta \tag{2}$$

2. Prove Kepler's 2nd law by expressing dA/dt in terms of h, where $h = |\mathbf{h}| = |\mathbf{r} \times \dot{\mathbf{r}}|$.

3. Integrate dA/dt for a full time period P and show that the orbital time period is:

$$P = 2\pi \frac{ab}{h} \tag{3}$$

(remember that the area of an ellipse is given by πab)

4. Deduce Newton's correct version of Kepler's 3rd law using expressions for h and b found in the lecture notes:

$$P^2 = \frac{4\pi^2}{G(m_1 + m_2)}a^3 \tag{4}$$

5. Explain what was wrong with Kepler's original law:

$$P^2 = a^3 \tag{5}$$

where $[P] =$ yrs and $[a] =$ AU.

2. Comparing your Orbits to Kepler's Laws

1. Use equation (2) to determine the area swept out by your planet in some time interval Δt.

   (a) Compare two areas: one where the planet is close to the aphelion and one where the planet is close to the perihelion. To what degree are the areas equal?

   (b) What distance did the planet travel as it swept out these two areas?

   (c) What mean velocity did the planet travel with?

2. Check if all your orbits are consistent with Kepler's 3rd law. Is there a measurable difference between Kepler's original version and Newton's corrected version?

C. Can Extraterrestrials Discover the Planets in your Solar System?

We are now going to take a second look at our planetary orbits. This time however, you are going to include the gravitational pull on your star by its planets.

Be careful not to confuse your orbits later in the project; **the orbits you simulate for this challenge should not be used in anywhere else.**

Furthermore, half of this challenge involves the exchange of data between you and another group. Make sure you have someone to cooperate with!

1. The Solar Orbit

1. Choose one of the planets in your solar system: preferably one of the largest and as close to the star as possible (to maximize gravitational pull). Place the star at the origin for $t = 0$ and let its initial velocity be zero. Doing this places you in a frame of reference where the center of mass drifts; find a way to change your frame of reference to the center of mass frame.

   Ignore all other planets and simulate the orbits of the star and your chosen planet using your favorite integration method. Choose your integration parameters based on your experience with simulating the orbits.

2. With the solar orbit in hand we are now going to test its accuracy using an analytic approach. Your first task is to study the two-body system in general, then use your findings in order to study your specific solar orbit.

(a) Show that the total energy of the two-body system as seen from the center of mass frame can be written as

$$E = \frac{1}{2}\hat{\mu}v^2 - \frac{GM\hat{\mu}}{r} \qquad (6)$$

where $v = |\dot{\mathbf{r}}|$ is the relative speed, $r = |\mathbf{r}|$ is the relative distance, $\hat{\mu}$ is the reduced mass and M is the total mass of the system.

(b) Show that the angular momentum of the two-body system as seen from the center of mass frame can be written as

$$\mathbf{P} = \mathbf{r} \times \hat{\mu}\dot{\mathbf{r}} \qquad (7)$$

(c) Take a moment and study equations (6) and (7). If you were given these equations and nothing else, what physical system would you guess they described?

(d) Use equation (6) to test the numerical accuracy of your new orbits. Is the energy conserved to a reasonable degree?

2. The Radial Velocity Curve

1. The radial velocity curve of a star is dependent on three parameters: the peculiar velocity of the center of mass, line of sight, and the inclination of the solar system from the line of sight.

   Choose an arbitrary set of parameters and design the radial velocity curve for your star based on your newly simulated solar orbit. With the exception of inclination angles $i \approx 0$, you are free to choose any physically consistent values for your parameters.

   Now add some independent Gaussian noise to the data. Let the mean noise be zero and the standard deviation be $1/5$ of the maximum value of your velocity curve.

2. Contact the other group you are cooperating with and exchange your radial velocity curves. Your task is to analyse their velocity curve and provide an estimated lower boundary for the mass of the planet in their solar system.

   Your analysis should be based on the method of least squares as described in section 5 of Lecture Notes 1C. However, before you can employ the least squares algorithm you need to adjust the data for the star's peculiar velocity by removing any existing drift.

3. Once both groups have finished their analyses, compare your results to the actual masses. Do you expect to find exact numbers? Why, why not?

3. The light curve

This is optional for those working alone.

1. Let the inclination of your solar system as seen from an outside observer be $i = 90°$. You need to design an approximate light curve, remember that the received light depends on the area of the planet in front of the star (this is very easy if you do some approximations, remember to explain and justify). That is, you do not need to estimate the actual flux from your star; only the relative change in the flux, caused by the planets eclipsing the star. Define your flux relative to the maximum flux emitted by your star (such that flux $= 1$ when no planet is eclipsing the star).

   Again, add some independent Gaussian noise to the data. Let the mean noise be zero and the standard deviation be 0.2 relative flux. Do you think it is possible to detect your planet?

2. Contact the other group you are cooperating with and exchange light curves. Your task is to analyse their light curve and attempt to estimate the radius and density of the planet eclipsing their star. In order to do this, you need data from your analysis of the same other group's radial velocity curve.

   As opposed to your analysis of the velocity curve, you do *not* need to use the method of least squares; it is sufficient to plot an approximate light curve.

3. Once both projects have finished the analysis, compare your results to the actual radii and densities.

4. The radial velocity curve with more planets

This is optional for those working alone.

1. You are now going to revisit your simulation of the solar orbit and include 2-4 additional large planets from your solar system. The simulation is in principle exactly the same as before, but you may have to redesign your integration loop depending on how you structured your initial program.

2. Again, choose an arbitrary (yet physically plausible) set of parameters and design a new radial velocity curve. Exchange this curve with the other group you are cooperating with and attempt to resolve the number of planets the other project included in their simulations. Once both groups have made their guesses, compare with the actual number of planets in the simulations.

VERIFICATION/CONFIRMATION

The Part 2 challenges are now complete! All that remains now is to verify the numerical accuracy of your orbits. Check your orbits using the `verify_planet_positions` method from the `SpaceMission` class. Your calculations are expected to be within a 1% margin.

Note that the orbits you need to verify are the original orbits where the star is *stationary* (i.e. the orbits from challenge A).

VISUALISING THE SOLAR SYSTEM

Use the `generate_orbit_video` method from the `SpaceMission` class to generate an XML file containing your numerically simulated orbits. Assuming you have downloaded MCAst, place the XML file in the `data` directory within the MCAst installation directory. You can now explore your solar system using the SolarSystemViewer software (Note that you need to start the application called `SSView` in the MCAst directory for this, not `MCAst`). Does it live up to your expectations?

EXTRA CHALLENGE

If you finished all the challenges and are still hungry for more, fear not! [2]

⋆ Can you explain why the gravitational constant is $G = 4\pi^2$ in Astronomical Units?

⋆ Are you able to redo the analytic part of challenge B 1 without using infinitesimals such as dA? Use the theory of limits and the definition of the derivative from MAT 1100.

⋆ Try to write a program that automagically analyses any light curve based on an approximate model you derive. (Hint: You already know the max and min flux values. You need to create a model for the change in flux as the planet enters/leaves the radiating disks.)

FIG. 2. Orbital motivational duck.

AST 2000 - Part 2
Tips, Hints & Guiding Questions

I. IN GENERAL

→ Note that the first challenge is the most important challenge in Part 2. Without the planetary orbits you are effectively incapable of continuing the project.

→ Be extra careful with your units! We actually mean it when we *"encourage you to use AU units"*.

II. HINTS FOR SOME OF THE CHALLENGES

A. The Gravitational System

Since the orbits are decoupled (they do not interact), each orbit is described by the same differential equation:

$$m_p\ddot{\mathbf{r}} = -G\frac{m_s m_p}{|\mathbf{r}|^3}\mathbf{r}, \tag{a}$$

Here, m_s is the mass of the star, m_p is the mass of the planet and $\mathbf{r}$ is the position vector of the planet. Initial conditions for the planetary orbits are accessed through the `SolarSystem` object held by your `SpaceMission` instance.

You are most likely better off simulating all your planets' trajectories simultaneously using NumPy arrays. The numerical accuracy of your calculations are really important here; we advise the use of higher order integration methods such as the Leap Frog scheme. You can find more hints in the Numerical Compendium.

B. Deriving Kepler's Laws

→ You are allowed to use *"physicist math"* and play with differentials:

$$\mathrm{d}f = c\,\mathrm{d}x \iff \frac{\mathrm{d}f}{\mathrm{d}t} = c\frac{\mathrm{d}f}{\mathrm{d}t} = c\dot{f},$$

$$\frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\mathrm{d}f}{\mathrm{d}t}\frac{\mathrm{d}x}{\mathrm{d}x} = \frac{\mathrm{d}f}{\mathrm{d}x}\frac{\mathrm{d}x}{\mathrm{d}t}, \quad \text{etc.}$$

→ The area circumscribed by an ellipse with semi-major and semi-minor axes a and b is equal to:

$$A = \pi ab \tag{b}$$

→ Be careful with your variables! For example: while p is a constant dependent on the shape of the orbit, $\mathbf{p}$ is the *momentum* of the body. They are not related. Specifically: $p \neq |\mathbf{p}|$.

→ What is the mass of our star (*The Sol*) in AU units, i.e. in solar masses $\mathrm{M}_\odot$?

C. Simulating the New Orbits

How does the the gravitational system differ from the original simulation? Is equation (a) still valid for any or all bodies? What has changed?

To change your frame of reference to the center of mass frame, you need to construct the position vectors of the bodies as seen from the center of mass. How do the orbits look from the center of mass? Can you calculate the position vector of the center of mass as seen from your simulation's frame of reference?

D. Center of Mass Frame

In challenges 1.2a & 1.2b in C1, use the definitions of E and $\mathbf{P}$ (theory from FYS-MEK 1110) with the position vector transformations:

$$\mathbf{r}_1^{\mathrm{cm}} = -\frac{\hat{\mu}}{m_1}\mathbf{r} \quad \text{and} \quad \mathbf{r}_2^{\mathrm{cm}} = \frac{\hat{\mu}}{m_2}\mathbf{r} \tag{c}$$

Are there similar transformations for $\dot{\mathbf{r}}_1^{\mathrm{cm}}$ and $\dot{\mathbf{r}}_2^{\mathrm{cm}}$ to $\dot{\mathbf{r}}$?

E. Designing the Radial Velocity Curve

Here you need to use your solar orbit from challenge C1 as seen from the center of mass frame. Your first task is to add an arbitrary peculiar velocity to the center of mass. Next comes the inclination of the orbit; you should be able to deduce the effect of tilting the orbital plane using simple trigonometry. Last but not least, you need to choose your line of sight. To guide your intuition, take a look at figure 3: You can think of the line of sight as placing the stick-man somewhere on the (x, y) plane, recalling that he is only able to observe the radial velocity along his line of sight. The easiest cases to implement are when the line of sight is either along the x-axis or the y-axis. Can you explain why this is the case?

Unless you have a strong passion for a specific programming style, I would like to personally recommend checking out `JIT`! It's fast and simple. Otherwise standard `NumPy` vectorization is key here. Regardless of your chosen style, be careful to mix `JIT` and `NumPy`, check out the ζ Half-Life example in the Numerical Compedium.

F.   Analysing the Radial Velocity Curve

Your first task is to remove any drift from the data. *What is it* that is drifting? How did you add the drift to your radial velocity in the first place?

The analysis of the radial velocity curve is based on theory from Section 5 in Lecture Notes 1C, specifically the method of least squares:

$$\Delta(v_{*r}, P, t_0) = \sum_{i=1}^{N} \Big[v_i - v_{\mathrm{model}}(t_i) \Big]^2 \qquad \text{(d)}$$

where

$$v_{\mathrm{model}}(t_i) = v_{*r} \cos \left[\frac{2\pi}{P} \big(t_i - t_0 \big) \right] \qquad \text{(e)}$$

is the theoretical model for the curve. Here, v_{*r}, P and t_0 are the model's parameters that you need to optimize using your favorite algorithm.

You will learn more about the method of least squares in MAT 1120 (and in practically every other course with any connection to statistical analysis).
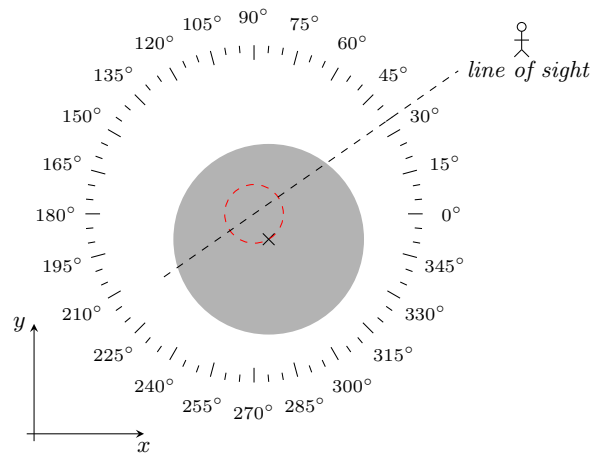

FIG. 3. An illustration of an observer's line of sight to a distant solar system. The star (gray disk) follows its circular orbit (dashed red line) in the (x, y) plane. An observer (stickman) sees the star system from an arbitrary position. Note that in this illustration the inclination angle i is zero, this is for illustrative purposes only.

[1] Assuming you ignore everything after $4\pi^2$.
[2] If you are actually hungry for more, and not just because of 'dem sweet grades, consider speaking to someone.