國立清華大學
NATIONAL TSING HUA UNIVERSITY

10820ASTR660000
Computational Astrophysics
計算天文物理

Release date: 2021.03.04
Due in class: 2021.03.18
(email to Kuo-Chuan before lecture)

Problem Set 2

Reading Assignments

1. Writing Makefiles: read chapter 2 "An introduction to Makefiles" (http://www.gnu.org/software/make/manual/make.html#Introduction).
2. Debugging with *gdb*: read "https://www.cs.cmu.edu/~gilpin/tutorial/".
3. *(Optional)* Read Chapter 2 of "Gnuplot in Action" (https://livebook.manning.com/book/gnuplot-in-action/chapter-2/1)
4. *(Optional)* Create scientific plots using gnuplot: Read "http://www.gnuplotting.org/"

Written Assignments

1. Derive the Kepler's second law
$$A = \frac{1}{2}\frac{L}{\mu}P, \tag{1}$$
and third law
$$P^2 = \frac{4\pi^2}{G(m_1 + m_2)}a^3, \tag{2}$$
where $A = \pi ab$ is the area of an ellipse, L is the angular momentum, μ is the reduced mass, P is the orbital period, a is the binary separation, and m_1 and m_2 are masses. Do not assume a circular orbit. [30pts]

Programming Assignments

1. Modify the initial condition in your first order binary evolution code to the Sun-Earth system. Set $m_1 = 1M_\odot$, $m_2 = 5.97219 \times 10^{27}$ g, $a = 1$ AU, $\Delta t = 1,000$ s, and $t_{max} =$ 5 yrs. Plot $x_2(t)$ and compare your computed orbital period with a year. [30pts]

國立清華大學
NATIONAL TSING HUA UNIVERSITY

2. Now artificially multiply the initial $v_{y,2}$ by a factor of 1.25. (a) Plot the trajectory of both stars on the x-y plane. (b) The eccentricity e of the new orbit can calculated by $r_p = a(1-e)$, where r_p is distance to the perihelion. Plot $v_{y,2}^2(t)$ together with two lines with $v^2 = \frac{GM_\odot}{a}\frac{1+e}{1-e}$ and $v^2 = \frac{GM_\odot}{a}\frac{1-e}{1+e}$. What do you observed? (c) modify the file *"output.f90"* to compute and store the total angular momentum $L(t)$ and total energy $E(t)$ as functions of time. Rerun simulations with different time step $\Delta t = 0.01, 0.001$, and 0.00001 yrs. Are angular momentum and total energy conserved? If not, why? [30pts]

3. Lets move back to the original setup of the binary evolution code in class. Now, there is a third star ($m_3 = 1M_\odot$) located at 10 AU away from the center of mass of the original binary. Assuming the third star has a circular orbit and three stars are aligned on the x-axis initially (see the figure below). Extend your binary evolution code to handle this three-body problem. (a) plot the trajectory of the three body system on the orbital plane ($t_{max} = 100$ yrs). Run your simulation with different time step $\Delta t = 0.1, 0.01, 0.001$, and 0.00001 yrs. (b) A simple upgrade on your numerical scheme to second order is to do two Euler steps and then take an average (Runge-Kutta 2). For example,

$$x^* = x^n + v_x^n \times \Delta t,$$
$$x^{n+1} = 0.5 \times [x^n + (x^* + v_x^* \times \Delta t)]\,.$$

Modify the `update()` subroutine in the `physics` module to enable RK2. You might need to add new temporal star variables in module `Simulation_data` to store the starred variables. Redo part (a) with the second order RK2 method and compare the convergence behavior with the original first order scheme. [20pts]

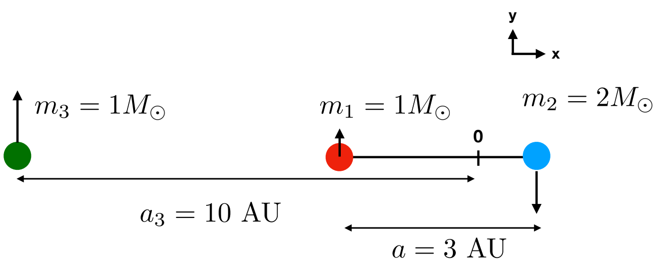

Figure 1: The initial setup of the triple star system in Problem 3.

4. *(Optional)* If you write your code properly, your code should be able to handle n-body simulations by only changing the number of stars N in your simulation data module `Simulation_data` and the initial condition. Build a solar system simulator with parameters taken from `https://nssdc.gsfc.nasa.gov/planetary/factsheet/` (Sun, Moon, and 9 planets). Can your simulator maintain the Earth's orbit for 1000 orbits? [10pts]