

Orbital behavior around a nonuniform celestial body

This content has been downloaded from IOPscience. Please scroll down to see the full text.

2016 J. Phys.: Conf. Ser. 750 012022

(<http://iopscience.iop.org/1742-6596/750/1/012022>)

View [the table of contents for this issue](#), or go to the [journal homepage](#) for more

Download details:

IP Address: 134.94.112.226

This content was downloaded on 15/05/2017 at 14:46

Please note that [terms and conditions apply](#).

You may also be interested in:

[Timeresolved Spectroscopy of V Sagittae](#)

Marcos P. Diaz

[Physical Nature and Orbital Behavior of V523 Cas](#)

Ronald G. Samec, Danny R. Faulkner and David B. Williams

[Searching sequences of resonant orbits between a spacecraft and Jupiter](#)

J K S Formiga and A F B A Prado

[Orbital Stability of Extrasolar Planets](#)

M. Noble, Z. E. Musielak and M. Cuntz

[PITCH ANGLE RESTRICTIONS IN LATE-TYPE SPIRAL GALAXIES BASED ON CHAOTIC AND ORDERED](#)

[ORBITAL BEHAVIOR](#), E. Pichardo, E. Moreno et al.

[Chaotic Orbits of Disk-Star-Planet Systems](#)

Ing-Guey Jiang and Li-Chin Yeh

[THE PHYSICAL NATURE AND ORBITAL BEHAVIOR OF THE ECLIPSING SYSTEM DK CYGNI](#)

Jae Woo Lee, Jae-Hyuck Youn, Jang-Ho Park et al.

[TEMPORARY CAPTURE OF PLANETESIMALS BY A PLANET FROM THEIR HELIOCENTRIC ORBITS](#)

Ryo Suetsugu, Keiji Ohtsuki and Takayuki Tanigawa

[Comparing the behavior of orbits in different 3D dynamical models for elliptical galaxies](#)

Euaggelos E. Zotos

Orbital behavior around a nonuniform celestial body

Z Rosson, F Hall and T Vogel

Department of Physics, Stetson University, DeLand, FL 32723, USA

Email: zachrosson@gmail.com

Abstract. To effectively model the orbit around a nonuniform celestial body, detailed information needs to be determined of the perturbations. This research looked at one of the most crucial perturbations: the nonuniform gravitational field of a celestial body. Given an orbiting particle (a satellite), we utilized numerical methods to calculate its orbit in two dimensions around a discretized center mass structure. The gravitational acceleration imposed on the particle due to each mass point sums vectorally as the particle completes each infinitesimal time step of one orbit. There are noticeable effects on the orbit as the conditions of the center mass change. The development of a simulation code allows for the modelling of the orbit about an irregular body with satisfactory accuracy.

1. Introduction

The gravitational effects due to nonuniformities have been crucial to understanding orbits around the Moon, and the presence of mass-concentrations (“mascons”) has been the focus of NASA’s recent GRAIL (Gravity Recovery and Interior Laboratory) missions [1]. Over billions of years, mascons formed on the surface of the Moon as lunar material melted due to the heat of asteroids’ impact, which changed the density of the surface rock. This caused an increase in the gravitational field above the mascons due to the increased mass density. Additionally, the European Space Agency’s recent Rosetta mission, (the successful orbit around the 67P/Churyumov-Gerasimenko comet), served as a source of motivation for this research due to the comet’s irregular shape and its particularly nonuniform gravitational field.

Orbital course adjustments must be made to account for these gravitational perturbations due to shape or mass density irregularities. In this research, we model these anomalies and simulate orbits around the nonuniform celestial bodies. For this, advanced numerical methods and gravitational potential functions have been used [2], but we attempt to approach the physical situation in a more straightforward manner while still keeping sufficient computational accuracy. This is not only because valuable orbital information due to nonuniformities can still be garnered using a more simple discretization method, but because this approach presents a novel teaching tool for students to understand and visualize orbits around more complex celestial bodies.

2. Methods

2.1 Basic physics and algorithm

The fundamental law guiding the physics of our research was Isaac Newton’s Universal Law of Gravitation (1687),



$$F = -\frac{GMm}{r^2}$$

where G is the gravitational constant, M is the mass of the center body, m is the mass of the orbiting satellite, and r is the distance separating the two bodies. The inverse square distance dependence gives rise to the elliptical behavior which one expects in gravitational orbits. For the purposes of modeling, we utilized a two-dimensional plane, since relevant orbital motion in three dimensions can often be reduced to motion in a suitably chosen orbital plane. We determined the x and y components of the acceleration experienced by the orbiting particle (making the assumption that its mass is negligible compared to the center body):

$$a_x = -\frac{GMx}{(x^2 + y^2)^{3/2}}$$

$$a_y = -\frac{GMy}{(x^2 + y^2)^{3/2}}$$

Then, to represent the center body, we utilized a flood-fill algorithm which populates a grid of mass points based on a defined shape and point density. To account for all these infinitesimal mass contributions, we summed vectorally the acceleration due to each of the n center mass points in the discretized grid:

$$a_{nx, tot} = \sum_{i=1}^n a_{ix}$$

$$a_{ny, tot} = \sum_{i=1}^n a_{iy}$$

The orbiting particle was given an initial y -velocity and x -position at the start of its orbit, and we utilized the structure of the Euler-Cromer algorithm to solve these differential equations for its equations of motion [3]:

$$v_x = v_x' + a_{nx, tot} * dt$$

$$v_y = v_y' + a_{ny, tot} * dt$$

$$x = x' + v_x * dt$$

$$y = y' + v_y * dt$$

The prime notation represents the velocity and position values from the previous time step to be used in the current computation. Essentially, the cumulative effect of the gravitational acceleration from each mass point is used to actively compute the velocity and position of the orbiting particle for every infinitesimal time step (dt). This process is performed iteratively until one full orbital period has passed. Our procedure deviates from the Euler-Cromer source equations only with the aforementioned summation of acceleration. Furthermore, we plotted the orbit using lists of x and y coordinates compiled during the iterations and printed values of acceleration, velocity, position, and time within small intervals at apoapsis and periapsis to test basic orbital energy conservation (i.e. convergence of initial conditions).

2.2 Justifications

To achieve sufficient accuracy with this method, the infinitesimal time step had to be chosen with regard to its ability to close an orbit but also to limit excessive computational run times. Table 1 shows a basic error analysis of a simple physical system to demonstrate this algorithm's accuracy for one orbital period at increasingly small time steps.

	Time (days)	x-position (km)	y-position (km)	x-velocity (km/day)	y-velocity (km/day)	x-acceleration (km/day ²)	y-acceleration (km/day ²)
Theoretical (initial values):	87.969000	57,910,000.000000	0.000000	0.000000	5,095,600.000000	-468,147.636200	0.000000
dt = 10 ⁻³ used, one orbit:	87.979000	57,909,999.993600	134.268711	-12.327646	5,095,600.000670	-468,147.627927	50.489716
Absolute error:	0.010000	0.006400	134.268711	12.327646	0.000670	0.008273	50.489716
dt = 10 ⁻⁴ used, one orbit:	87.979000	57,909,999.999200	137.723204	-12.652953	5,095,600.000050	-468,147.636161	3.784052
Absolute error:	0.010000	0.000800	137.723204	12.652953	0.000050	0.000039	3.784052
dt = 10 ⁻⁵ used, one orbit:	87.978970	57,910,000.000000	-15.110316	1.388230	5,095,600.000000	-468,147.636214	0.672334
Absolute error:	0.009970	0.000000	15.110316	1.388230	0.000000	0.000014	0.672334

Table 1. Values for an orbiting particle (Mercury) around a single mass point (Sun), with basic error analysis of the Euler-Cromer method for one orbital period to test convergence back to initial values used from analytical expressions. In this case, computational times increase by a factor of ten with each decrease in infinitesimal time step, with $dt = 10^{-3}$ requiring ~30 seconds for the code to run. As such, we utilized $dt = 10^{-4}$ to compensate accuracy with computational run time.

The choice to utilize the Euler-Cromer algorithm instead of a more stable, advanced method was multifaceted. First, we didn't study long term stability of our produced orbits and thus didn't encounter significant error accumulation issues. We wanted to properly visualize the changes in an orbit due to changing gravitational conditions, which was evident in just one orbital period. Furthermore, to accomplish these simulations, we needed to construct a functional code which would do the necessary computations of the vector sum of gravitational acceleration at each time step. The well-known 4th-order Runge-Kutta method requires the calculations of multiple coefficients within each time step [4]. Utilizing more advanced differential equation solvers such as this could have slowed code development unnecessarily when considered in addition to producing code which accounted for the summed acceleration from each mass point. Second, given a small enough time step, the Euler-Cromer algorithm produces a sufficiently closed orbit as evident in table 1. It has also been shown to produce similarly accurate results to the Runge-Kutta method with these exceedingly small time steps for short periods of run time [5].

In the literature, others have studied more advanced physical systems of small celestial bodies with nonuniform gravitational fields using spherical harmonics in potential functions, and triaxial ellipsoid or polyhedrons for their geometric setups while presumably using more accurate differential equation solvers [6]. Others have created detailed density and shape models of moons like Phobos which include gravitational information [7]. Our research doesn't delve into these advanced mathematics or topography but does create an elementary way to model mass distribution changes with discretized finite elements while providing a possible teaching mechanism. There are existing interactive physical models for teaching simple Galilean trajectory problems, but not for gravity in such a way where nonuniformities can be easily created and their effects observed and analyzed [8].

3. Results

The creation of our code and simulation system allowed us to model two-dimensional representations of small celestial bodies and orbits around them. We utilized our flood-fill algorithm described previously to populate a grid of some shape, which produced a list of mass point coordinates to be read into our program. We were able to manipulate these lists of values to produce highly irregular shapes and mass-concentrations. As expected, there were clear effects on the orbit as the conditions of the center mass distribution changed. As the complexity of these physical systems increased, so did the sensitivity to the initial conditions and infinitesimal time step for the orbiting particle. We used basic analytical

equations for energy conservation, gravitational force, and orbital period to produce guesses for these initial conditions, but had to interpolate on some of these values manually to close the orbit when the mass distributions changed. In particular, the orbiting particle's motion was very sensitive to its initial y-velocity and its x-distance from the surface of the center body. As such, these are the values that showed the clearest changes (along with the orbital period) due to increased mass distribution complexity when other initial conditions were held constant. Figure 1 shows this behavior occurring for a mass distribution only slightly misshapen compared to a large orbital radius and not covering a large spatial extent compared to that of a single mass point. Given the same initial conditions used for one single mass point, we observe that the orbit does not close for the same time interval and would require a slightly higher value for the initial y-velocity to do so.

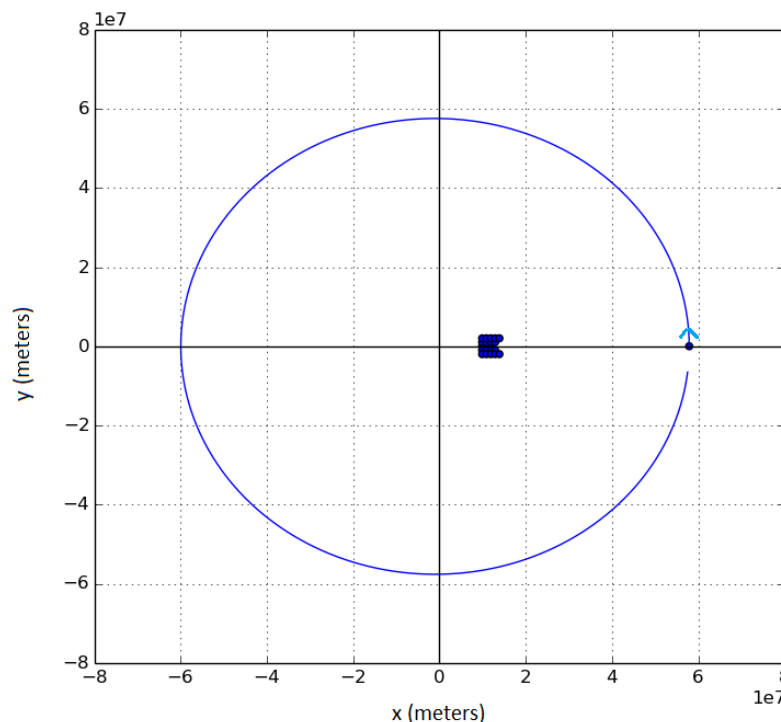


Figure 1. Arbitrary center body shape with 21 center mass points. Using the initial values from table 1 with $dt = 10^{-4}$, we observe lack of closure of the orbit due to the changing shape of the center body.

In the examples shown in Figure 2, we greatly increase the number of discretized mass points for an ellipse-shaped moon of mass on the order of 10^{22} kg. As we deviate from circular symmetry, the initial values require interpolation as described in the previous paragraph. Moreover, further deviations in mass density distribution also require more extensive interpolation. After discerning the proper initial velocity for the first arbitrary ellipse case on the next page, we manipulated the lists of x and y coordinates of the mass points to include mass-concentrations in asymmetric locations. Without changing the initial velocity before running the simulation, we observe that the orbiting particle crashes into the center body (Figure 2(b)). This is analogous to the issues being studied by NASA's GRAIL missions described in the introduction. Figure 2(c) shows a closure of the orbit around the celestial body with mass-concentrations given the proper initial velocity which was on the order of 10^2 m/s greater than that of figure(a) and figure(b). This illustrates not only the fundamental sensitivity of orbital motion around nonuniform bodies to the initial velocity of the orbiting satellite, but also that these bodies can allow closed orbits around them when the orbiting particle is projected at the correct speed given its distance from the surface.

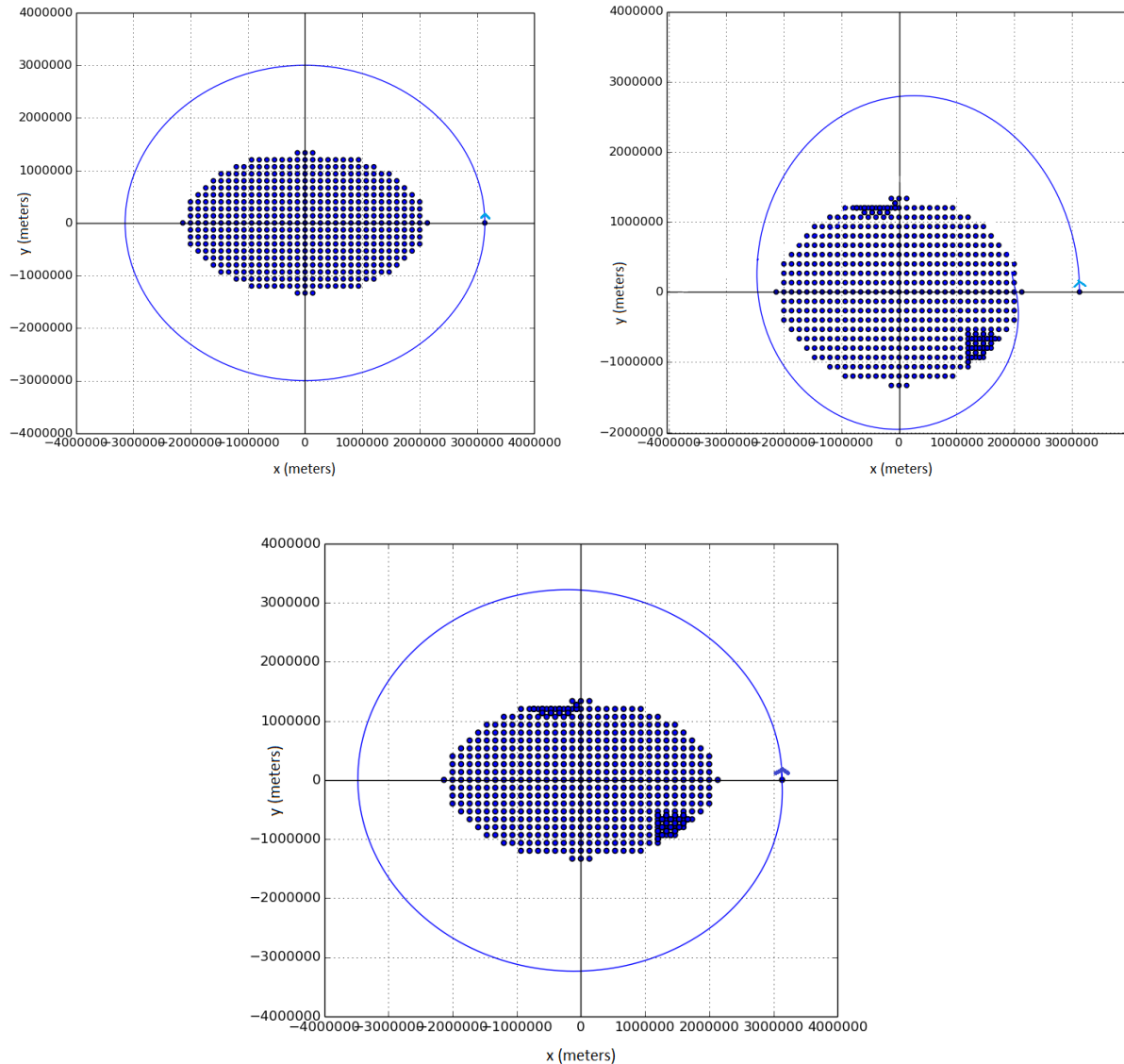


Figure 2. (a) Top left: Closed orbit around uniform moon ellipse shape with 501 center mass points. (b) Top right: Attempted orbit around non-uniform body with 629 center mass points, (including “mascons”), with the same initial conditions for the speed and position as in (a), resulting in a crash. (c) Bottom: Orbit around same distribution as (b), with adapted initial velocity for a closed orbit.

4. Conclusion

This research succeeded in modeling gravitational anomalies using basic physical properties in a logical but enhanced manner. The technique of summing the gravitational influence of each center mass point in the grid for each time step to represent shape and density changes could be applied in other computational physics settings. This method could also be used as a novel teaching tool to depict gravitational nonuniformities of celestial bodies in an easily understandable manner, using basic Newtonian mechanics and numerical methods.

Furthermore, much can be done to optimize code performance using parallel processing or through employing more accurate differential equation solvers to solve related problems of long term

orbital stability and error accumulation reduction. Also, the physical problem could be extended to include other perturbations including orbital inclination in the third dimension, outgassing, or relativity.

Acknowledgments

We would like to thank the Stetson University College of Arts and Sciences Dean's Fund, the Stetson Physics Department, and the Center for Simulation Physics of the University of Georgia for funding.

References

- [1] Melosh H, Freed A et al 2013 The origin of lunar mascon basins *Science* **340** 6140 pp 1552-1555
- [2] Rossi A, Marzari F and Farinella P 1999 Orbital evolution around irregular bodies *Earth, Planets and Space* **51** 11 pp 1173-1180
- [3] Gould H, Tobochnik J and Christian W 1988 *An Introduction to Computer Simulation Methods: Applications to Physical Systems* (Reading: Addison-Wesley)
- [4] Newman M 2013 *Computational Physics* (Createspace) chapter 8
- [5] Burden R and Faires J 1993 *Numerical Analysis* (Boston: PWS-Kent Publishing)
- [6] Cui P and Qiao D 2014 The present status and prospects in the research of orbital dynamics and control near small celestial bodies *Theoretical and Applied Mechanics Letters* **4** 1 013013
- [7] Willner K, Shi X and Oberst J. 2014 Phobos' shape and topography models *Planetary and Space Science* **102** pp 51-59
- [8] Jimoyiannis A and Komis V 2001 Computer simulations in physics teaching and learning: a case study on students' understanding of trajectory motion *Computers & Education* **36** 2 pp 183-204