Computer Modeling Homework in Introductory Mechanics

Michael F. Schatz[1], Marcos D. Caballero[1], Matthew A. Kohlmyer[2] and John B. Burk[3]

[1]School of Physics, Georgia Institute of Technology, Atlanta, GA 30332 [2]Department of Physics, North Carolina State University, Raleigh, NC 27695

[3]High School Physics, The Westminster Schools, Atlanta, GA 30327

Georgia Institute of Technology

Abstract

We present an overview of the computational component in our large (N ∼ 500) introductory physics course and the development of homework exercises to enhance students' understanding of numerical computation and visualization introduced in the mechanics course.

Computation in Intro. Physics

Students taking introductory physics are rarely exposed to computation (using numerical methods to model systems and solve complex problems). Computation can provide students with new opportunities not afforded to them by a standard approach to physics [1]. A course that includes computation allows students to explore "intractable" systems, simulate "impossible" experiments and visualize problems more readily.

At Georgia Tech, we have taught computation using VPython [2] in an introductory course based on the Matter and Interactions (M&I) textbook. VPython is conveniently coupled to M&I allowing us to leverage our years of experience with teaching M&I. While our implementation builds on our M&I experience, it is not limited to it.

With the increasing demand on our graduates to be fluent with computation in the work place, we have begun to extend the computational experience beyond the laboratory. Students taking the M&I course at Georgia Tech utilize their lab-developed code to solve new and different problems on their homework sets.

References

[1] R. Chabay, B. A. Sherwood. *Matter and Interactions*, 3rd ed., Wiley and Sons, 2010

[2] Freely available at `vpython.org`.

Funding

Supported by the National Science Foundation DUE-0618519 & DUE-0942076

Design Philosophy

No Programming Experience

We aimed to develop an instructional strategy that helps computation permeate the course, *but does not require that students have previous programming experience*.

Easily Deployable

This implementation had to be easily *deployable across large lecture sections*; the setting in which most introductory calculus-based courses are taught.

Informed by Professional Practice

Our philosophy was informed by what research scientists do this quite often; they write a program to solve a problem and then alter that program to solve a different problem that is of interest to them.

Lots of Practice

We envisioned developing computational activities that would *start with guided inquiry* and exploration in the laboratory *followed by independent practice* on homework.

Final Result

Students *work in groups and with TAs in the laboratory* to develop a program that solves a problem. Students then *use that program individually* to solve a variety of problems on their homework by *making any modifications that are necessary*.

Sample Computational Lab

In this lab, students write a program that integrates the system of two gravitationally interacting bodies.

They must prepare the objects, setup the loop structure, write the force law and update the momentum and position of the less massive object (i.e., a spacecraft).

After completing the lab, students have a VPython code that will integrate this system for any arbitrary amount of time.

```
from __future__ import division
from visual import *

craft = sphere(pos = vector(10e7,0,0), color = color.white, radius = 1e6)
Earth = sphere(pos = vector(0,0,0), color = color.blue, radius = 6.3e6)
trail = curve(color = craft.color)

G = 6.67e-11
mcraft = 1500
mEarth = 5.97e24

vcraft = vector(0,2400,0)
pcraft = mcraft*vcraft

t = 0
deltat = 60
tf = 365*24*60*60

while t < tf:

    r = craft.pos-Earth.pos
    rhat = r/mag(r)
    Fgrav = -G*mEarth*mcraft/mag(r)**2*rhat

    pcraft = pcraft+Fgrav*deltat
    craft.pos = craft.pos + pcraft/mcraft*deltat

    trail.append(pos = craft.pos)
    t = t + deltat

print 'Craft final position: ', craft.pos, 'meters.'
```

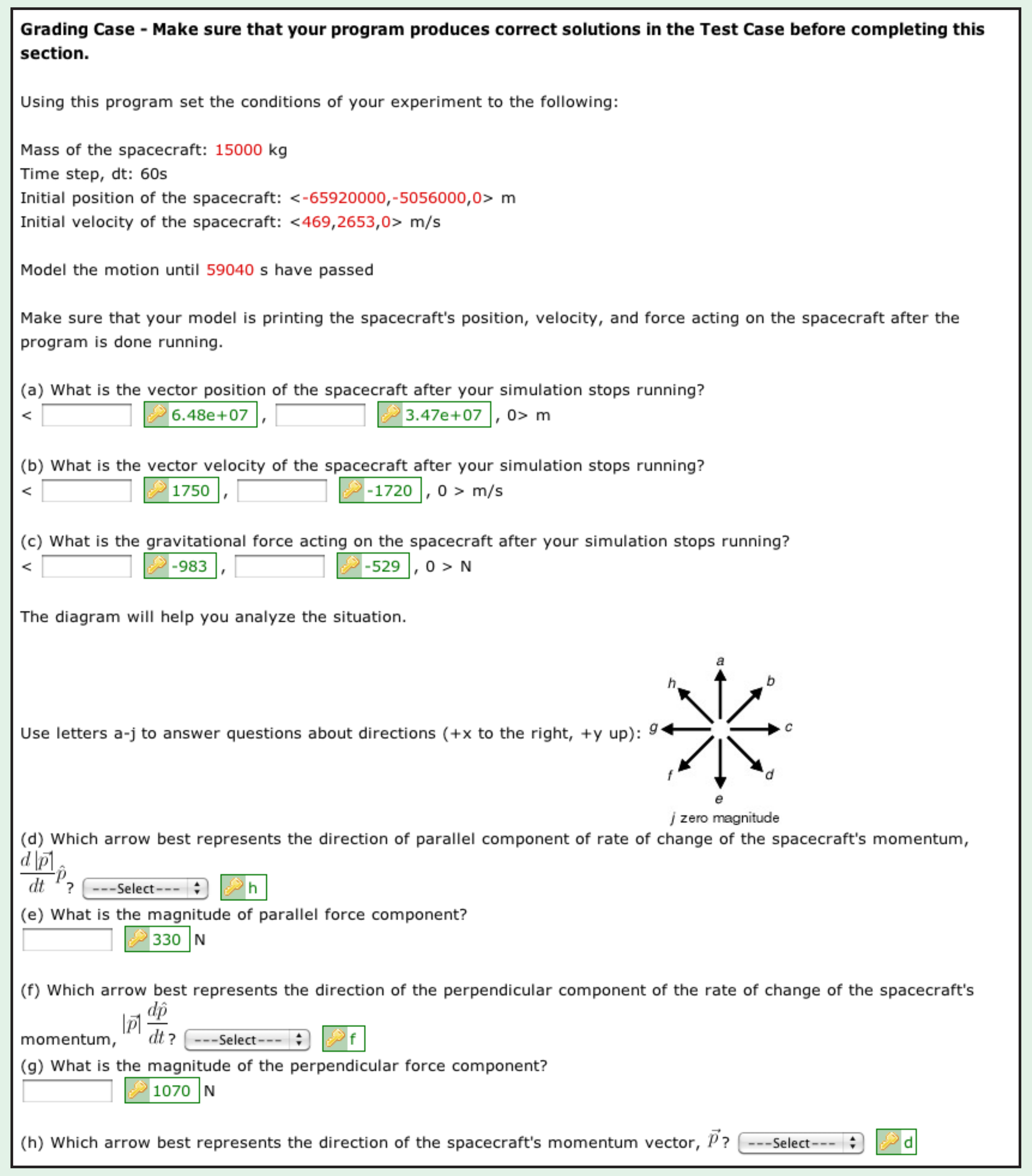
Initial Conditions (lines 4–17) **Force Calculation** (lines 21–23) **Newton's Second Law** (line 25) **Position Update** (line 26)

Sample Computational Homework

We have written more than a dozen laboratory and homework questions using the WebAssign homework system.

Initial conditions for these problems are randomized on a per student basis.

In the sample problem below, students were asked to write additional code to compute (and represent as arrows) $\frac{d|\vec{p}|}{dt}\hat{p}$ and $|\vec{p}|\frac{d\hat{p}}{dt}$ (i.e., the radial and tangential components of the net force).



Grading Case - Make sure that your program produces correct solutions in the Test Case before completing this section.

Using this program set the conditions of your experiment to the following:

Mass of the spacecraft: 15000 kg
Time step, dt: 60s
Initial position of the spacecraft: <-65920000,-5056000,0> m
Initial velocity of the spacecraft: <469,2653,0> m/s

Model the motion until 59040 s have passed

Make sure that your model is printing the spacecraft's position, velocity, and force acting on the spacecraft after the program is done running.

(a) What is the vector position of the spacecraft after your simulation stops running?
< [ ] 6.48e+07 , [ ] 3.47e+07 , 0> m

(b) What is the vector velocity of the spacecraft after your simulation stops running?
< [ ] 1750 , [ ] -1720 , 0 > m/s

(c) What is the gravitational force acting on the spacecraft after your simulation stops running?
< [ ] -983 , [ ] -529 , 0 > N

The diagram will help you analyze the situation.

Use letters a-j to answer questions about directions (+x to the right, +y up):

(d) Which arrow best represents the direction of parallel component of rate of change of the spacecraft's momentum, $\frac{d|\vec{p}|}{dt}\hat{p}$? ---Select--- h

(e) What is the magnitude of parallel force component?
[ ] 330 N

(f) Which arrow best represents the direction of the perpendicular component of the rate of change of the spacecraft's momentum, $|\vec{p}|\frac{d\hat{p}}{dt}$? ---Select--- f

(g) What is the magnitude of the perpendicular force component?
[ ] 1070 N

(h) Which arrow best represents the direction of the spacecraft's momentum vector, $\vec{p}$? ---Select--- d

Schedule

Laboratory Activity	Homework Exercise
Introduction to Computation	No Homework
1D Constant Force	2D Constant Force, *Alter Initial Conditions (ICs)*
Gravitational Orbit I	Gravitational Orbit I, *Alter ICs & Create Arrows for $\vec{p}$ and $\vec{a}$*
No Associated Lab	Electric Force HW, *Incomplete Code: Add ICs & Num. Integration*
Gravitational Orbit II	Gravitational Orbit II, *Alter ICs & Create Arrows for $\frac{d|\vec{p}|}{dt}\hat{p}$ and $|\vec{p}|\frac{d\hat{p}}{dt}$*
Gravitational Orbit III	Gravitational Orbit III, *Alter ICs & Add'l Calculations*
Spring Motion	Spring Motion I, *Alter ICs & Create Arrows for $\frac{d|\vec{p}|}{dt}\hat{p}$ and $|\vec{p}|\frac{d\hat{p}}{dt}$*
	Spring Motion II, *Alter ICs & Add'l Calculations*
Air Resistance	Air Resistance, *Alter ICs & Create Arrows for $\frac{d|\vec{p}|}{dt}\hat{p}$ and $|\vec{p}|\frac{d\hat{p}}{dt}$*
Spring Energy	Anharmonic Spring HW, *Inc. Code: Add ICs & Num. Integration*
No Associated Lab	Lennard-Jones HW, *Inc. Code: Add ICs & Num. Integration*
Evaluation Assignment	Central Force Problem, *Inc. Code: Add ICs & Num. Integration*

Challenges to Student Success

Students' programs are not reviewed, only their numerical solutions are graded.

Some programs are susceptible to round-off error; numerical solutions are marked incorrect by WebAssign.

Current & Future Work

We have begun to:

- determine how computation challenges students,
- assess students' attitudes towards computation
- and diversify implementation beyond colleges.