Computation in Classical Mechanics

J. E. Hasbun

Department of Physics
University of West Georgia

Scientific advances create the need to become computationally adept to tackling problems of increasing complexity. The use of computers in attaining solutions to many of science's difficult problems is inevitable. Therefore, educators face the challenge to infuse the undergraduate curriculum with computational approaches that will enhance students' abilities and prepare them to meet the world's newer generation of problems. Computational physics courses are becoming part of the undergraduate physics landscape and learned skills need to be honed and practiced. A reasonable ground to do so is the standard traditional upper level physics courses. I have thus developed a classical mechanics textbook[1] that employs computational techniques. The idea is to make use of numerical approaches to enhance understanding and, in several cases, allow the exploration and incorporation of the "what if environment" that is possible through computer algorithms. The textbook uses Matlab because of its simplicity, popularity, and the swiftness with which students become proficient in it. The example code, in the form of Matlab scripts, is provided not to detract students from learning the underlying physics. Students are expected to be able to modify the code as needed. Efforts are under way to build OSP[2] Java programs that will perform the same tasks as the scripts. Selected examples that employ computational methods will be presented.

[1] To be published, Jones and Bartlett Publishers.

[2] Open Source Physics: http://www.opensourcephysics.org/.

Table of Contents

Chapter 1 Highlights

Why we need computational physics? We can go beyond solvable problems. We can get more insight. We can explore situations beyond classroom examples.

Start with the iterative Euler method.

Chapter 1 Highlights

Why we need computational physics? We can go beyond solvable problems. We can get more insight. We can explore situations beyond classroom examples.

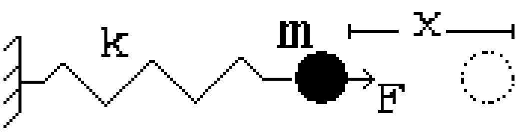
If we know the acceleration of an object,

$$\int dv = \int a\, dt$$

If the acceleration is constant, an object's velocity is

$$v(t) = v_0 + at \quad \Rightarrow \quad \int dx = \int v(t)dt \quad \Rightarrow \quad x(t) = x_0 + v_0 t + \frac{1}{2} a t^2$$

However, if the acceleration is not constant, say a mass at the end of a spring,



$$F_s = -kx(t) \qquad a(t) = -k\,x(t)/m = \frac{d^2x}{dt^2} = \frac{dv}{dt}$$

The analytic solution is done in a later chapter. Let's look at a numerical solution. MATLAB code is provided. Students are encouraged to run it and explore it.

•The Euler Method to solve a 2nd order DE: convert it to two 2nd order DE's

$$\frac{dx}{dt} = v(t,x) \quad \text{and} \quad \frac{dv}{dt} = a(t,v)$$

So that we do,

$$v_{i+1} = v_i + a_i\,\Delta t, \qquad x_{i+1} = x_i + v_{i+1}\,\Delta t$$

to be solved on $[t_0, t_f]$ with

$$t_{i+1} = t_i + \Delta t, \qquad a_i = -k\,x_i / m.$$

For N steps $\Delta t = \left(t_f - t_0\right)/N$

Given initial conditions: x_0, v_0

First example, by calculator (reproduced by a general force MATLAB code): let

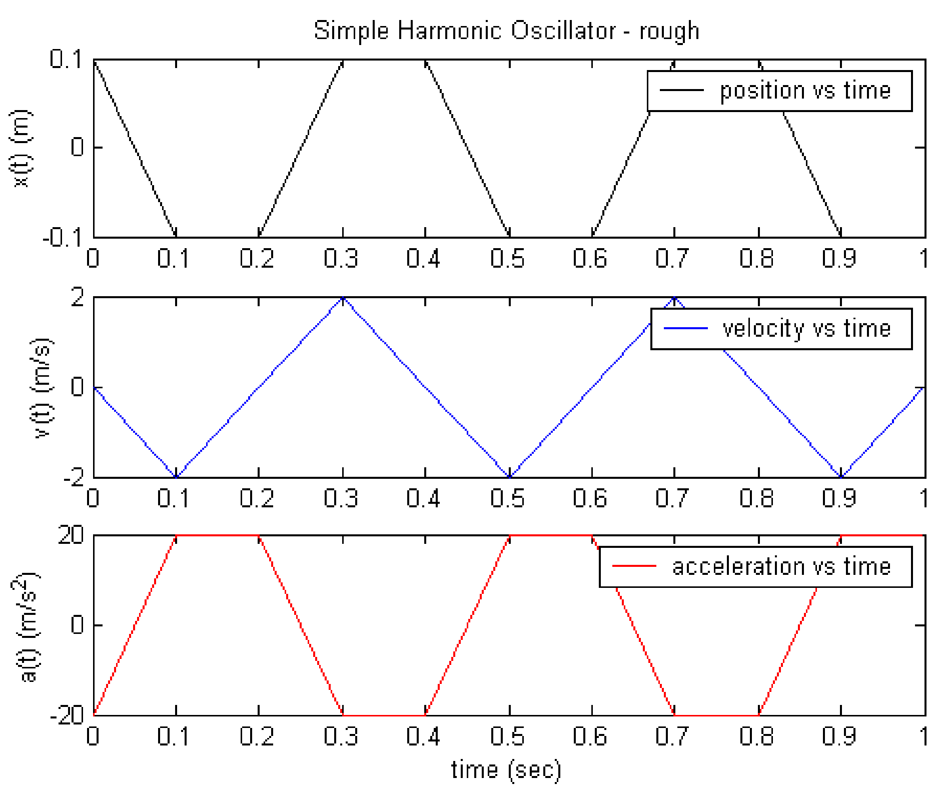
$k = 1000 N/m$, $m = 5kg$, $x_0 = 0.1m$, $v_0 = 0.0 m/s$ on time interval $[0, 1s]$

for $N = 10$ so that $\Delta t = 0.1$

- Create a table of the calculations

i	$t_i = i\,\Delta t$	$v_{i+1} = v_i + a_i\,\Delta t$	$x_{i+1} = x_i + v_{i+1}\,\Delta t$	$a_i = -200 x_i / m$
0	0.0	0.0	0.1	-20
1	0.1	-2.0	-0.1	20
2	0.2	0.0	-0.1	20
3	0.3	2.0	0.1	-20
4	0.4	0.0	0.1	-20
5	0.5	-2.0	-0.1	20
6	0.6	0.0	-0.1	20
7	0.7	2.0	0.1	-20
8	0.8	0.0	0.1	-20
9	0.9	-2.0	-0.1	20
10	1.0	0.0	-0.1	20

- Can create a plot of this rough calculation



Matlab Code for the above example

```
%ho1.m
%Calculation of position, velocity, and acceleration for a harmonic
%oscillator versus time. The equations of motion are used for small time intervals
clear;
%NPTS=100;TMAX=1.0;%example Maximum number of points and maximum time
TTL=input(' Enter the title name TTL:','s');%string input
NPTS=input(' Enter the number calculation steps desired NPTS: ');
TMAX=input(' Enter the run time TMAX: ');
NT=NPTS/10;%to print only every NT steps
%K=1000;M=5.0;C=0.0;E=0.0;W=0.0;x0=0.1;v0=0.0;% example Parameters
K=input(' Enter the Spring contant K: ');
M=input(' Enter the bob mass M: ');
C=input(' Enter the damping coefficient C: ');
E=input(' Enter the magnitude of the driving force E: ');
W=input(' Enter the driving force frequency W: ');
x0=input(' Enter the initial position x0: ');% Initial Conditions
v0=input(' Enter the initial velocity v0: ');% Initial Conditions
t0=0.0;% start at time t=0
dt=TMAX/NPTS;%time step size
fprintf(' Time step used dt=TMAX/NPTS=%7.4f\n',dt);%the time step being used
F=-K*x0-C*v0+E*sin(W*t0); % initial force
a0=F/M;% initial acceleration
fprintf('    t       x       v       a\n');%output column labels
v(1)=v0;
x(1)=x0;
a(1)=a0;
t(1)=t0;
fprintf('%7.4f %7.4f %7.4f %7.4f\n',t(1),x(1),v(1),a(1));%print initial values
for i=1:NPTS
   v(i+1)=v(i)+a(i)*dt;                 %new velocity
   x(i+1)=x(i)+v(i+1)*dt;                %new position
   t(i+1)=t(i)+dt;                      %new time
   F=-K*x(i+1)-C*v(i+1)+E*sin(W*t(i+1));  %new force
   a(i+1)=F/M;                          %new acceleration
% print only every NT steps
   if(mod(i,NT)==0)
      fprintf('%7.4f %7.4f %7.4f %7.4f\n',t(i+1),x(i+1),v(i+1),a(i+1));
   end;
end;
```

ho1.m continued on next page

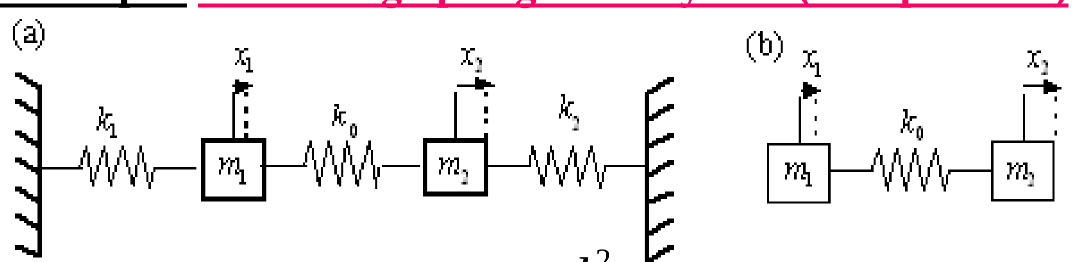
ho1.m continued from previous page

```
subplot(3,1,1)
plot(t,x,'k-');
ylabel('x(t) (m)','FontSize',14);
h=legend('position vs time'); set(h,'FontSize',14);
title(TTL,'FontSize',14);
subplot(3,1,2)
plot(t,v,'b-');
ylabel('v(t) (m/s)','FontSize',14);
h=legend('velocity vs time'); set(h,'FontSize',14)
subplot(3,1,3)
plot(t,a,'r-');
ylabel('a(t) (m/s^2)','FontSize',14);
xlabel('time (sec)','FontSize',14);
h=legend('acceleration vs time'); set(h,'FontSize',14)
```

•We also need to be able to visualize analytical solutions – so use small MATLAB scripts provided or modify available ones

•**Harmonic Motion example:** **Interacting Spring-Mass System (Computation)**

Interaction mass-spring system a)with walls, and b) without walls



$$m_1 \frac{d^2 x_1}{dt^2} = -k_1 x_1 - k_0 (x_1 - x_2)$$ and $$m_2 \frac{d^2 x_2}{dt^2} = -k_2 x_2 - k_0 (x_2 - x_1)$$

•Case 1: No Walls - Single Mode $k_1 = k_2 = 0$ The analytic solution is:

$$x_1(t) = x_{cm}(t) - x_{cm0} - \frac{m_2}{m_1 + m_2} x_r(t)$$

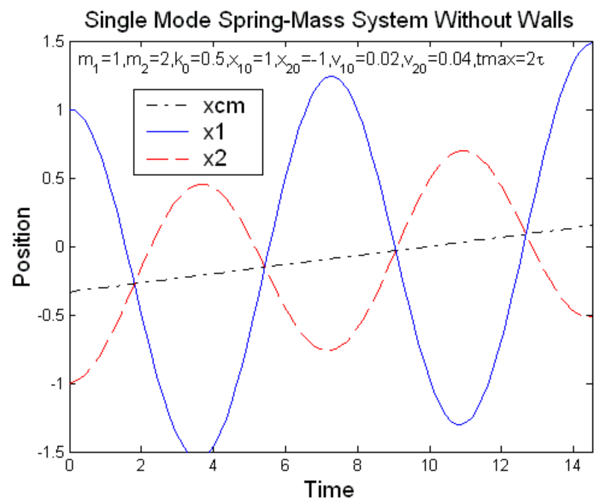
$$x_2(t) = x_{cm}(t) - x_{cm0} + \frac{m_1}{m_1 + m_2} x_r(t)$$

$$x_r = A \sin \omega t + B \cos \omega t, \quad \omega A = v_{r0} = v_{20} - v_{10}, \quad B = x_{r0} = x_{20} - x_{10}$$

$$x_{cm}(t) - x_{cm0} = \frac{m_1 x_1 + m_2 x_2}{m_1 + m_2} = v_{cm} t \qquad v_{cm} = \frac{m_1 v_1 + m_2 v_2}{m_1 + m_2} = \frac{m_1 v_{10} + m_2 v_{20}}{m_1 + m_2}$$

•Can create a plot of this calculation inter_spr1.html

The coupled mass-spring system without walls with a single mode of vibrations



•Case2: Full System - Bimodal $m_1 = m_2 = m, \quad k_1 = k_2 = k \neq k_0$

Write the equations in the matrix form $m\,\ddot{x} = -k\,x - k_0 M\,x$

where $m \equiv \begin{pmatrix} m & 0 \\ 0 & m \end{pmatrix} \quad x \equiv \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad k \equiv \begin{pmatrix} k & 0 \\ 0 & k \end{pmatrix} \quad M \equiv \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$

Solve using eigenvalue-eigenvector method, get two modes

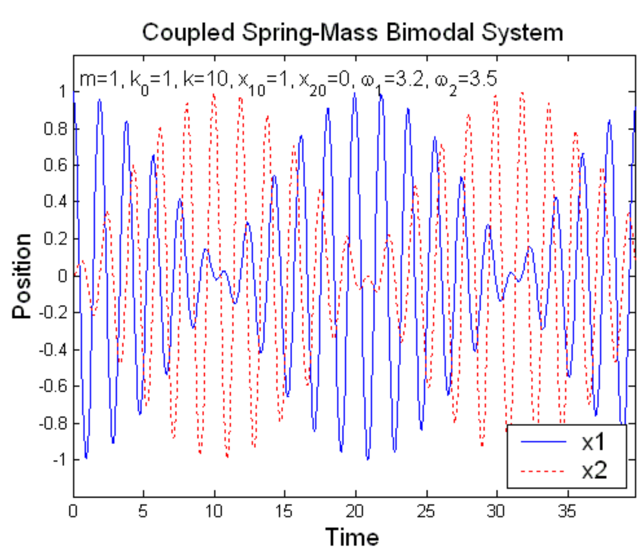
$$x_1(t) = x_{10}\cos\bar{\omega}t\cos\omega_m t + x_{20}\sin\bar{\omega}t\sin\omega_m t$$

$$x_2(t) = x_{10}\sin\bar{\omega}t\sin\omega_m t + x_{20}\cos\bar{\omega}t\cos\omega_m t$$

Average frequency $\bar{\omega} = (\omega_1 + \omega_2)/2$ Modulation frequency $\omega_m = (\omega_2 - \omega_1)/2$

•Can create a plot of this calculation inter_spr2.html

The solution of the full coupled spring-mass bimodal system



- **Three Dimensional Motion** of a **charged Particle in an Electromagnetic Field (Computation)** - This follows the two dimensional analytic solutions of the charge in Electric, magnetic, and joint E& B fields

We have

$$\vec{F} = q\vec{v} \times \vec{B} + q\vec{E} = m\vec{a} \qquad \text{or}$$

$$\frac{d^2x}{dt^2} = q(v_y B_z - v_z B_y + E_x)/m, \quad \frac{d^2y}{dt^2} = q(v_z B_x - v_x B_z + E_y)/m, \quad \frac{d^2z}{dt^2} = q(v_x B_y - v_y B_x + E_z)/m$$

In MATLAB write these as

$$x \to r(1), \quad \dot{x} \to r(2); \quad y \to r(3), \quad \dot{y} \to r(4); \quad z \to r(5), \quad \dot{z} \to r(6)$$

Obtain six 1st order equations given by

$$\frac{dr(1)}{dt} = r(2), \quad \frac{dr(2)}{dt} = q\left[r(4)B(3) - r(6)B(2) + E(1)\right]/m$$

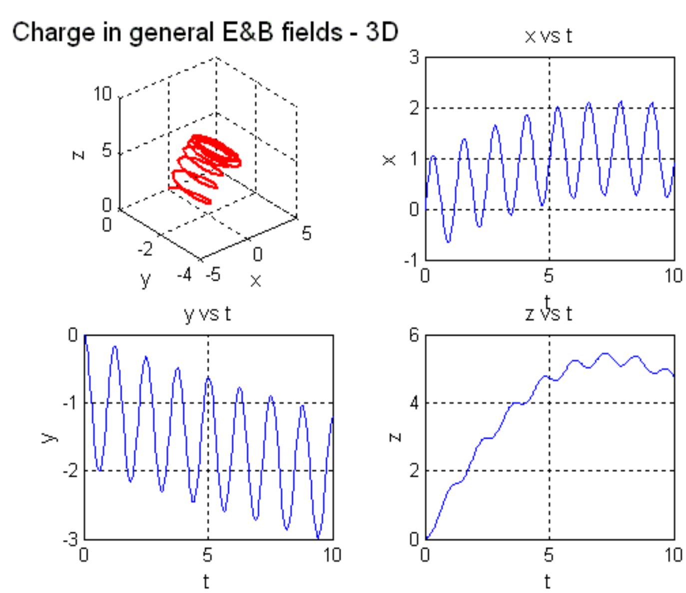
$$\frac{dr(3)}{dt} = r(4), \quad \frac{dr(4)}{dt} = q\left[r(6)B(1) - r(2)B(3) + E(2)\right]/m$$

$$\frac{dr(5)}{dt} = r(6), \quad \frac{dr(6)}{dt} = q\left[r(2)B(2) - r(4)B(1) + E(3)\right]/m$$

where we have the field arrays $E = (E_x, E_y, E_z) = E(1,2,3)$, $B = (B_x, B_y, B_z) = B(1,2,3)$

Field values example: $E = [0.5\times10^{-8}, 1\times10^{-9}, \text{-}3\times10^{-9}]$, $B = [1\times10^{-8}, \text{-}1\times10^{-9}, 5.13\times10^{-8}]$

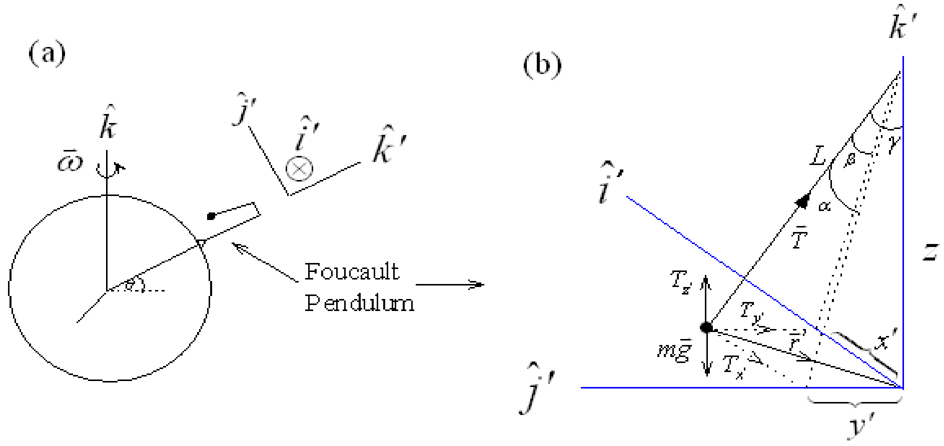
A charged particle moving in the presence of a three dimensional electromagnetic field



- see cycloid3d.html

Systems of Coordinates - Foucault pendulum (computation)

a) The Foucault pendulum and b) the forces on it.



S-frame (Earth's center) acceleration: $\vec{a} = -g\,\hat{k}' + \vec{T}/m = \ddot{\vec{r}}' + 2\vec{\varpi}\times\dot{\vec{r}}' + \vec{\varpi}\times\left(\vec{\varpi}\times\vec{r}'\right)$

Look at x-y plane motion, and ignore $\vec{\varpi}\times\left(\vec{\varpi}\times\vec{r}'\right)$, but keep the Coriolis term, and

$\vec{T} = T_x\hat{i}' + T_y\hat{j}', \quad \vec{r}' = x'\hat{i}' + y'\hat{j}',$

$$T_{x'} = -T\sin\alpha = -T\cos\beta = -T\,x'/L$$
$$T_{y'} = -T\sin\beta = -T\cos\alpha = -T\,y'/L$$

get
$$\ddot{x}' = 2\dot{y}'\omega_0\sin\theta - g\,x'/L$$
$$\ddot{y}' = -2\dot{x}'\omega_0\sin\theta - g\,y'/L$$

where for Earth $\omega_0 = 7.272\times10^{-5}\,rad/s$

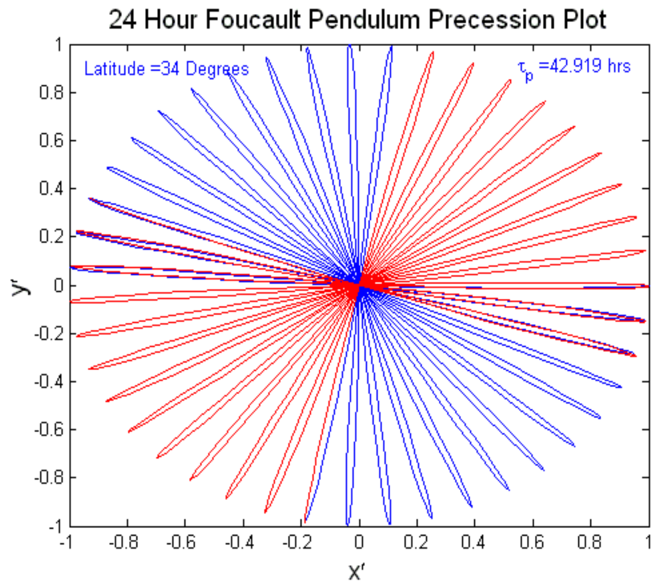
Solve and get:
$$x'(t) = x_0'\cos\omega_1 t\cos\omega_2 t + y_0'\sin\omega_1 t\cos\omega_2 t$$
$$y'(t) = -x_0'\sin\omega_1 t\cos\omega_2 t + y_0'\cos\omega_1 t\cos\omega_2 t$$

with

$\omega_f = \sqrt{g/L}$ ← Pendulum frequency $\qquad \omega_2 \equiv \sqrt{\omega_1^{\ 2} + \omega_f^{\ 2}}$

$\omega_1 = \omega_0\sin\theta$ ← Precessional frequency $\qquad \theta$ ← Latitude angle

Equations (7.8.9) for a Foucault pendulum with a 24 hour period



This is for a Foucault pendulum with a swinging period of one hour (very long!)

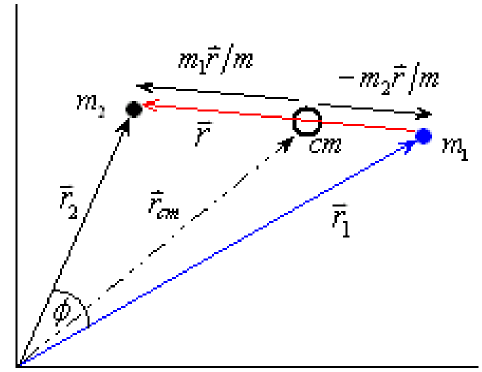
See Foucault.html

Gravitation: Binary Mass System Simulation

$$\vec{r}_1 = \vec{r}_{cm} - \frac{m_2}{m}\vec{r}$$

$$\vec{r}_2 = \vec{r}_{cm} + \frac{m_1}{m}\vec{r}$$

Center of mass of a binary mass system



Can write an equation for each mass

$$m_1\frac{d^2\vec{r}_1}{dt^2} = -\frac{Gm_1m_2\,\vec{r}_{12}}{r_{12}^3}, \quad m_2\frac{d^2\vec{r}_2}{dt^2} = -\frac{Gm_1m_2\,\vec{r}_{21}}{r_{21}^3}$$

$$\vec{r} \equiv \vec{r}_{21} = -\vec{r}_{12} = \vec{r}_2 - \vec{r}_1$$

But can also use Center of Mass - Relative Coordinate Method

$$\begin{pmatrix}\vec{r}_{cm}\\ \vec{r}\end{pmatrix} = \begin{pmatrix} m_1/m & m_2/m \\ -1 & 1\end{pmatrix}\begin{pmatrix}\vec{r}_1\\ \vec{r}_2\end{pmatrix}$$

and convert to an equation for the reduced mass $\rightarrow \frac{1}{\mu} = \frac{1}{m_1} + \frac{1}{m_2}$

$$\mu\frac{d^2\vec{r}}{dt^2} = -\frac{Gm_1m_2\vec{r}}{r^3}$$

← whose solution we know ⇒

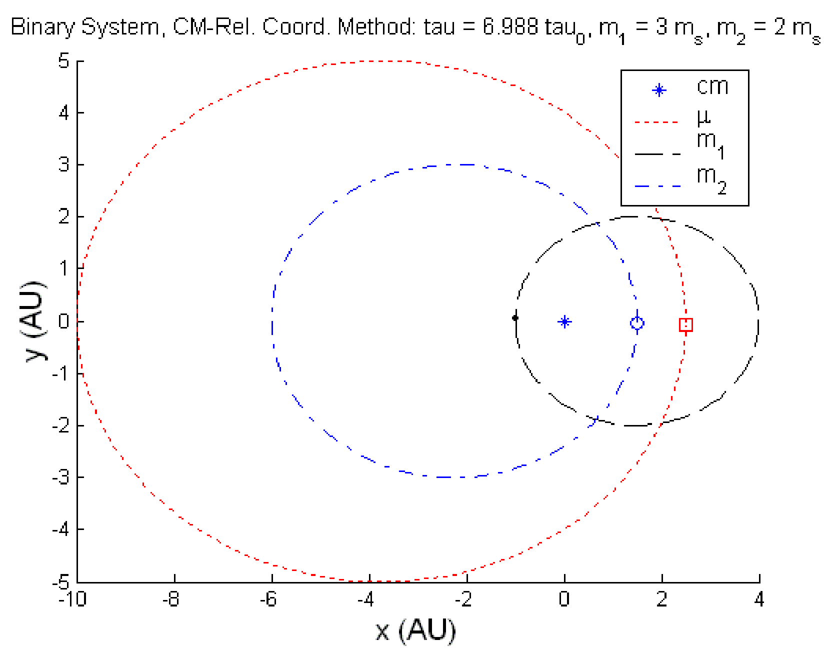
$$r = -\frac{L_\mu^2}{\mu K_\mu}\frac{1}{\left(1-\left[u_0L_\mu^2/\mu K_\mu\right]\cos\theta\right)}$$

or

$$r = \frac{v^2r^2}{G(m_1+m_2)}\frac{1}{\left(1+\frac{u_0v^2r^2}{G(m_1+m_2)}\cos\theta\right)} = r_{\min}\left(\frac{1+e}{1+e\cos\theta}\right)$$

Then get r_1 and r_2. Example follows:

Binary system simulation using analytic formulas



Using astronomical units

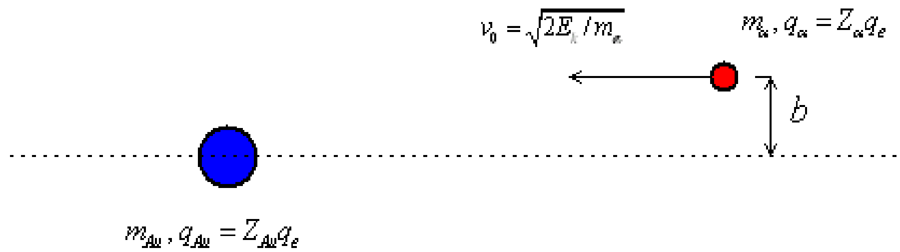
see binary1.html and binary1.avi

Rutherford Scattering (simulation)

$$q_{\text{target}} = Z_t q_e$$

$$q_{\text{projectile}} = Z_p q_e$$

Alpha particle with impact parameter directed at a target



$$m_p \frac{d}{dt}(v_x \hat{i} + v_y \hat{j}) = \frac{k q_e^2 Z_t Z_p}{\left(x^2 + y^2\right)^{3/2}} (x \hat{i} + y \hat{j})$$

← Projectile equation of motion with a fixed target

Use dimensionless units:
$$\frac{d}{d\bar{t}}(\bar{v}_x \hat{i} + \bar{v}_y \hat{j}) = \left[\frac{k q_e^2 \tau^2}{m_\alpha a_b^3}\right] \frac{\bar{K}(\bar{x} \hat{i} + \bar{y} \hat{j})}{\bar{m}\left(\bar{x}^2 + \bar{y}^2\right)^{3/2}}$$

take $a_b = 1\, fm$

$\bar{K} = Z_{Au} Z_\alpha$

$\bar{m} = 1$

and let
$$\frac{k q_e^2 \tau^2}{m_\alpha a_b^3} \equiv 1 \Rightarrow \tau = \sqrt{\frac{m_\alpha a_b^3}{k q_e^2}} = \sqrt{\frac{4(1.66 \times 10^{-27}\, kg)(10^{-15}\, m)^3}{(9 \times 10^9 \frac{Nm^2}{C^2})(1.602 \times 10^{-19}\, C)^2}} = 1.695 \times 10^{-22}\, s$$

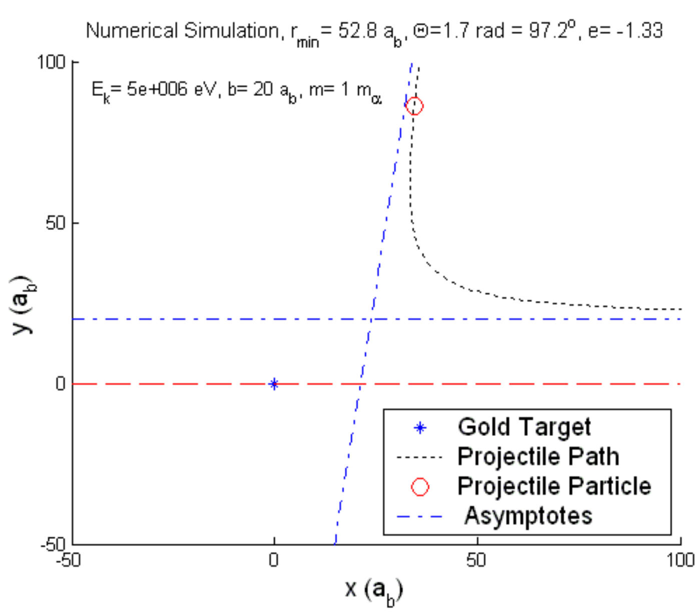
Speed unit: $v_b = a_b / \tau = 1 \times 10^{-15}\, m / 1.695 \times 10^{-22}\, s = 5.898 \times 10^6\, m/s = 0.01965c$

Solve these numerically ➡

$$\frac{d\bar{x}}{d\bar{t}} = \bar{v}_x, \quad \frac{d\bar{v}_x}{d\bar{t}} = \frac{\bar{K}\bar{x}}{\bar{m}\left(\bar{x}^2 + \bar{y}^2\right)^{3/2}}; \quad \frac{d\bar{y}}{d\bar{t}} = \bar{v}_y, \quad \frac{d\bar{v}_y}{d\bar{t}} = \frac{\bar{K}\bar{y}}{\bar{m}\left(\bar{x}^2 + \bar{y}^2\right)^{3/2}}$$

$$\bar{r}_{\min} = \min\left(\sqrt{\bar{x}(\bar{t})^2 + \bar{y}(\bar{t})^2}\right)$$

Numerical simulation of a projectile alpha particle onto a gold target



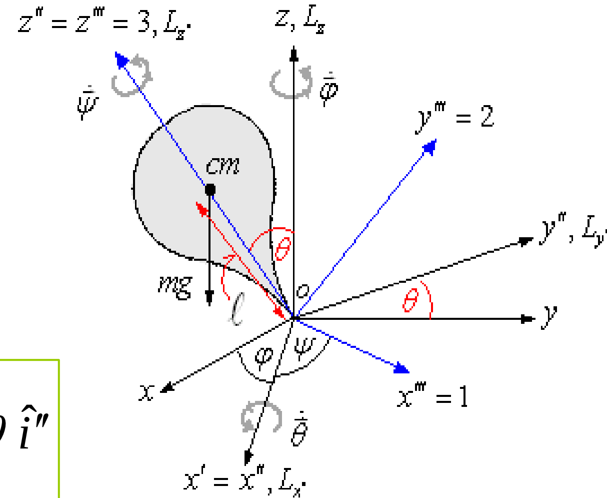
See ruther.html and ruther.avi

Motion of Rigid Bodies – Symmetric Top (simulation)

Using Eulerian angles

$$\phi,\theta,\psi$$

Spinning symmetric top with its symmetry axis (), which is its spin axis as well as its principal axis of symmetry, at angle from the fixed axis



$$\vec{\tau} = \left(\frac{d\vec{L}}{dt}\right)_{fixed} = \left(\frac{d\vec{L}}{dt}\right)_{rot} + \vec{\omega}'' \times \vec{L} = mg\ell\sin\theta\,\hat{i}''$$

$$\vec{L} = I\dot{\theta}\hat{i}'' + I\dot{\varphi}\sin\theta\,\hat{j}'' + I_s\omega_s\hat{k}'', \quad \vec{\omega}'' = \dot{\theta}\,\hat{i}'' + \dot{\varphi}\sin\theta\,\hat{j}'' + \dot{\varphi}\cos\theta\,\hat{k}''$$

$$I_3\left(\dot{\varphi}\cos\theta + \dot{\psi}\right) \equiv I_s\omega_s$$

or

$$mg\ell\sin\theta = I\ddot{\theta} + I_s\omega_s\dot{\varphi}\sin\theta - I\dot{\varphi}^2\cos\theta\sin\theta$$

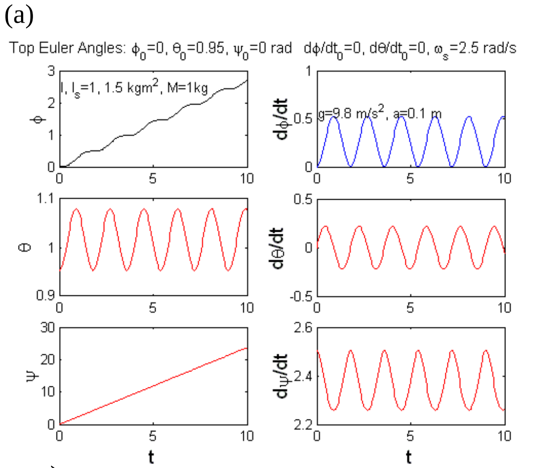
$$0 = I\frac{d}{dt}\left(\dot{\varphi}\sin\theta\right) - I_s\omega_s\dot{\theta} + I\dot{\theta}\dot{\varphi}\cos\theta$$

$$0 = I_s\dot{\omega}_s$$

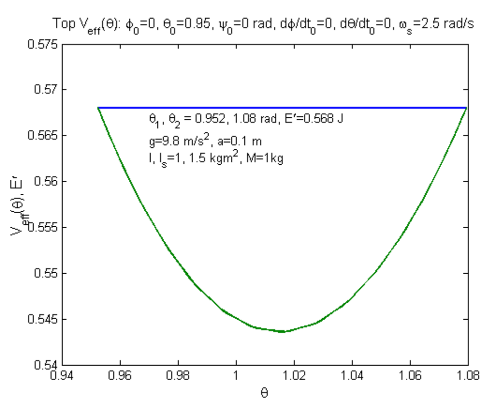
→ solve numerically for

$$\phi(t),\theta(t),\psi(t)$$

(a)

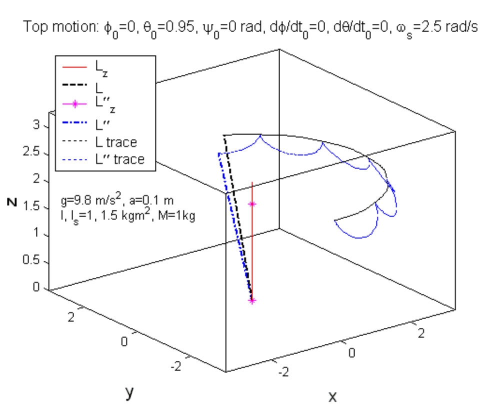


(b)



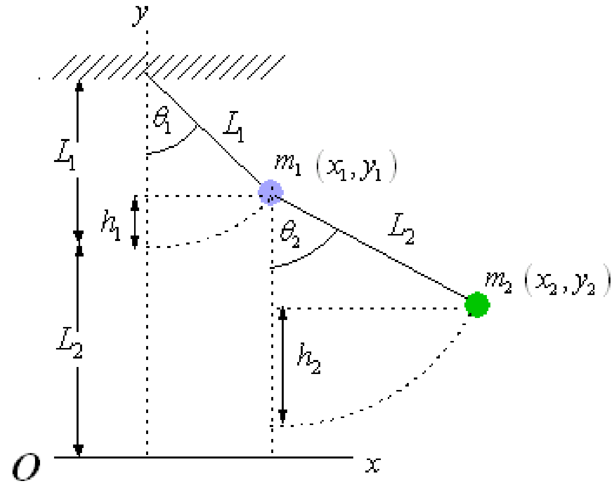
See [top.html]() and [top.avi]()

(c)



spinning fixed point symmetric top a) Numerical solution, b) Plot of the energy and the effective potential, and c) a snapshot of the simulated motion of the top's total angular momentum as well as the body angular momentum vs time

LAGRANGIAN DYNAMICS – Double Pendulum (simulation)



Double Pendulum

Coordinates

$$x_1 = L_1 \sin\theta_1 \qquad x_2 = x_1 + L_2 \sin\theta_2$$
$$y_1 = L_1 \cos\theta_1 \qquad y_2 = y_1 + L_2 \cos\theta_2$$

kinetic, potential energies

$$T = \frac{1}{2} m_1 \left(\dot{x}_1^2 + \dot{y}_1^2\right) + \frac{1}{2} m_2 \left(\dot{x}_2^2 + \dot{y}_2^2\right), \quad V = m_1 g L_2 + m_1 g h_1 + m_2 g \left(h_1 + h_2\right)$$

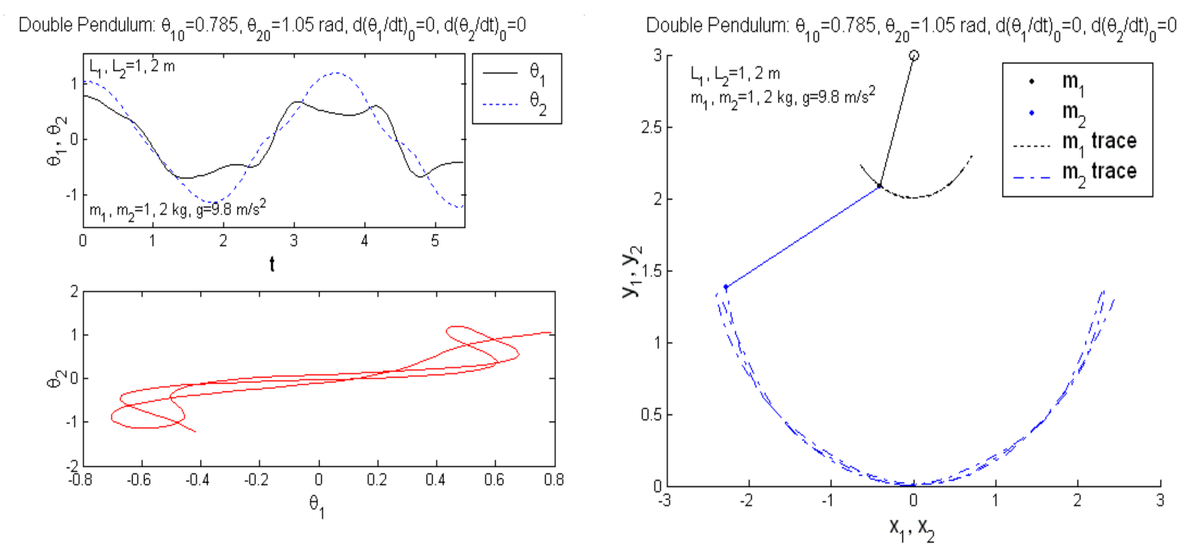
Lagrangian →

$$L = T - V = \frac{1}{2} m_1 L_1^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 \left[L_1^2 \dot{\theta}_1^2 + L_2^2 \dot{\theta}_2^2 + 2 L_1 L_2 \dot{\theta}_1 \dot{\theta}_2 \cos\left(\theta_1 - \theta_2\right)\right] - m_1 g L_2 - \left(m_1 + m_2\right) g L_1 \left(1 - \cos\theta_1\right) - m_2 g L_2 \left(1 - \cos\theta_2\right).$$

Lagrange's equations →

- solve numerically

$$\left(m_1 + m_2\right) L_1 \ddot{\theta}_1 + m_2 L_2 \ddot{\theta}_2 \cos\left(\theta_1 - \theta_2\right) = -m_2 L_2 \dot{\theta}_2^2 \sin\left(\theta_1 - \theta_2\right) - \left(m_1 + m_2\right) g \sin\theta_1$$
$$m_2 L_2 \ddot{\theta}_2 + m_2 L_1 \ddot{\theta}_1 \cos\left(\theta_1 - \theta_2\right) = m_2 L_1 \dot{\theta}_1^2 \sin\left(\theta_1 - \theta_2\right) - m_2 g \sin\theta_2$$



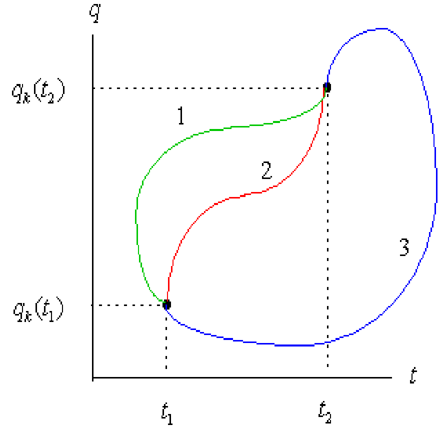
The double pendulum a) Eulerian angles plotted versus time (upper figure) and versus each other (lower figure) b) simulation of the pendulum for the initial conditions shown.

See doublep.html and doublep.avi

LAGRANGIAN DYNAMICS – Principle of Least Action (simulation)

Three possible paths in the evolution process of the action integral

$$S \equiv \int_{t_1}^{t_2} L dt$$



Hamilton's principle

$$I = \delta \int_{t_1}^{t_2} L dt = 0$$

Hamilton's principle: the motion followed by a mechanical system as it moves from a starting point to a final point within a given time will be the motion that provides an extremum for the time integral of the Lagrangian.

Example – case of a particle in free fall, we have the Lagrangian and the action:

$$L = T - V = \frac{1}{2} m v_y^2 - mgy \qquad S = \int_{t_0}^{t_f} L dt = \int_{t_0}^{t_f} \left(\frac{1}{2} m v^2 - mgy \right) dt$$

Numerically, make the approximation:

$$S = \int_{t_0}^{t_f} L dt \approx \Delta t \sum_{k=1}^{N-1} L_k$$

with $L_k \equiv L(t_k) \approx \frac{1}{2} m \left(\frac{y_{k+1} - y_k}{\Delta t} \right)^2 - mgy_k$

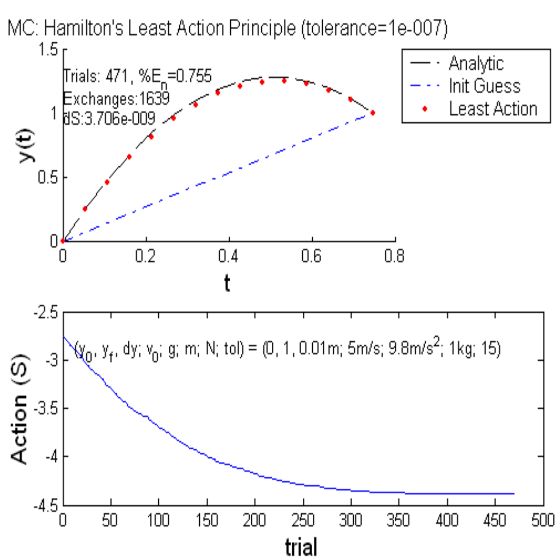
and the initial guess $\longrightarrow$ $y_k = y_0 + \frac{(y_f - y_0)}{(t_f - t_0)} (t_k - t_0)$

Modify the guess randomly, accept steps that lead to s decrease in $dS = S_{n,N-1} - S_{n-1,N-1}$

until dS is small. Compare numerical results against the exact solution

$$y = y_0 + v_0 t - \frac{1}{2} g t^2$$

Simulation of Hamilton's least action principle for the case of the motion of a single particle free falling near Earth's surface, in one dimension



See least_action.html and least_action.avi

Other Highlights

•Harmonic oscillator (undamped, damped, and forced)

•Projectile Motion (analytic and numerical)

•The pendulum (small, and large angles)

•Central Forces
-Planetary Motion (analytic, numerical, and simulations) and comparison with data

•Eulerian Angle Frame Rotation (visualization)

•More on Rutherford Scattering
--Comparison with the 1913 Geiger Marsden Data for Silver and Gold

Conclusion

•A junior level mechanics textbook has been developed that incorporated computational physics: “Intermediate Classical Mechanics with MATLAB applications.”

•The text makes use of the valuable traditional analytic approach in pedagogy. It further incorporates computational techniques to help students visualize, explore, and gain insight to problems beyond idealized situations.

•Some programming background is expected and most physics/engineering majors have had programming experience by their junior year.

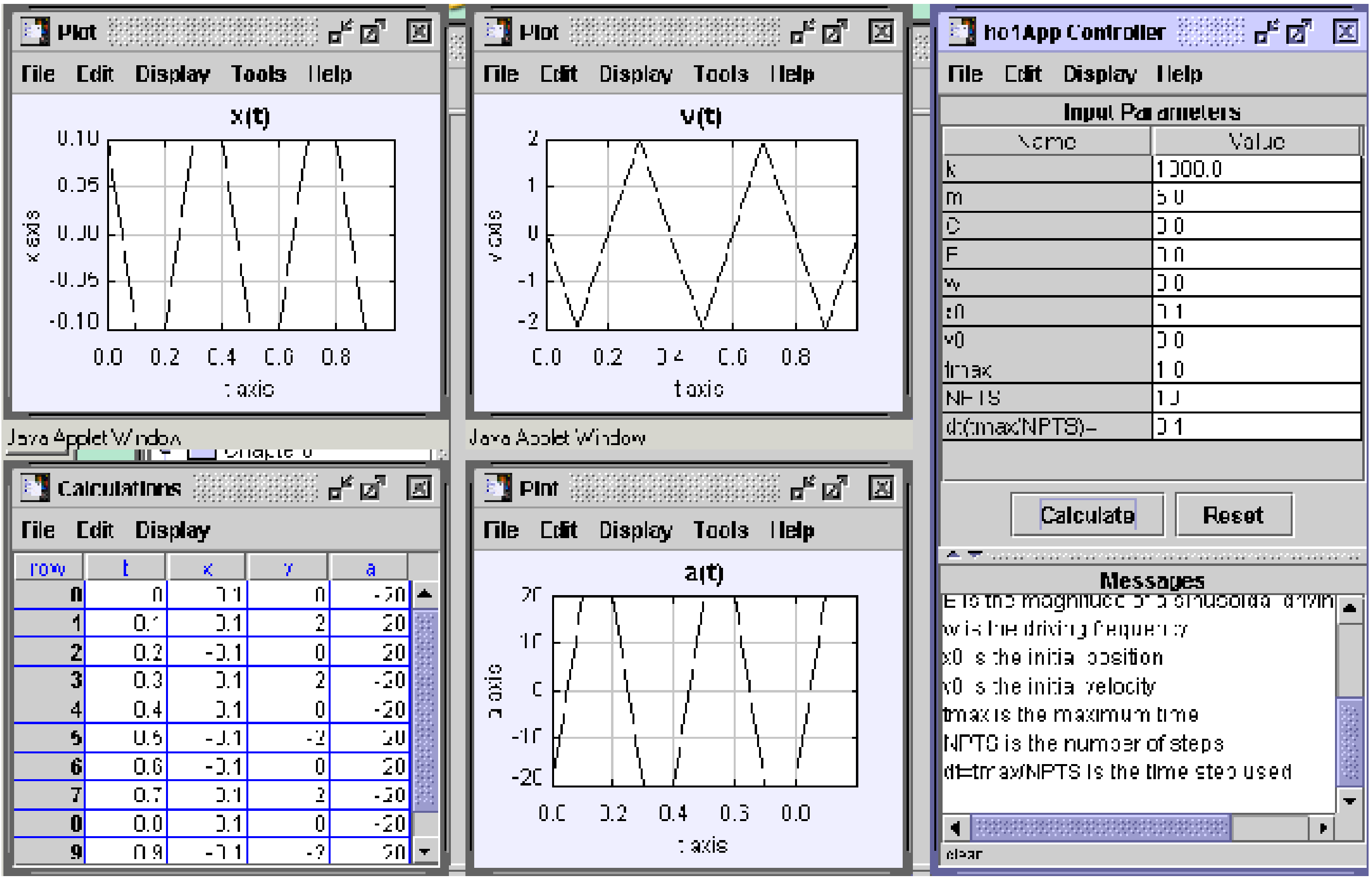
•The emphasis is placed on understanding. The analytic approach is supported and complemented by the computational approach.

•Java applications analogue to the MATLAB scripts are available (under development) see below. They use the Open Source Physics (OSP) library of W. Christian and co-workers.
http://www.westga.edu/~jhasbun/osp/osp.htm
http://www.opensourcephysics.org/

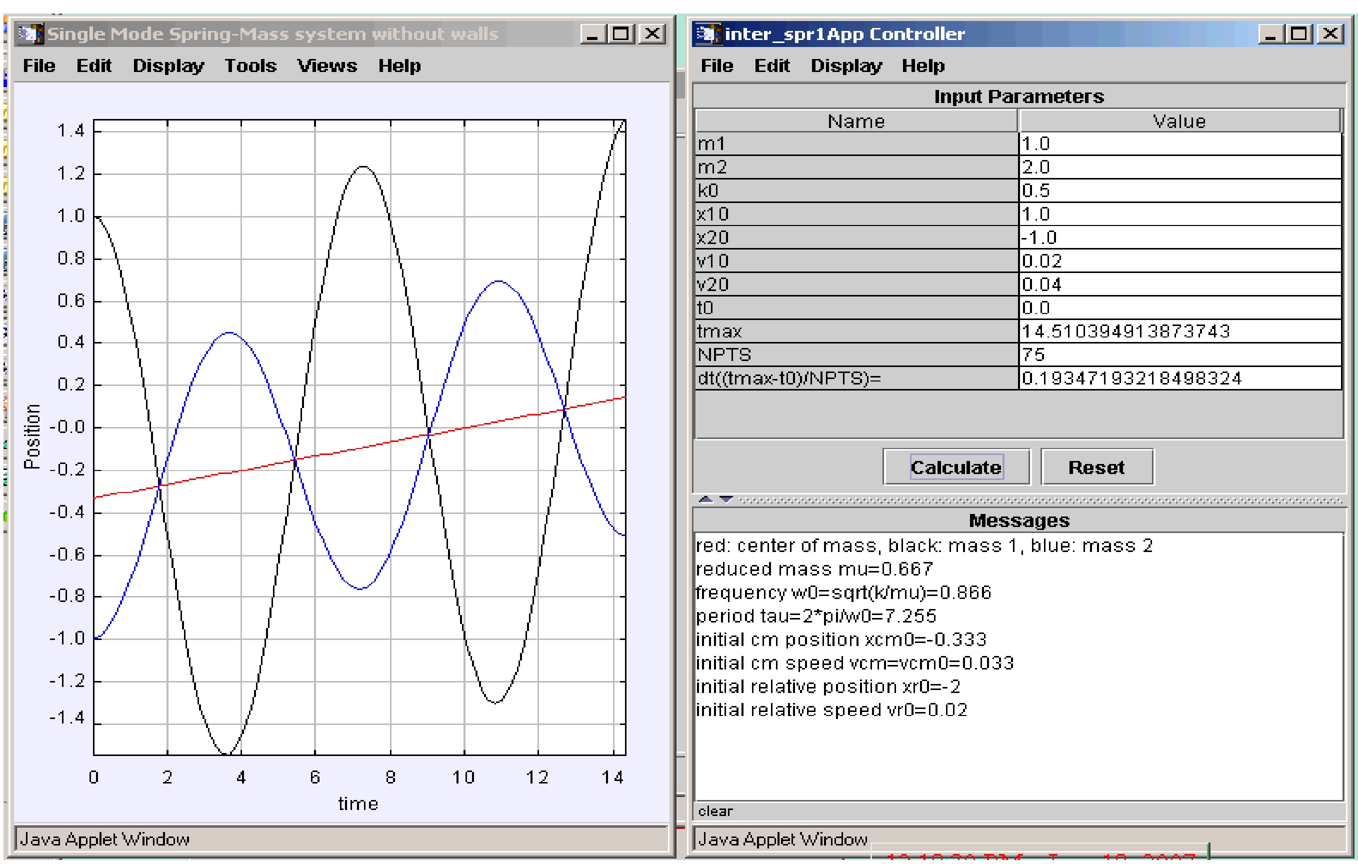
•Comments are welcome. Please contact J. E. Hasbun
jhasbun@westga.edu

OPEN SOURCE PHYSICS (OSP) JAVA APPLICATIONS

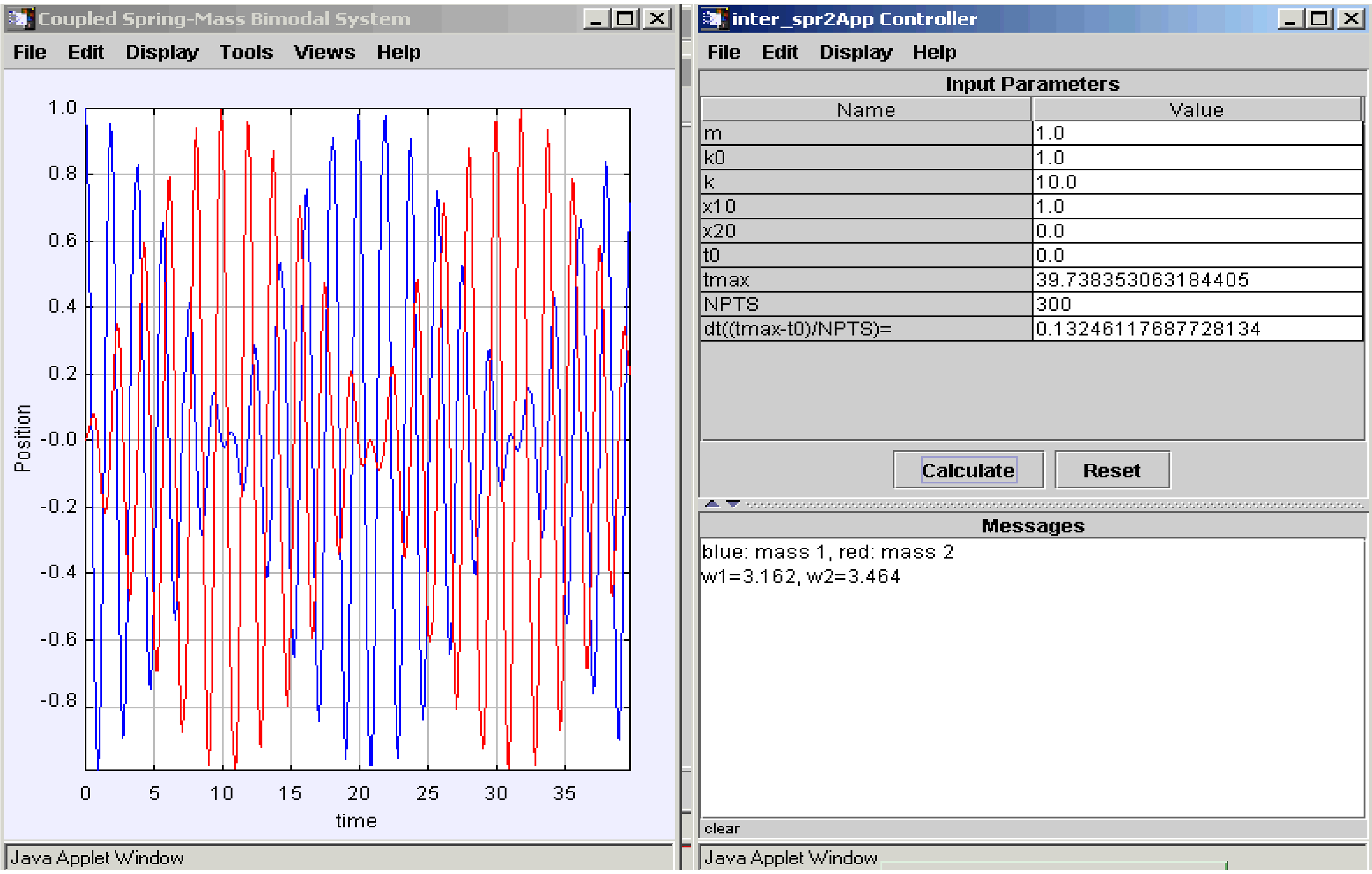
ho1app



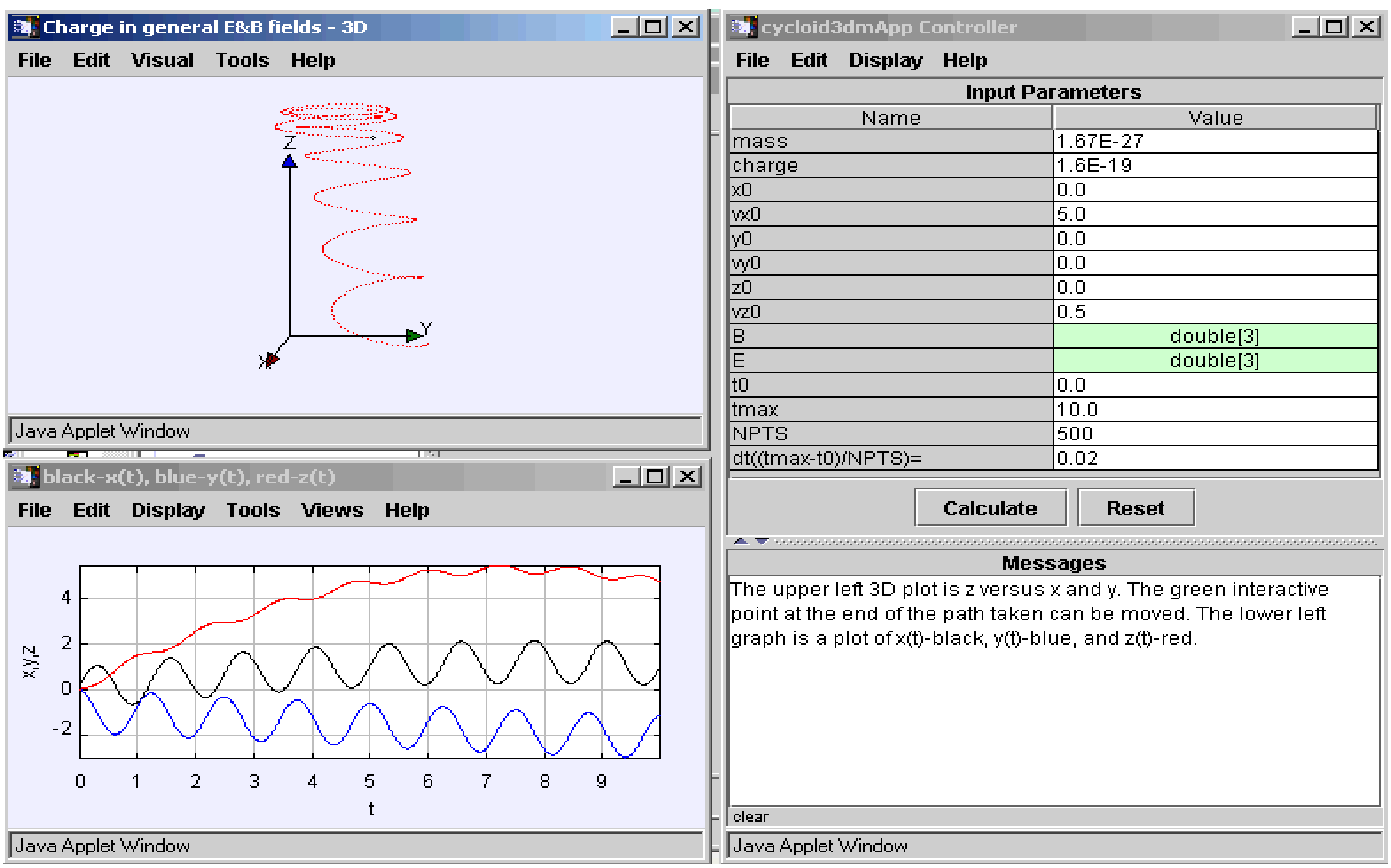
inter_spr1App



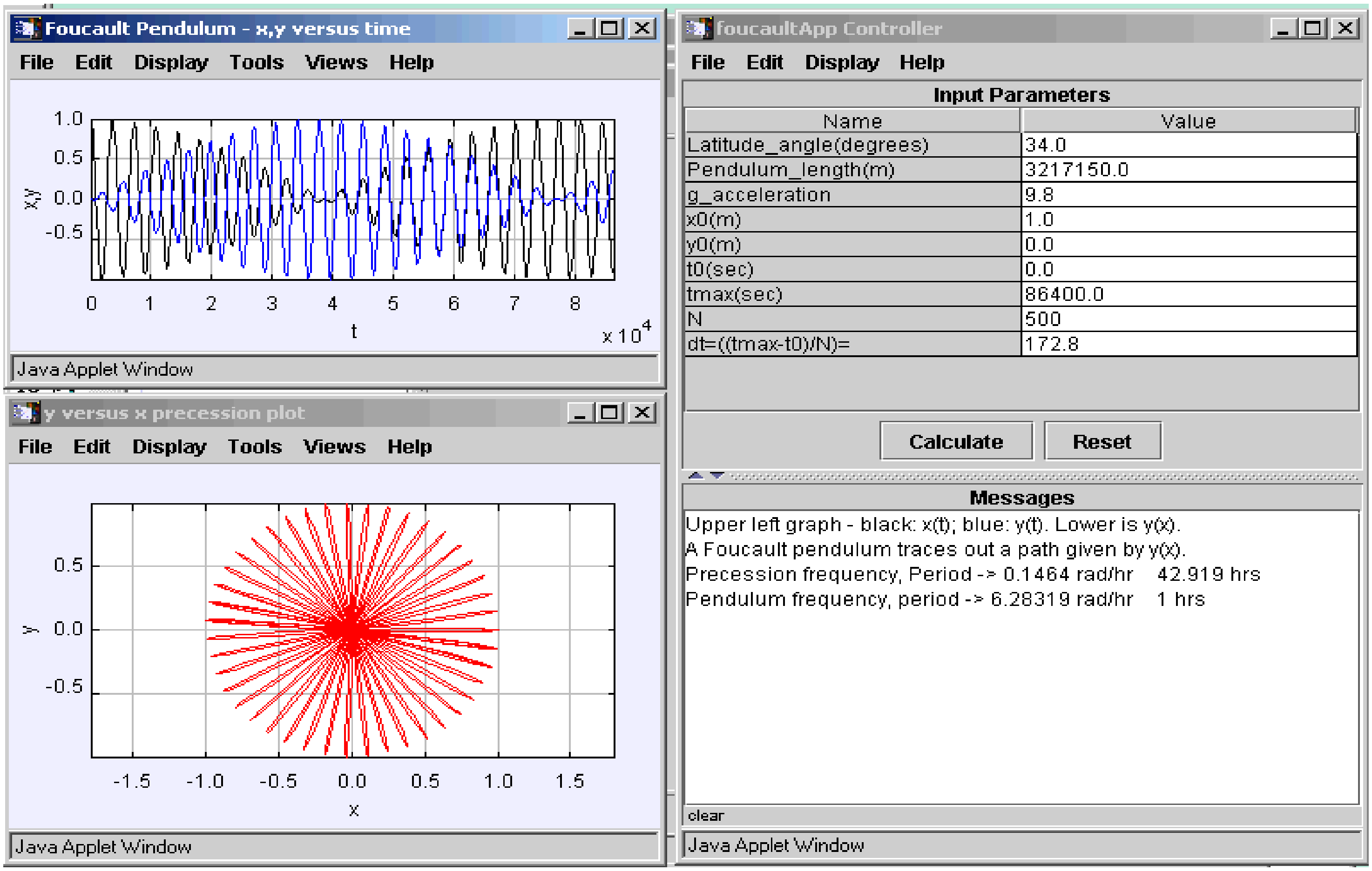
inter_spr2App



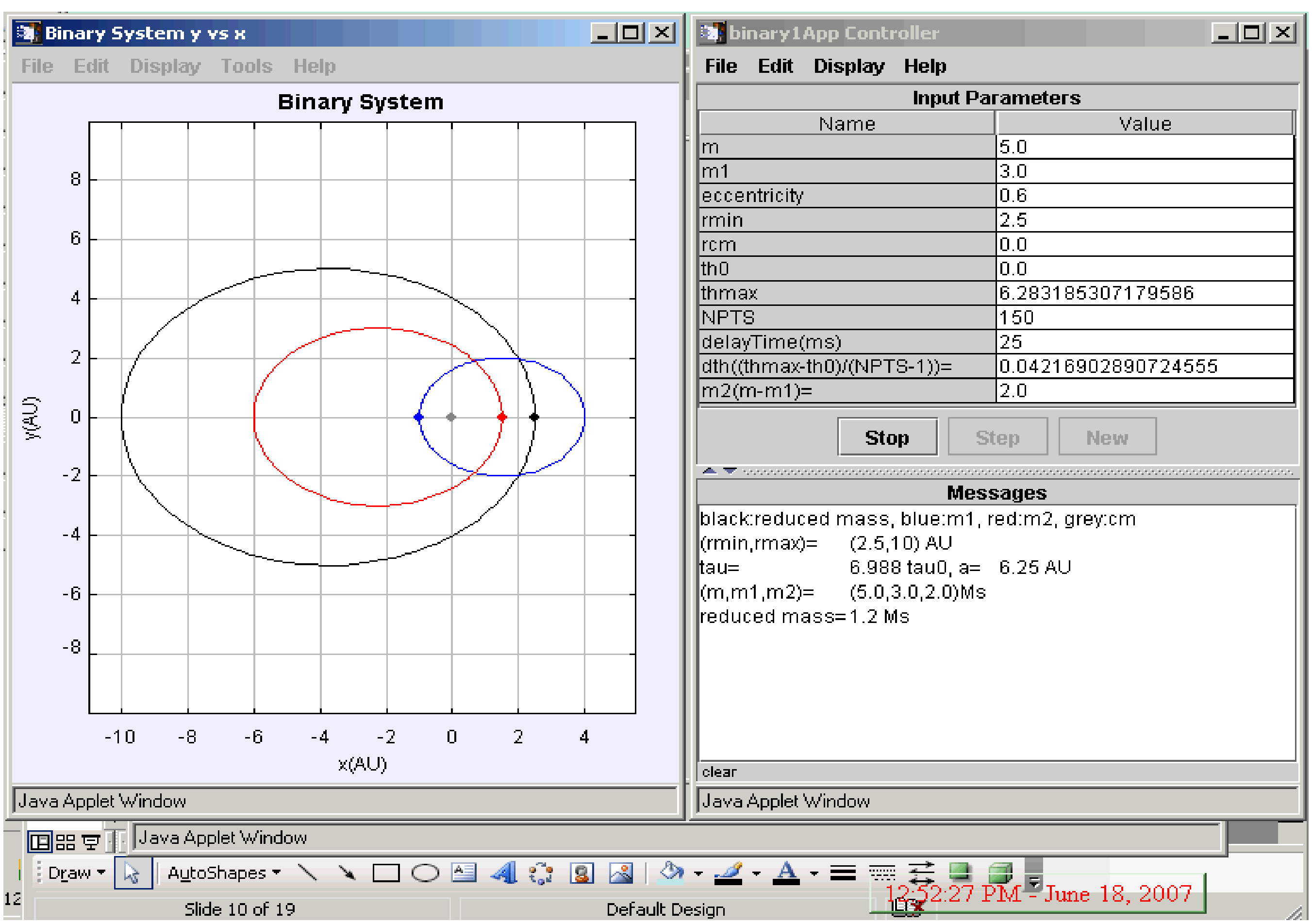
cycloid3dApp



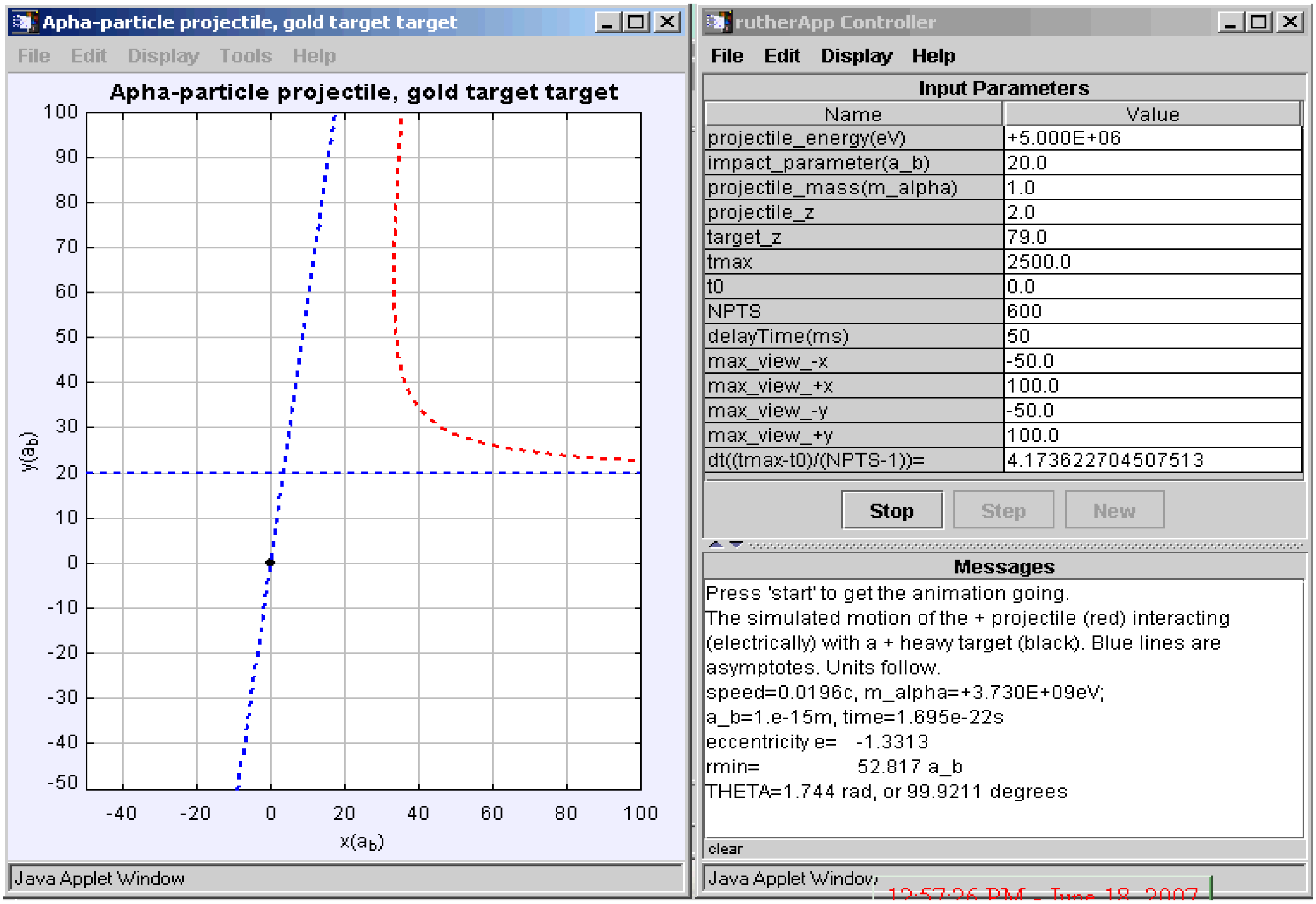
foucaultApp



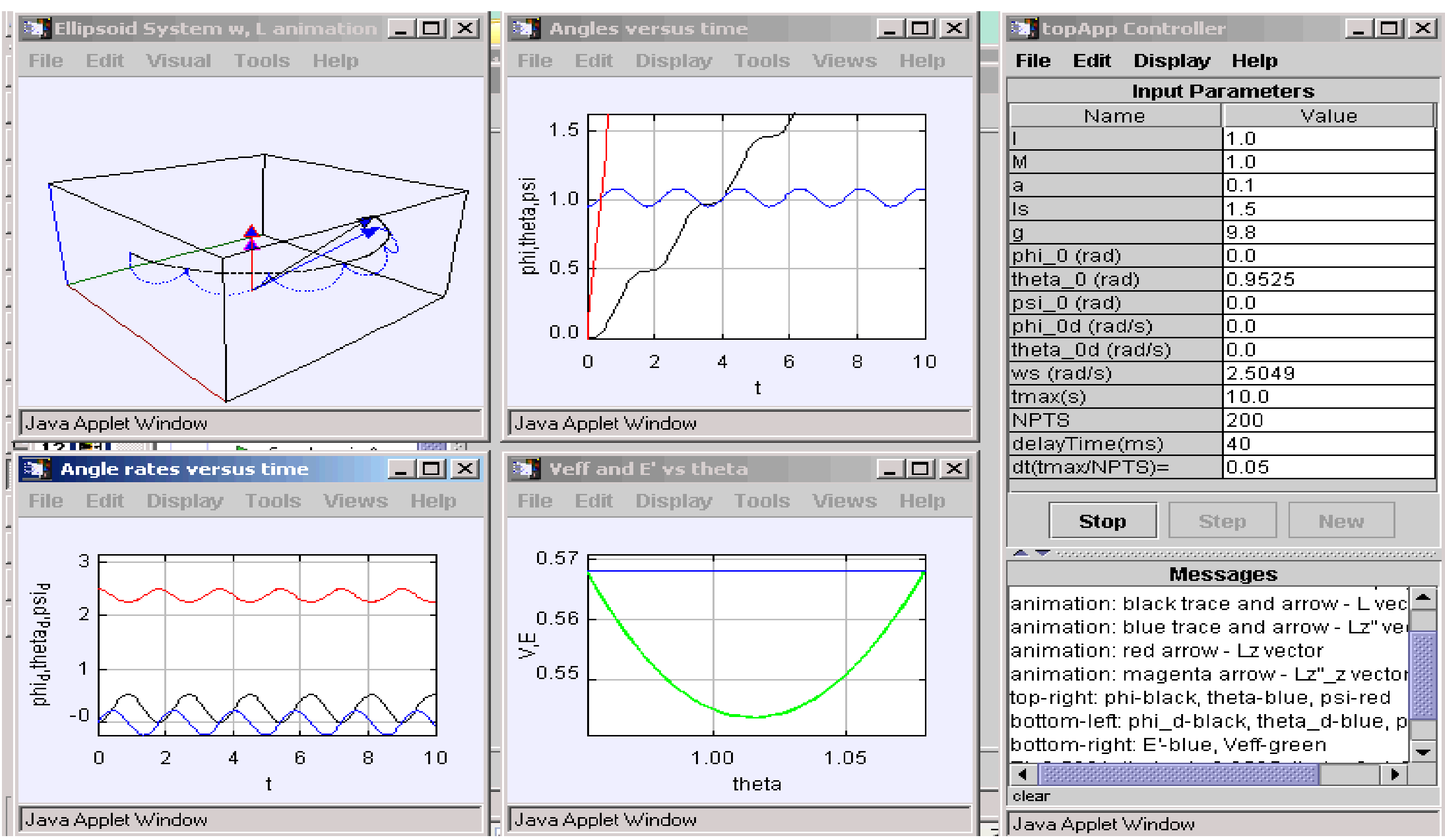
binary1App



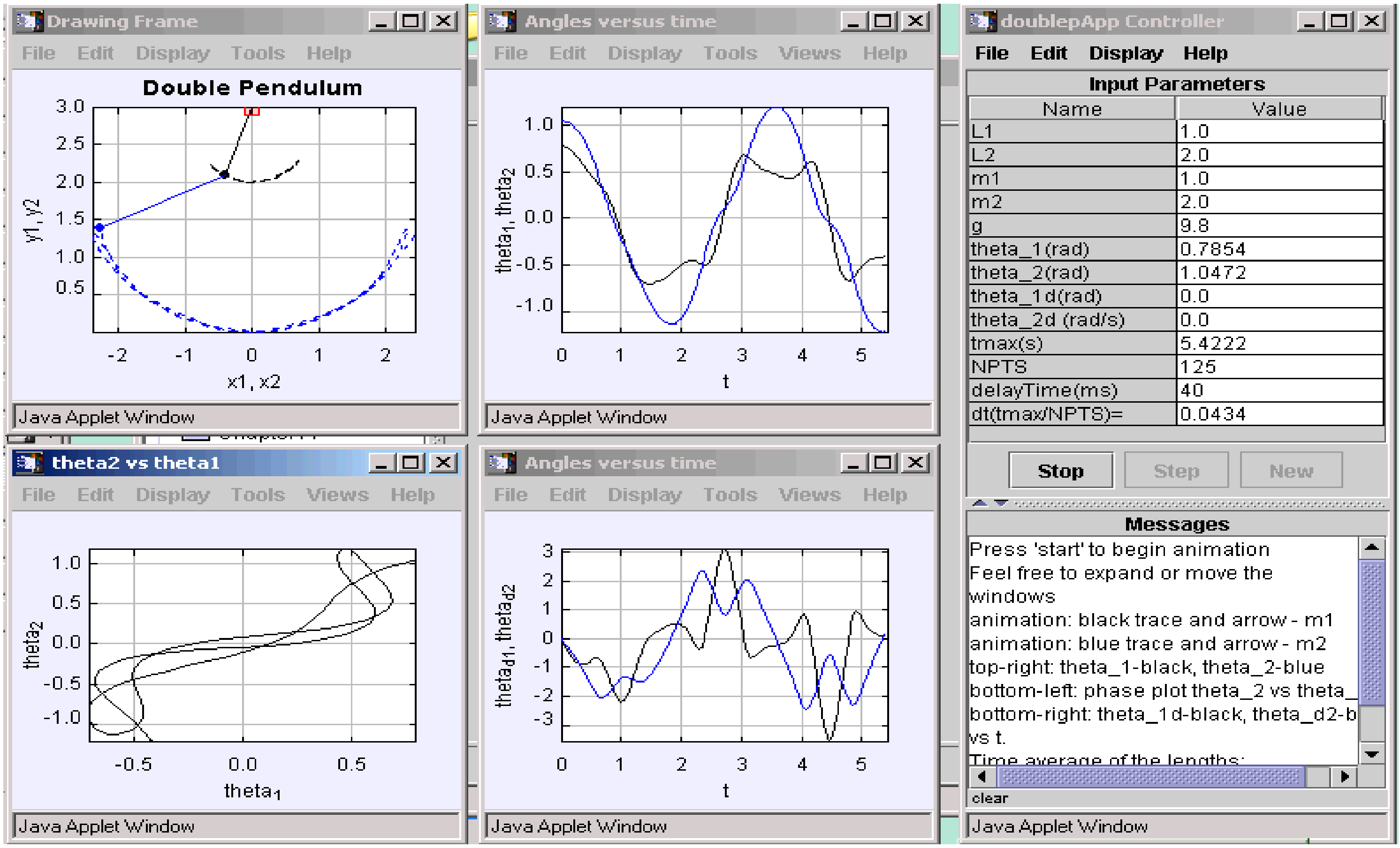
rutherApp



topApp



doublepApp



least_actionApp