Johnny VanLandingham

Problem Set #5

1. For the following differential equation:

$$\frac{d^2x}{dt^2} + x = 0$$

The coupled first order equations are:

$$\frac{dx}{dt} = v, \frac{dv}{dt} = -x$$

(a) The program `integrator1.c` integrates these equations with initial conditions $x = 0, v = 1$ for any integration time and step size, using either the Euler, Leapfrog, or Runge-Kutta 4 method. For example, using the Leapfrog method to integrate from 0 to 15 with a step size of 1 would produce this:

```
Reading input...
Creating file Sinusoid15_15_3.dat...
Writing initial values to file...
Integrating using Leapfrog method...
Integration complete, values written to file.
```

The first few lines of this file look like this:

```
x1 x2 t
0.000000 1.000000 0.000000
1.0000000000 0.0000000000 1.000000
1.0000000000 -1.0000000000 2.000000
```

(b) See figures 1, 2 and 3. The Euler method requires a very small time step to get accurate results. It is only close to the analytic result for a time step of 0.01. Leapfrog and Runge-Kutta 4 are much more accurate, and Runge-Kutta 4 is even quite accurate for a time step of 1.

(c) See figure 4. Euler is first order so the error should be on the order of h^2. Leapfrog is 2nd order so the error should be on the order of h^3. RK4 is 4th order, so the error should be on the order of h^5. On a log-log plot this means that for Euler, slope ≈ 2. For Leapfrog, slope ≈ 3, and for RK4 slope ≈ 5. This is about right for my graph. The only difficulty is with RK4. The error is smaller than it should be for a step size of 1, and the error was to small to observe for a step size of 0.01. With that small of a step size, the integration result was indistinguishable from the analytical result.

2. Find accelerations from the potential:

$$\phi = -\frac{1}{\sqrt{1+2x^2+2y^2}}, \vec{a} = -\nabla\phi$$

$$\vec{a_x} = \frac{d}{dx}[-\frac{1}{\sqrt{1+2x^2+2y^2}}] = -\frac{1}{2}\frac{1}{(1+2x^2+2y^2)^{3/2}}(4x) = -\frac{2x}{(1+2x^2+2y^2)^{3/2}}$$

$$\vec{a_y} = \frac{d}{dy}[-\frac{1}{\sqrt{1+2x^2+2y^2}}] = -\frac{1}{2}\frac{1}{(1+2x^2+2y^2)^{3/2}}(4y) = -\frac{2y}{(1+2x^2+2y^2)^{3/2}}$$

Reduce these to first order:

$$\frac{dx}{dt} = v_x, \frac{dy}{dt} = v_y$$

$$\frac{dv_x}{dt} = -\frac{2x}{(1+2x^2+2y^2)^{3/2}}, \frac{dv_y}{dt} = -\frac{2y}{(1+2x^2+2y^2)^{3/2}}$$

(a) The program `integrator2.c` integrates this system with initial conditions $x = 1, y = 0, v_x = 0, v_y = 0.1$. The output for RK4 integrating from 0 to 100 with a step size of 0.25 would be `Orbit100_400_2.dat`. The first few lines look like this:

```
x1 x2 x3 x4 t
1.000000 0.000000 0.000000 0.100000 0.000000
0.987949 0.024899 -0.096592 0.098785 0.250000
0.951519 0.049179 -0.195410 0.094995 0.500000
```

See figure 5 for Runge-Kutta and figure 6 for Leapfrog.

(b) See figures 7 and 8 for RK4 and figures 9 and 10 for Leapfrog. Over time, energy is lost when you use RK4, but for Leapfrog, the energy is conserved over the long term.

3. The program `integrator3.c` integrates the Lotka-Volterra equations. See figure 11 for the phase diagrams where $d = e = 0$. If I set $d = e = 1.25$ I get the file `Preyhunt100_1000_2.dat`. Since the foxes die out much more quickly than the rabbits, I graphed the rabbit population from this data in figure 12. By $t = 100$ the rabbit population is just under 10^{-9}.

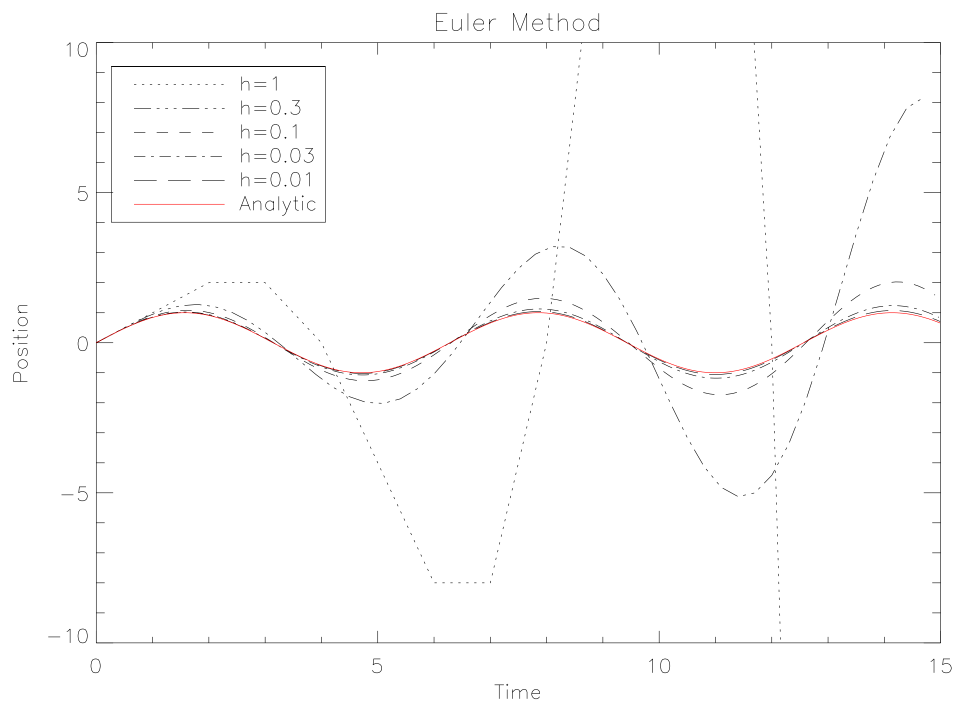


Figure 1:

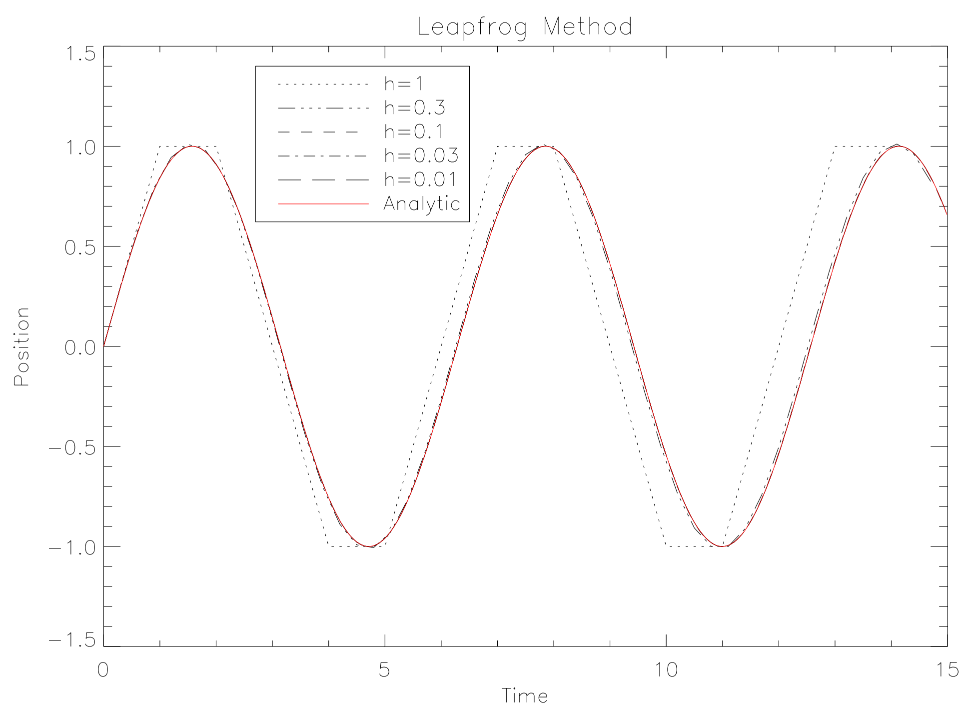


Figure 2:

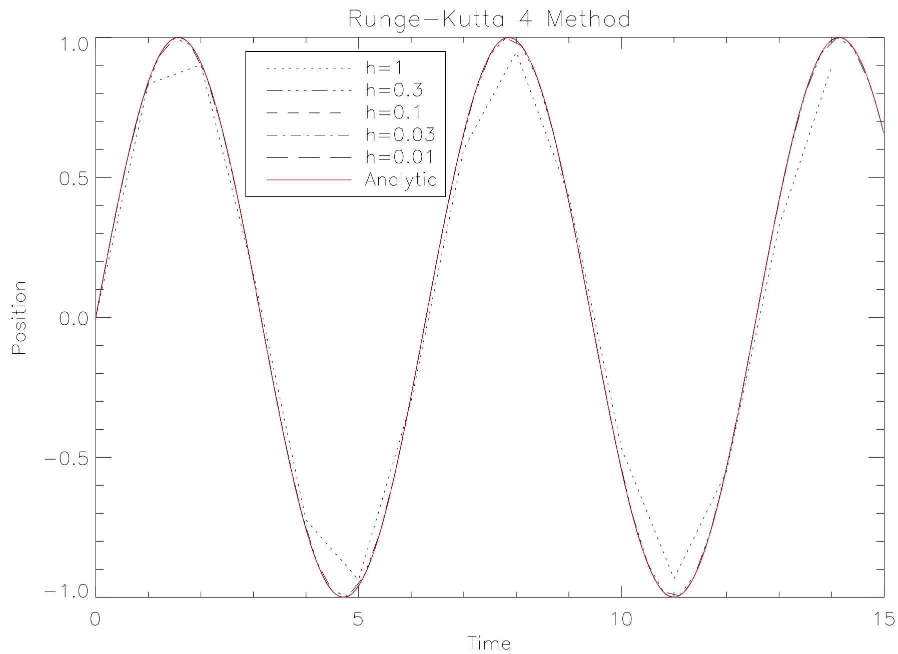


Figure 3:

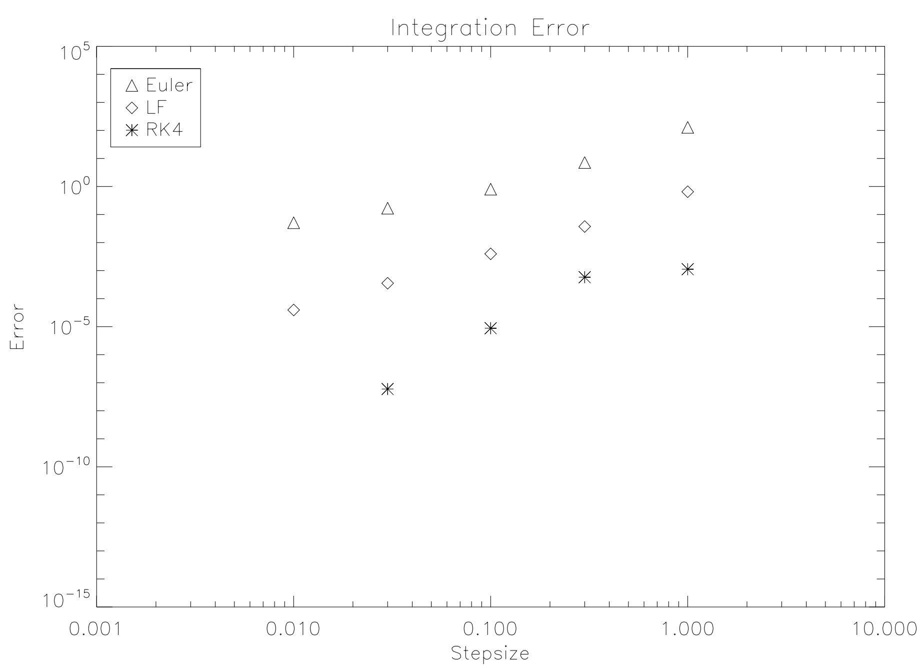


Figure 4:

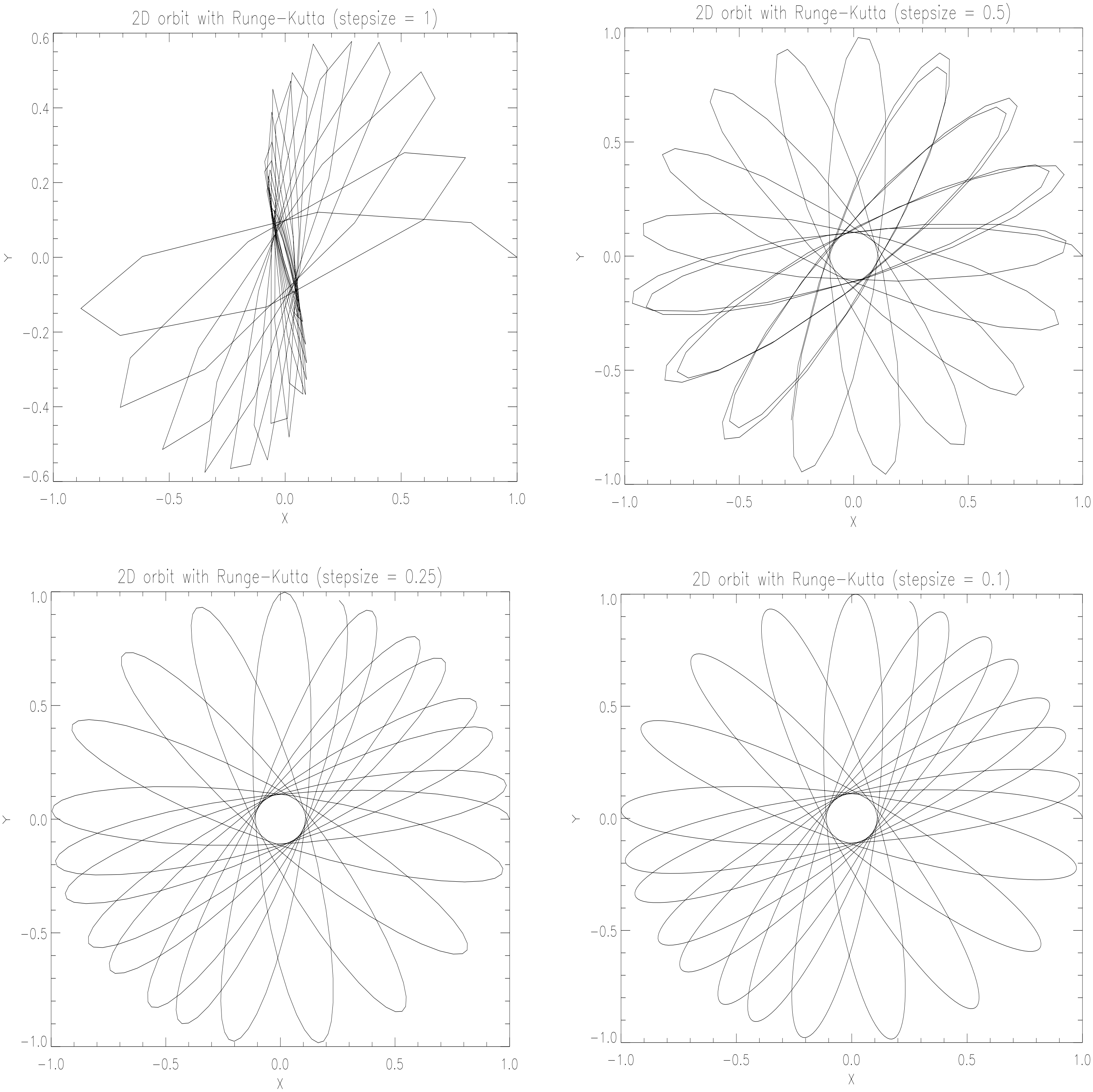

2D orbit with Runge–Kutta (stepsize = 1)
2D orbit with Runge–Kutta (stepsize = 0.5)
2D orbit with Runge–Kutta (stepsize = 0.25)
2D orbit with Runge–Kutta (stepsize = 0.1)
X
Y

Figure 5:

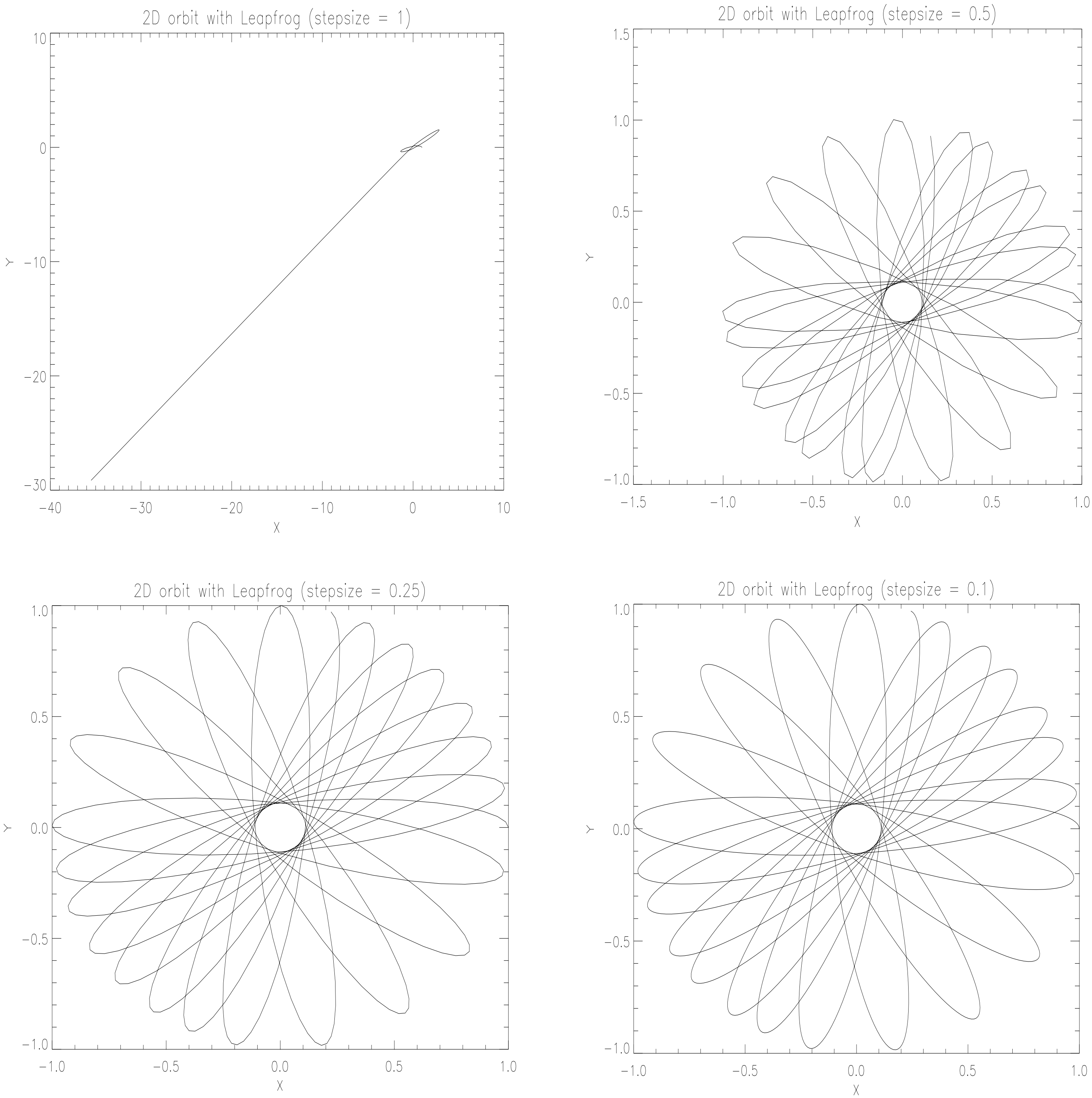

2D orbit with Leapfrog (stepsize = 1)
X
Y
2D orbit with Leapfrog (stepsize = 0.5)
X
Y
2D orbit with Leapfrog (stepsize = 0.25)
X
Y
2D orbit with Leapfrog (stepsize = 0.1)
X
Y

Figure 6:

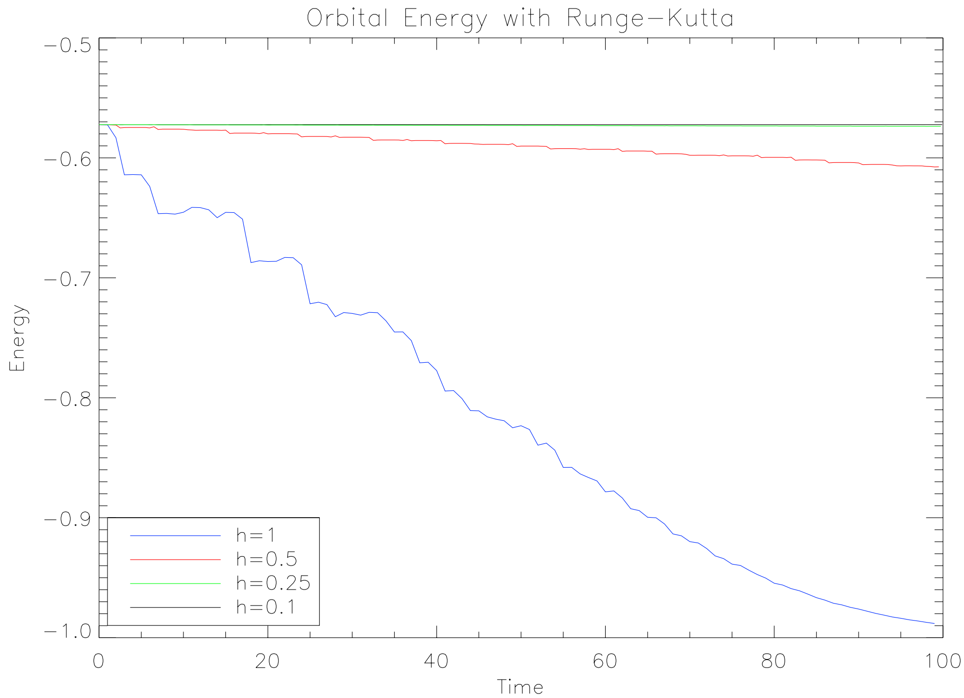


Figure 7:

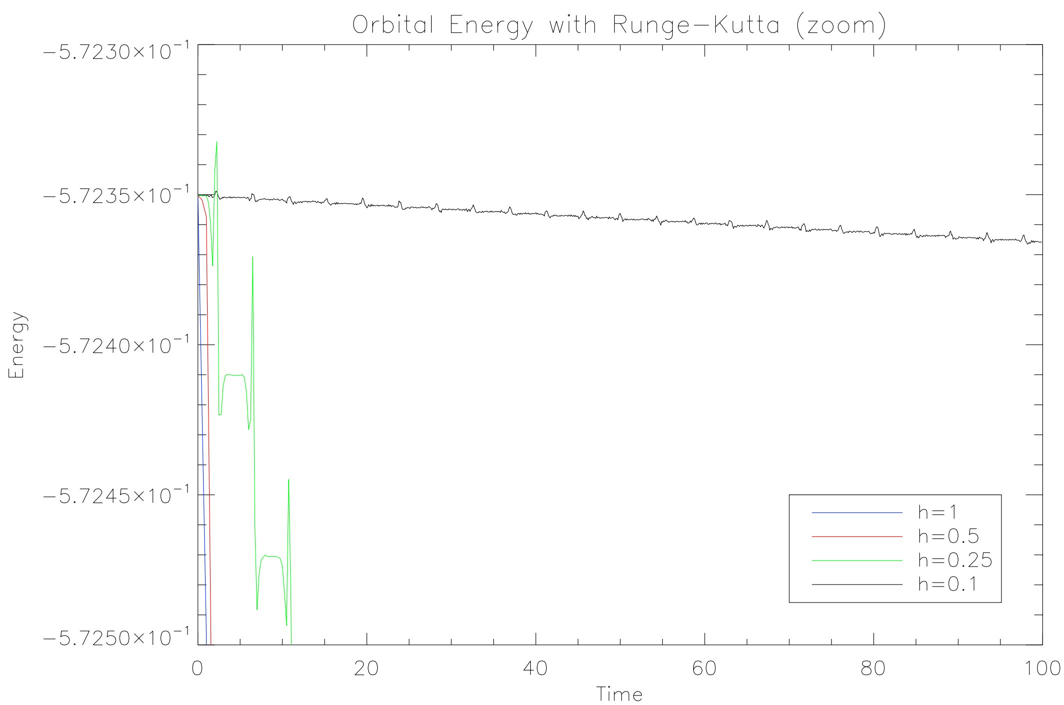


Figure 8:

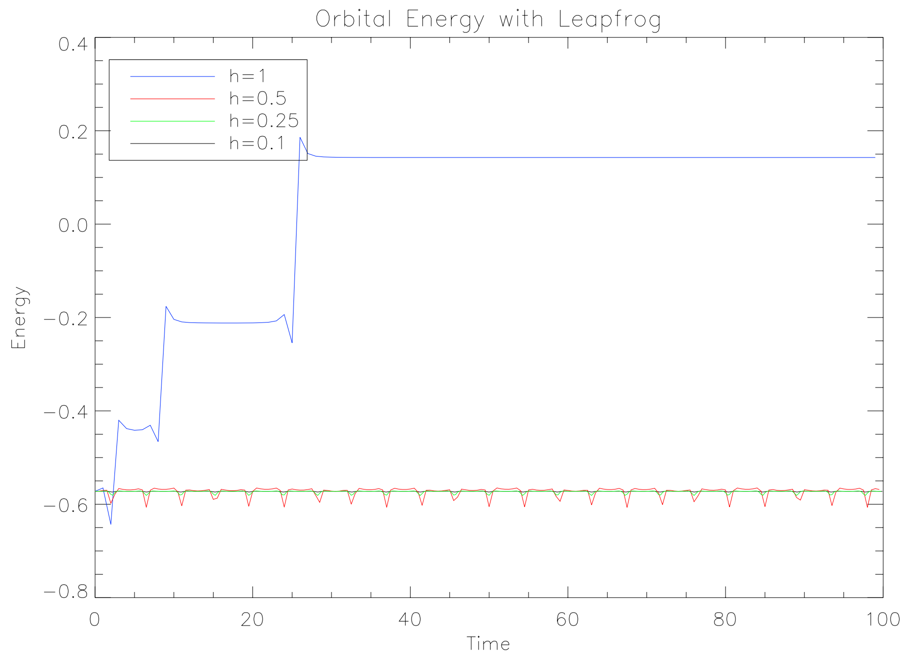


Figure 9:

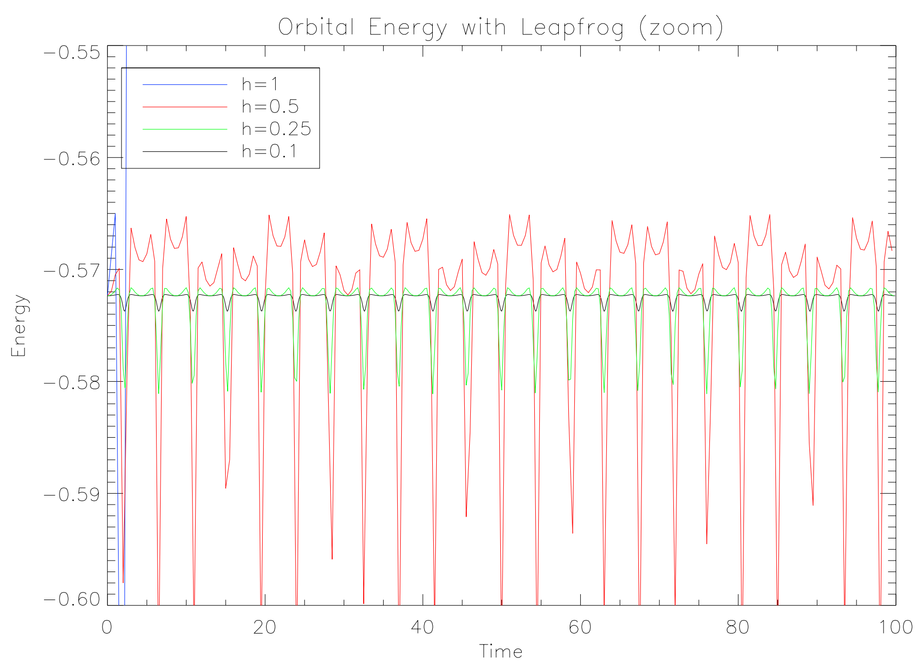


Figure 10:

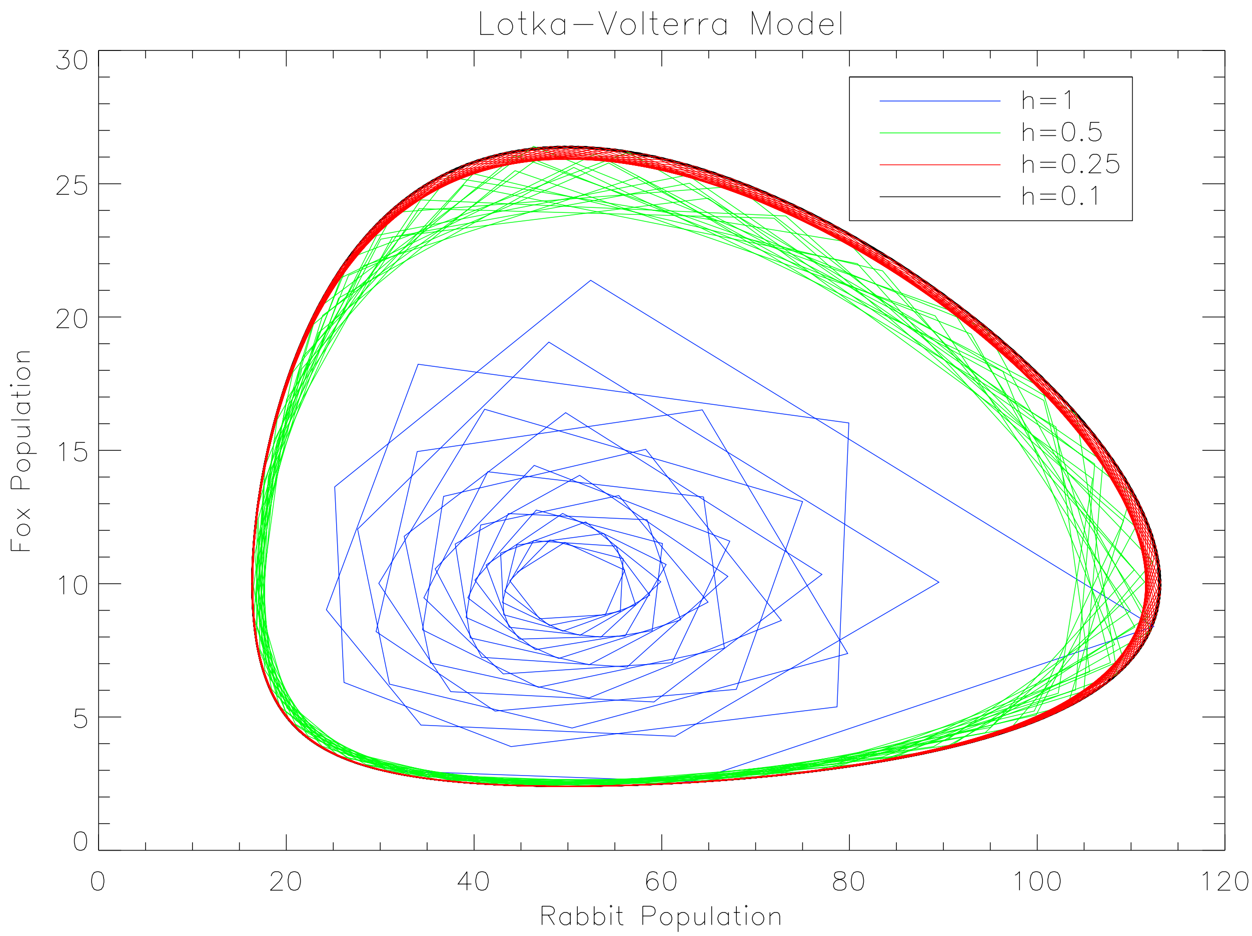
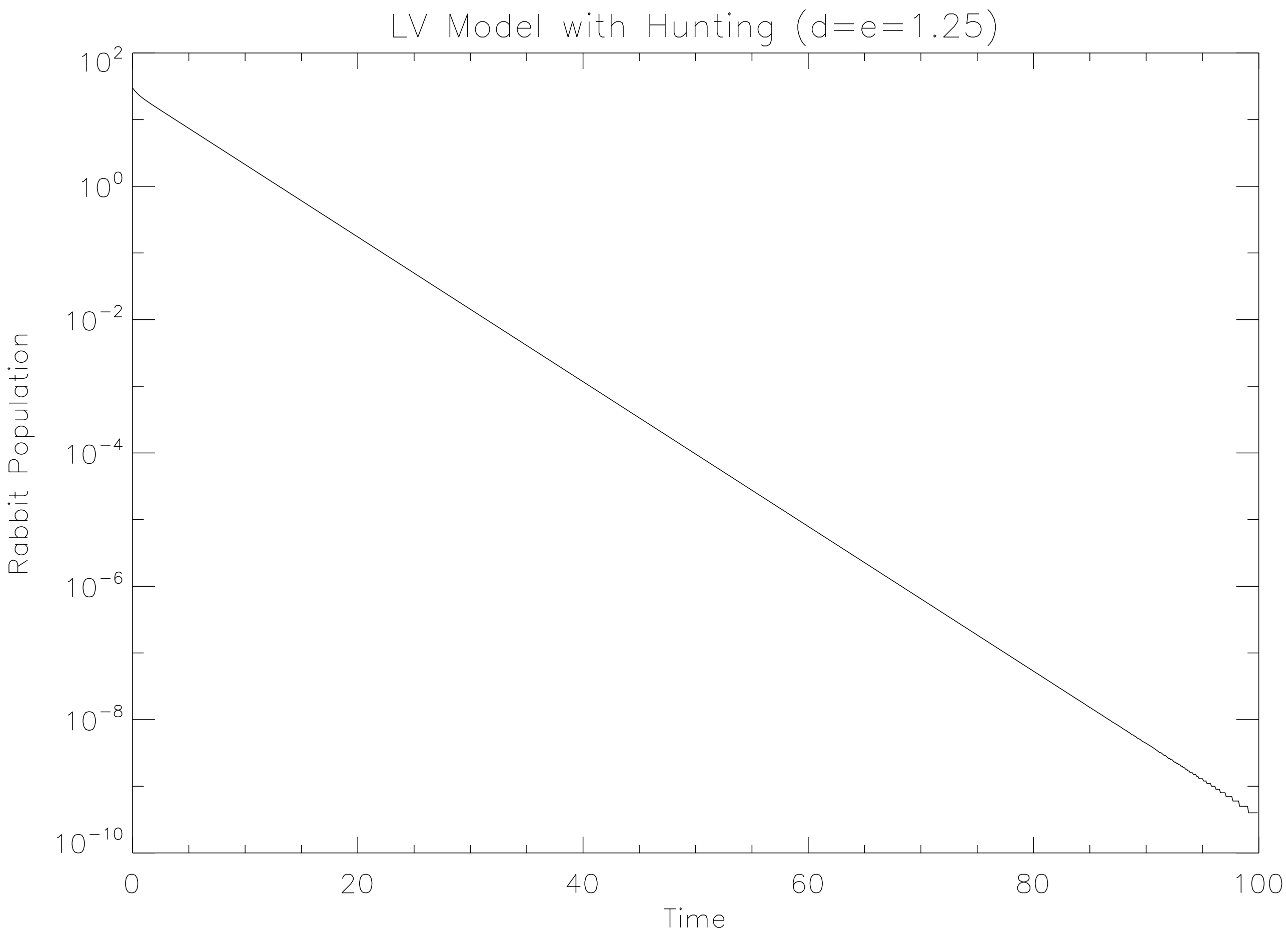

Lotka−Volterra Model
h=1
h=0.5
h=0.25
h=0.1
Fox Population
Rabbit Population
0
5
10
15
20
25
30
0
20
40
60
80
100
120

Figure 11:



LV Model with Hunting (d=e=1.25)
Rabbit Population
Time
10^{2}
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
0
20
40
60
80
100

Figure 12: