1 hour

7. Montgomery's eight

Three equal masses $(m=1)$ are attracted by an inverse-square gravity law with $G=1$. That is, each mass is attracted to the other by $F = Gm_1m_2/r^2$ where r is the distance between them. Use these unusual and special initial positions.

$$(x_1, y_1) = (-.97\ldots)$$

$$(x2, y2) = (-x_1, -y1)$$

$$(x_3, y3) = (0, 0)$$

and initial velocities:

$$(vx3, v_y3) = (0.93\ldots)$$

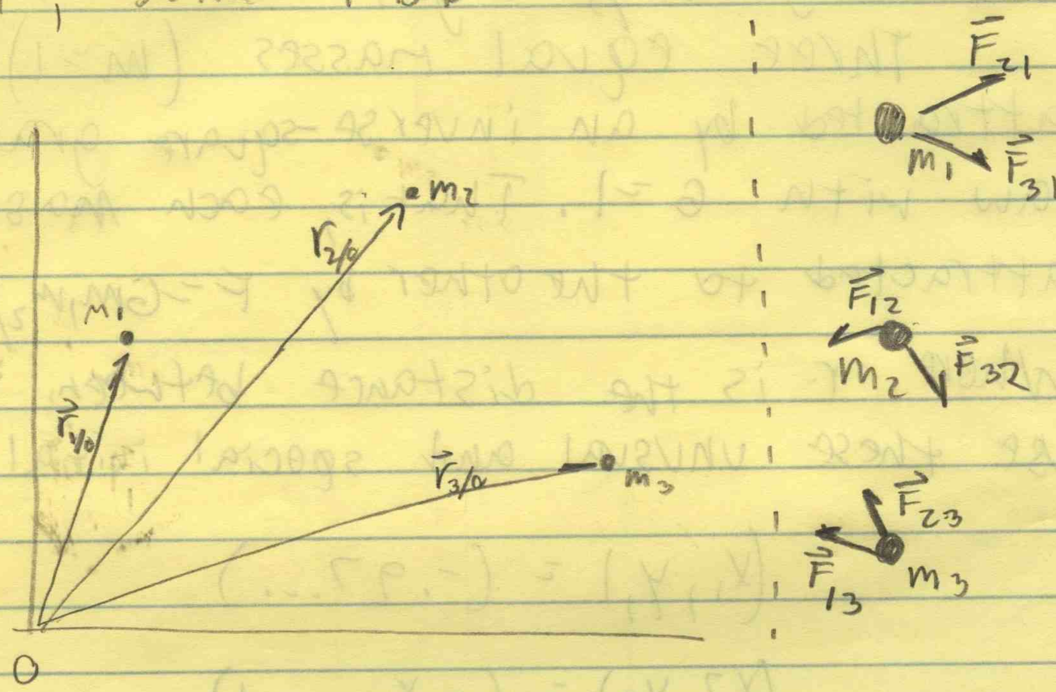
$$(vx1, vy1) = -(vx3, vy3)/2$$

$$(vx2, vy2) = -(vx3, vy3)/2$$

For each of the problems below show accurate plots and explain any curiosities.

a) Find the motion of the particles. Plot each with a different color. Run for 2.1 time units.

First, some FBD:



where $\vec{F}_{ij} = \dfrac{-G m_i m_j}{|r_{j/0} - r_{i/0}|^2} \dfrac{r_{j/0} - r_{i/0}}{|r_{j/0} - r_{i/0}|}$

so, a force balance for each particle yields:

① $m_1 \vec{a}_{1/0} = \dfrac{-G m_1 m_2}{|\vec{r}_{1/0} - \vec{r}_{2/0}|^2} \dfrac{r_{1/0} - r_{2/0}}{|r_{1/0} - r_{2/0}|} - \dfrac{G m_1 m_3}{|\vec{r}_{1/0} - \vec{r}_{3/0}|^2} \dfrac{\vec{r}_{1/0} - \vec{r}_{3/0}}{|\vec{r}_{1/0} - \vec{r}_{3/0}|}$

② $m_2 \vec{a}_{2/0} = -\dfrac{G m_2 m_1}{|\vec{r}_{2/0} - \vec{r}_{1/0}|^2} \dfrac{\vec{r}_{2/0} - \vec{r}_{1/0}}{|\vec{r}_{2/0} - \vec{r}_{1/0}|} - \dfrac{G m_2 m_3}{|\vec{r}_{2/0} - \vec{r}_{3/0}|^2} \dfrac{r_{2/0} - r_{3/0}}{|r_{2/0} - r_{3/0}|}$

③ $m_3 \vec{a}_{3/0} = \dfrac{-G m_3 m_1}{|\vec{r}_{3/0} - \vec{r}_{1/0}|^2} \dfrac{\vec{r}_{3/0} - \vec{r}_{1/0}}{|\vec{r}_{3/0} - \vec{r}_{1/0}|} - \dfrac{G m_3 m_2}{|\vec{r}_{3/0} - \vec{r}_{2/0}|^2} \dfrac{\vec{r}_{3/0} - \vec{r}_{2/0}}{|\vec{r}_{3/0} - \vec{r}_{2/0}|}$

setting up a numerical solution with Euler's method,

$$\vec{r}_j[n+1] = \vec{r}_j[n] + \vec{v}_j[n]\Delta t$$

$$\vec{v}_j[n+1] = \vec{v}_j[n] + \vec{a}_j[n]\Delta t$$

$$\vec{a}_j[n+1] = \underbrace{\vec{a}_{j/o}\left(\vec{v}_j[n], \vec{r}_j[n]\right)}_{1,\ 2,\ \text{cr}\ 3}$$

This is coded up in 'MontsEight.m'

SOLUTION - Attached plot. Interesting trajectory

b) same as above, but run for 10 time units

SOLUTION. Attached. Remarkably stable!

c) same as above, but change the ICs slightly.

SOLUTION

Attached, I varied one different parameter in two separate trials. You can see the results for slightly increasing the initial x position of m_2, or increasing its mass. The one where we changed position seems to rotate around the center, where the one with adjusted velocity seems to translate ✓

d) same as above, but change ICs more and run for a much longer time.

SOLUTION - Attached. After 50 seconds, we indeed see the position-adjusted case rotate around the center of mass, where the velocity-adjusted <u>one flies away!</u>

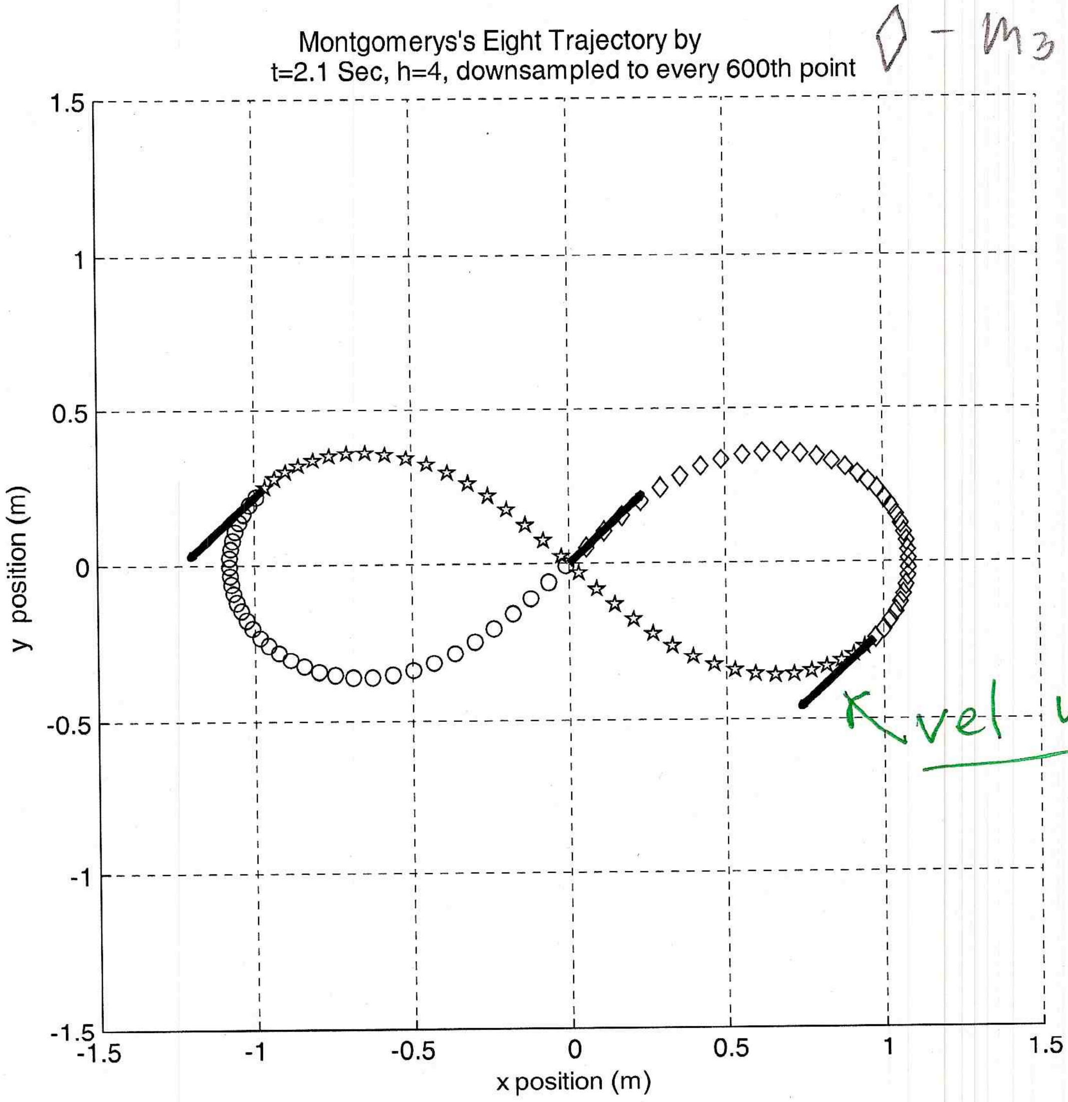
✓

```
function MontsEight(h) %by

%this function outputs the orbit of three particles subjected to the
%specific initial conditions gven from time 0 to 'totalTime'
%at a timestep of 10^-h

deltaT=10^-h;
totalTime=50;

numSteps=totalTime/deltaT;

G=1; %define parameters
m1=1;
m2=1;
m3=1;

dmax=1.5; %set up plotting
figure(1)
hold on
axis('square');
axis([-dmax, dmax,-dmax,dmax,-dmax,dmax]);
grid on
title('Montgomerys''s Eight Trajectory')

%2*pi*.5/sqrt(4*pi^2/(G*(m1+m2)));


p1x0=-.97000436; %define initial position on particle 1
p1y0=.24308753;
p2x0=-p1x0; %define initial position on particle 2
p2y0=-p1y0;
p3x0=0; %define initial position on particle 3
p3y0=0;

r1=[p1x0,p1y0]; %put the position ICs in vector form
r2=[p2x0,p2y0];
r3=[p3x0,p3y0];

v3x0=.93240737; %initial velocity on particle 3
v3y0=.86473146;
v1x0=-v3x0/2*1.1;    %initial velocity on particle 1
v1y0=-v3y0/2;
v2x0=-v3x0/2;   %initial velocity on particle 2
v2y0=-v3y0/2;


v1=[v1x0,v1y0]; %put the velocity ICs in vector form
v2=[v2x0,v2y0];
v3=[v3x0,v3y0];

quiver(p1x0,p1y0,v1x0,v1y0,.5,'r','LineWidth',3) %show initial trajectories
```

```
quiver(p2x0,p2y0,v2x0,v2y0,.5,'b','LineWidth',3)
quiver(p3x0,p3y0,v3x0,v3y0,.25,'g','LineWidth',3)

for n=1:numSteps %actually perform Euler's

    %plot them, but only every 1500th point
    if mod(n,1500)==0
        scatter(r1(1),r1(2),'ro');
        scatter(r2(1),r2(2),'bp');
        scatter(r3(1),r3(2),'gd');
    end

    %update
    a1=-G*m2/norm(r1-r2)^3*(r1-r2)-G*m3/norm(r1-r3)^3*(r1-r3);
    a2=-G*m1/norm(r2-r1)^3*(r2-r1)-G*m3/norm(r2-r3)^3*(r2-r3);
    a3=-G*m2/norm(r3-r2)^3*(r3-r2)-G*m1/norm(r3-r1)^3*(r3-r1);

    r1=r1+v1*deltaT;
    r2=r2+v2*deltaT;
    r3=r3+v3*deltaT;

    v1=v1+a1*deltaT;
    v2=v2+a2*deltaT;
    v3=v3+a3*deltaT;
end
```

a)

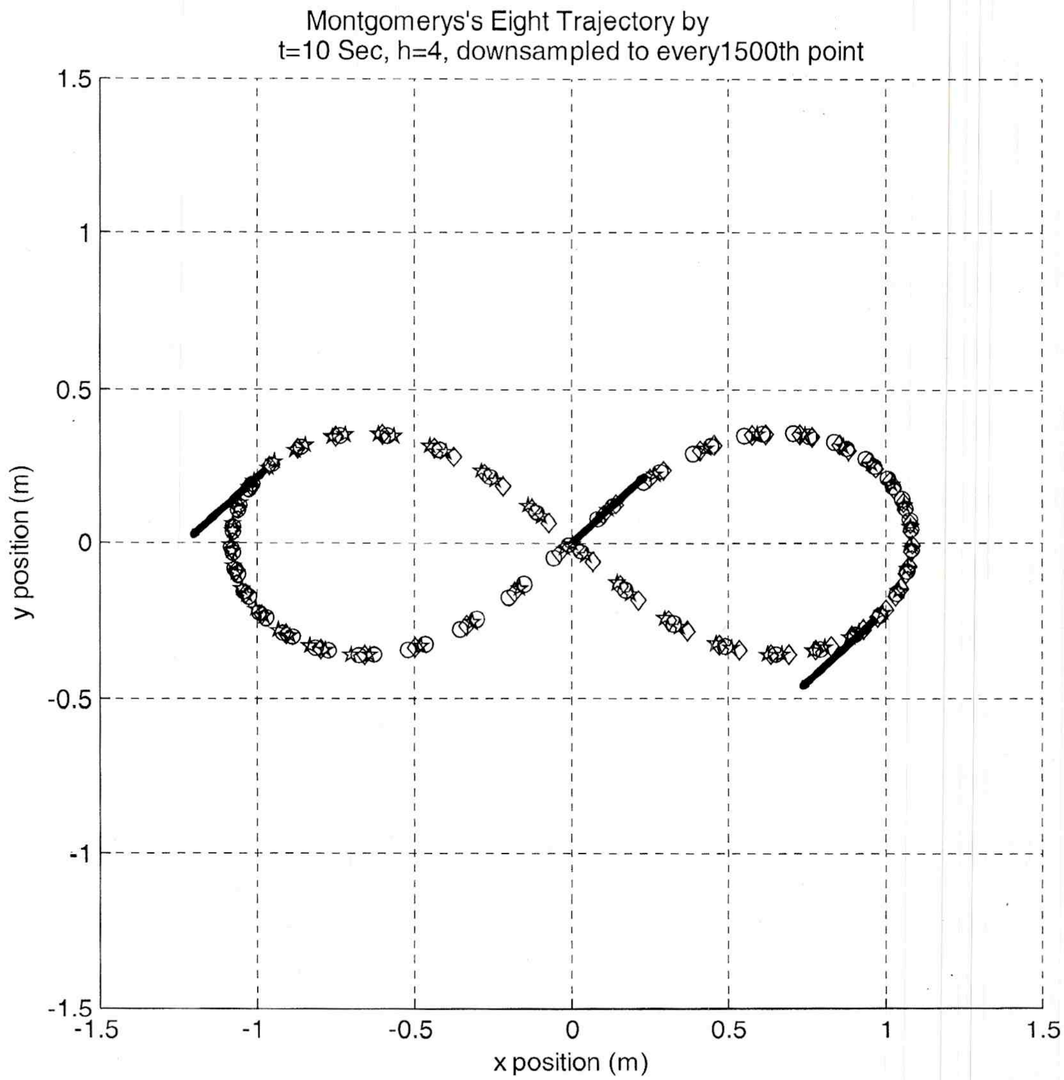
O - m_1
☆ - m_2
◇ - m_3



vel vector?

36

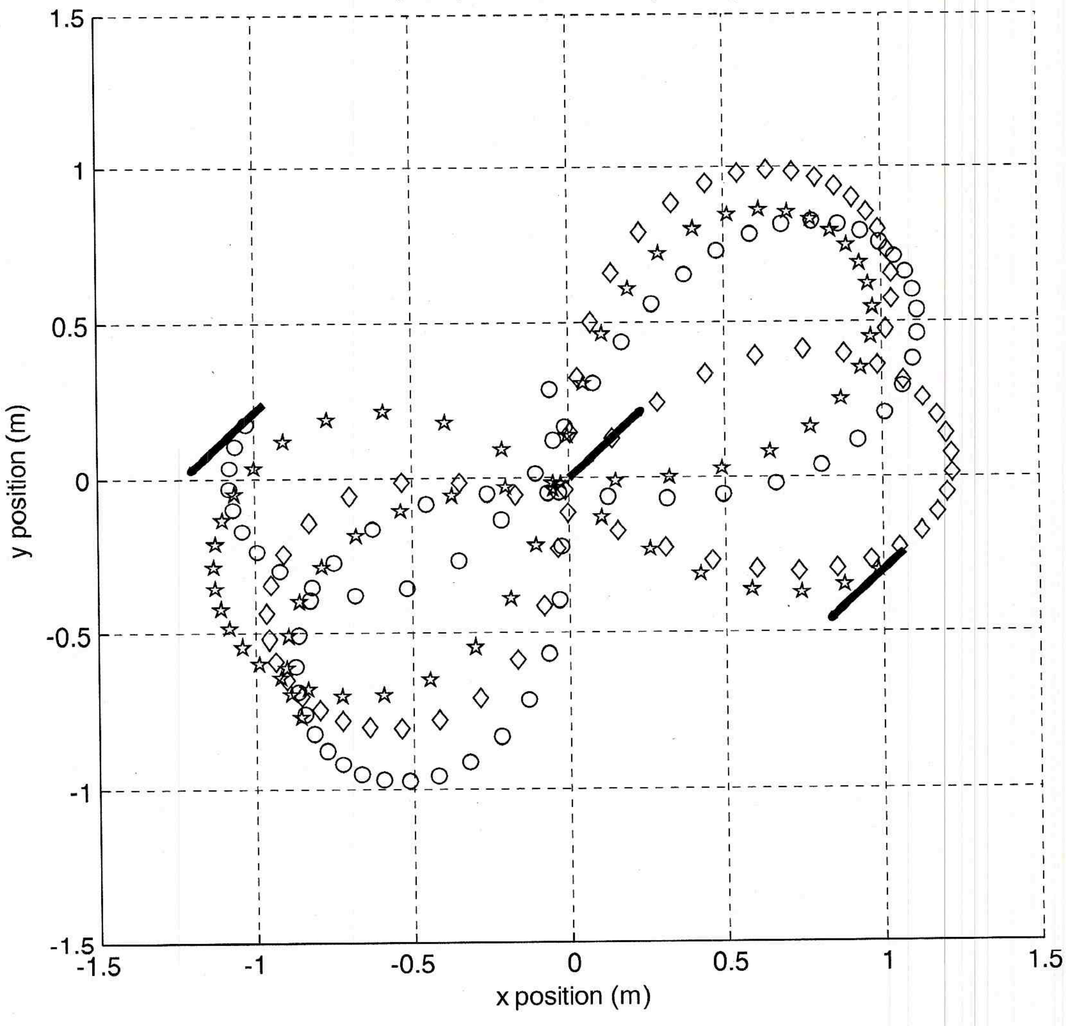
b)


Montgomerys's Eight Trajectory by
t=10 Sec, h=4, downsampled to every1500th point
1.5
1
0.5
0
-0.5
-1
-1.5
-1.5
-1
-0.5
0
0.5
1
1.5
y position (m)
x position (m)

c)

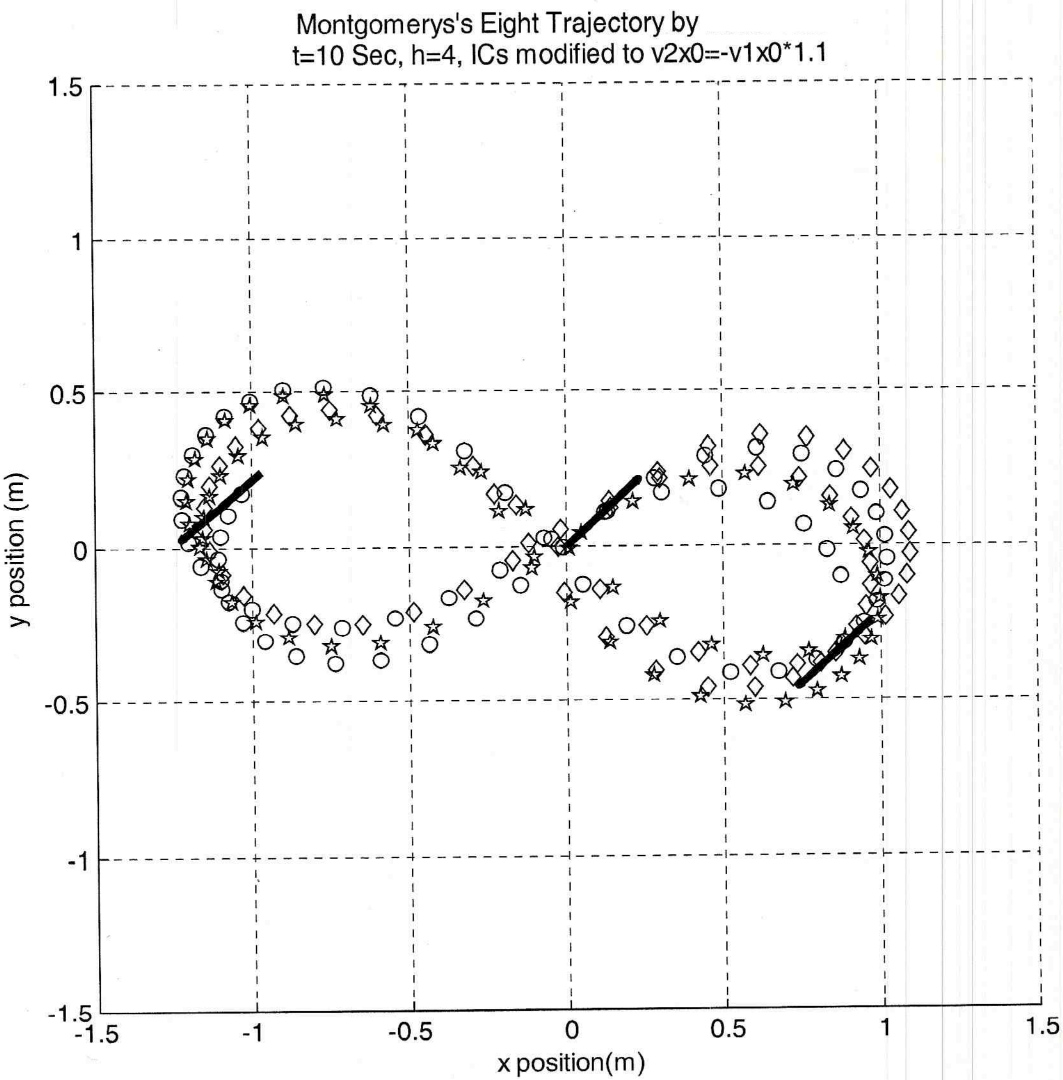
position mod

Montgomerys's Eight Trajectory by
t=10 Sec, h=4, ICs modified to p2x0=-p1x0*1.1



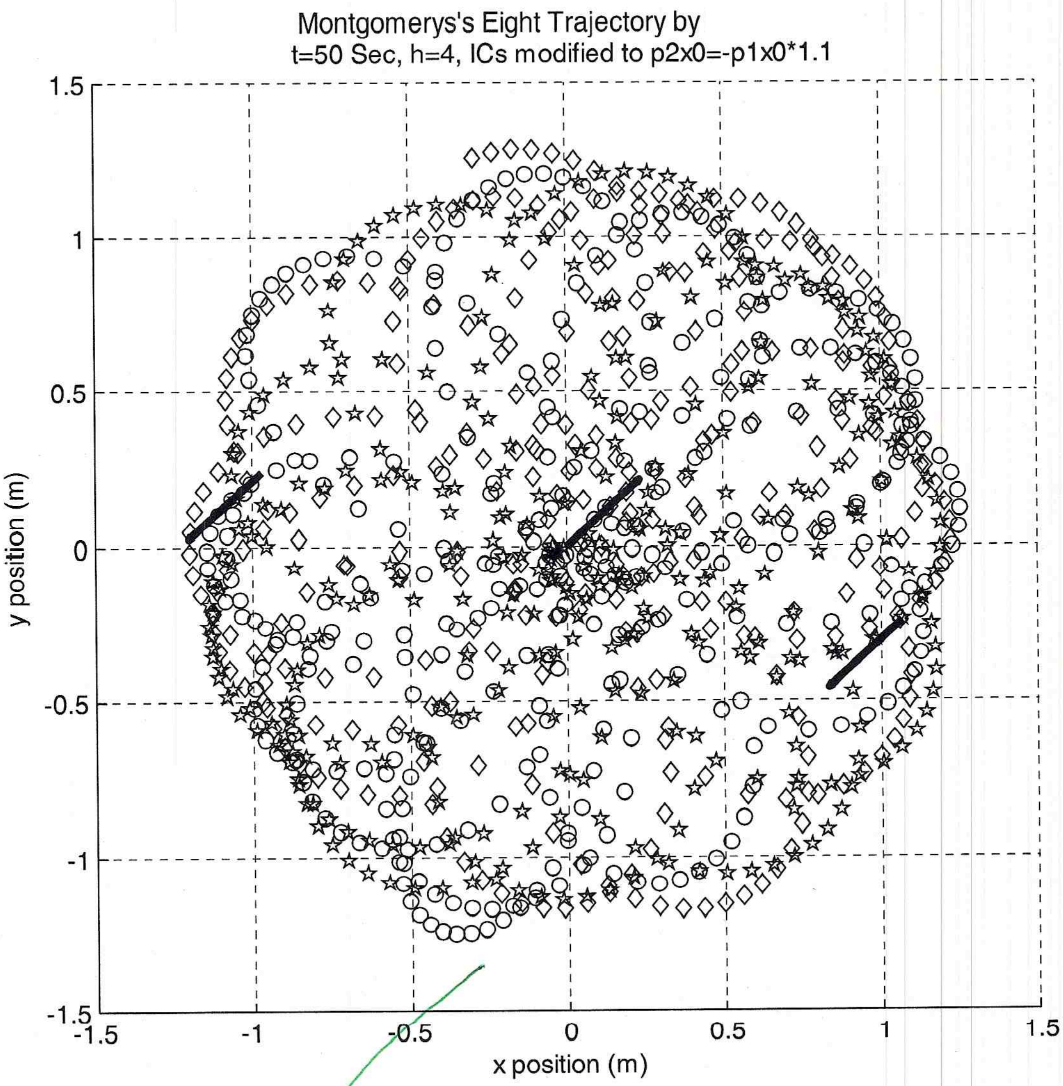
c)

velocity rod



d)

Position mod



Maybe more clean if you drew curves w/ no ◇, ☆, ○.

velocity mod d)

Montgomerys's Eight Trajectory by
t=50 Sec, h=4, ICs modified to v2x0=-v1x0*1.1