Foucault Pendulum

-by Group 2G

Method:

In each method, there will be two vectors, the State vector and the Function vector constantly present. The State vector stores the x and y coordinates along with the x and y velocities, while the Function vector stores the differentiated elements of the State vector.

The equations were:
$x'' - 2\Omega y' + \omega*\omega x=0$
$y'' + 2\Omega x' + \omega*\omega y=0$
(Here prime notation indicates differentiation with respect to time)

State Vector $Y= [x, x', y, y']$
Function Vector $f= [x', x'', y', y'']= [x', 2\Omega y'-\omega*\omega x, y', -2\Omega x'-\omega*\omega x]$

So each element correspondence can be taken to be a differential equation-i.e, we are solving 4 different differential equations.

1) Euler's Method:
In Euler's method,
$Yn+1 = Yn + dt*f(Yn)$ is a step advancement.

2) Heun's Method:
$k1 = f(Yn)$
$k2 = f(Yn+dt*k1)$
$Yn+1 = Yn+(dt/2)*(k1+k2)$ describes step advancement.
k1 and k2 are two step vectors

3) Runge-Kutta's Method:
$k1 = f(Yn)$
$k2 = f(Yn + (dt/2)*k1)$
$k3 = f(Yn + (dt/2)*k2)$
$k4 = f(Yn+dt*k3)$
$Yn+1 = Yn + (dt/6)*(k1 +2k2 +2k3 +k4)$

4) RK45 (Fehlberg's) Method:
Here the coefficients used were developed by Cash and Karp- a shot is put up here:

$$k_1 = f(x_i, y_i)$$

$$k_2 = f\left(x_i + \frac{1}{5}h,\ y_i + \frac{1}{5}k_1 h\right)$$

$$k_3 = f\left(x_i + \frac{3}{10}h,\ y_i + \frac{3}{40}k_1 h + \frac{9}{40}k_2 h\right)$$

$$k_4 = f\left(x_i + \frac{3}{5}h,\ y_i + \frac{3}{10}k_1 h - \frac{9}{10}k_2 h + \frac{6}{5}k_3 h\right)$$

$$k_5 = f\left(x_i + h,\ y_i - \frac{11}{54}k_1 h + \frac{5}{2}k_2 h - \frac{70}{27}k_3 h + \frac{35}{27}k_4 h\right)$$

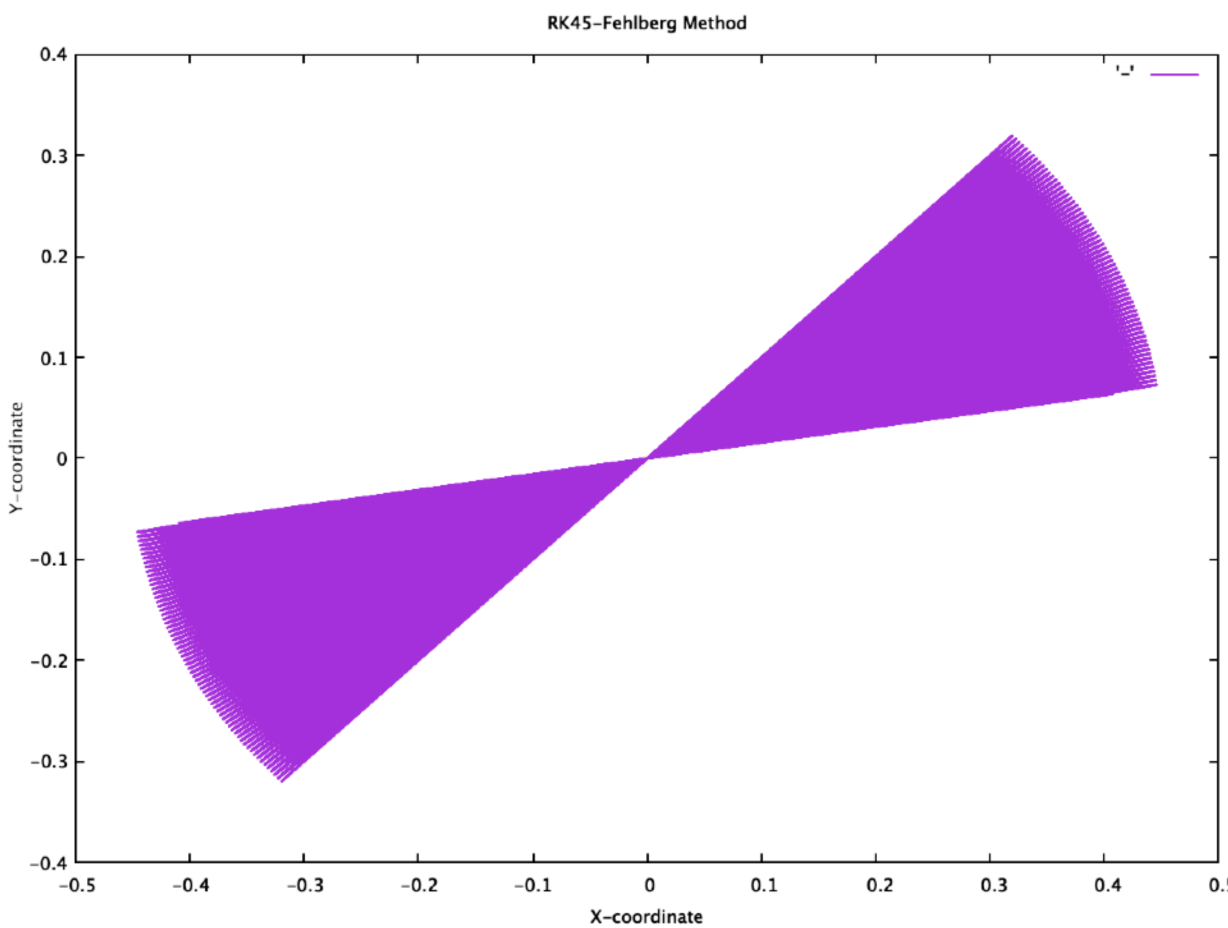
$$k_6 = f\left(x_i + \frac{7}{8}h,\ y_i + \frac{1631}{55{,}296}k_1 h + \frac{175}{512}k_2 h + \frac{575}{13{,}824}k_3 h + \frac{44{,}275}{110{,}592}k_4 h + \frac{253}{4096}k_5 h\right)$$

$$y_{i+1} = y_i + \left(\frac{2825}{27{,}648}k_1 + \frac{18{,}575}{48{,}384}k_3 + \frac{13{,}525}{55{,}296}k_4 + \frac{277}{14{,}336}k_5 + \frac{1}{4}k_6\right)h$$

This method gives a better error estimate (although it is not asked for here).

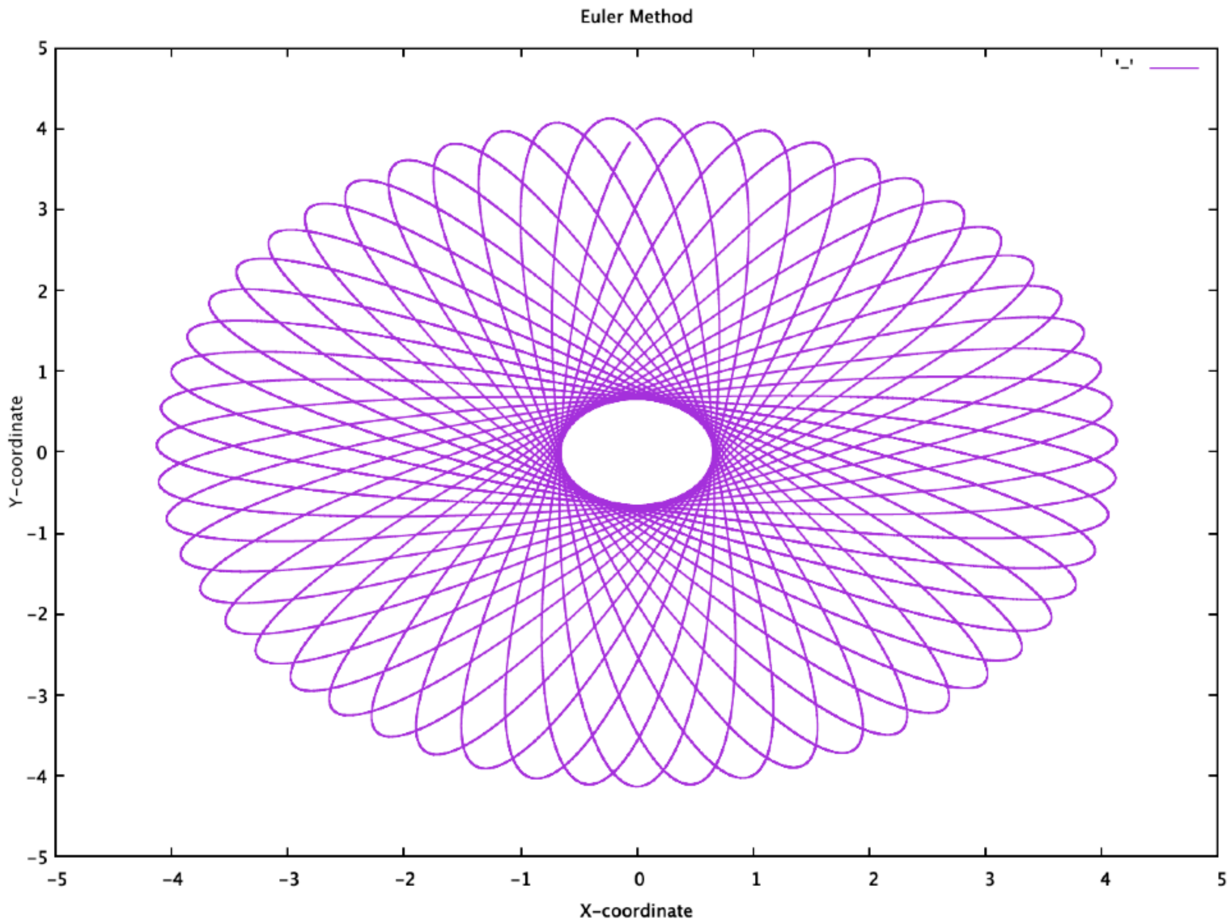
Graphs and Observations:

By taking the Coriolis coefficient to be that of earth, 0.000073, and length of pendulum to be 1m, we obtain a graph as follows from RK45 method:



So this shows the plane of oscillations changing throughout the day- although the above graph is only for a few hours. This is a successful simulation of the Foucault pendulum.

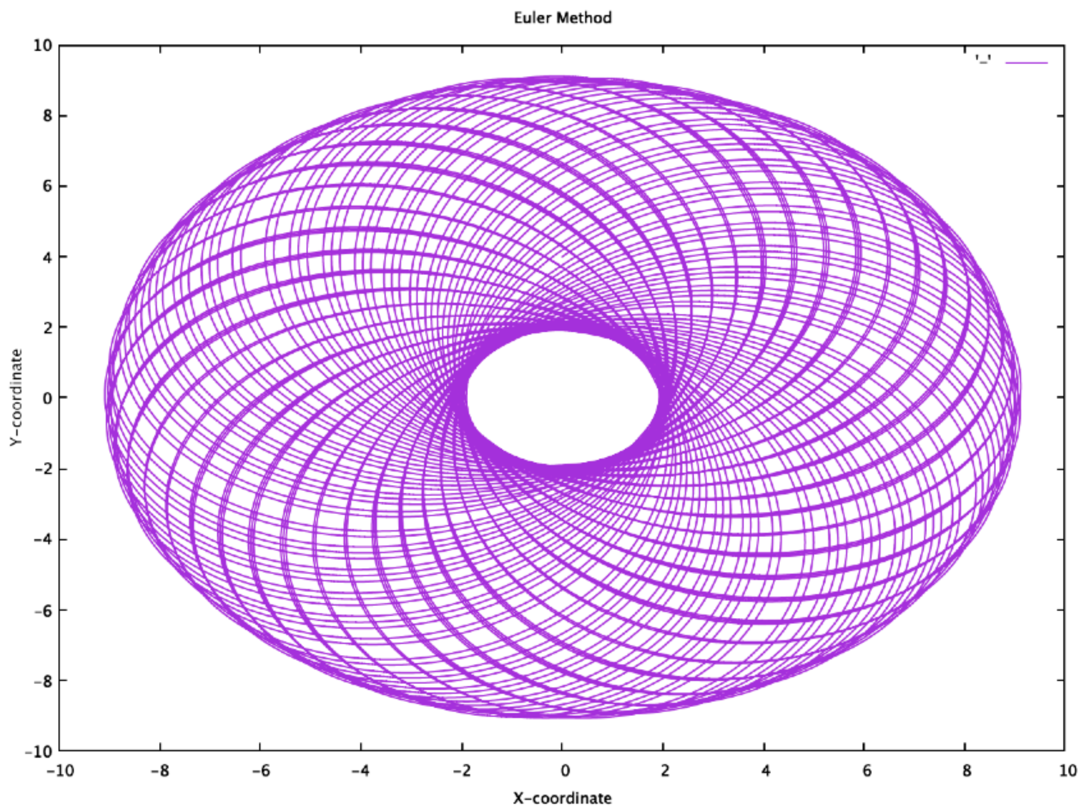
However, the plane of rotation of the previous pendulum changes slowly. For a greater value of Coriolis coefficient (0.073),



Here we can clearly observe the precessing ellipses. This graph was generated at the same natural frequency of 3.13 and same initial velocities of 1m/s and 1m/s.

Accuracy Comparisons:

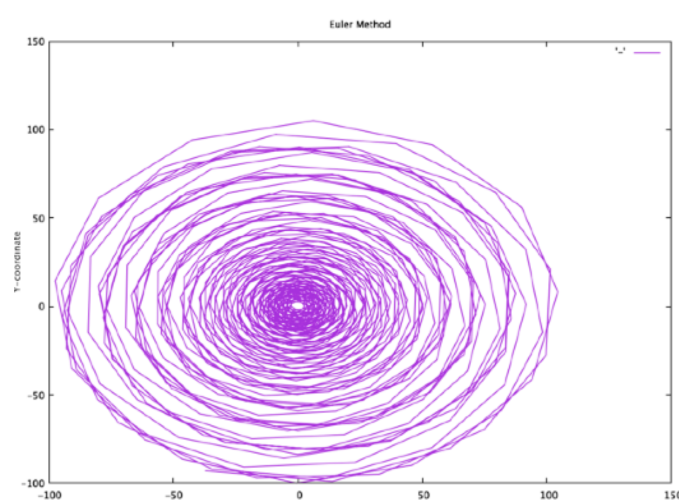
We can make some comparisons between these methods by noticing the time steps required to obtain a clean graph.
For example,



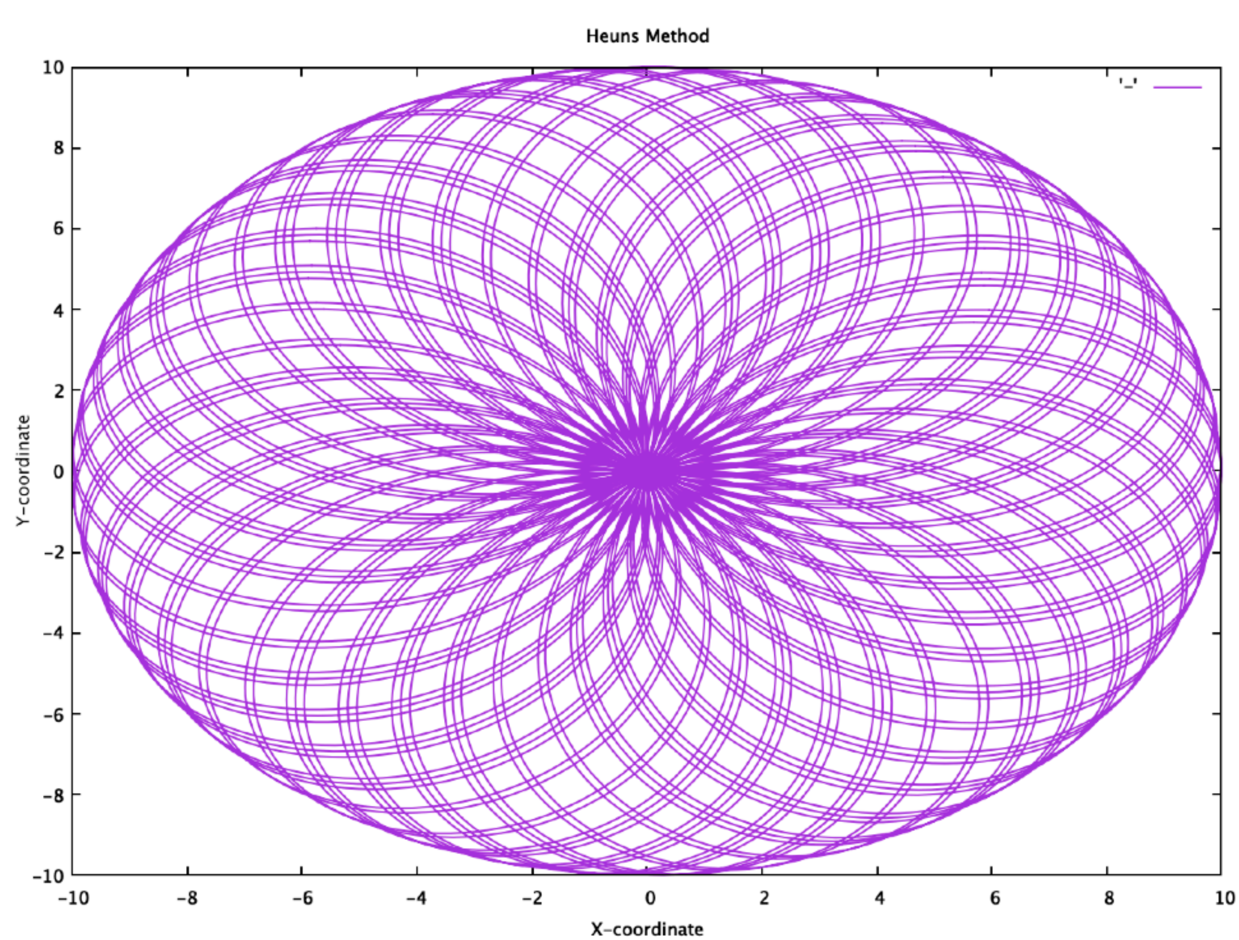
This graph was obtained by taking value of time step as 0.000411:

```
./a.out e 0.1 0.1 1 1 0.000411 6000000
```

Any greater time step resulted in this, which is, I promise, not from a horror movie.



Using Heun’s,

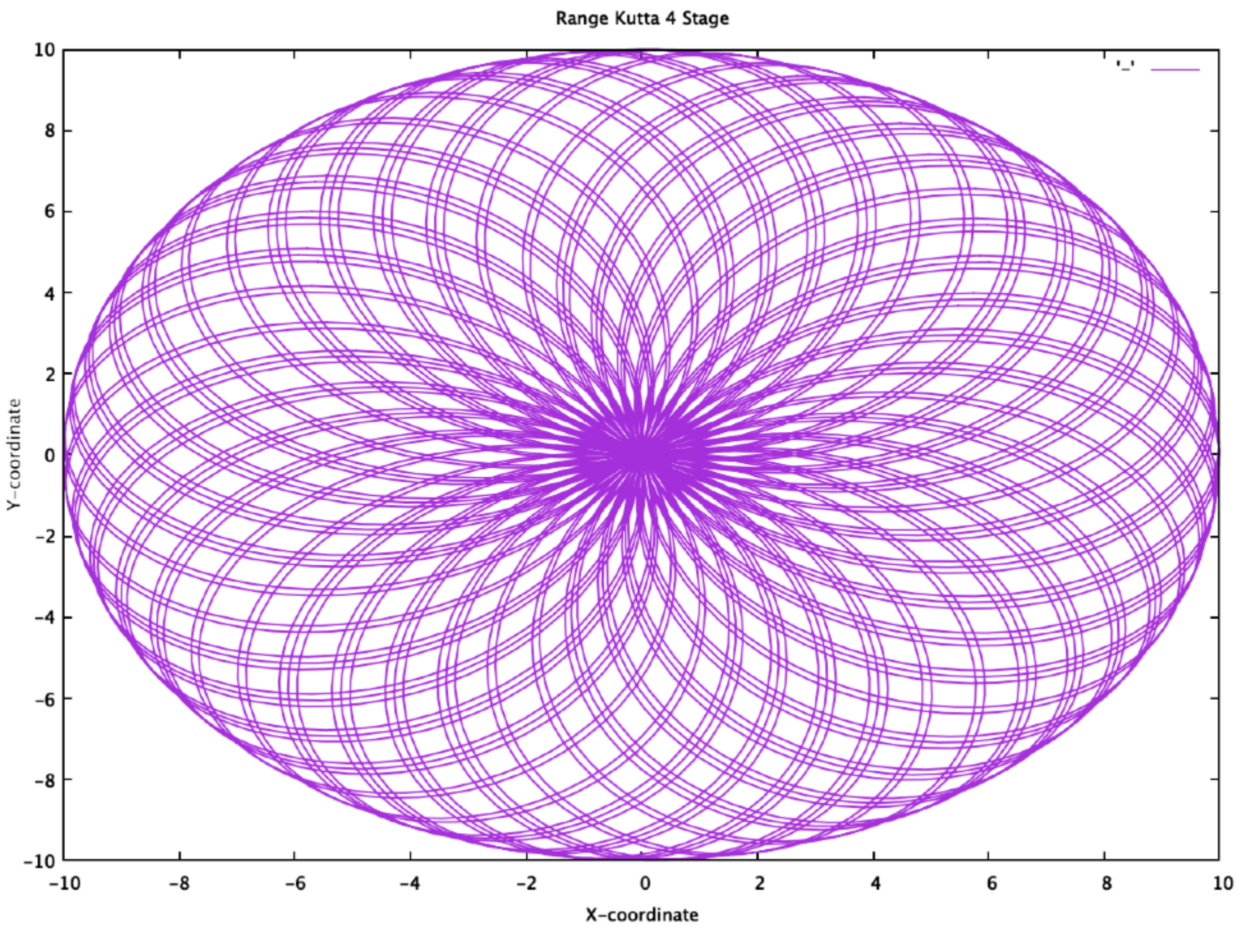


This was obtained for a time step of 0.00411, for values:

```
./a.out h 0.1 0.1 1 1 0.00411 600000
```

Notice that this graph has more resolution and the centre is also blocked.

Range Kutta improved on the accuracy even more:

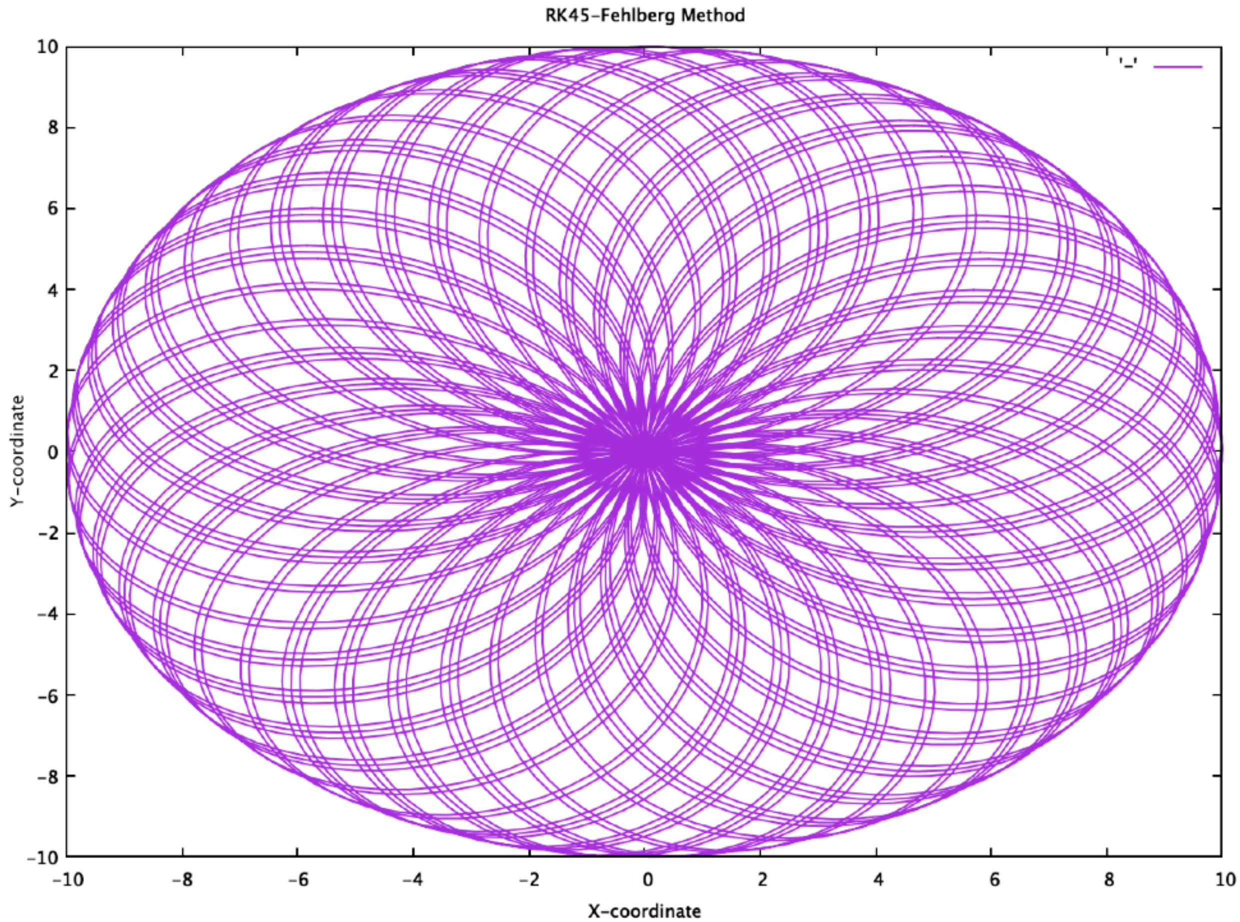


This graph came for a time step of 0.0411, at values:

```
./a.out r 0.1 0.1 1 1 0.0411 60000
```

(Here a slight modification to code was made: instead of 1:50 being printed we printed 1:10)

This same graph was obtained by RK45 method:



Here the step size was largest: 0.411
The values were:
`./a.out f 0.1 0.1 1 1 0.411 6000`

We see that of all the methods used, RK45 is most capable of handling large step sizes, while Euler is least capable.

Thank You!