EE1103: QUIZ-2

Group: 9
Date: 21st October, 2018
Members: Anirudh R(EE18B103)
Arjun Menon Vadakkeveedu(EE18B104)
Aswin Rajesh(EE18B106)
Name of c file attached: **Quiz2_v2.c**

Plots for Given Initial Conditions:

A. EULER'S METHOD(FORWARD DIFFERENCE)

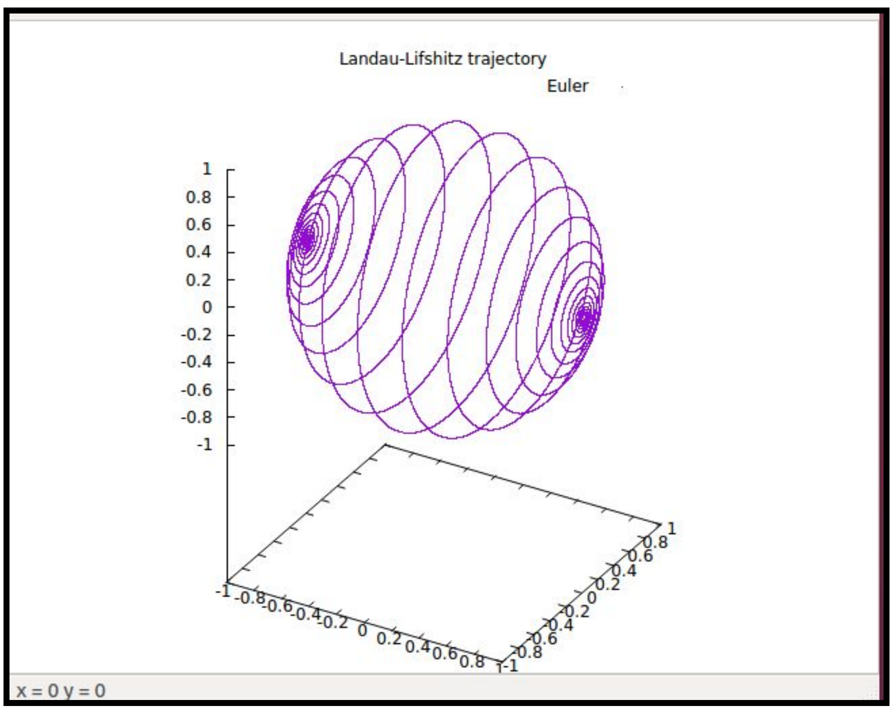


FIG1: Graph showing trajectory of M vector in 3D space.

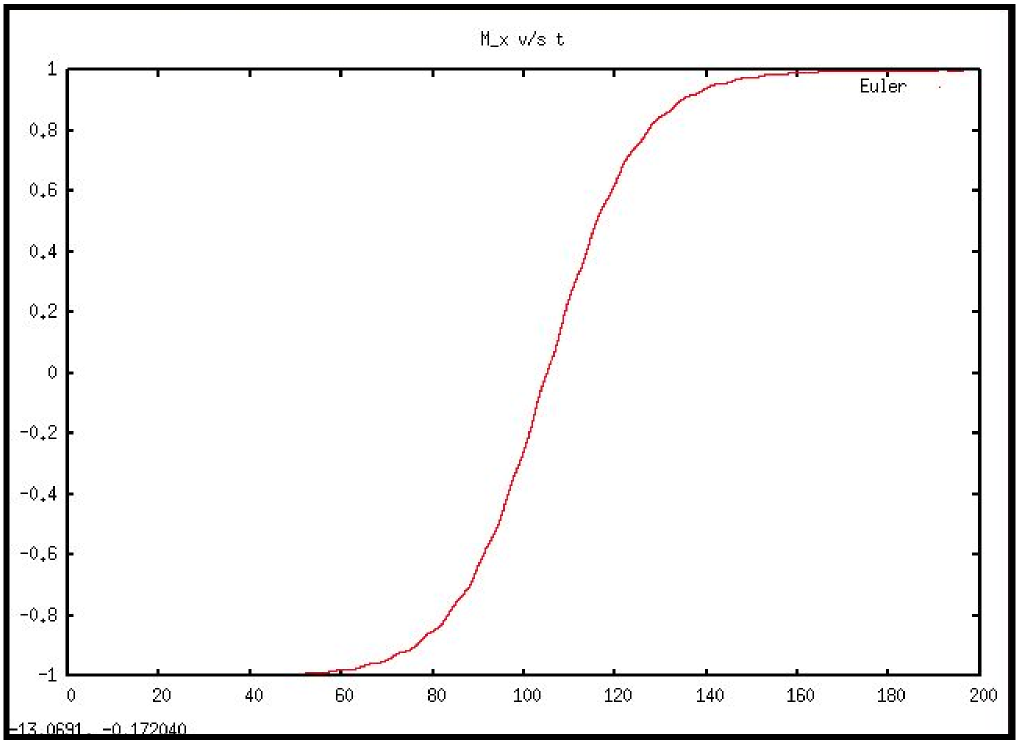


FIG2: Graph showing the variation of **M_x** with respect to **t**.

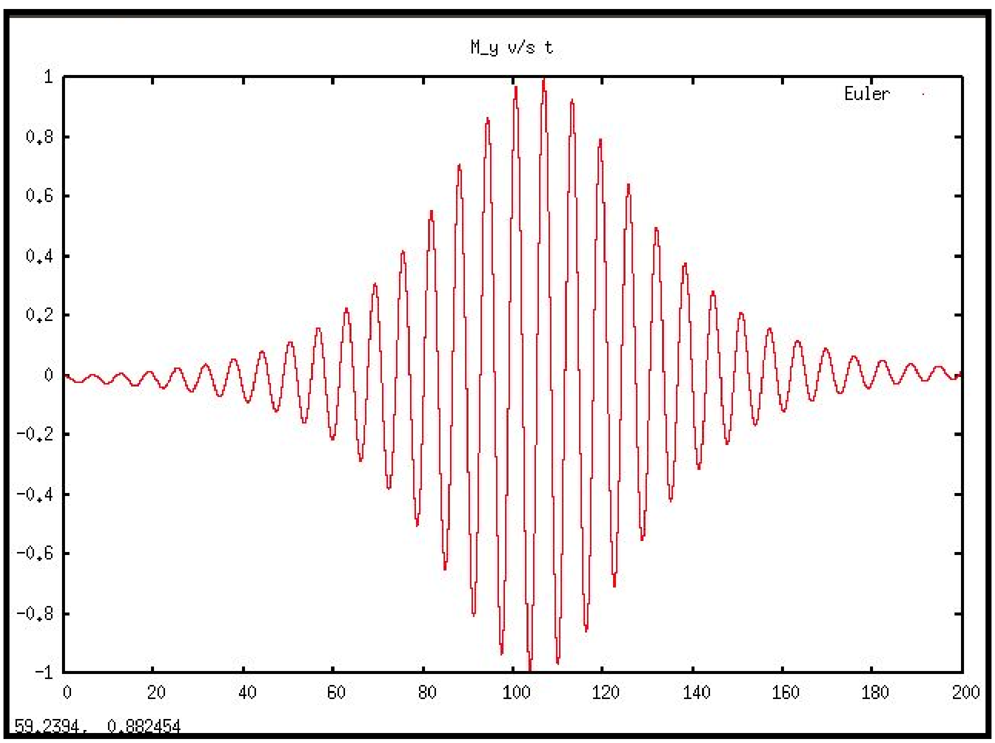


FIG3: Graph showing the variation of **M_y** with respect to **t**.

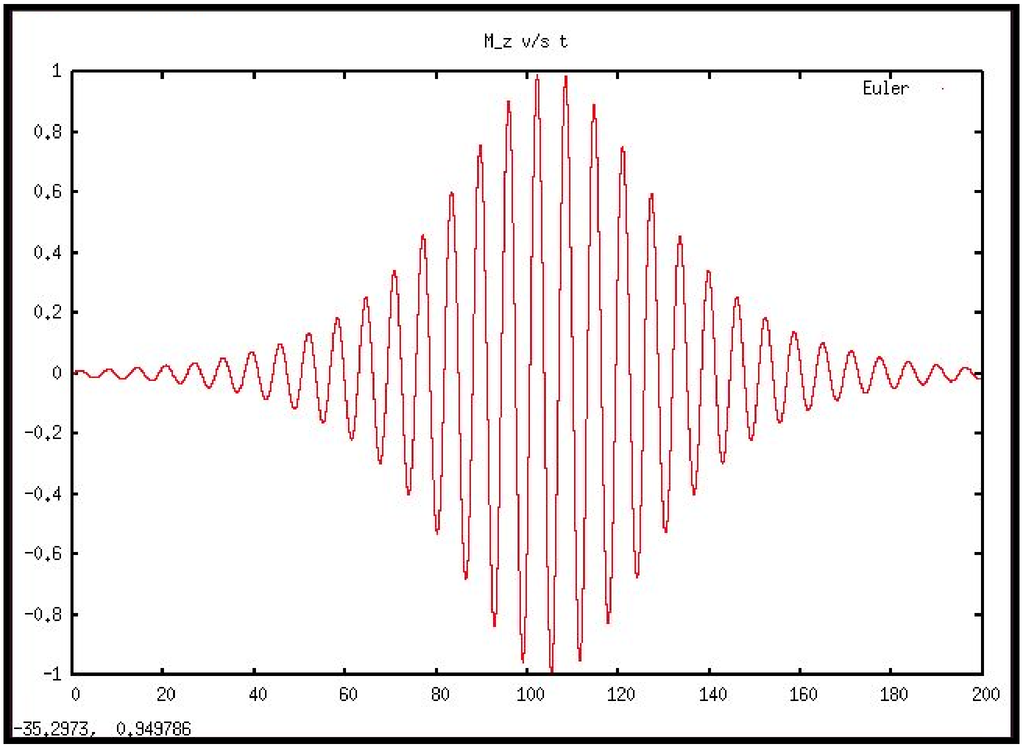


FIG4: Graph showing the variation of **M_z** with respect to **t**.

B. FOURTH-ORDER RUNGE-KUTTA'S METHOD(RK4)

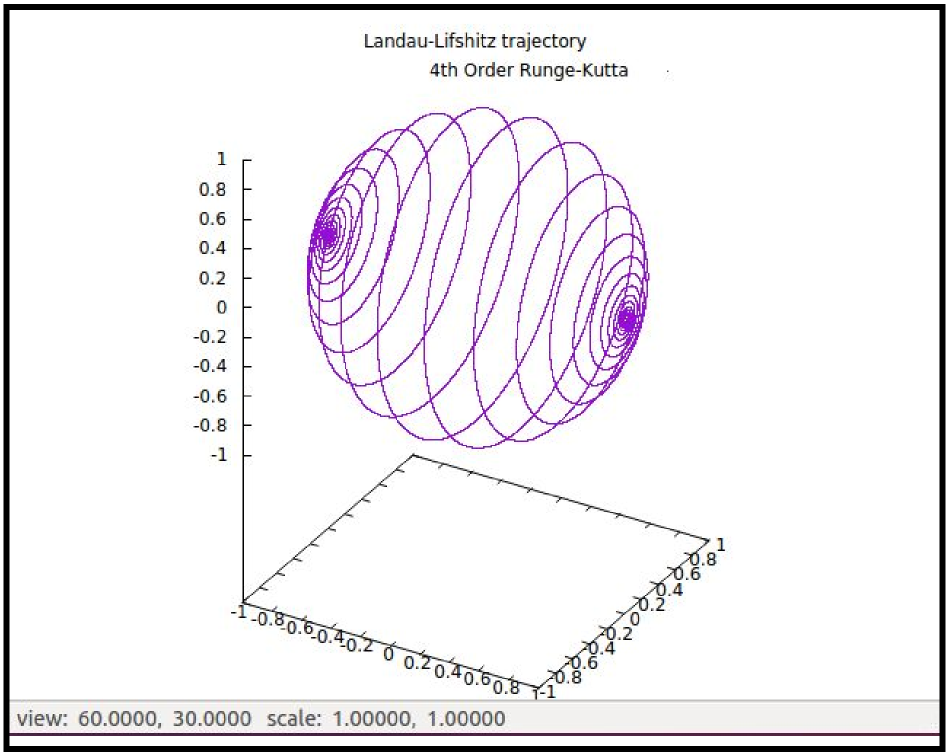


FIG5: Graph showing trajectory of M vector in 3D space.

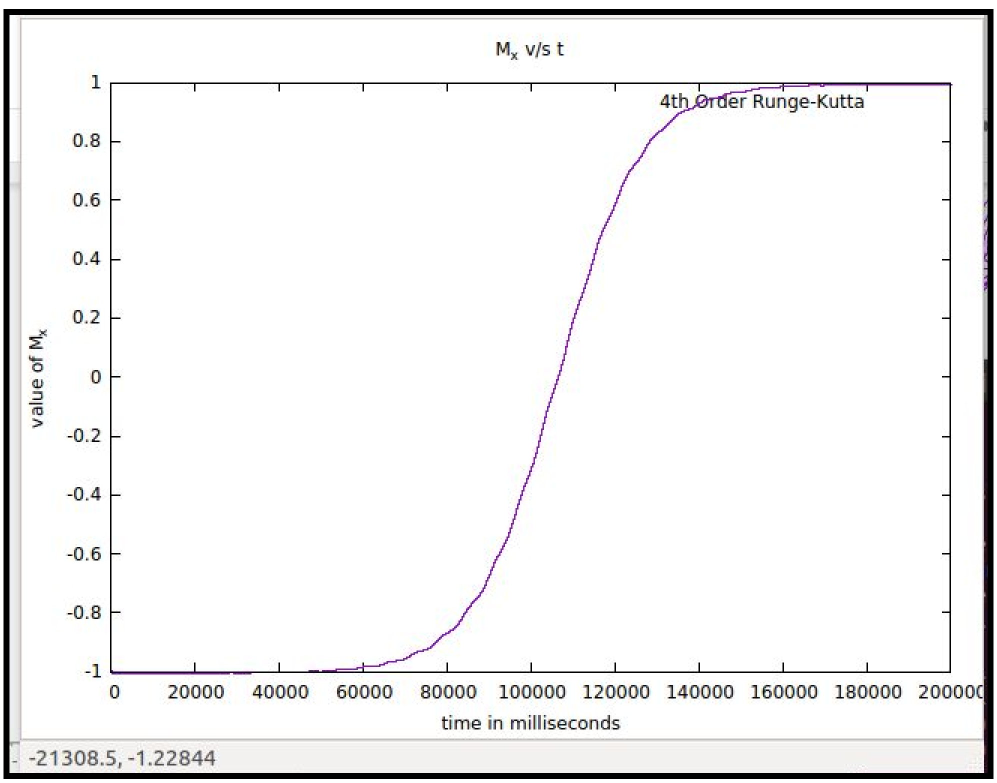


FIG6: Graph showing the variation of **M_x** with respect to **t**.

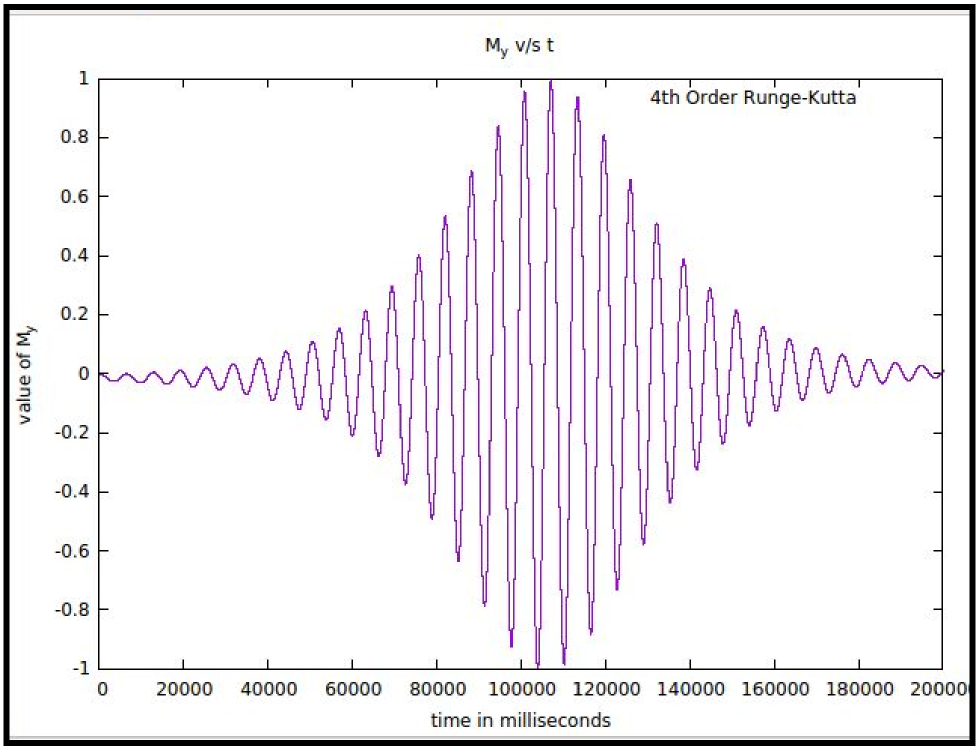


FIG7: Graph showing the variation of **M_y** with respect to **t**.

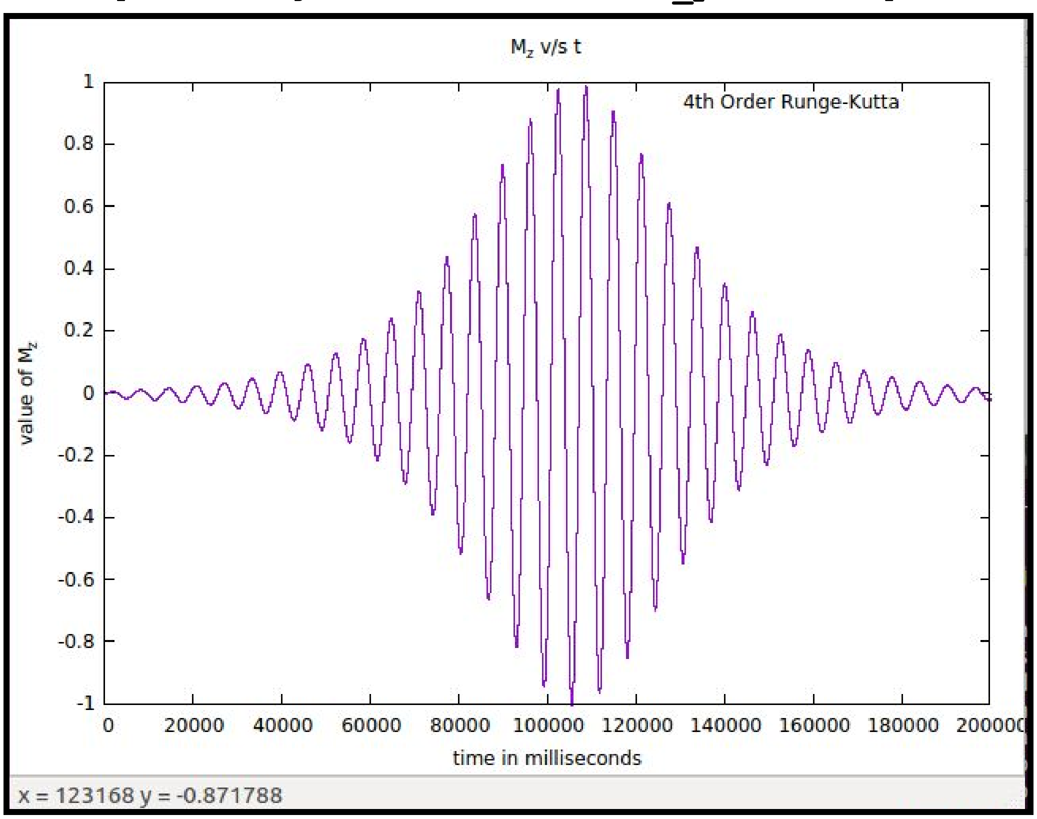


FIG 8: Graph showing the variation of **M_z** with respect to **t**.

Behaviour of the function for different values of Gamma and Alpha:

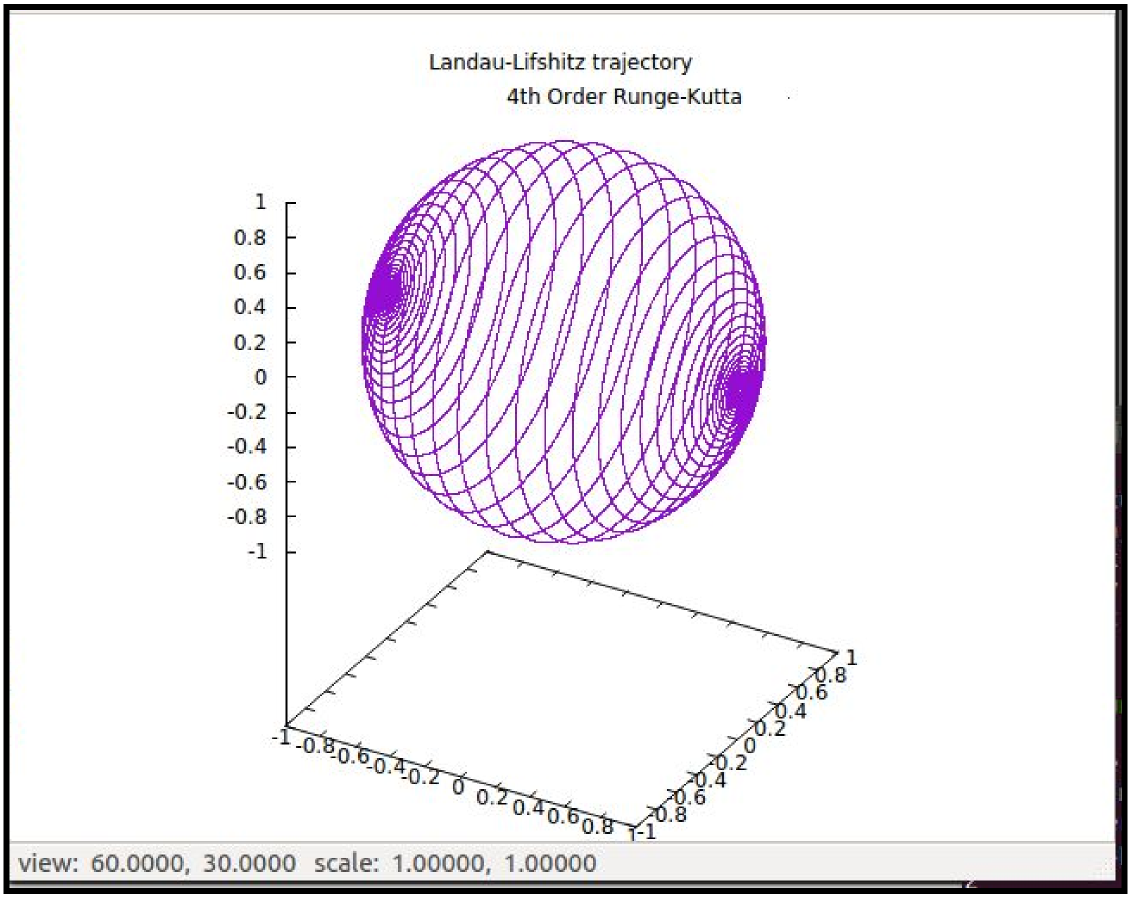


FIG. 1: Trajectory for gamma = 2.0 and alpha = -0.05

NOTICE THAT THE NUMBER OF REVOLUTIONS MADE BY THE VECTOR HAS INCREASED, AS IS EVIDENT FROM FIGURE 2

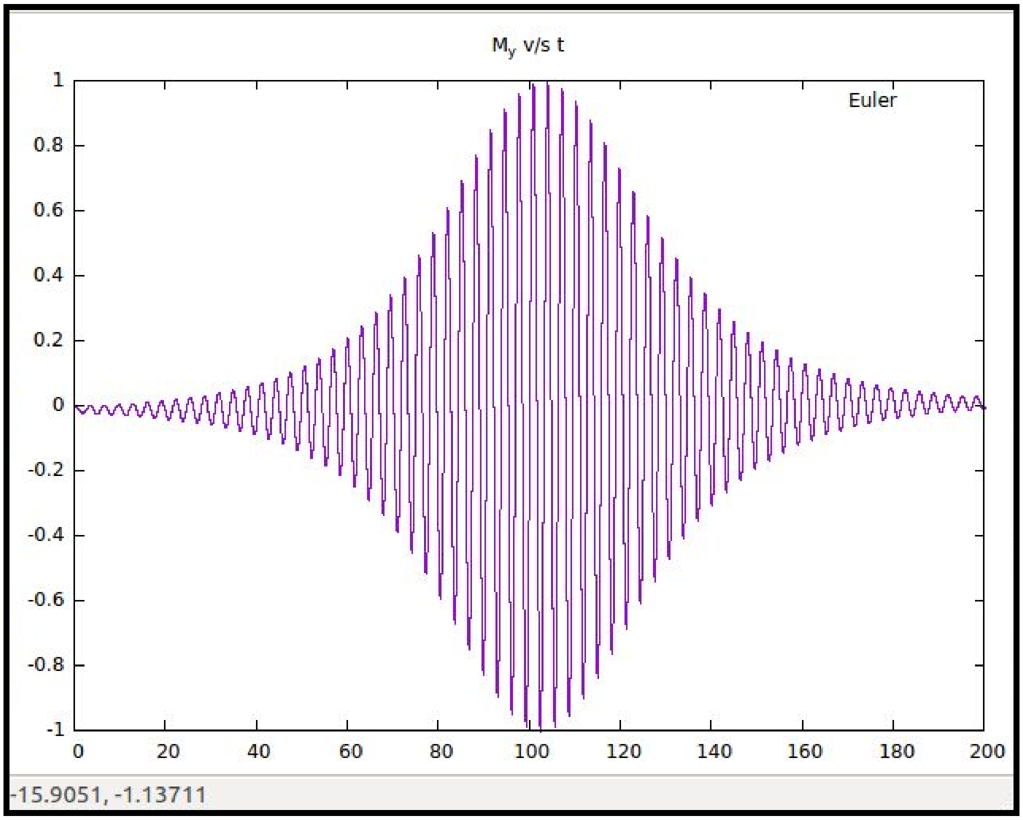


FIG 2: M_y v/s time

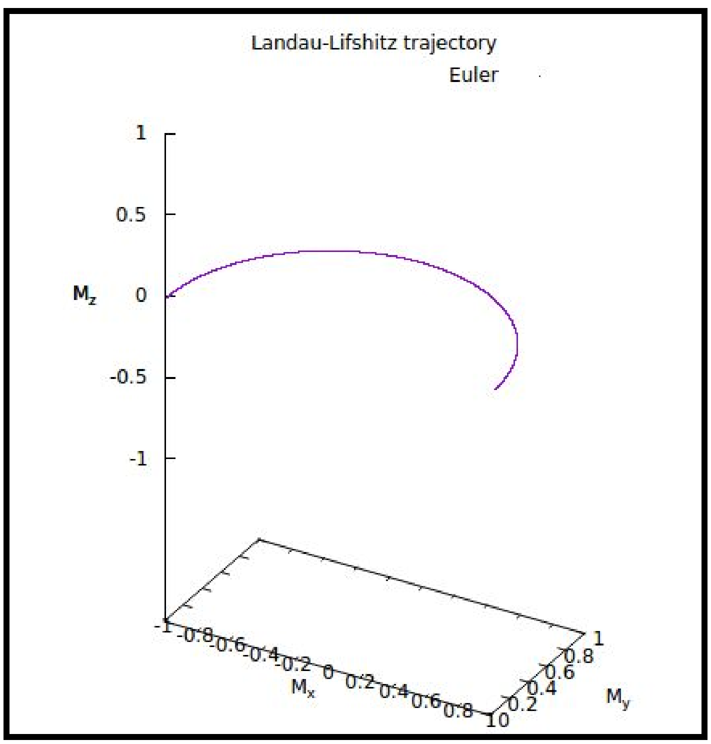


FIG 3: Trajectory for Gamma = 0 and Alpha = -0.05. No precession occurs

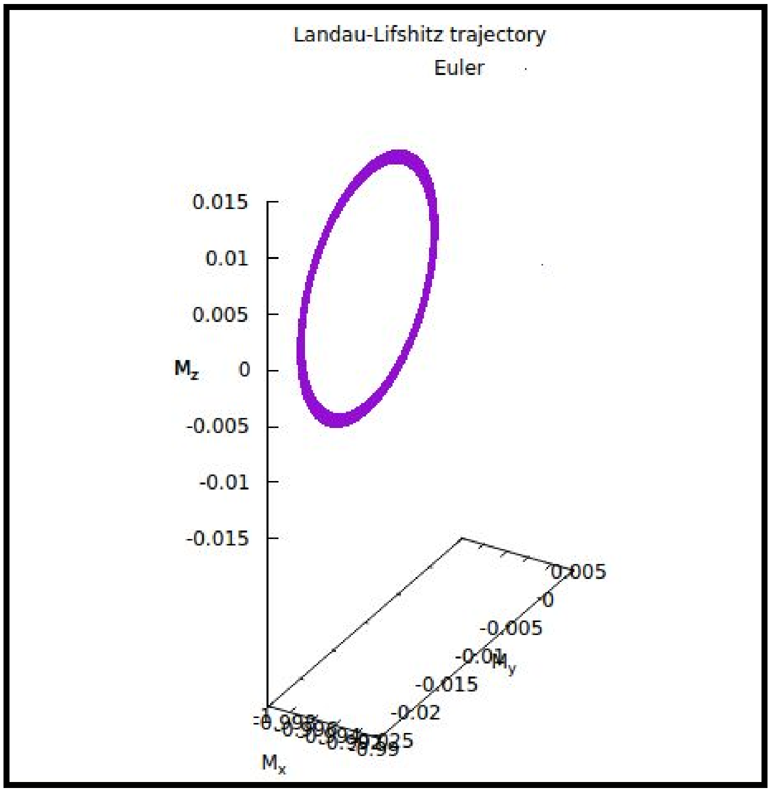


FIG 4: Trajectory for Alpha = 0 and Gamma = 1. Since only the precession term is present, the magnetisation vector rotates without any translation in x-direction

Varying Alpha

We ran a function "ALVAR()" that varies alpha and finds time for the magnetisation to switch, i.e.- M_x crosses 0. The following plots are the values of time of switching versus alpha for both Forward Difference method and RK4 method.

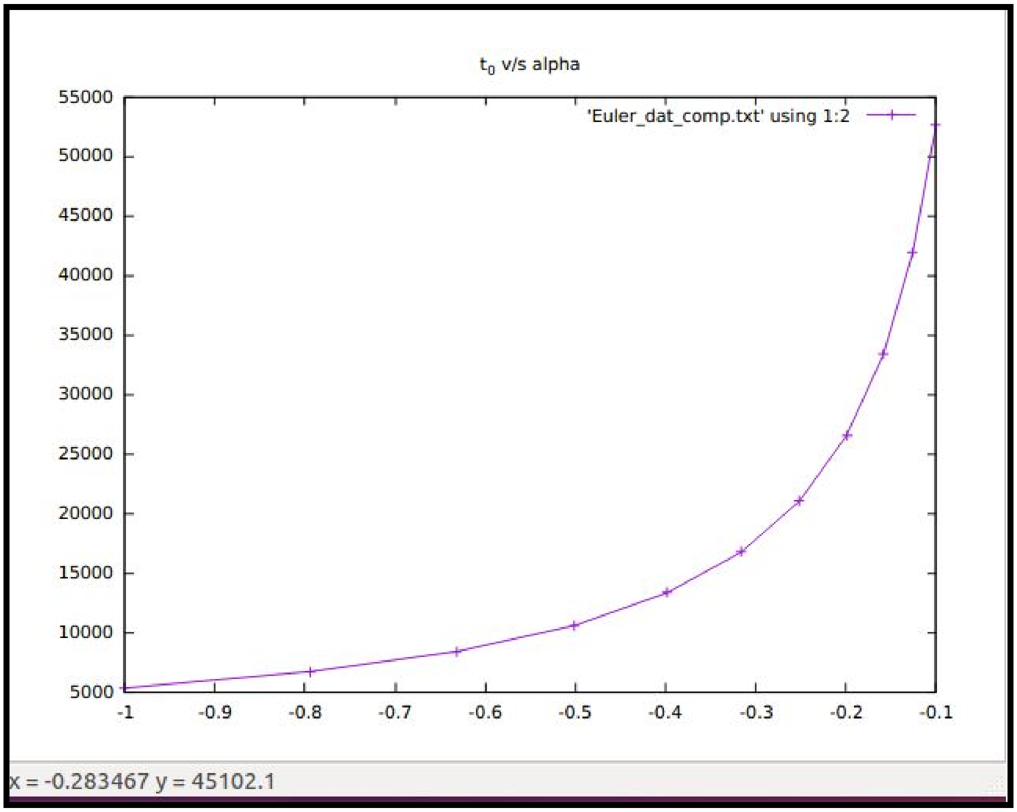


FIG 5: t-switch versus alpha for Euler's method

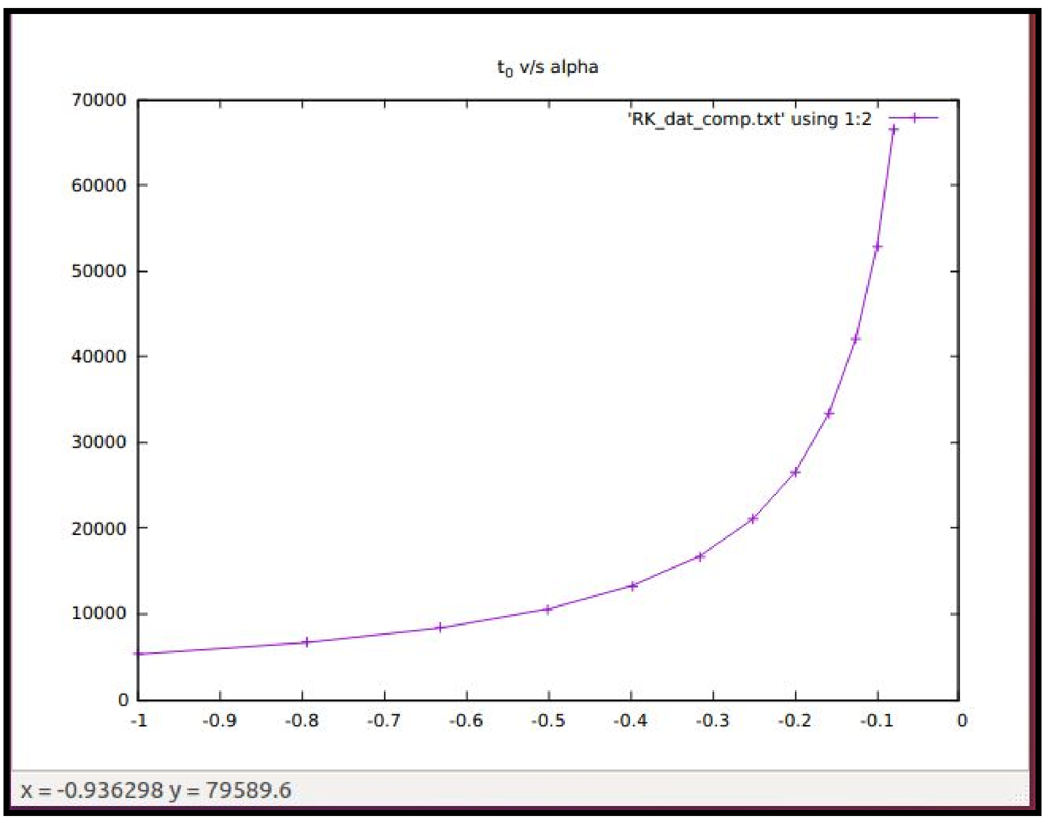


FIG 6: t-switch versus alpha for RK4 method

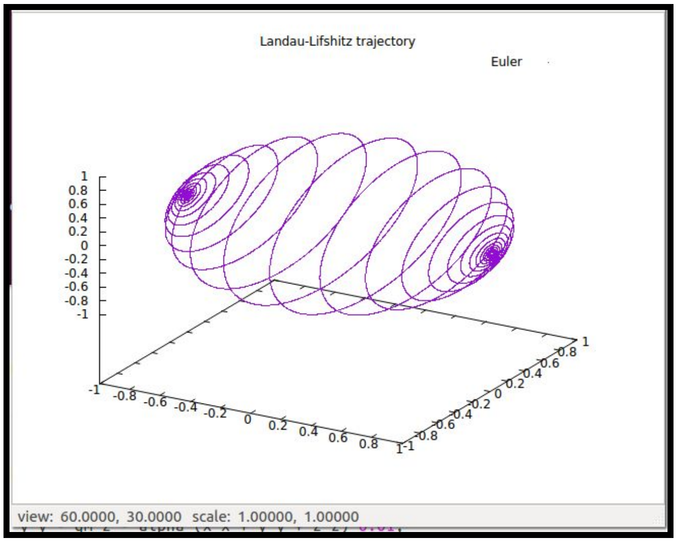
OBSERVATIONS FROM PLOTS OF t-switch versus alpha:

1. THE BEHAVIOUR FOR BOTH THE METHODS IS SOMEWHAT EXPONENTIAL. FOR HIGHER VALUES OF ALPHA, THE M_x MAY NOT EVEN CROSS 0(y-z PLANE).
   Physically, alpha behaves like a “loss” factor. Higher the alpha, the longer it takes to align the magnetic needle to the applied magnetic field.
2. A HIGHER VALUE OF t-switch WAS OBSERVED FOR THE SOLUTION OBTAINED BY RK4 METHOD WHEN COMPARED TO THAT OBTAINED BY EULER’S METHOD, FOR EVERY VALUE OF ALPHA THAT WAS CHECKED BY THE FUNCTION.
3. THE SLOPE OF THE CURVE IS LARGER FOR RK4 SOLUTION THAN THAT OF THE FORWARD DIFFERENCE SOLUTION.

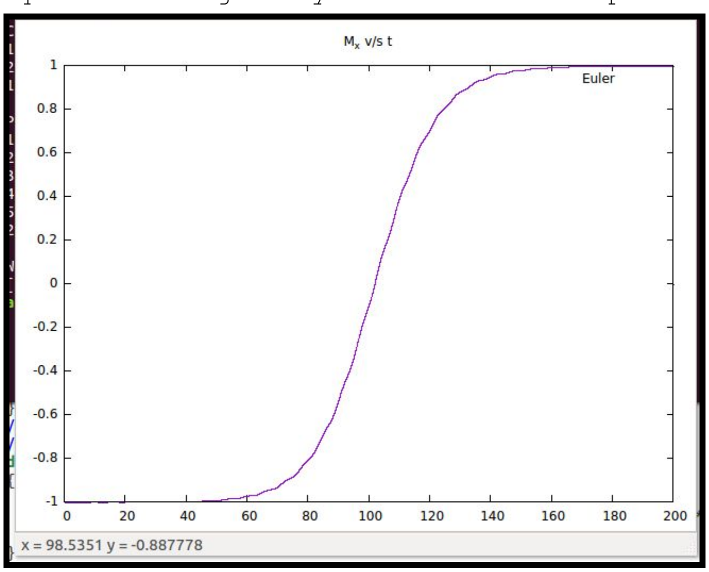
Variation in Step-Size (h):

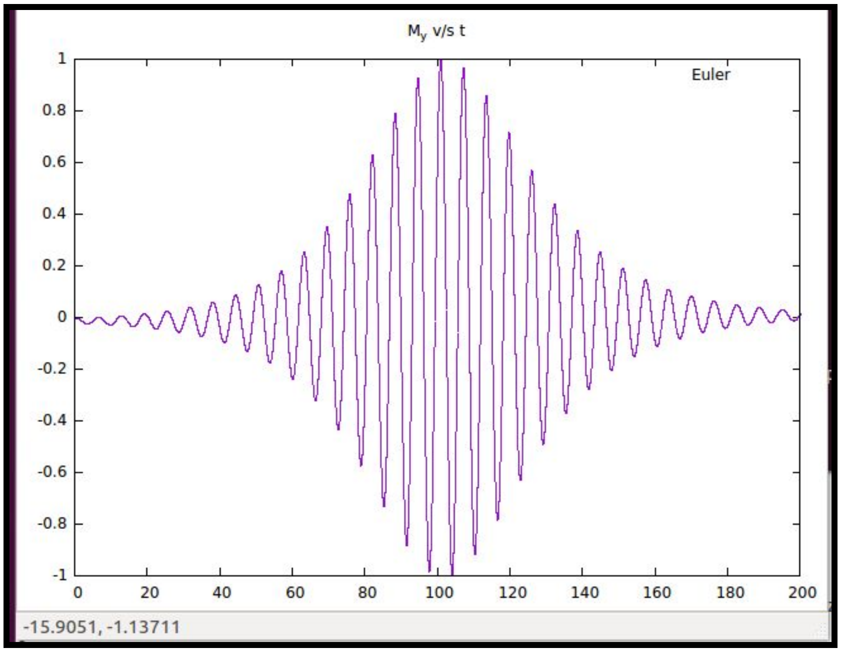
A few plots obtained for various 'h' by Euler's method

1. h = 0.005

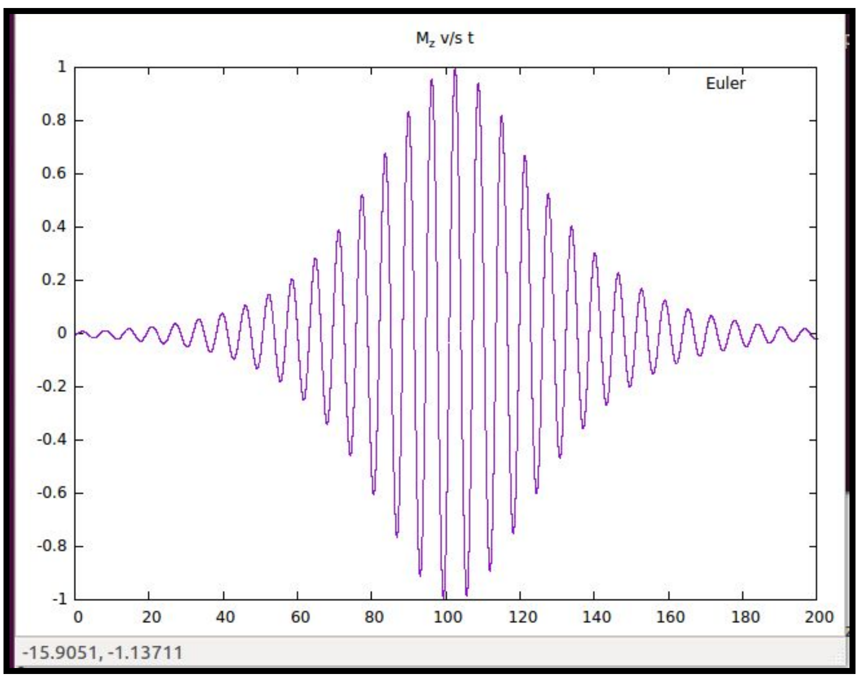


*Note that the trajectory here seems distorted and "looks" like an ellipsoid. This is so because the ratio of scales on gnuplot for x, y and z were not made equal. The trajectory here is still a sphere



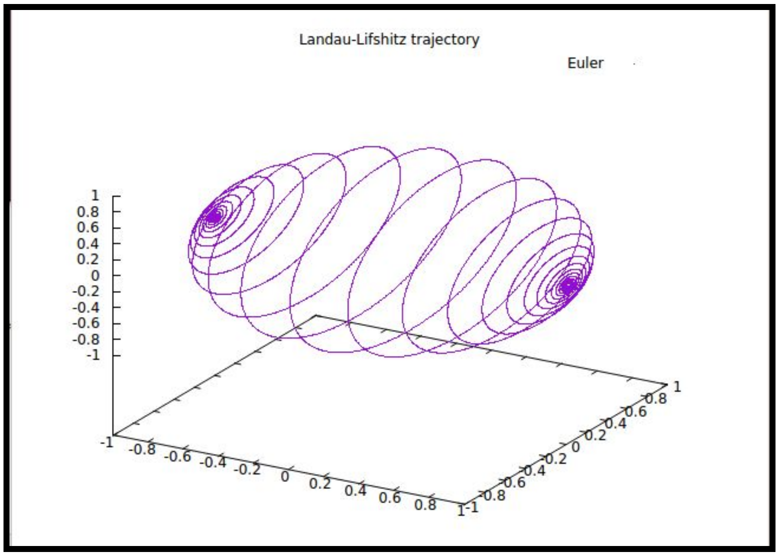


M_y v/s t
Euler
-15.9051, -1.13711

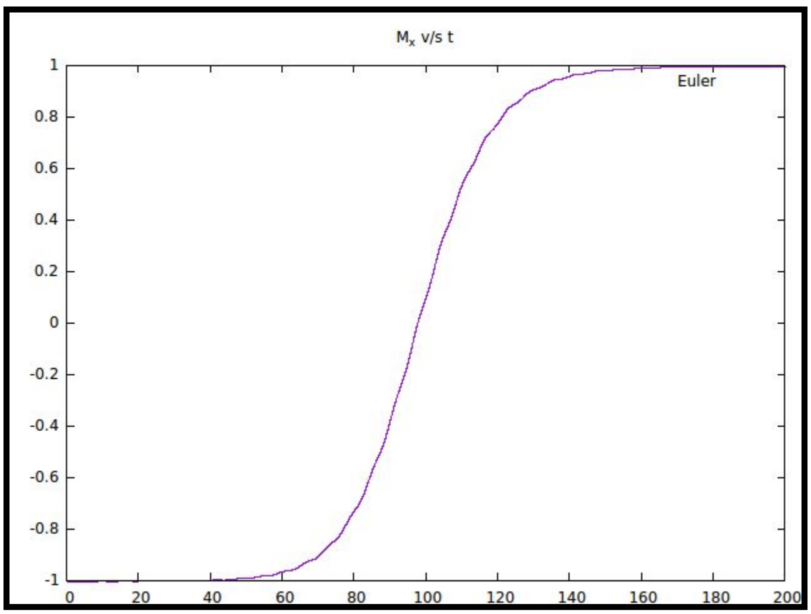


M_z v/s t
Euler
-15.9051, -1.13711

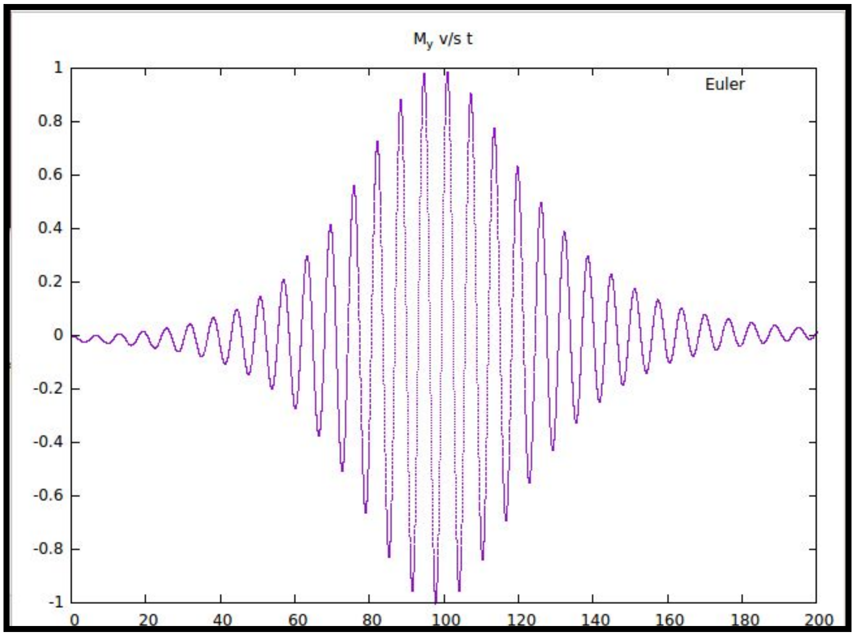
2. h = 0.01



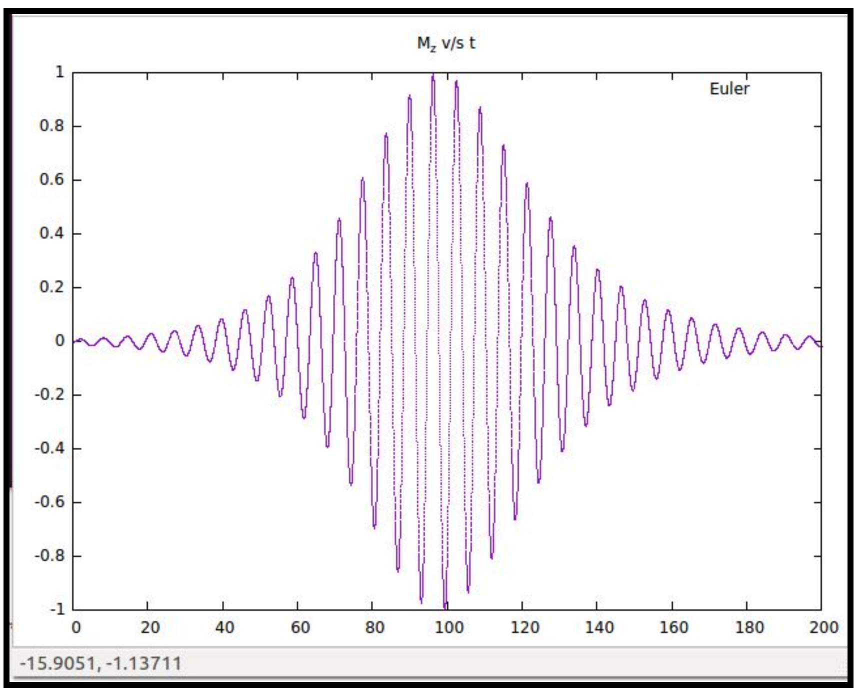
Landau-Lifshitz trajectory
Euler
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1



M_x v/s t
Euler
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
20
40
60
80
100
120
140
160
180
200

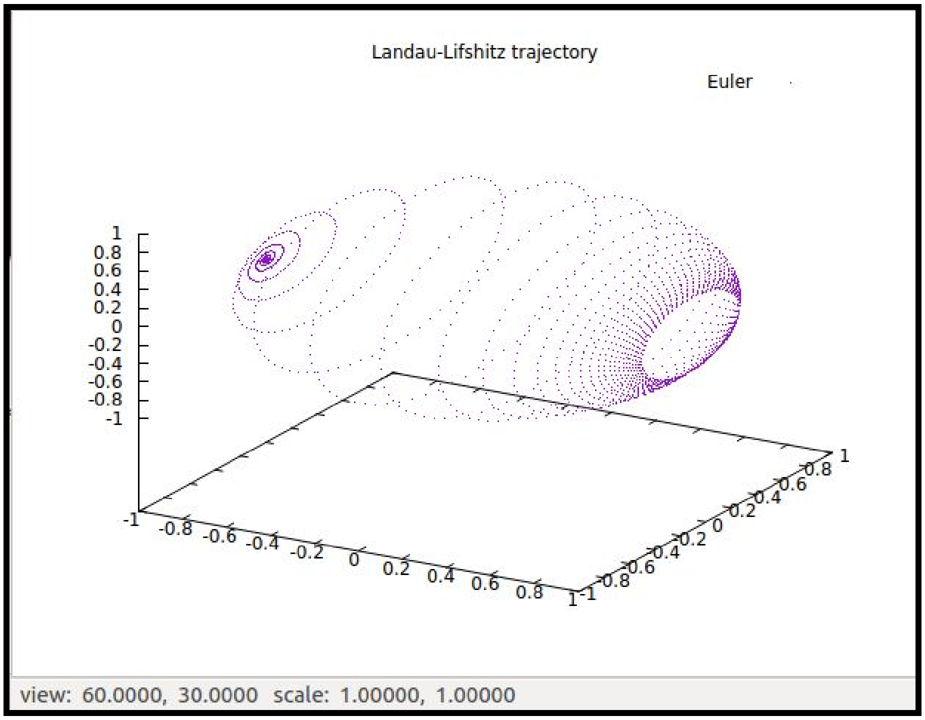


M_y v/s t
Euler

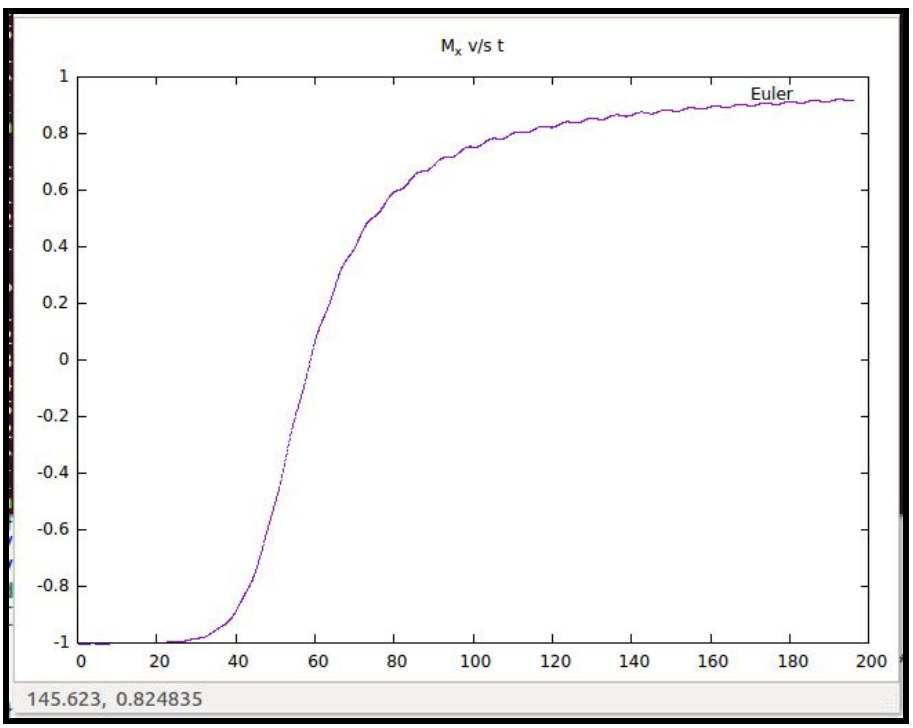


M_z v/s t
Euler
-15.9051, -1.13711

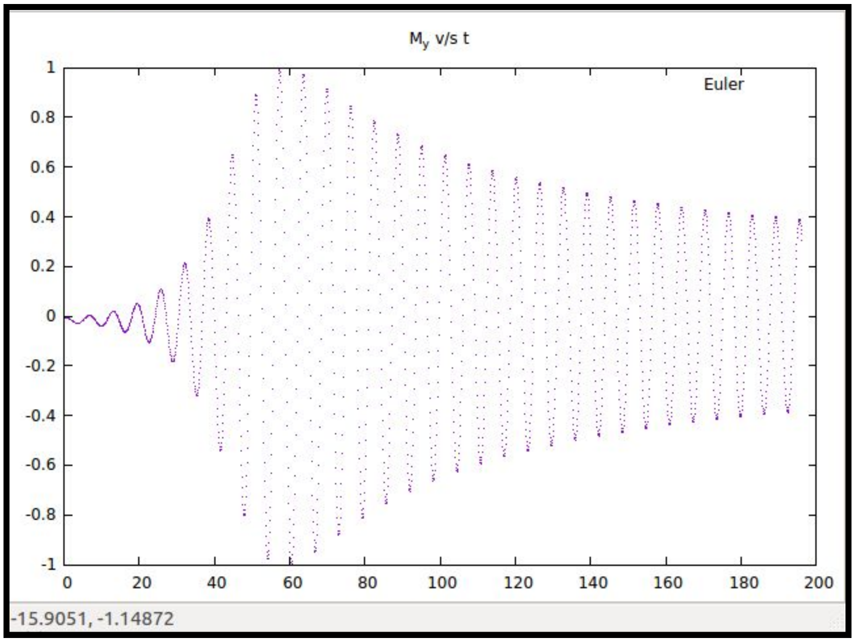
3. h = 0.1



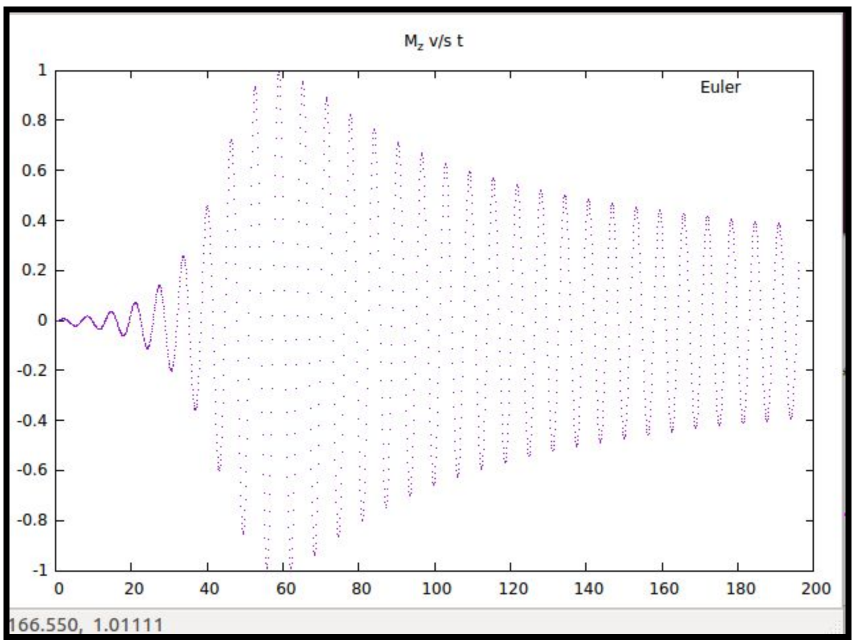
Landau-Lifshitz trajectory
Euler
view: 60.0000, 30.0000 scale: 1.00000, 1.00000



M_x v/s t
Euler
145.623, 0.824835



M_y v/s t
Euler
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
20
40
60
80
100
120
140
160
180
200
-15.9051, -1.14872



M_z v/s t
Euler
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
20
40
60
80
100
120
140
160
180
200
166.550, 1.01111

INFERENCES FROM THE ABOVE PLOTS:

As is visible from the graphs above, the accuracy of Euler's method reduces as h increases, with significant errors for higher values of h such as 0.1.

The following conclusions may be drawn:

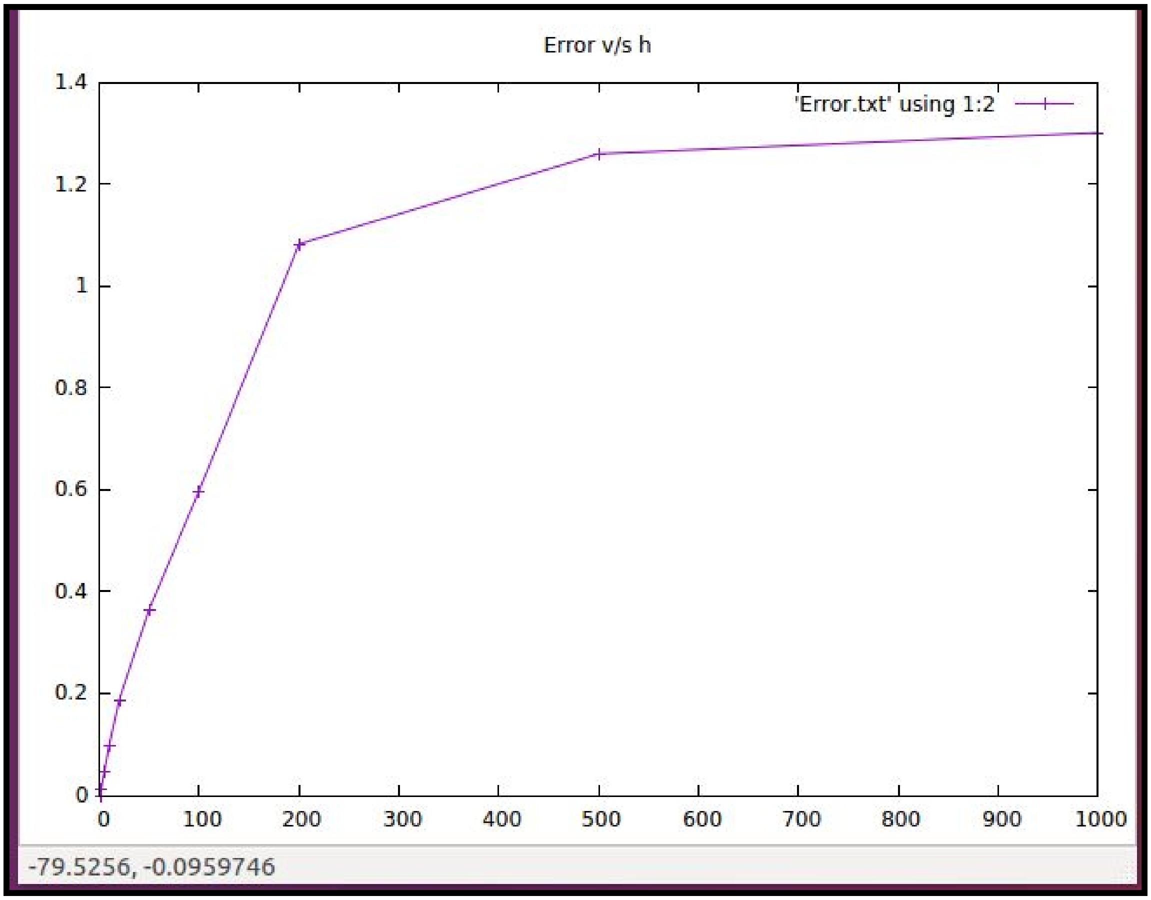
a. As h increases, the points tend to become stagnant near the end causing the trajectory to look like a circle there(no significant change in x, y, or z).
b. For h = 0.1 as t tends to 200s, it is observed that x stagnates at a value near 0.9. Y and Z vary sinusoidally without much change in their amplitudes.
c. It can hence be claimed that for Euler's method, the maximum affordable step-size that gives "good" results is 0.01.

Plotting Error v/s 'h' for Euler and RK4:

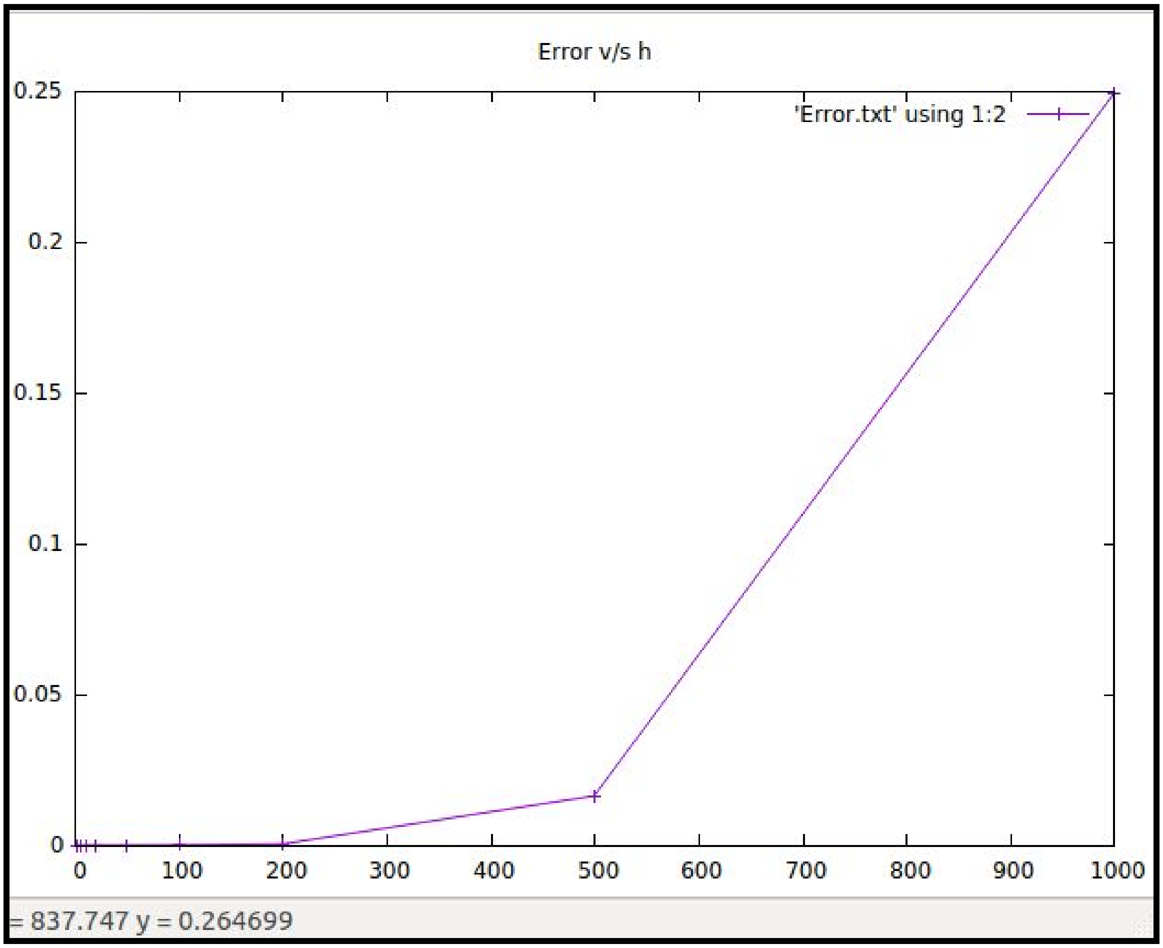
USING THE FUNCTIONS vary_h() AND error_find(), AND CONSIDERING THE DATA FOR h = 0.001 AS ACCURATE, TO WE FOUND THE ERRORS FOR DATA CORRESPONDING TO HIGHER h. THE ERROR v/s h HAS BEEN PLOTTED FOR BOTH EULER'S METHOD AND RK4 METHODS.

AS WAS EXPECTED, THE ERROR FOR HIGHER h WAS HIGHER IN THE EULER METHOD WHEN COMPARED TO RK4. WE USED 10 DIFFERENT VALUES OF h STARTING FROM h = 1 ms TO h = 1 s.

SINCE THE RANGE OF VALUES TAKEN WERE LOW, WE CANNOT DRAW STRONG INFERENCES ABOUT THE BEHAVIOUR OF ERROR WITH h. WE CAN, HOWEVER, MAKE ONE-TO-ONE COMPARISONS BETWEEN THE ERRORS FOR THE TWO METHODS AND DECIDE ON A MAXIMUM POSSIBLE TIME-STEP THAT MODELS THE SYSTEM WITHOUT ADDING SIGNIFICANT AMOUNTS OF ERROR. THE PLOT BELOW DOES NOT SHOW THE REAL FUNCTION OF ERROR VERSUS h AS gnuplot (IN LINESP MODE) CONNECTED THE DATA POINTS THAT WE OBTAINED BY STRAIGHT LINES.



ERROR v/s h FOR EULER'S METHOD



ERROR v/s h FOR RK4

FOR RK4, THE ERROR IS MUCH LESSER THAN 0.05 FOR h BELONGING TO (5, 200)ms. AT h = 500ms, WE OBSERVE A SIGNIFICANT RISE IN ERROR, ALBEIT VERY SMALL COMPARED TO THE CORRESPONDING ERROR IN EULER.

WHILE THE MAXIMUM AFFORDABLE TIME-STEP FOR EULER MAY BE AROUND 10ms FOR EULER'S METHOD, WE MAY HAVE A TIME-STEP AS LARGE AS 500ms(200 ms WOULD BE HIGHLY ACCURATE) TO GET DATA WITHIN THE SAME ERROR.

Adding Noise to the values of M_x, M_y and M_z:

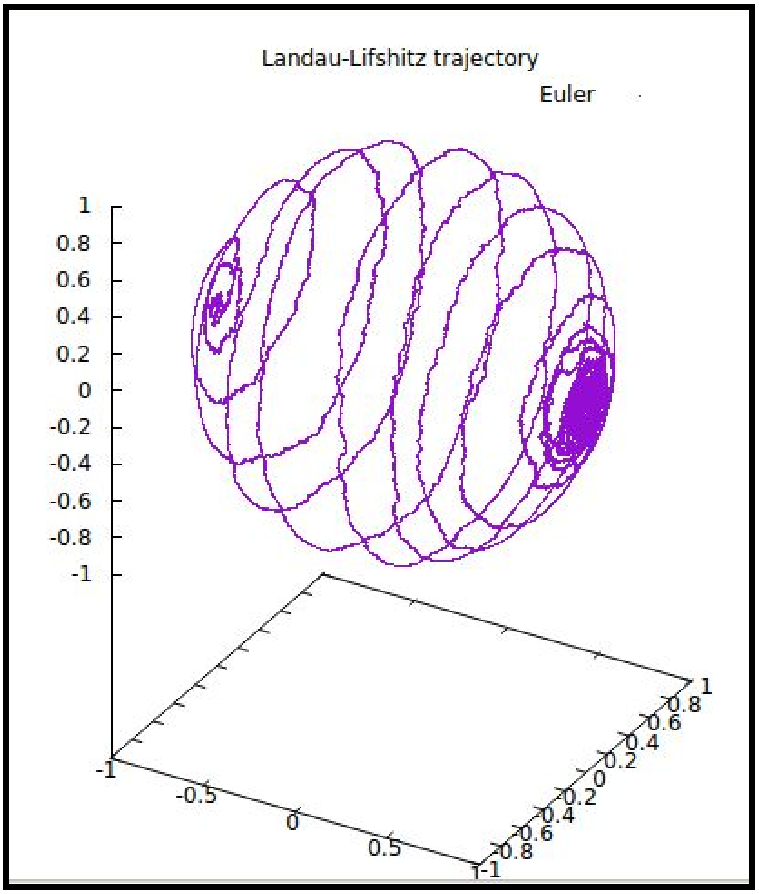
WE USED THE FUNCTION “Noise_gen()” TO GENERATE RANDOM NUMBERS FALLING IN A GAUSSIAN DISTRIBUTION. (THE FUNCTION WE DEVISED IS SIMILAR TO THE POPULAR BOX-MULLER METHOD)

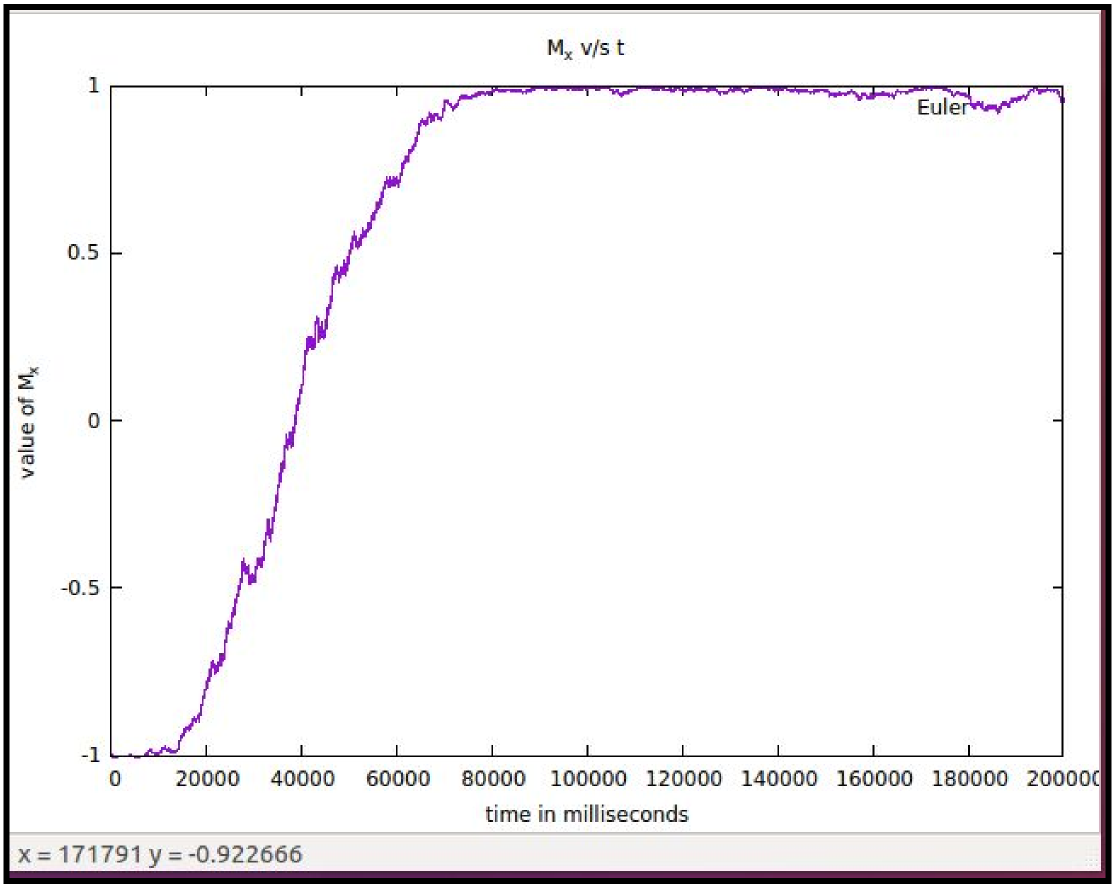
THE NOISE, THUS GENERATED WAS ADDED TO THE VALUES OF M_x, M_y AND M_z AND THE DIFFERENTIAL EQUATIONS WERE SOLVED USING THE UPDATED VALUES. A NEW SET OF NOISE VARIABLES WERE FURTHER ADDED TO THE SOLVED VALUES. THESE VALUES WERE TAKEN AS THE VALUES OF M_x, M_y AND M_z CORRESPONDING TO THE NEXT ITERATION:

```
 M_x(i+1) = Euler(M_x(i) + noise1_x) + noise2_x
 M_y(i+1) = Euler(M_y(i) + noise1_y) + noise2_y
 M_z(i+1) = Euler(M_z(i) + noise1_z) + noise2_z
```

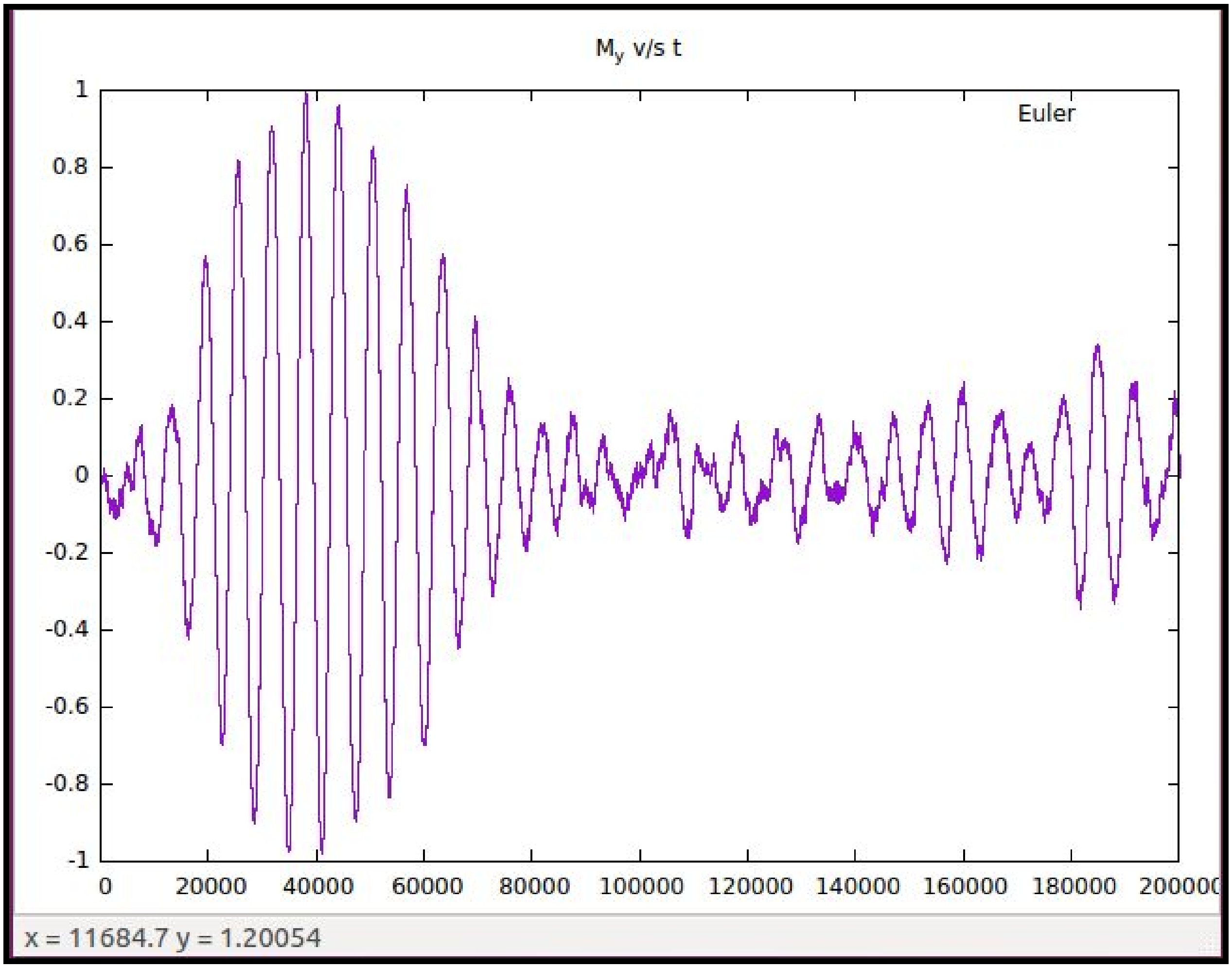
THE FOLLOWING PLOTS WERE OBTAINED AFTER ADDING NOISE:

Euler's Method

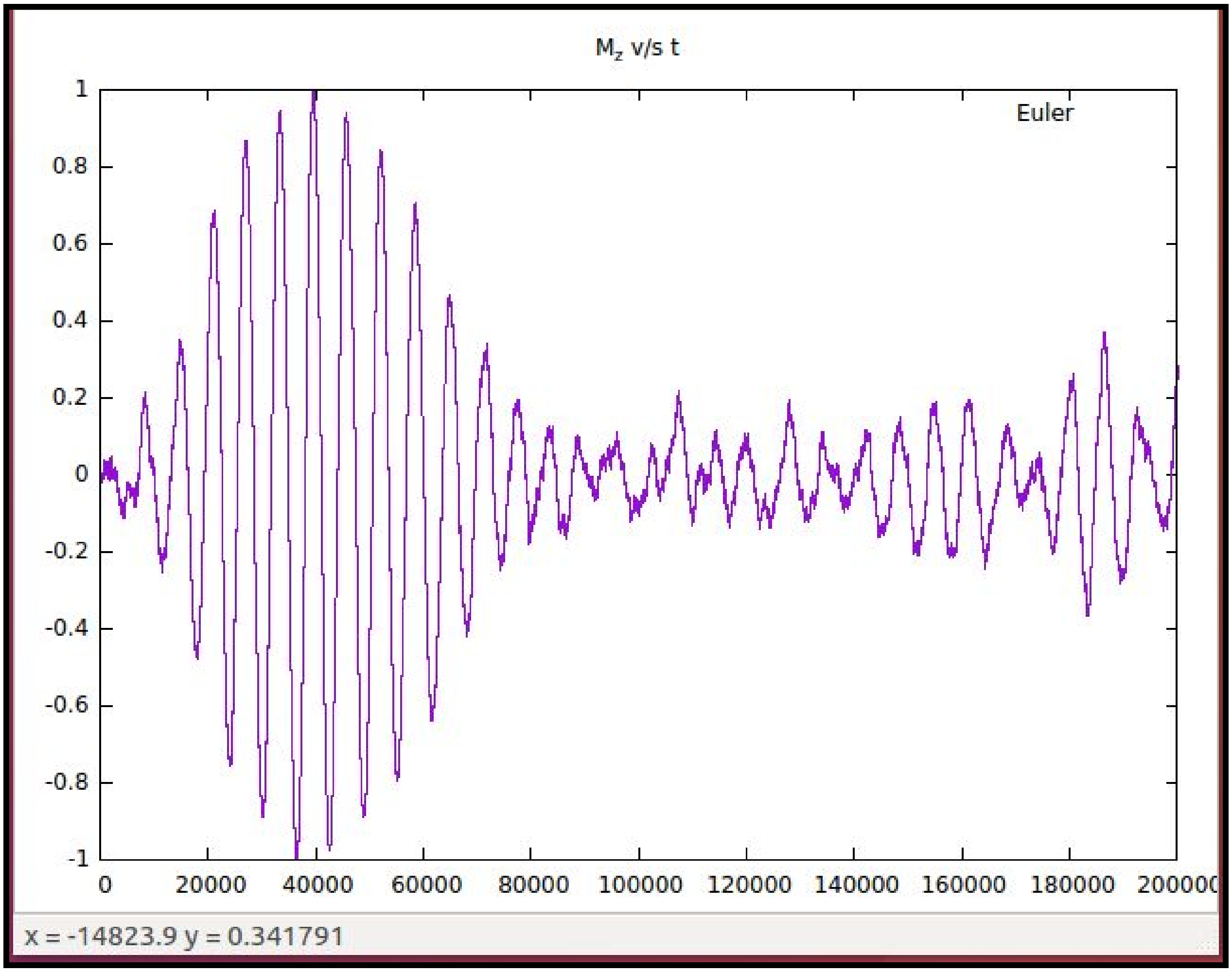




M_x v/s t
value of M_x
time in milliseconds
Euler
1
0.5
0
-0.5
-1
0
20000
40000
60000
80000
100000
120000
140000
160000
180000
200000
x = 171791 y = -0.922666

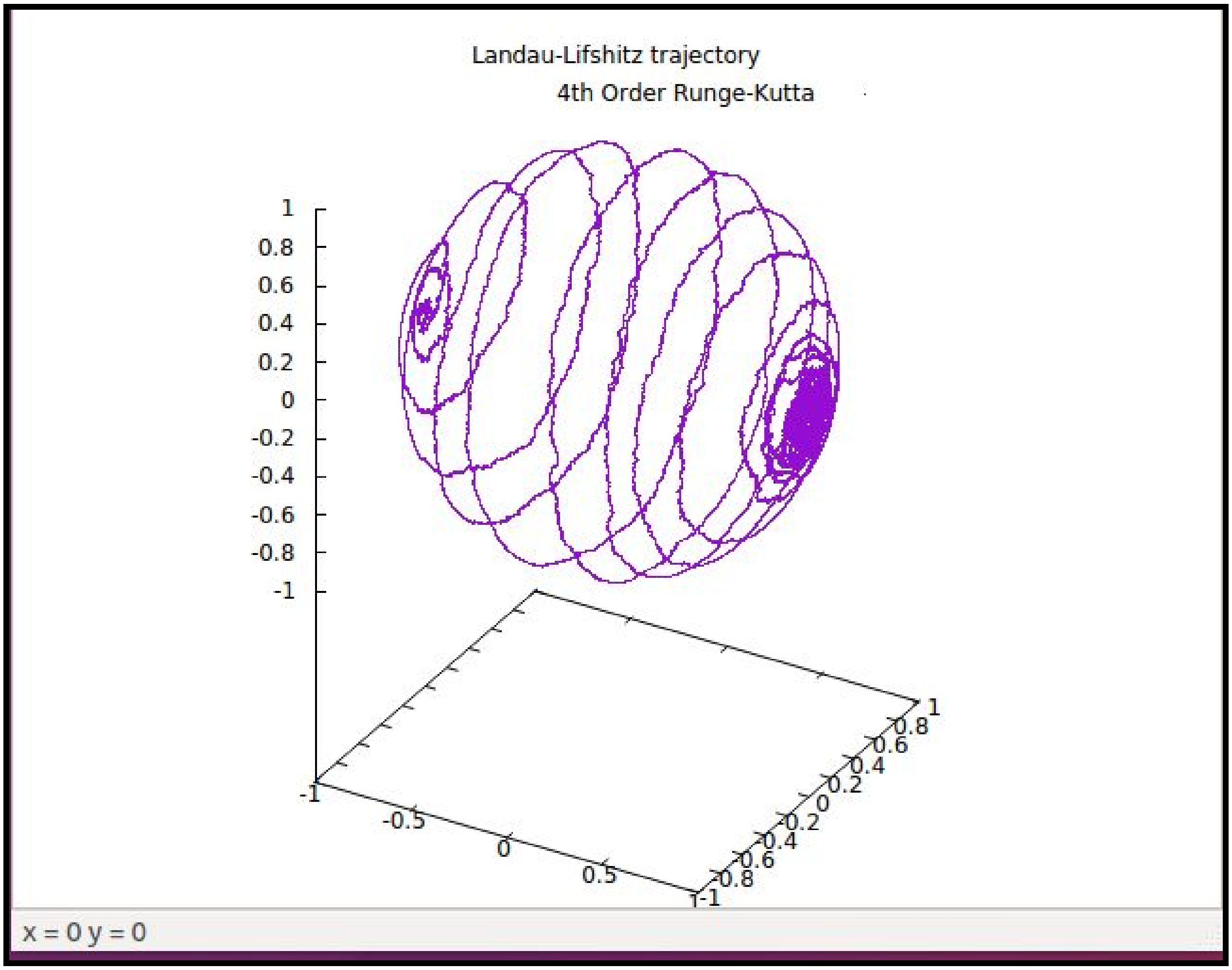


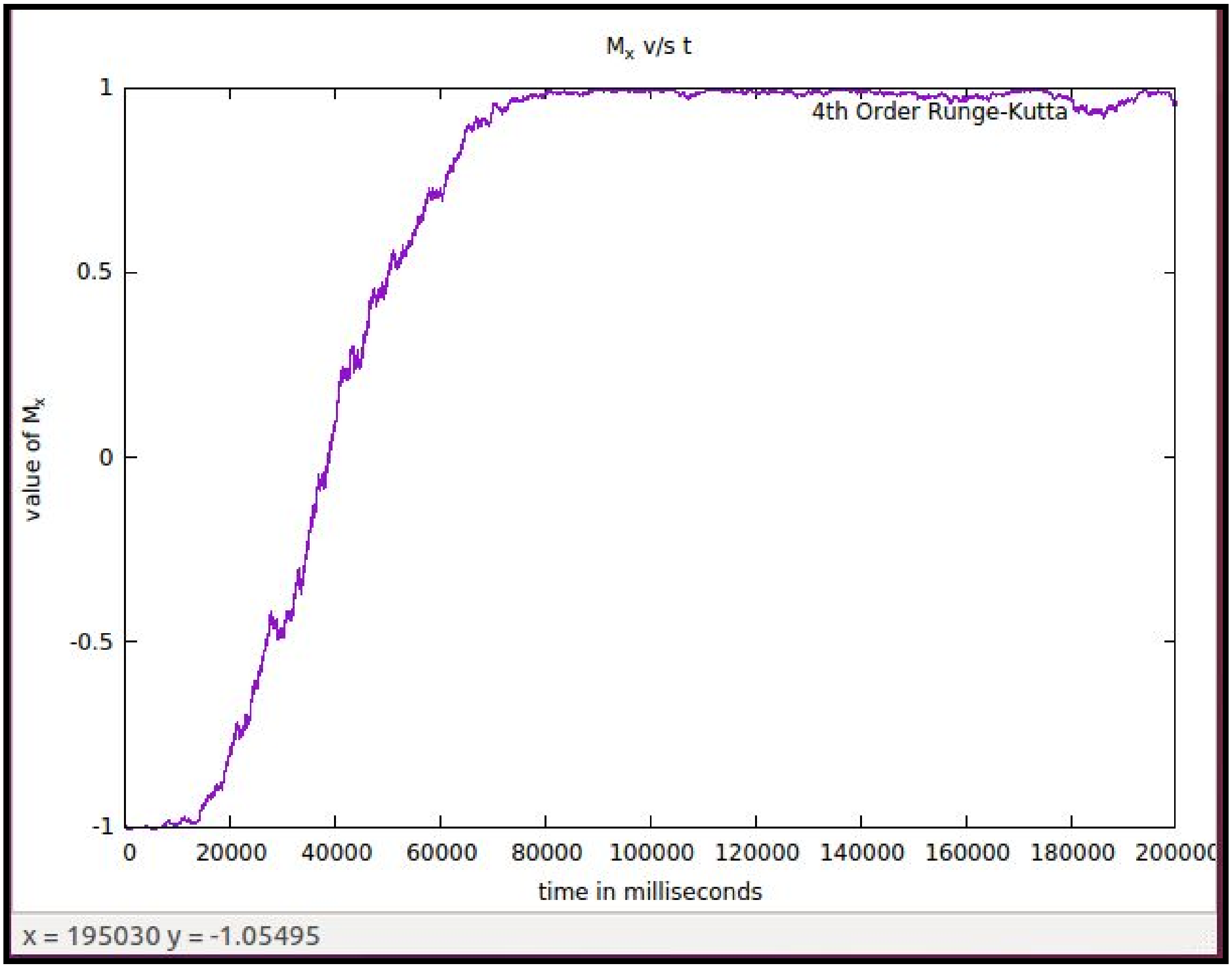
M_y v/s t
Euler
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
20000
40000
60000
80000
100000
120000
140000
160000
180000
200000
x = 11684.7 y = 1.20054



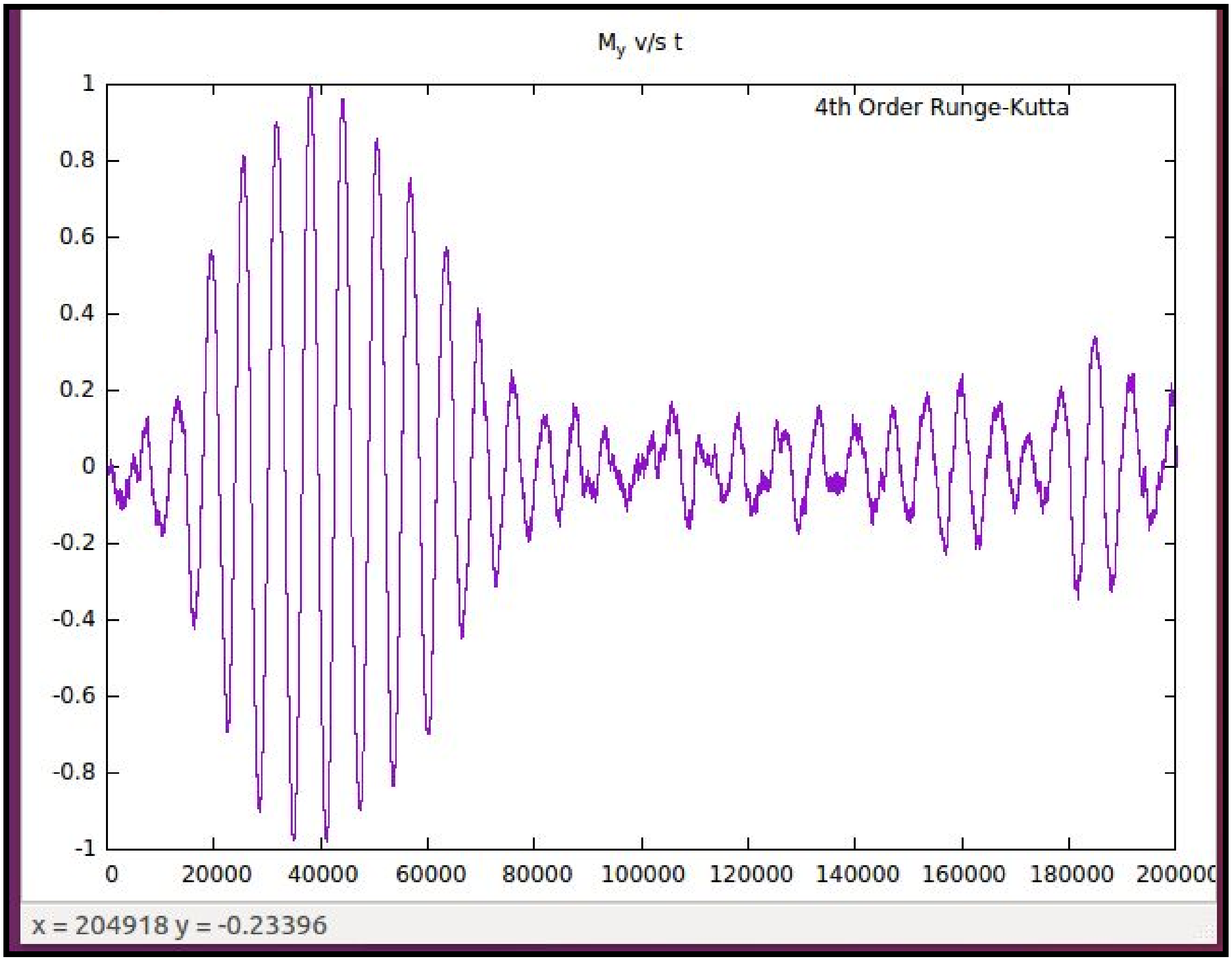
M_z v/s t
Euler
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
20000
40000
60000
80000
100000
120000
140000
160000
180000
200000
x = -14823.9 y = 0.341791

FOURTH-ORDER RUNGE-KUTTA METHOD

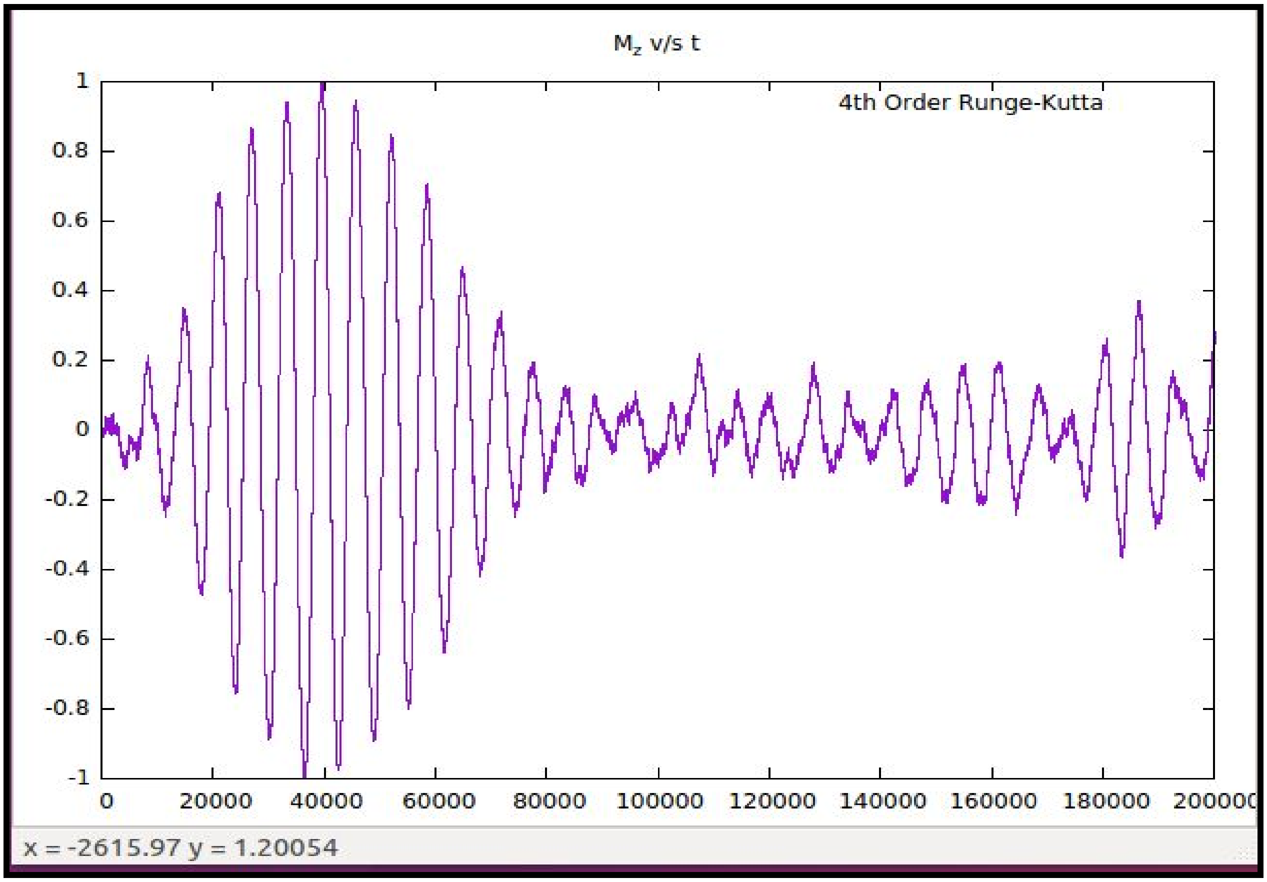




M_x v/s t
4th Order Runge-Kutta
value of M_x
time in milliseconds
1
0.5
0
-0.5
-1
0
20000
40000
60000
80000
100000
120000
140000
160000
180000
200000
x = 195030 y = -1.05495



M_y v/s t
4th Order Runge-Kutta
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
20000
40000
60000
80000
100000
120000
140000
160000
180000
200000
x = 204918 y = -0.23396



INFERENCES FROM THE ABOVE PLOTS:

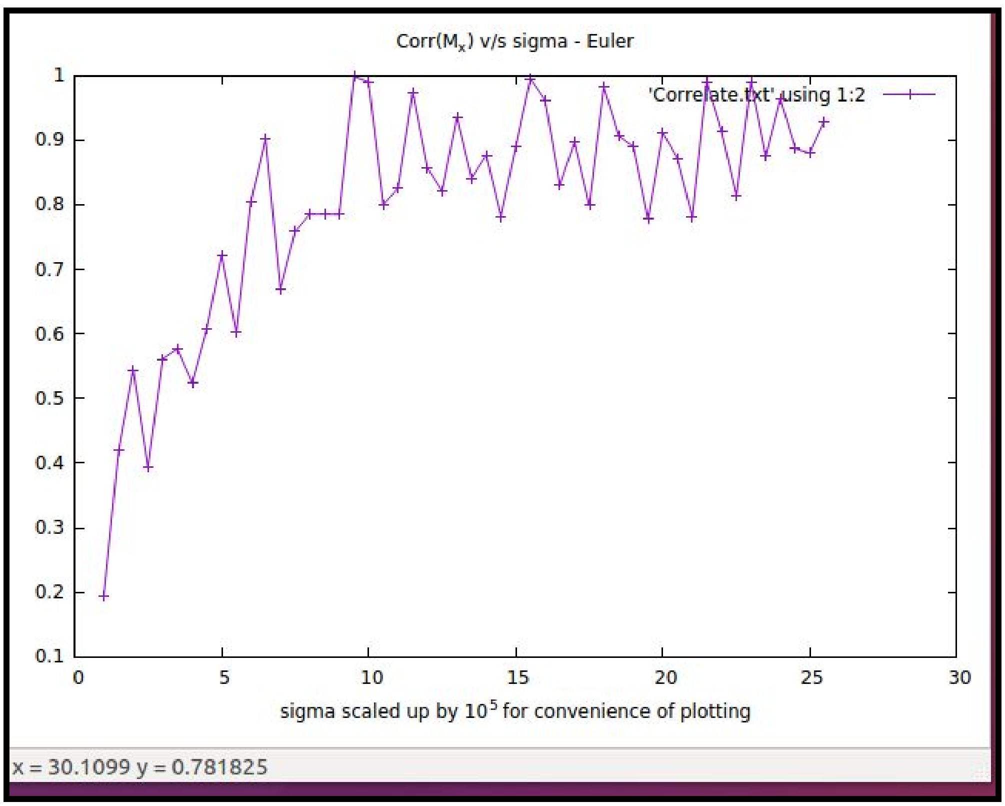
1. The noise added to the data changes the values of M_x, M_y and M_z more rapidly for smaller values of these variables.
2. It is hence observed that all these variables shoot up to the maximum much faster than in the noise-free system.
3. The values of M_y and M_z are observed to fall again after attaining high values. This may be explained by the reasoning that the slope of M_y and M_z wrt time become more negative as M_y and M_z increase. This can validated by looking at the differential equation.

Correlation between the noisy data and the noise-free data:

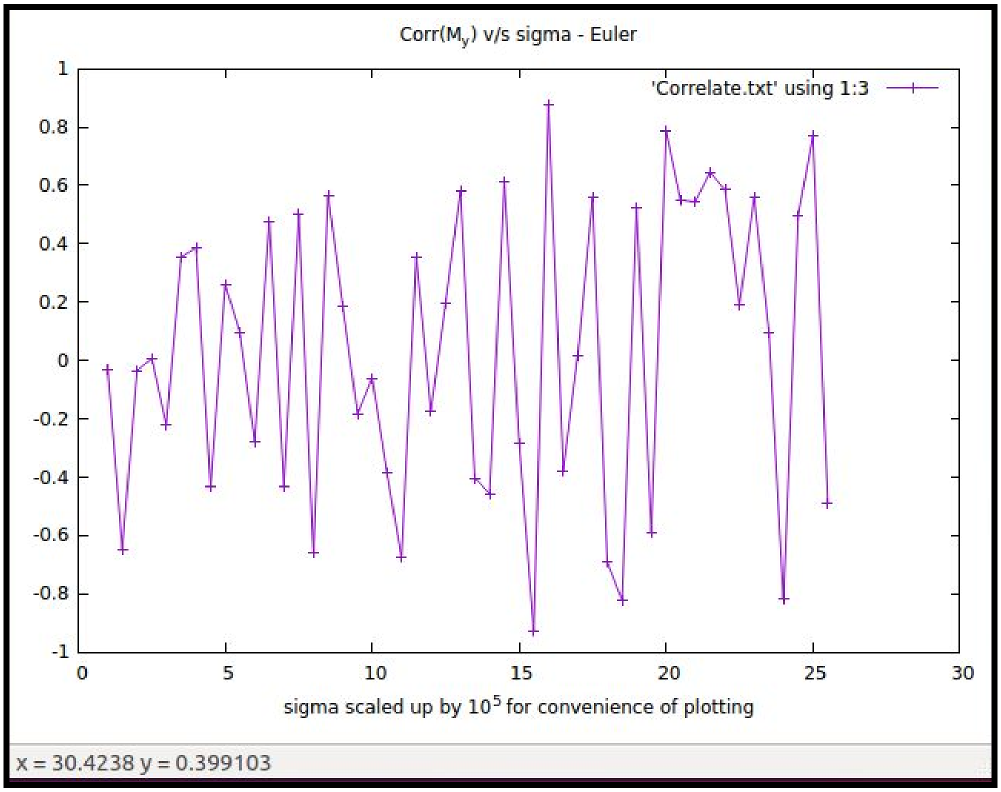
We computed the cross-correlation between the noisy data and the noise-free data. The obtained value was divided by the autocorrelation value for the noise-free data.

For both the methods, this ratio was observed to fall between -1 and +1. The correlation ratio has been plotted for different values of standard deviation.

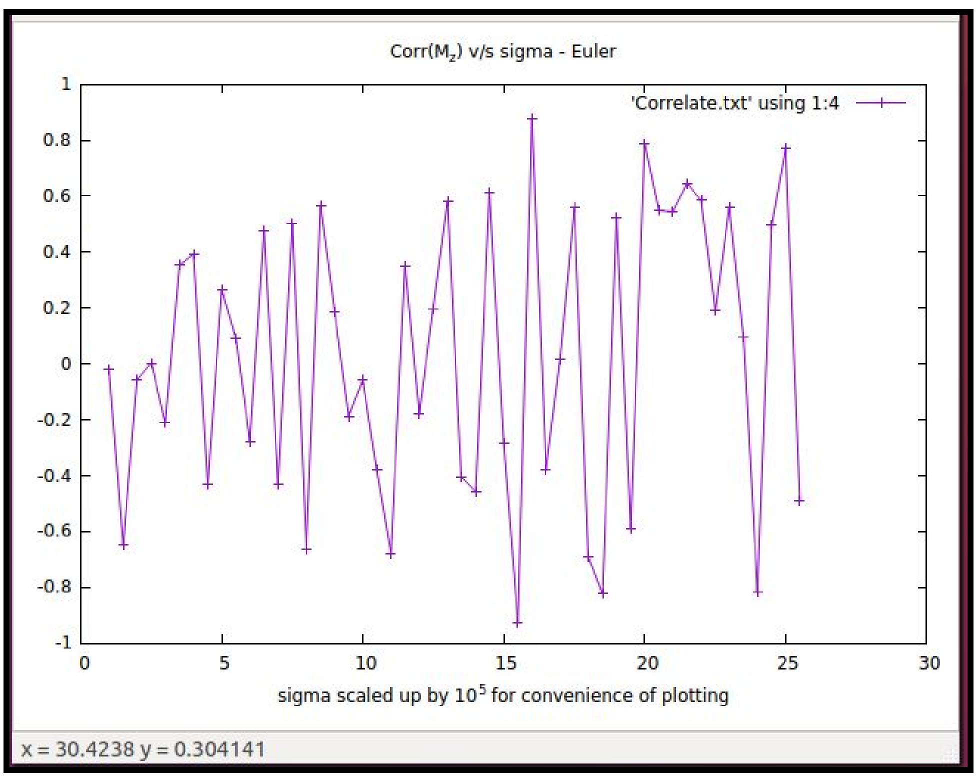
Correlation ratio plots for Euler's method:



Correlation ratio for M_x

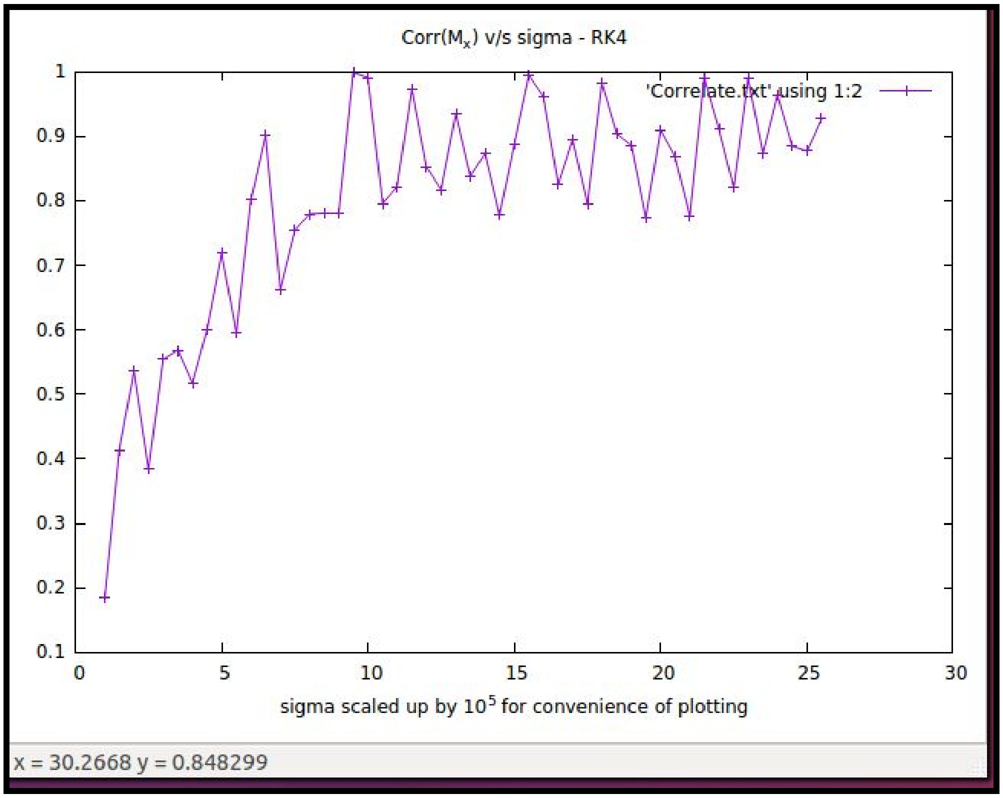


Correlation for M_y

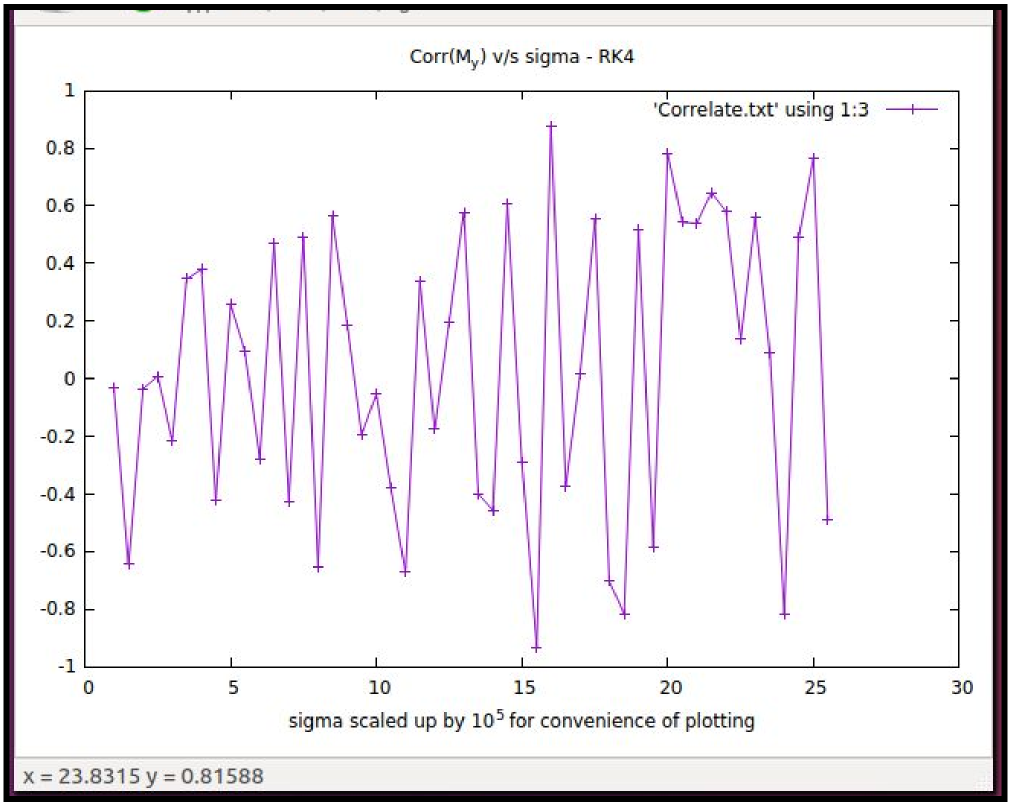


Correlation for M_z

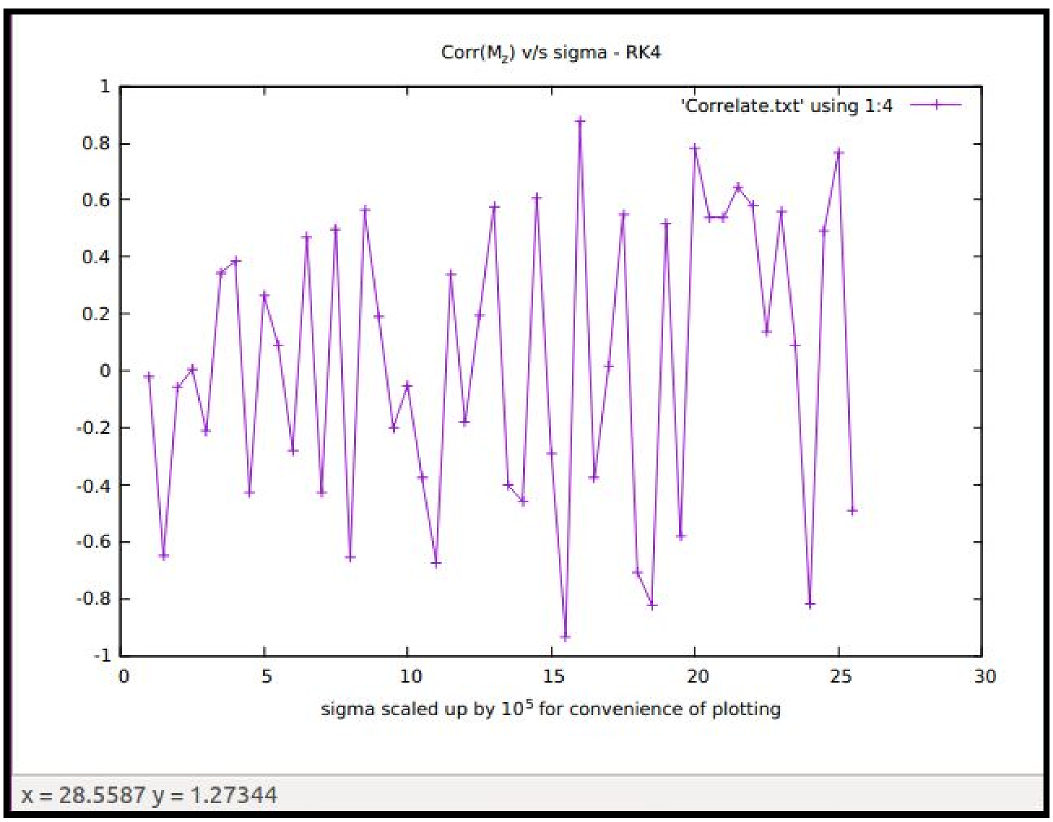
Correlation for 4th order Runge-Kutta



Correlation for M_x



Correlation for M_y



Correlation for M_z