A Comparative Error Analysis of Different Numerical Methods for Solving Ordinary Differential Equations

Harshank Nimonkar (219110) Simran Ghag (219125) Tarun Jhangyani (219112)

St. Xavier's College, Mumbai

"We are presenting the theory and Python codes for solving Ordinary Differential Equations numerically using different Numerical Methods such as ***'Euler's Method', 'Heun's Method', 'Midpoint Method'*** *and* ***'4th Order Runge-Kutta Method'*** *with the help of different libraries from Python. Adding on it, we did comparative* ***Error Analysis*** *of these numerical methods to study their accuracy and by increasing/decreasing the step size we analysed the methods to see which method gives most accurate result. Finally, we applied all these methods to a physical system which is,* ***'A body falling under the influence of gravity acted upon by drag force****'. We also recorded a video representing the physical system and analysed it using* ***TRACKER****, a video analysis software. Finally, we conclude that the 4th Order Runge-Kutta Method provides the best results.*

Keywords: Euler's method, Heun's method, Midpoint method, 4th Order Runge Kutta, Error analysis, drag force

1. INTRODUCTION

Differential equations, either ordinary derivatives or partial derivatives, are equations which contain derivatives. An ordinary differential equation together with initial condition is called an initial value problem (IVP) which specifies the value of the unknown function at a given point in the domain. There are a lot of physical problems in Science and Engineering which exist in the form of differential equations.

To determine the solution of differential equations, there are different analytical methods available. In certain cases, however, analytical methods are not capable of solving some complicated or complex differential equations. Numerical methods are proved to be useful in getting the solution to complicated differential equations. With the help of computer programming, numerical methods are very valuable tools for solving complex problems.

Numerous numerical methods for solving ordinary differential equations with initial value problems have been developed. Many authors have attempted to solve initial value problems to obtain high accuracy rapidly by using a numerous method, such as Euler's method and Heun's method, Midpoint method and Runge-Kutta methods. Euler's method uses the line tangent to the function at the beginning of the interval as an estimate of the slope of the function over the interval. However, Heun's method considers the tangent lines to the solution curve at both ends of the interval. And some of them have attempted to enhance these precision methods, where others have improved these methods for better accuracy, stability, and consistency.

2. NUMERICAL METHODS

a. Euler's Method:

Euler's method is based on approximating the graph of a solution *y(x)* with a sequence of tangent line approximations computed sequentially, in "steps". Our task then, is to derive a useful formula for the tangent line approximation in each step.

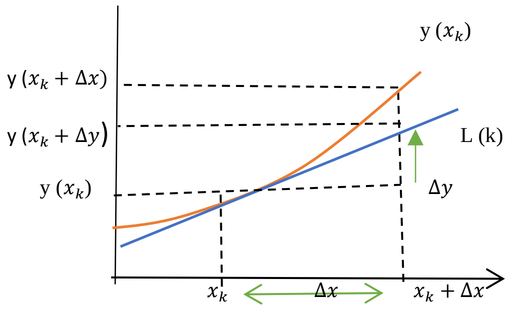


Fig 1.a

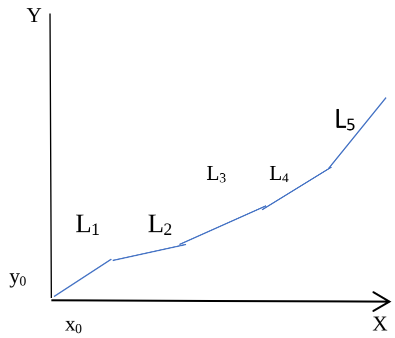


Fig 1.b

Let *y* = *y(x)* be the desired solution to some first-order differential equation

$$\frac{dy}{dx} = f(x, y)$$

and let x_k be some value for *x* on the interval of interest. As illustrated in figure 1.a, $(x_k, y(x_k))$ is a point on the graph of *y* = *y(x)*, and the nearby points on this graph can be approximated by corresponding points on the straight-line tangent at point $(x_k, y(x_k))$ (line L_k in figure 1.b).

As with the slope lines, the differential equation can give us the slope of this line

The slope of the approximating line $= \frac{dy}{dx}$ at $(x_k, y(x_k))$ = f$(x_k, y(x_k))$

Now let Δx be any positive distance in the X direction. Using our tangent line approximation (again, see figure 1.a), we have that $y(x_k + \Delta x) \approx y(x_k) + \Delta y$

where $\frac{\Delta y}{\Delta x}$ = slope of the approximating line = f $((x_k, y(x_k))$.

So, $$\Delta y = \Delta x \cdot f(x_k, y(x_k))$$

And $$y(x_k + \Delta x) \approx y(x_k) + \Delta x \cdot f(x_k, y(x_k)) \text{ .... (1)}$$

Approximation equation (1) is the fundamental approximation underlying each basic step of Euler's method. However, in what follows, the value of $y(x_k)$ will usually only be known by some approximation y_k . With this approximation, we have

$$y(x_k) + \Delta x \cdot f(x_k, y(x_k)) \approx y_k + \Delta x \cdot f(x_k, y_k)),$$

which, combined with approximation (1), yields the approximation that will actually be used in Euler's method,

$$y(x_k + \Delta x) \approx y_k + \Delta x \cdot f(x_k, y_k)) \text{ .... (2)}$$

The distance Δx in the above approximations is called the step size. We also see that choosing a good value for the step size is important.

The Steps in Euler's Method:

1. Get the differential equation into derivative formula form

$$\frac{dy}{dx} = f(x, y)$$

2. Set x_0 and y_0 equal to the x and y values of the initial data.
3. Pick a distance Δx for the step size, a positive integer N for the maximum number of steps, and a maximum value desired for x, x_{max}. These quantities should be chosen so that

$$x_{max} = x_0 + N\Delta x$$

4. Write out the equations,

$$x_{k+1} = x_k + \Delta x$$

and

$$y_{k+1} = y_k + \Delta x \cdot f(x_k, y_k)$$

using the information from the previous steps. Because of this, y_k each generated by Euler's method is an approximation of $y(x_k)$.

b. Heun's method:

Let us consider the first-order differential equation with initial value problem,

$$\frac{dy}{dx} = f(x, y) \text{where } y(x_0) = y_0 \quad \text{......... (1)}$$

The Improved Euler's method, also known as the Heun formula or the average slope method, gives a more accurate approximation than the Euler rule and gives an explicit formula for computing y_{n+1}. The basic idea is to correct some error of the original Euler's method. The syntax of the Improved Euler's method is similar to that of the trapezoid rule, but the y value of the function in terms of y_{n+1} consists of the sum of the y value and the product of h and the function in terms of x_n and y_n

Improved Euler formula or the average slope method is commonly referred as Heun's Method

$$y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, y_n + hf(x_n, y_n))], \qquad n = 0,1,2,3 \ldots.$$

Since it is actually the simplest version of predictor-corrector method, the recurrence can be written as,

$$p_{n+1} = y_n + hf(x_n, y_n),$$

$$y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, p_{n+1})], \quad n = 0,1,2,3\ldots$$

c. Midpoint Method

The midpoint method is a refinement of the Euler's method, this is also known as second order Runge- Kutta method. This method improves the Euler method by adding a midpoint in the step which increases the accuracy by one order

$$y_{n+1} = y_n + hf\,(x_n, y_n) \;\ldots.\; (1)$$

The major to derive the Euler's method is the approximate equality which is obtained from slope formula,

$$y'(x) \approx \frac{y(x+h)-y(x)}{h} \;\ldots.\; (2)$$

also considering that y'=f (x, y).

for midpoint method we replace eqn. 1 with the more accurate.

$$y'\left(x+\frac{h}{2}\right) \approx \frac{y(x+h)-y(x)}{h} \quad \ldots.\; (3)$$

so instead of (1) we find,

$$y(x+h) \approx y(x)+hf\left(x+\frac{h}{2}, y\left(x+\frac{h}{2}\right)\right) \ldots\ldots\; (4)$$

Thus, we have,

$$y\left(x+\frac{h}{2}\right) \approx y(x)+\frac{h}{2}y'(x) = y(x)+\frac{h}{2}f\,(x, y(x))$$

So, when we plugin eqn (4) we get, the explicit midpoint method,

$$y(x+h) \approx y(x)+hf\left(x+\frac{h}{2}, y(x)+\frac{h}{2}f(x, y(x))\right)$$

for implicit midpoint method, it obtained by approximating the value at the half step $x+\frac{h}{2}$ by the midpoint of the line segment from y(x) to y(x+h).

$$y\left(x+\frac{h}{2}\right) \approx \frac{1}{2}(y(x) + y(x+h))$$

And thus,

$$\frac{y(x+h)-y(x)}{h} \approx y'\left(x+\frac{h}{2}\right) \approx k=f\left(x+\frac{h}{2}, \frac{1}{2}(y(x) + y(x+h))\right)$$

Inserting the approximation y_n+ hk for $y(x_{n+h})$ results in Implicit Runge-Kutta method

$$k=f\left(xn+\frac{h}{2}, yn+\frac{h}{2}k\right)$$

d. 4th Order Runge Kutta Method-

The most widely known method in numerical analysis from Runge–Kutta family is generally referred to as "RK4", or “Runge Kutta Method".

Let us consider the first order differential equations with initial value,

$$\frac{dy}{dx} = f(x, y) \text{where } y(x_0) = y_0$$

Here y is an un1nown function of x

At the initial value of x_0 the corresponding y value is $y_{0,}$ the function f and the initial conditions $x_{0,}y_0$, are given.

Let us pick a step-size *h* > 0 and define,

$$Y_{n+1=}y_n+\frac{1}{6}h(k_1 + 2k_2 + 2k_3 + k_4)$$

$$X_{n+1=} x_n+h$$

Thus, for k=0,1,2,3….

We have,

$$k_1=f(x_n, y_n)$$

$$k_2=f(x_n+\frac{h}{2}, y_n+h\frac{k_1}{2})$$

$$k_3=f(x_n+\frac{h}{2}, y_n+h\frac{k_2}{2})$$

$$k_4=f(x_n+h, y_n+hk_3)$$

Here y_{n+1} is the RK4 approximation of $y(x_{n+1})$.

4. PHYSICAL SYSTEM

Let us consider a case of a body falling under the influence of gravity and also acted upon by drag force,

$$f = kv$$

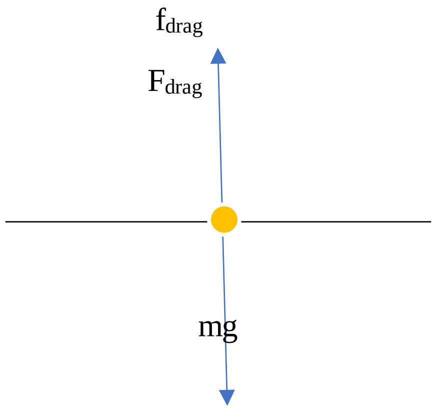
Where drag force k is constant and k depends upon the medium in which the body is travelling and also shape, volume of body. Here we are considering a special case of drag force where speeds are less and drag force is proportional to velocity, when speed increases the drag force is proportional to v^2. This drag force is applicable when speeds are smaller.

At start, initially v=0, so f_{drag}=0

Since total force is mg,

ma = mg

a=g



Later at time ‘t’, v$\neq$ 0, thus $f_{drag} \neq$ 0,

mg- f_{drag}= Total force

mg-kv= ma

this gives, m$\frac{dv}{dt} = mg - kv$= g-$\frac{k}{m}v$ …… (1)

Thus, as v increases f_{drag} also increases until f_{drag}=mg at this point v=v_t

Because Net force, f_{drag}-mg=0, this gives us v=0 and v=constant=v_t , Where v_t , is the terminal velocity.

Therefore, k v_t =mg

$$\frac{mg}{k} = v_t$$

$$\boxed{\frac{k}{m} = \frac{g}{v_t}}$$

From (1) we get,

$$\boxed{\frac{dv}{dt} = g - \frac{g}{v_t}(v) = g\left(1 - \frac{v}{v_t}\right)}$$

Integrating we get,

$$\int_{v_0}^{v} \frac{v_t dv}{(v_t - v)} = \int_{0}^{t} g dt$$

$$v_t \int_{v_0}^{v} \frac{dv}{(v_t - v)} = gt + C$$

Solving Further,

$$-v_t \ln|v_t - v| = gt + C \qquad \ldots\ldots (2)$$

At t=0, we have $v=v_0$

Thus, $-v_t \ln|v_t - v_0| = C$

Therefore, from equation (2) we get,

$$-v_t \ln|v_t - v| = gt - v_t \ln|v_t - v_0|$$

Thus, $$-gt = v_t\, ln|v_t - v| - v_t\, ln|v_t - v_0|$$

$$-gt = v_t\, ln\left|\frac{v_t - v}{v_t - v_0}\right|$$

therefore $\boxed{e^{\frac{-gt}{v_t}} = \left|\frac{v_t - v}{v_t - v_0}\right|}$

Rearranging we get,

$$\boxed{v = v_t - (v_t - v_0)e^{-gt/v_t}}$$

Substituting $\frac{mg}{k} = v_t$ and finally we get,

$$\boxed{v = v_t - (v_t - v_0)e^{-(k/m)t}}$$

5. METHODOLOGY

1. We have considered two first order ordinary differential equations of the form,

$$\boldsymbol{\frac{dy}{dx} = f(x, y)}$$

   Here our first differential equation is

$$\boldsymbol{\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5}$$

   And the second differential equation is,

$$\boldsymbol{\frac{dy}{dx} = 4e^{0.8x} - 0.5y}$$

2. We have defined Python functions for Euler's Method, Heun's Method, Midpoint Method and 4th Order Runge-Kutta Method (RK-4).

3. Also, we solved and computed the analytical solution of the two ordinary differential equations.

4. Adding on, we plotted the analytically solved differential equation along with the solution obtained from different methods by varying the step size.

5. For each value of *y* obtained from different numerical methods, we have calculated the error by comparing it with analytically obtained '*y*' value by using the formula of percentage relative error.

$$\text{Error} = \left(\frac{y_{analytical} - y_{obtained\ from\ methods}}{y_{analytical}}\right) X\ 100$$

   And we plotted it against the common '*x*' values for the specified interval.

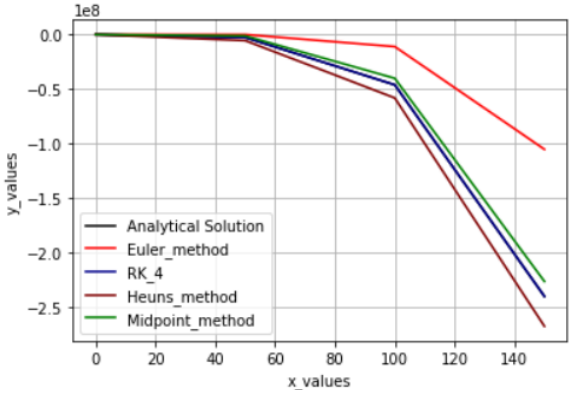
6. Also, we made an array of the '*error in y*' obtained from each method; averaged it and plotted it against specific step-size value. This is done for different step-size values.

1. $\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$

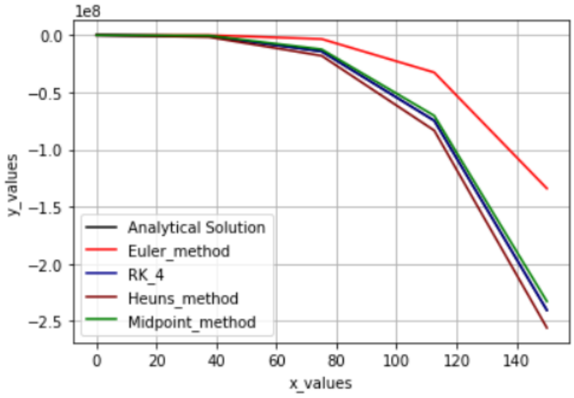
At $x_0 = 0$, $y_0 = 1$

Analytical Solution: $y = -\frac{1}{2}x^4 + 4x^3 - 10x^2 + 8.5x + 1$

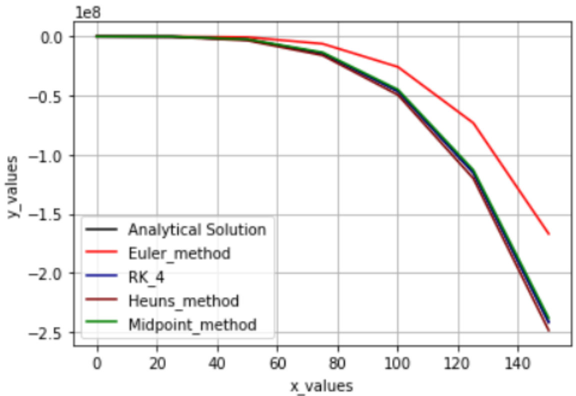
for h = 50-step size



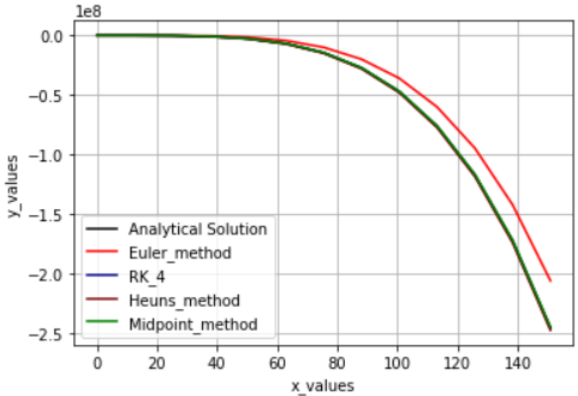
for h = 37.525-step size



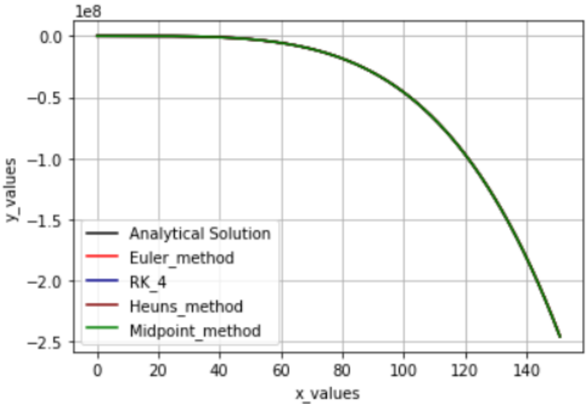
for h = 25.05-step size



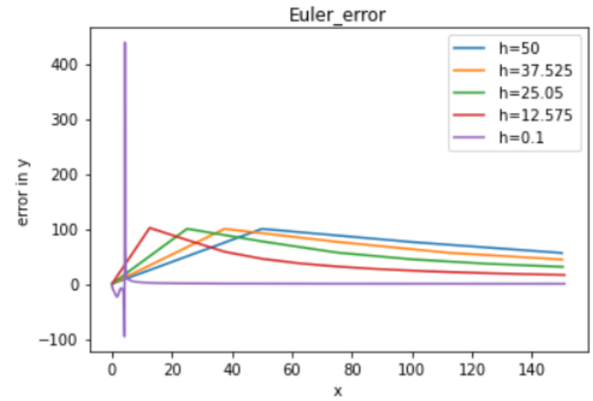
for h = 12.575-step size

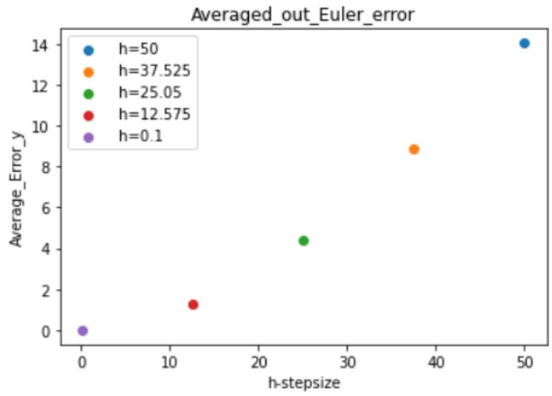


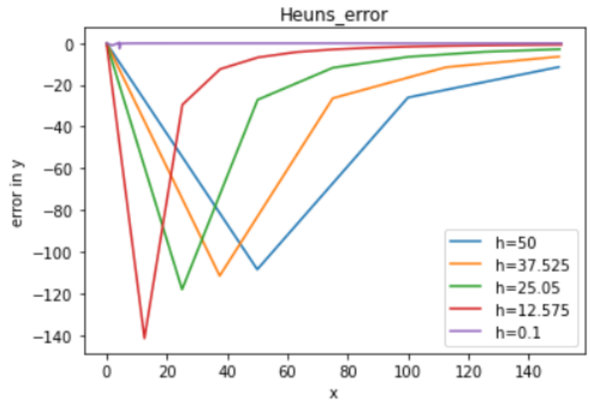
for h = 0.1-step size

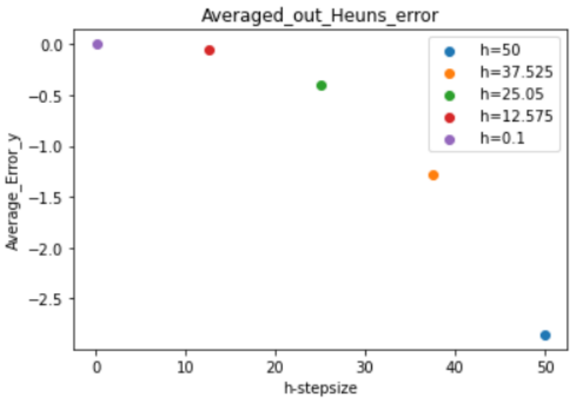


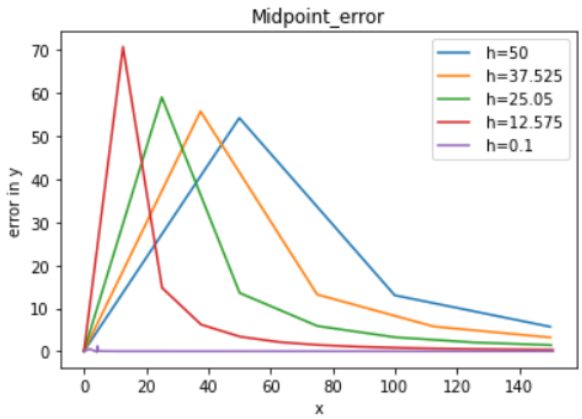
Error Analysis:

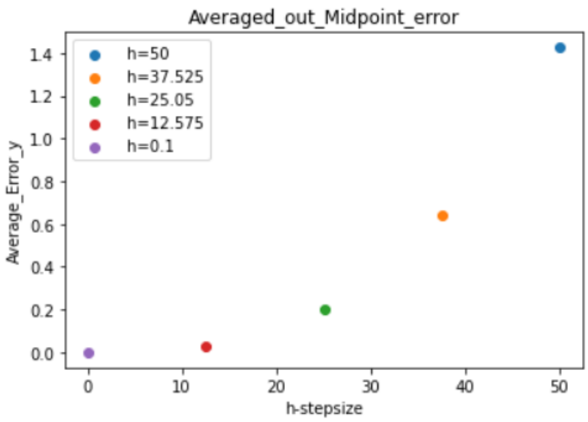


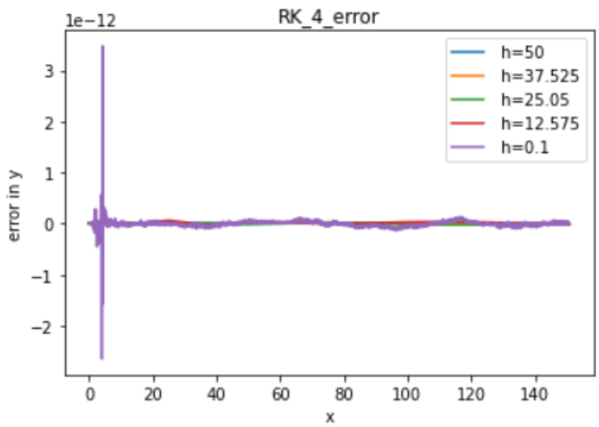


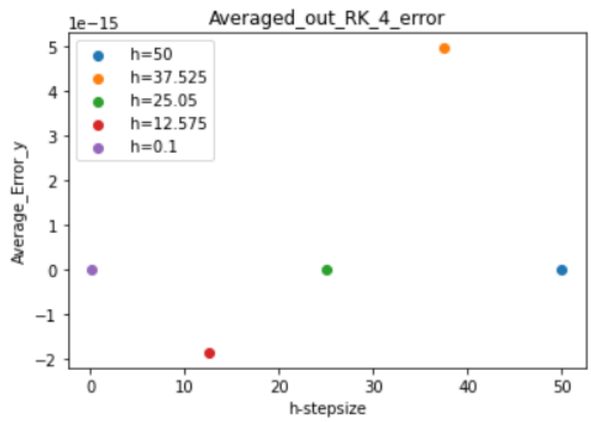










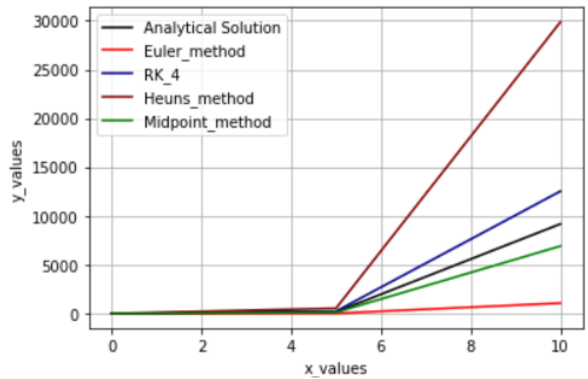


$$2.\ \frac{dy}{dx} = 4e^{0.8x} - 0.5y$$

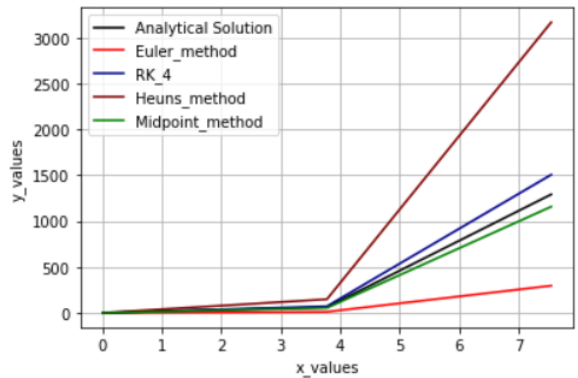
at $x_0 = 0$, $y_0 = 2$

Analytical Solution: $y = \frac{4}{1.3}e^{0.8x} - 1.076923e^{-0.5x}$

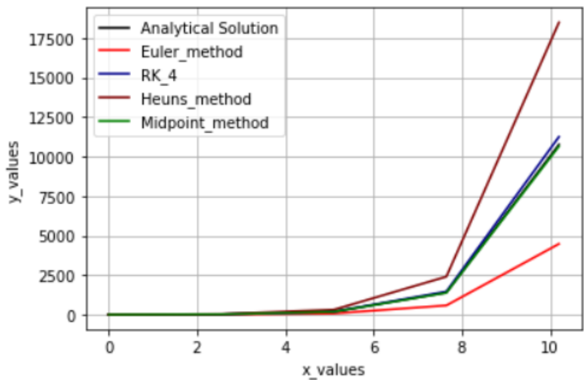
for h = 5-step size



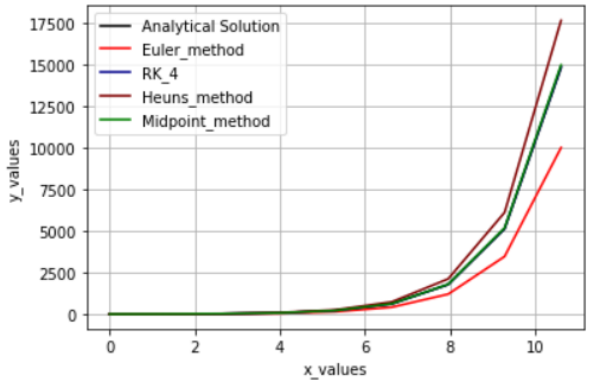
for h = 3.775-step size



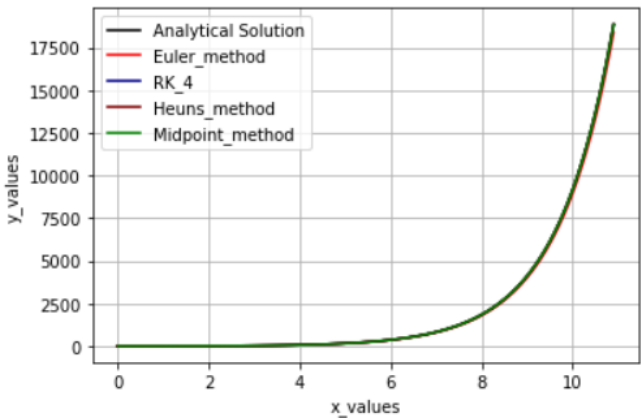
for h = 2.55-step size



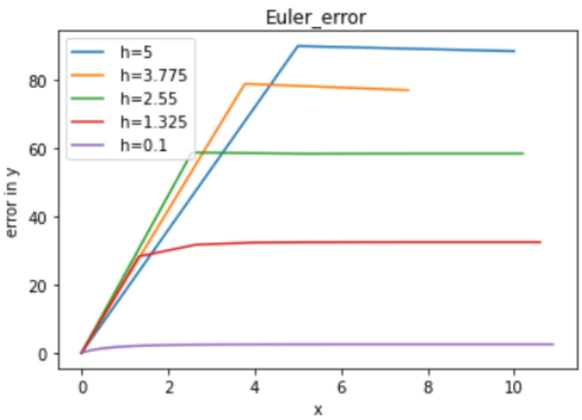
for h = 1.325-step size

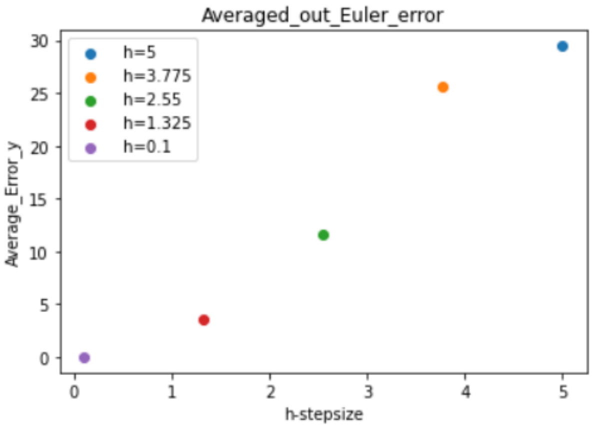


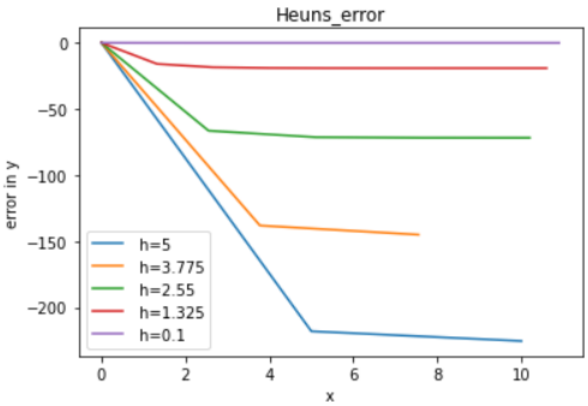
for h = 0.1-step size

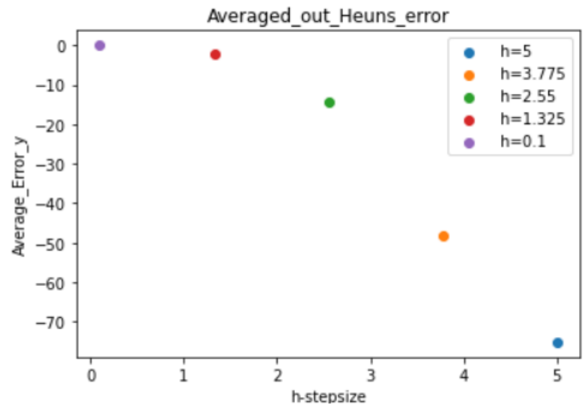


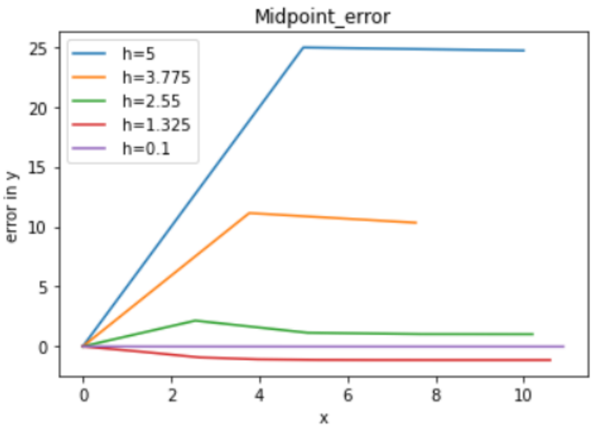
Error Analysis:

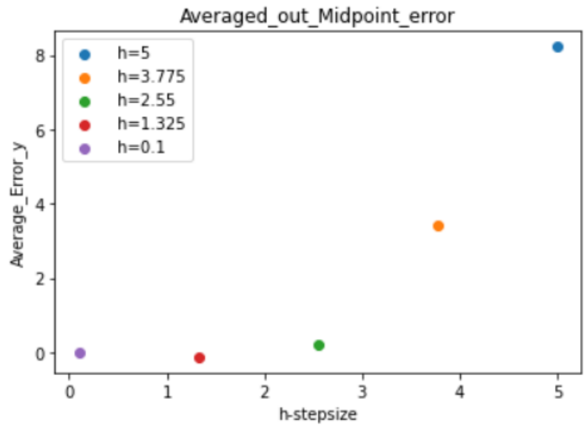


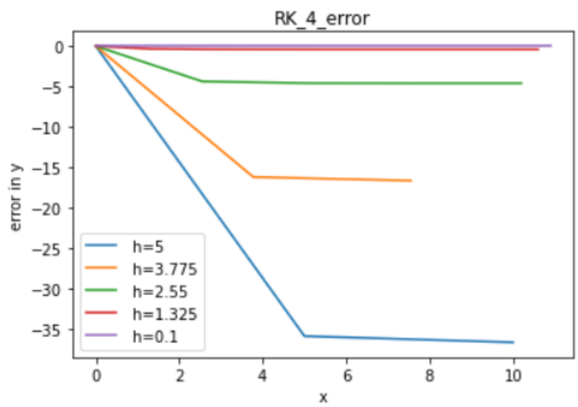


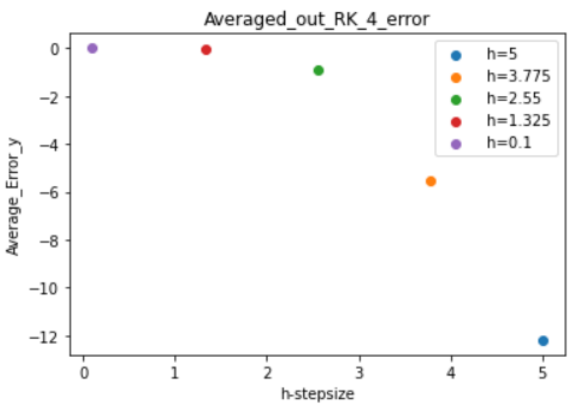












The Physical System

A body freely falling under the influence of gravity and experiencing a drag due to the resistance of the medium it is travelling through, eg. air, water, etc.

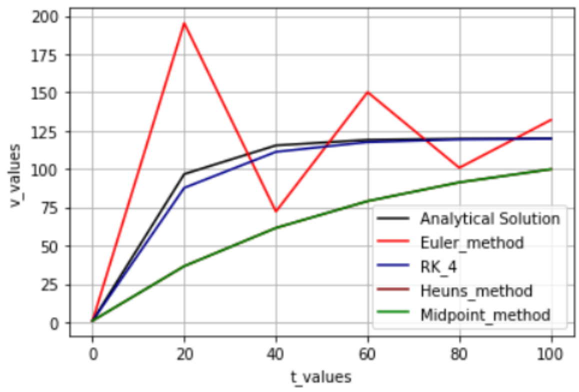
Maximum velocity attained by the body while falling through the medium is called **terminal velocity**.

$$\frac{dv}{dt} = g\left(1 - \frac{v}{v_t}\right)$$

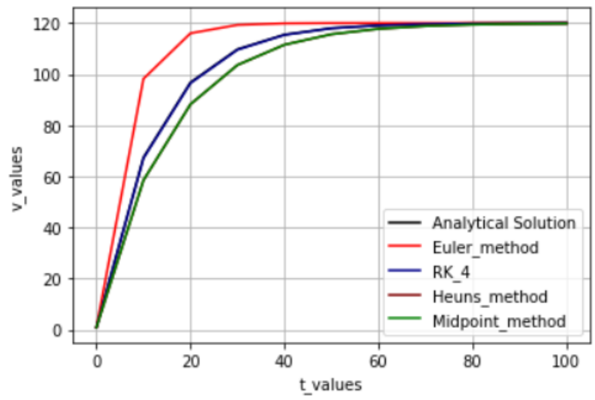
at $t = 0$, $v_0 = 1$

Analytical Solution: $$v = v_t - (v_t - v_0)e^{-gt/v_t}$$

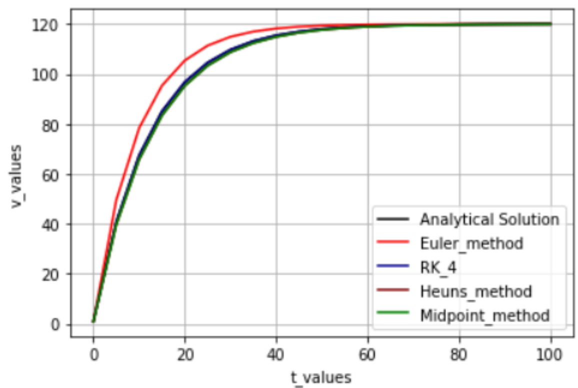
`for h = 20-step size`



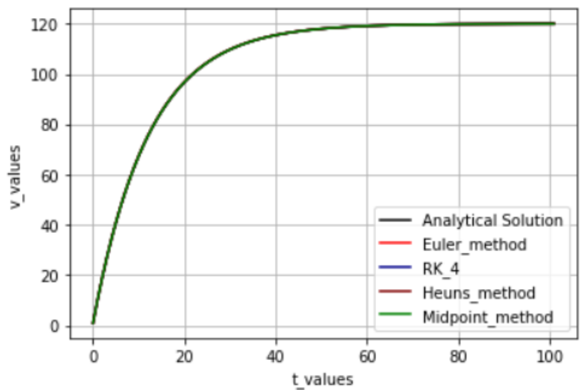
`for h = 10-step size`



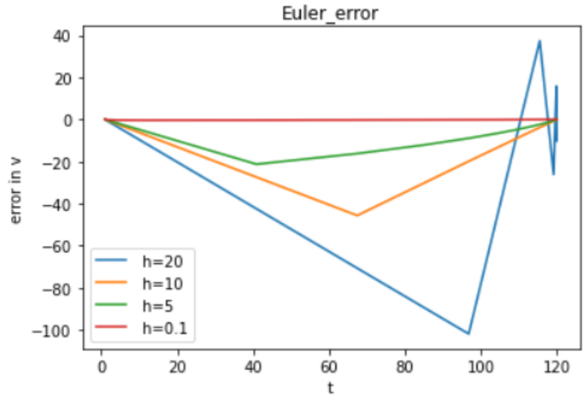
`for h = 5-step size`

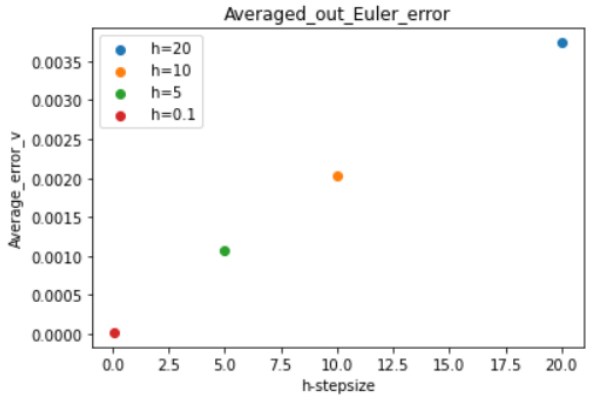


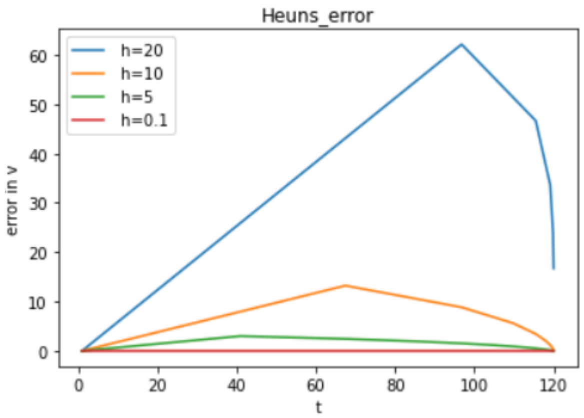
`for h = 0.1-step size`

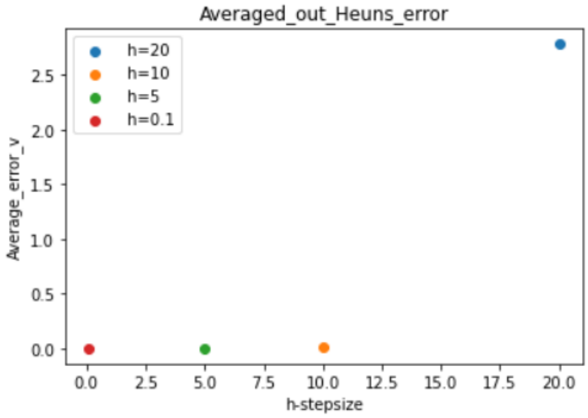


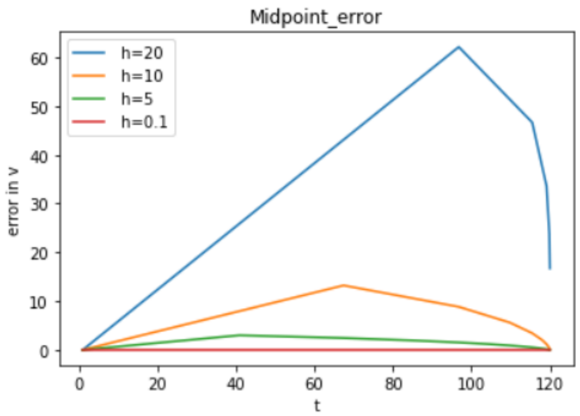
Error Analysis of Physical System:

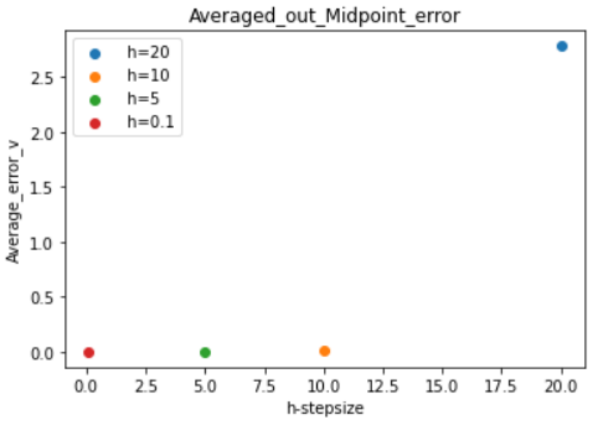


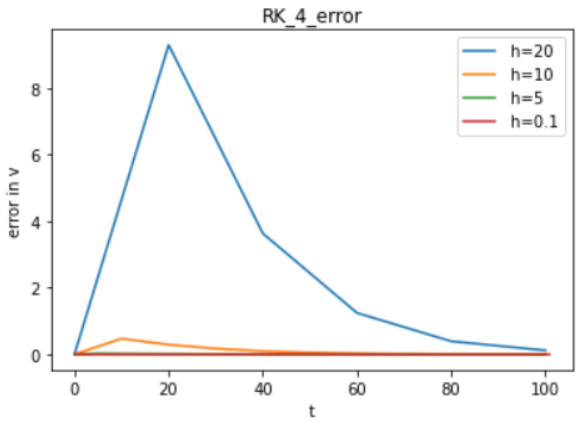


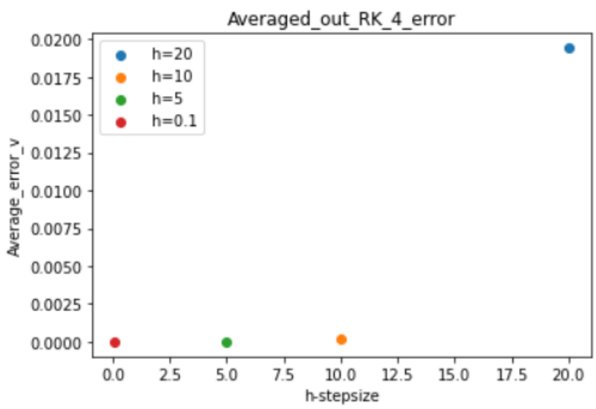






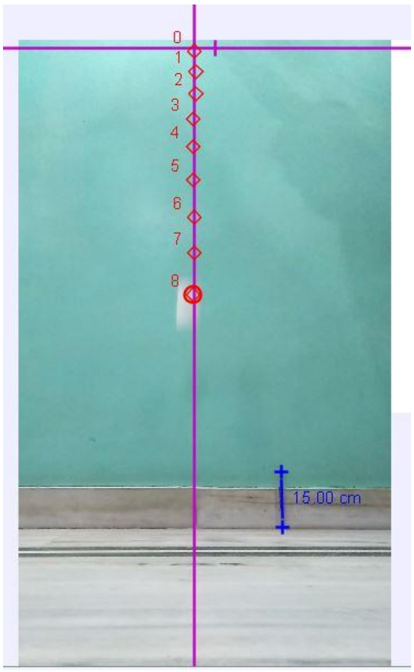


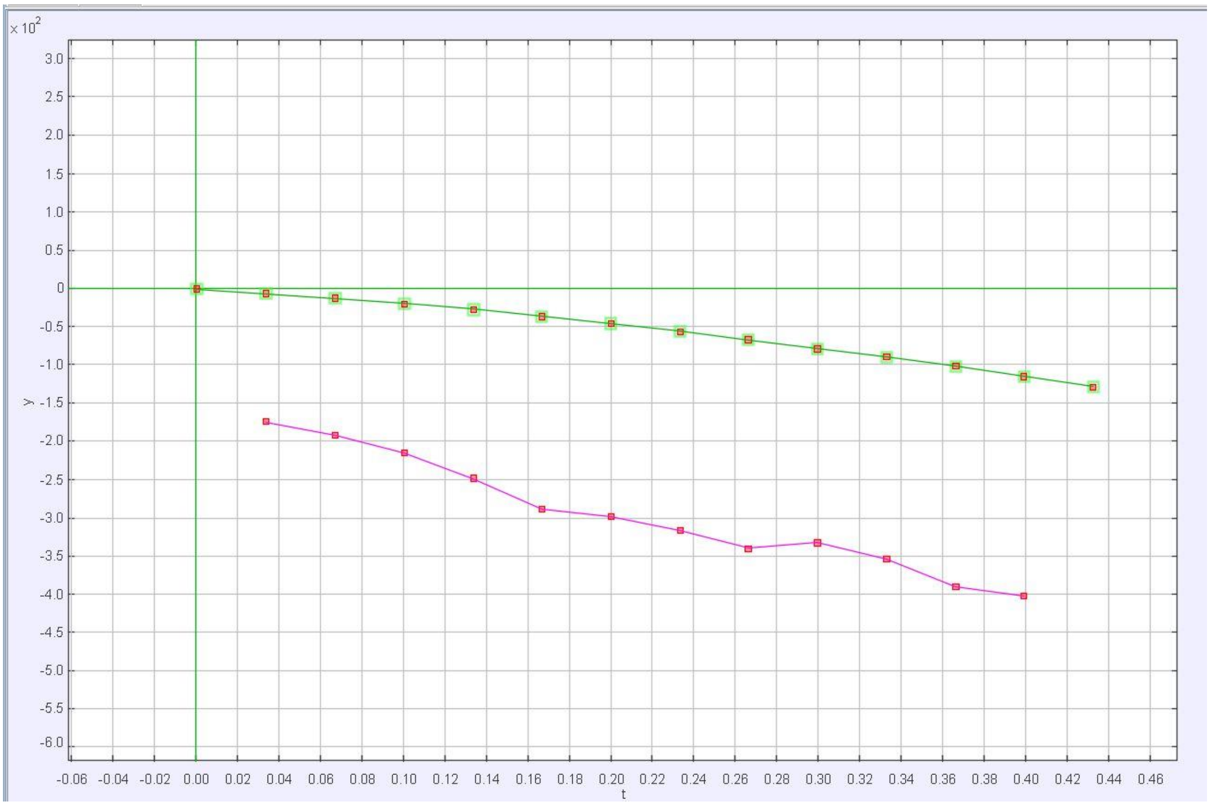




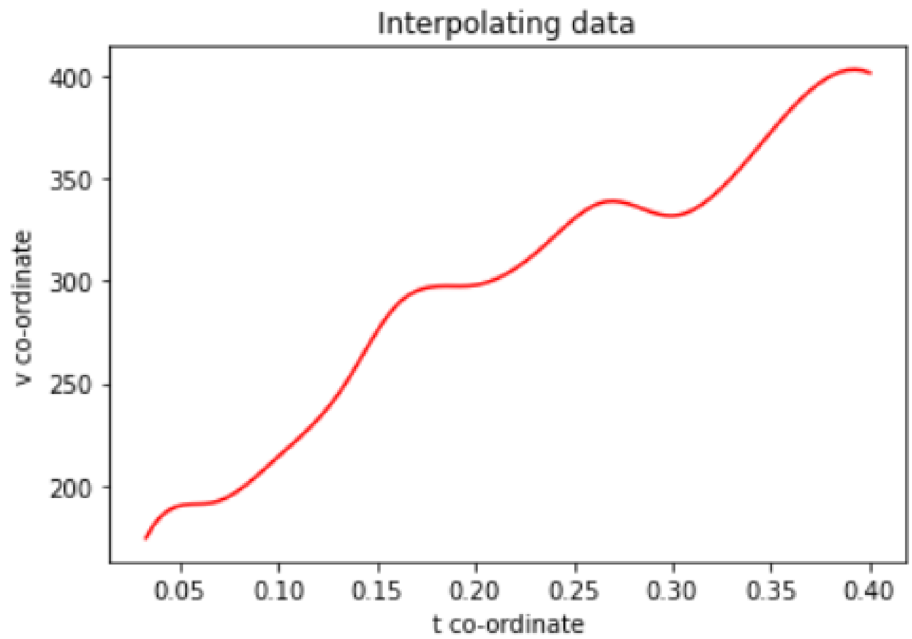
A body dropped from a height of 2 metres and analysed using **TRACKER** Software.

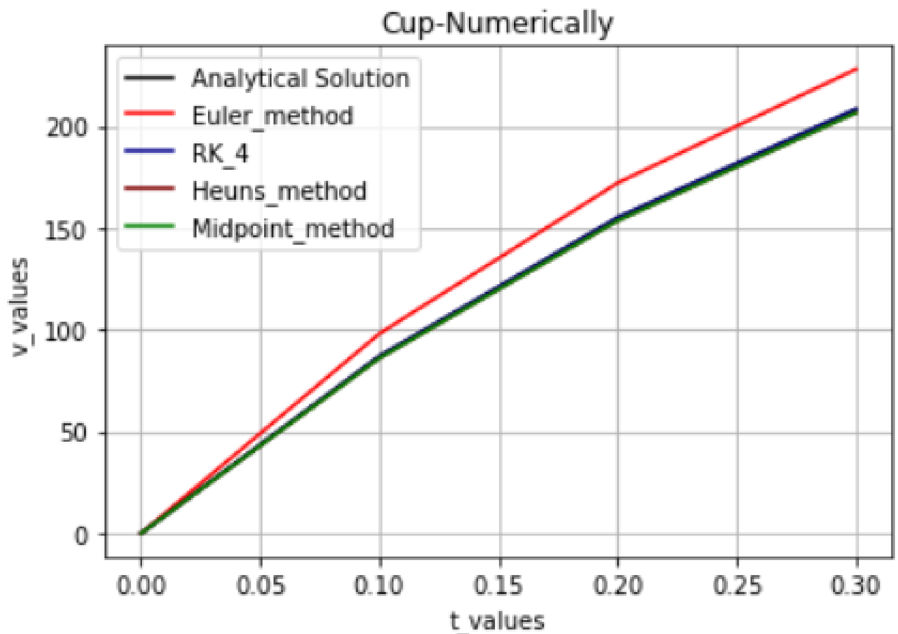
i. Paper Cup:
Mass of the cup ≈ **1.5 gm**
Area of the surface experiencing drag ≈ **16.65 cm^2**





Green plot represents y vs t & Pink plot represents v vs t





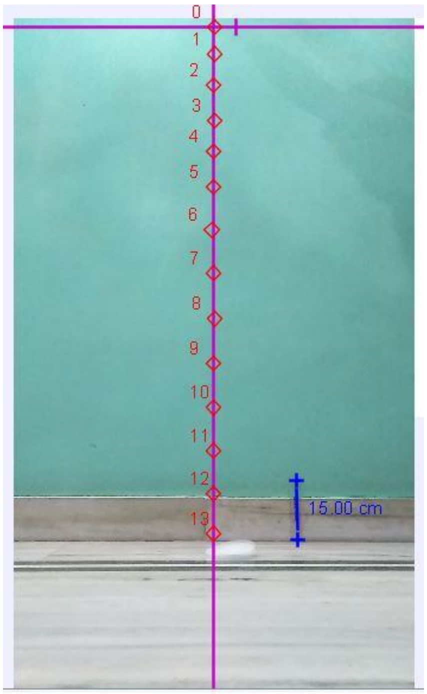
$$k = \frac{mg}{v_t}$$

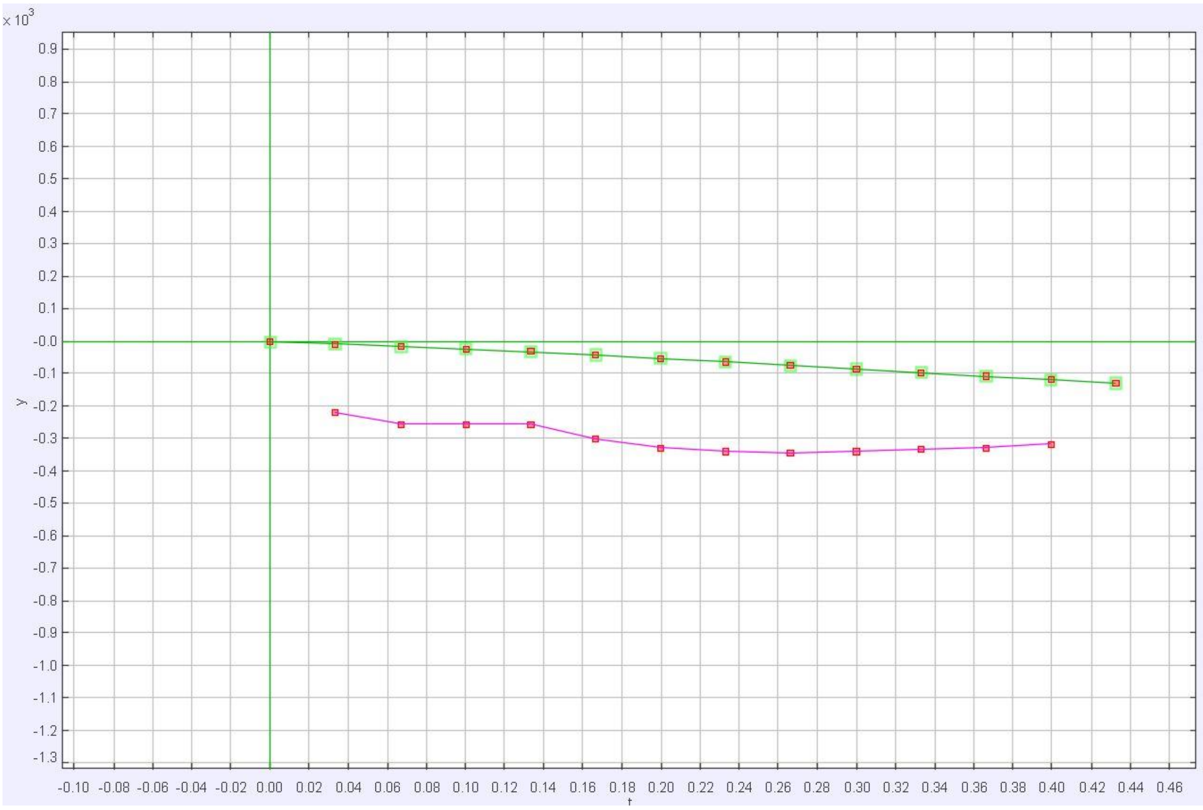
$$k_{cup} = \frac{1.5 \times 980}{400} = 3.68$$

ii. A plastic disk:

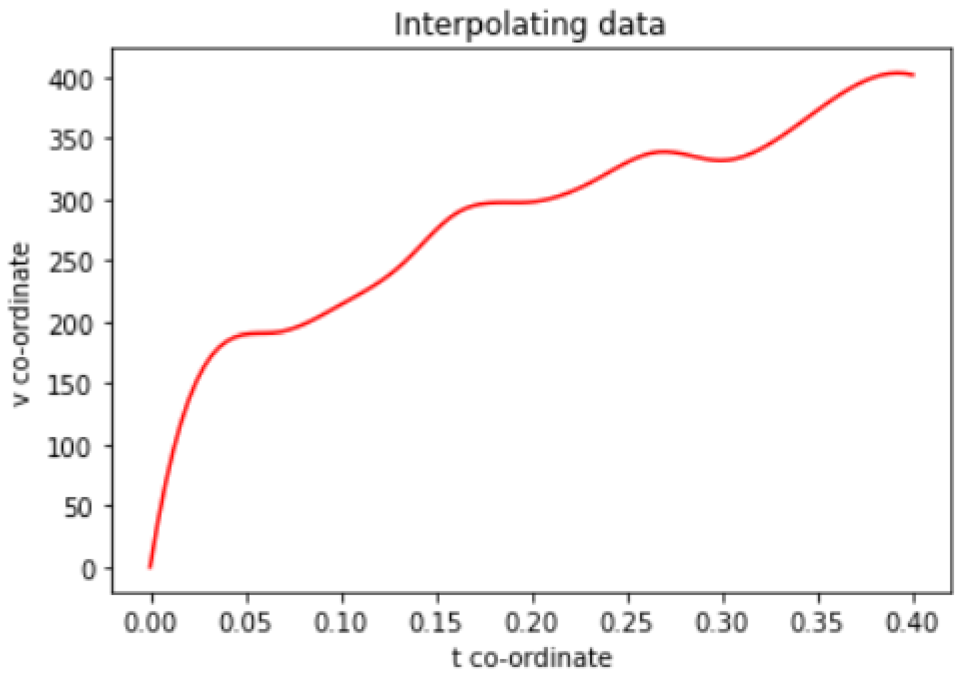
Mass of the disk ≈ **1.5 gm**

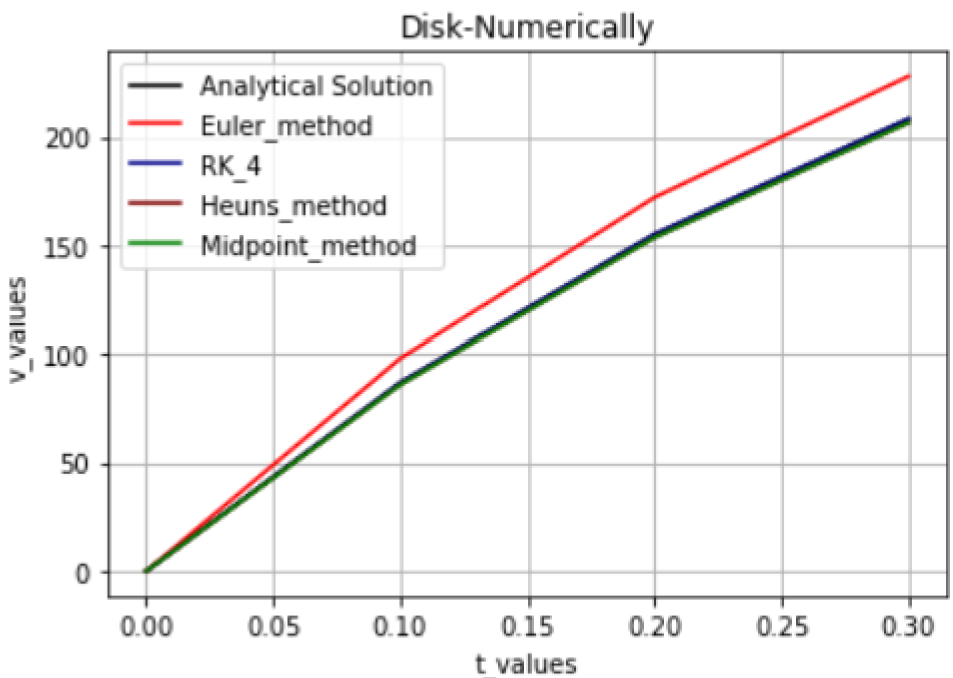
Area of the surface experiencing drag ≈ **30.16 cm^2**





Green plot represents y vs t & Pink plot represents v vs t





$$k = \frac{mg}{v_t}$$

$$k_{disk} = \frac{1.5 \times 980}{341} = 4.31$$

Conclusion

1. As expected, when step size is decreased, the estimated value of 'y' obtained from all four numerical methods approach the true value of y at that x_0.
2. The Euler- method shows significant amount of difference between true and estimated value.
3. The Heun's and Midpoint methods show intermediate difference between true and estimated value of 'y'.
4. The RK-4 method shows the least amount of difference between true and estimated value even at higher value of step-size which is evident from the graphs.
5. For simple differential equations such as polynomials, just like our first function, there is almost no difference between the solution obtained from RK-4 and analytically obtained solution. The Relative percentage error is almost in the range of 10^{-12}.
6. For complicated functions such as our second function, $\frac{dy}{dx} = f(x,y)$=$4e^{0.8\,x} - 0.5y$ the behavior of Euler, Heun's and Midpoint method is similar as in previous example, but for RK-4, there is a significant difference between numerically obtained and analytically obtained y value.
7. Looking at the velocity vs time graphs for the paper-cup and the plastic disk, we see that both the bodies have experienced significant drag, i.e., air resistance. However, since the disk has a larger area to mass ratio, it attains the terminal velocity quickly as compared to the paper-cup.
8. If we had released the cup from a greater height, it could have attained the terminal velocity.
9. Both the bodies have very less density, hence they experience a significant amount of upthrust, which is more evident in the case of the paper-cup where the upthrust force has overpowered the drag force.

Acknowledgement:

*We would like to extend our sincere gratitude towards **The Department of Physics, St. Xavier's College, Mumbai** for giving us this opportunity.*

*We are also thankful to **Dr. Manojendu Choudhury** Sir for his constant support and guidance.*

Bibliography:

1. **Steven C. Chapra and Raymond P. Canale, "Numerical Methods for Engineers", Seventh Edition.**

References:

1. **https://paulnotebook.net**
2. **http://connor-johnson.com/2014/02/21/numerical-solutions-to-odes/**

Python Code File