Nepal Journal of Mathematical Sciences (NJMS)
ISSN: 2738-9928 (online), 2738-9812 (print)
Vol. 2, No. 2, 2021 (August): 79-88
DOI: https://doi.org/10.3126/njmathsci.v2i2.40126
©School of Mathematical Sciences,
Tribhuvan University, Kathmandu, Nepal

Research Article
Received Date: June 20, 2021
Accepted Date: August 12, 2021
Published Date: August 30, 2021

Formulative Visualization of Numerical Methods for Solving Non-Linear Ordinary Differential Equations

Jeevan Kafle[1], Bhogendra Kumar Thakur[2] ,& Grishma Acharya[3]
[1,2,3]*Central Department of Mathematics, Tribhuvan University*
Kirtipur, Kathmandu, Nepal
Correspondence Author: Jeevan Kafle
Email: jeevankafle@yahoo.com

Abstract: *Many physical problems in the real world are frequently modeled by ordinary differential equations (ODEs). Real-life problems are usually non-linear, numerical methods are therefore needed to approximate their solution. We consider different numerical methods viz., Explicit (Forward) and Implicit (Backward) Euler method, Classical second-order Runge-Kutta (RK2) method (Heun's method or Improved Euler method), Third-order Runge-Kutta (RK3) method, Fourth-order Runge-Kutta (RK4) method, and Butcher fifth-order Runge-Kutta (BRK5) method which are popular classical iteration methods of approximating solutions of ODEs. Moreover, an intuitive explanation of those methods is also be presented, comparing among them and also with exact solutions with necessary visualizations. Finally, we analyze the error and accuracy of these methods with the help of suitable mathematical programming software.*

Keywords: *Ordinary Differential Equations, Numerical Methods, Error Function and Error Analysis.*
2020 Mathematics Subject Classification: 65L06, 65L10

1 Introduction

Doing mathematics and connecting mathematics with real life are two different things. When it comes to connecting, it is challenging even to great Mathematicians [8]. Differential equations involving its derivative of only one independent variable are called ordinary differential equations (ODEs) [2, 4, 17]. Differential equations are the most appropriate tool for analyzing the problem of many mathematical modelings mainly within the area of engineering, physics, aeronautics, medicine, environmental science, astronomy, chemistry, biology, and many other applied field [2, 17, 21, 22]. Most of the mathematical modeling problem was found to be non-linear [14, 20]. In the case of a non-linear equation it is very difficult or most of the time impossible to find the analytical solution of that equation [2, 5, 18]. The motion of the simple pendulum, Planetary motion under gravity, etc. are a typical example of second-order differential equation that is impossible to solve analytically [8, 17]. For such types of equations, we use the alternative approach to approximate the analytical solution. One such method is a numerical method [9, 13].

To approximate the solution of such types of the non-linear differential equation we will define the necessary numerical methods like, Explicit (Forward) and Implicit (Backward) Euler method, Classical second-order Runge-Kutta (RK2) method (Heun's method or Improved Euler method), third-order Runge-Kutta (RK3) method, fourth-order Runge-Kutta (RK4) method and Butcher fifth-order Runge-Kutta (BRK5) method which are popular iteration methods of approximating solutions of ODEs [7, 14, 26]. In this work, we compare the various numerical methods with the necessary visualization.

Kammruzzman and Nath [16] solved the initial value problem by considering three classical numerical solutions methods viz., Euler's method, modified Euler's method, and Runge Kutta method. Considering relative error as a difference of the approximate solution and analytic solution they observed only the relative error was nominal for the fourth-order Runge-Kutta (RK4) method. They also observed the rate of convergence of the Euler improved and modified Euler method in order of $O(h)$ and that of RK4 is $O(h^4)$. Their finding suggested that the RK4 method was more accurate and converged faster to the exact solution than other methods. Kafle et al. [13] studied the thermal diffusivity of heat equation of non-linear differential equation types by using the finite difference method. Aggarwal and Singh [1] studied the numerical solution of improper integrals containing error functions. These improper integral appears in the integral transformation of advanced non-linear differential equations such as radioactive decay, heat conduction problem, electrical circuit problem, and population growth problem. Rabiei and Ismail [23] found that the improved Runge-Kutta (IRK) methods are two-step in nature and require a lower number of stages per step compared to the classical Runge-Kutta methods and derive the fifth-order Improved Runge-Kutta method (IRK5) with only five stages. Improve Runge-Kutta (IRK) methods had the lower number of function evaluations per step. The stability region of the method is presented and numerical examples had given to illustrate the computational efficiency and accuracy of IRK5 compared to the RK5 method. Ahamad and Charan [2] in order to solve the initial value problem for fourth-order ODEs presented 5^{th} order Runge-Kutta (RK5) method.

Hossain et al. [9] considered a numerical example to solve second order IVP for ODE using fourth-order Runge Kutta (RK4) method and Butchers fifth-order Runge-Kutta (BRK5) method. They reduced the step size as much as possible to compare the approximate solution with the exact solution. They proposed that Butcher's fifth-order Runge-Kutta method was more appropriate and proficient for finding the numerical solution of IVP than the fourth-order Runge Kutta method. Ismail [12] derived a new sixth-order Runge-Kutta method, depend in on the pre-existing fifth-order Runge Kutta method. This method preluded a new five-function evaluation technique other than the six or seven function evaluation methods in the standard model. This method was more efficient than the standard Runge-Kutta method. Kafle et al. [14] compare the different iterative methods to analyze the damping conditions of series RLC circuits under the transient situations with DC source and they found the best iterative (BRK5) method to solve the second-order ODE of series RLC circuit. They observed the three damping conditions by using the BRK5 method. Kafle et al. [15] observed the application of iterative methods to analyze the different damping conditions of parallel RLC circuits. They found the Butcher fifth-order Runge-Kutta method is the best iterative method to solve the second-order ODE of parallel RLC circuit.

2 Numerical Methods and Simulated Results

To solve the initial value problem (IVP) for ordinary differential equations (ODE), we discuss different iterative methods here. We initiate with Euler's method, the simplest method, at the beginning. We exploit the solution of the proposed non-linear differential equation using explicit and implicit Euler method, classical second order Runge-Kutta (RK2) method, third-order Runge-Kutta (RK3) method, fourth-order Runge-Kutta (RK4) method, and Butcher's fifth-order Runge-Kutta (BRK5) method [6, 7]. Also, we compare the above-mentioned numerical methods with the necessary visualization. For the comparison of numerical methods we take the IVP [6]:

$$y'(t) = ty + t^2, y(0) = 1 \tag{1}$$

on the interval $0 \leq t \leq 1$ and with the exact solution.

$$y(t) = \sqrt{\frac{\pi}{2}}\, e^{t^2/2}\, erf(\frac{t}{\sqrt{2}}) + e^{t^2/2} - t \tag{2}$$

The error function (Gauss error function), often denoted by erf, is a complex function of a complex variable defined as [1].

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \tag{3}$$

2.1 Euler's (Explicit or Forward) Method

Euler put forward this method for initial value problems (IVP) at 1768. Euler's method is a numerical method that is used to approximate the solutions to explicit first-order equations [4, 25]. It is based on making successive linear approximations to the solution. Consider an initial value problem (IVP) for smooth function $f(t, y(t))$.

$$y'(t) = f(t, y(t)), \qquad y(t_0) = y_0 \tag{4}$$

Let, y_n be the position at time t_n with step size h. By the definition of the derivative, we have

$$\frac{dy}{dt} = \lim_{h \to 0} \frac{y(t+h) - y(t)}{h} \Rightarrow \frac{dy}{dt} \approx \frac{y(t+h) - y(t)}{h}$$

When $h \to 0$, from Fig. 1, it is clear that, at that time the slope at that point or the tangential approximation lies completely in the curve. Slope at the point (t_n, y_n) is

$$\frac{y_{n+1} - y_n}{h} \approx \frac{dy}{dt} = f(t_n, y_n)$$

$$\Rightarrow y_{n+1} = y_n + hf(t_n, y_n). \tag{5}$$

This formula is called Forward Euler's or Explicit Euler's Method [16].

The simulated result for the IVP (1) and exact solution given by equation (2) by using the Forward Euler method is shown below.

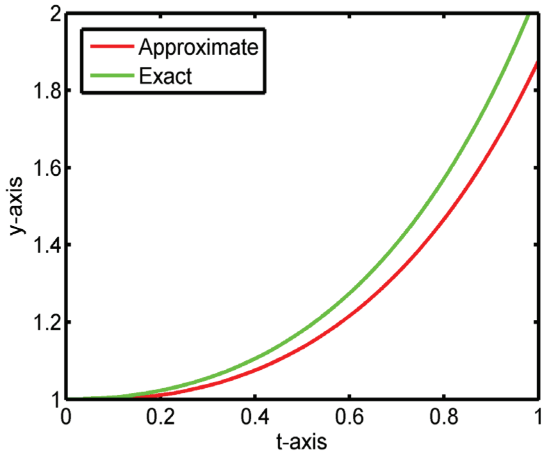

Figure 2: Explicit Euler with exact.

t_n	Explicit Euler Method		Error
	Approximate solution	Exact solution	
0.0	0.000000	0.000000	0.000000
0.1	1.000000	1.005347	0.005347
0.2	1.011000	1.022889	0.001189
0.3	1.035220	1.055192	0.019972
0.4	1.075277	1.105319	0.030042
0.5	1.134288	1.176975	0.042687
0.6	1.216002	1.274679	0.058677
0.7	1.324962	1.403988	0.079026
0.8	1.466710	1.571788	0.105078
0.9	1.648046	1.786666	0.138620
1.0	1.877370	2.059407	0.182037

Table 1: Explicit Euler with error.

As we understand about the numerical methods, if the step size h is smaller then the numerical method gives the more accurate result i.e. less error. Here, the Forward Euler method is more accurate if the step size h is smaller. Table 1 below shows the result of the Forward Euler method with the exact solution and approximate solutions for the step size $h = 0.1$. The last column in Table 1 contains the data of errors which is the modulus of difference between the exact solution and approximate solutions. From Fig. 2, we

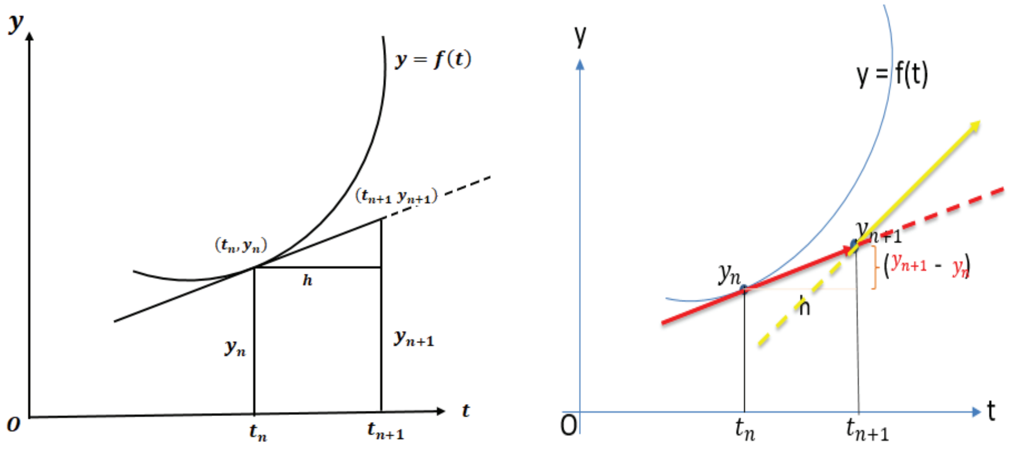

Figure 1: Visualization of Euler method Left: Explicit and Right: Implicit [25].

observed that the approximate solution curve represented by the red line is approaching the exact solution curve represented by the green line.

2.2 Euler's (Implicit or Backward) Method

The backward Euler method is almost similar to the forward Euler approach, backward Euler method uses the next point t_{n+1} as the point for calculating the derivative [19, 26]. From Fig. 1, slope at the point (t_{n+1}, y_{n+1}) is given by

$$\frac{y_{n+1} - y_n}{h} \approx \frac{dy}{dt} = f(t_{n+1}, y_{n+1})$$

$$\Rightarrow y_{n+1} = y_n + hf(t_{n+1}, y_{n+1}). \tag{6}$$

This procedure is then iterated until t_{n+1} converges onto a solution. Which is called Implicit (Backward) Euler Formula [25].

The simulated result for the IVP (1) and exact solution given by equation (2) by using the Backward Euler method is shown below:

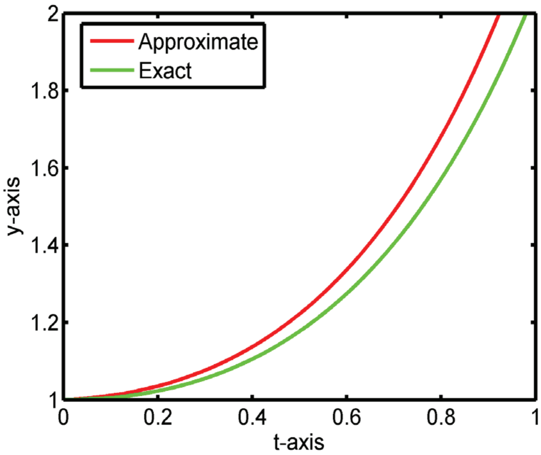

Figure 3: Implicit Euler with exact.

t_n	Implicit Euler Method		
	Approximate solution	Exact solution	Error
0.0	0.000000	0.000000	0.000000
0.1	1.011000	1.005347	0.005653
0.2	1.035442	1.022889	0.012553
0.3	1.076247	1.055192	0.021055
0.4	1.136948	1.105319	0.031629
0.5	1.221869	1.176975	0.044894
0.6	1.336347	1.274679	0.061668
0.7	1.487024	1.403988	0.083036
0.8	1.682233	1.571788	0.110445
0.9	1.932506	1.786666	0.145840
1.0	2.251250	2.059407	0.191843

Table 2: Implicit Euler with error.

Our understanding of any numerical method is that if the step size h is sufficiently reduced then the method gives more accurate results and lesser error. Here, the Implicit Euler method is more accurate if the step size h is smaller. Table 2 below shows the result of the Implicit Euler method with the exact solution and approximate solutions for the step size $h = 0.1$. The last column in Table 2 contains the data of errors which is the modulus of difference between the exact solution and approximate solutions. From Fig. 3, we observed that the approximate solution curve represented by the red line is approaching the exact solution curve represented by the green line. In this method, the approximate solution curve is converging faster towards the exact solution curve than the Explicit Euler method.

2.3 Classical Second-Order Runge-Kutta Method

The Runge-Kutta 2^{nd} order method is a numerical technique used to solve an ordinary differential equation of the form (4). Only first-order ordinary differential equations can be solved by using the Runge-Kutta 2^{nd} order method [3]. It is obvious from above Fig 4 that the our error will get minimized if we take an average of the slope at the point (t_n, y_n) represented by the red line and slope at the point (t_{n+1}, y_{n+1}) represented by the yellow line. Let k_1 be the slope at (t_n, y_n) i.e., $k_1 = f(t_n, y_n)$ and k_2 be the slope at the point (t_{n+1}, y_{n+1}) i.e., $k_2 = f(t_{n+1}, y_{n+1}) = f(t_n + h, y_n + hk_1)$. Thus, the average of k_1 and k_2 is given by

$$k_3 = \frac{k_1 + k_2}{2} = \frac{1}{2}[f(t_n, y_n) + f(t_{n+1}, y_{n+1})]$$

Therefore, we have

$$\Rightarrow y_{n+1} = y_n + hk_3$$

$$\Rightarrow y_{n+1} = y_n + \frac{h}{2}[f(t_n, y_n) + f(t_{n+1}, y_{n+1})]$$

$$\Rightarrow y_{n+1} = y_n + \frac{h}{2}[f(t_n, y_n) + f(t_n + h, y_n + hk_1)]. \tag{7}$$

This is the Classical second order Runge-Kutta method. It is also known as Heun's method or the Improved Euler method [12, 26].

The simulated result for the IVP (1) and exact solution given by equation (2) by using the Classical Second Order Runge-Kutta (Heun's or RK2) method is shown below.

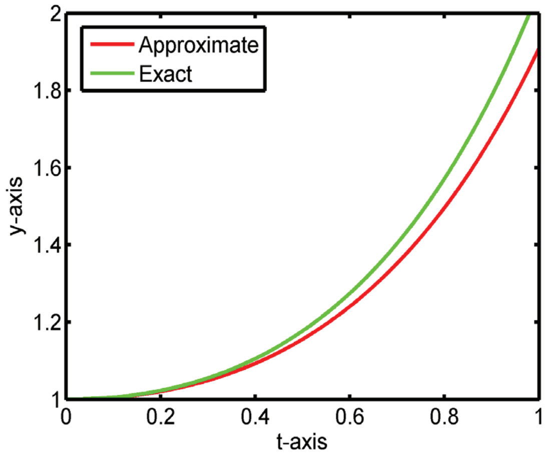

Figure 5: Heun's (RK2) with exact.

t_n	RK2 Method		Error
	Approximate solution	Exact solution	
0.0	0.000000	0.000000	0.000000
0.1	1.005500	1.005347	0.000153
0.2	1.023193	1.022889	0.000304
0.3	1.055640	1.055192	0.000448
0.4	1.105901	1.105319	0.000258
0.5	1.177672	1.176975	0.000697
0.6	1.275461	1.274679	0.000782
0.7	1.404804	1.403988	0.000816
0.8	1.572558	1.571788	0.000770
0.9	1.787266	1.786666	0.000600
1.0	2.059649	2.059407	0.000242

Table 3: RK2 with error.

Table 3 shows the result of RK2 method with exact solution and approximate solutions for the step size $h = 0.1$. The last column in Table 3 contains the data of errors which is the modulus of difference between the exact solution and approximate solutions. From Fig. 5, we observed that the approximate solution curve represented by the red line is approaching the exact solution curve represented by the green line. In this method, the approximate solution curve is converging faster towards the exact solution curve than the Explicit and Implicit Euler method.

2.4 Third Order Runge-Kutta (RK3) Method

The Third-order Runge-Kutta method (RK3) is widely used for solving initial value problems (IVP) for ordinary differential equations (ODE). The general formula for the Runge-Kutta third-order (RK3) method

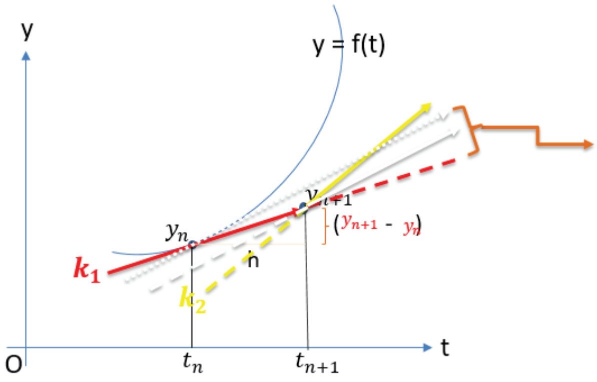

Figure 4: Visualization of Heun's method [25].

is shown below [12, 24].

$$y_{n+1} = y_n + h\left[\frac{k_1}{6} + \frac{2k_2}{3} + \frac{k_3}{6}\right] \tag{8}$$

$$\text{where} \quad k_1 = f(t_n, y_n), \quad k_2 = f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1\right), \quad k_3 = f(t_n + h, y_n - hk_1 + 2hk_2).$$

The simulated result for the IVP (1) and exact solution given by equation (2) by using the Third Order Runge-Kutta (RK3) method is shown below.

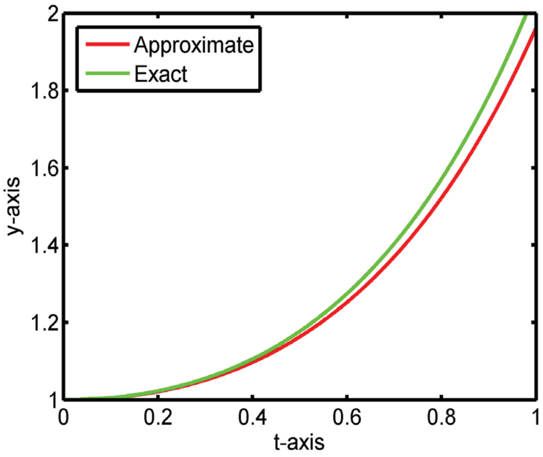

Figure 6: RK3 method with exact.

t_n	RK3 Method		
	Approximate solution	Exact solution	Error
0.0	0.000000	0.000000	0.000000
0.1	1.005351	1.005347	0.000004
0.2	1.022899	1.022889	0.000000
0.3	1.055208	1.055192	0.000016
0.4	1.105342	1.105319	0.000023
0.5	1.177006	1.176975	0.000031
0.6	1.274719	1.274679	0.000040
0.7	1.404039	1.403988	0.000051
0.8	1.571848	1.571788	0.000060
0.9	1.786736	1.786666	0.000070
1.0	2.059486	2.059407	0.000079

Table 4: RK3 with error.

Table 4 shows the result of the RK3 method with the exact solution and approximate solutions for the step size $h = 0.1$. The last column in Table 4 contains the data of errors which is the modulus of difference between the exact solution and approximate solutions. From Fig. 6, we observed that the approximate solution curve represented by the red line is approaching the exact solution curve is represented by the green line. In this method, the approximate solution curve is converging faster towards the exact solution curve than the Explicit Euler method Implicit Euler method and RK2 method. The error is also less than that of other mentioned methods.

2.5 Fourth Order Runge-Kutta (RK4) Method

This method was devised by two German mathematicians, Runge about 1894 A.D., and extended by Kutta a few years later. The Runge-Kutta method is most familiar because it is pretty accurate, steady, and simple to program [26]. This method is notable by their order in the logic that they concur with Taylor's series solution up to terms of h^r where r is the order of the method. The fourth-order Runge-Kutta method (RK4) is broadly used for solving initial value problems (IVP) for ordinary differential equation (ODE) [16]. The general formula for the fourth-order Runge-Kutta method is shown below [7, 12].

$$y_{n+1} = y_n + h[\frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}] \qquad \text{with} \tag{9}$$

$$k_1 = f(t_n, y_n), k_2 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1), k_3 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2), k_4 = f(t_n + h, y_n + hk_3).$$

The simulated result for the IVP (1) and exact solution given by equation (2) by using the Fourth Order Runge-Kutta (RK4) method are shown below.

Table 5 shows the result of the RK4 method with the exact solution and approximate solutions for the step size $h = 0.1$. The last column in Table 5 contains the data of errors which is the modulus of difference between the exact solution and approximate solutions. From Fig. 8, we observed that the approximate solution curve represented by the red line is approaching the exact solution curve represented by the green line. In this method, the approximate solution curve is converging faster towards the exact solution curve than the Explicit Euler method Implicit Euler method, RK2 method, and RK3 method. The error is also less than that of the other above-mentioned methods.

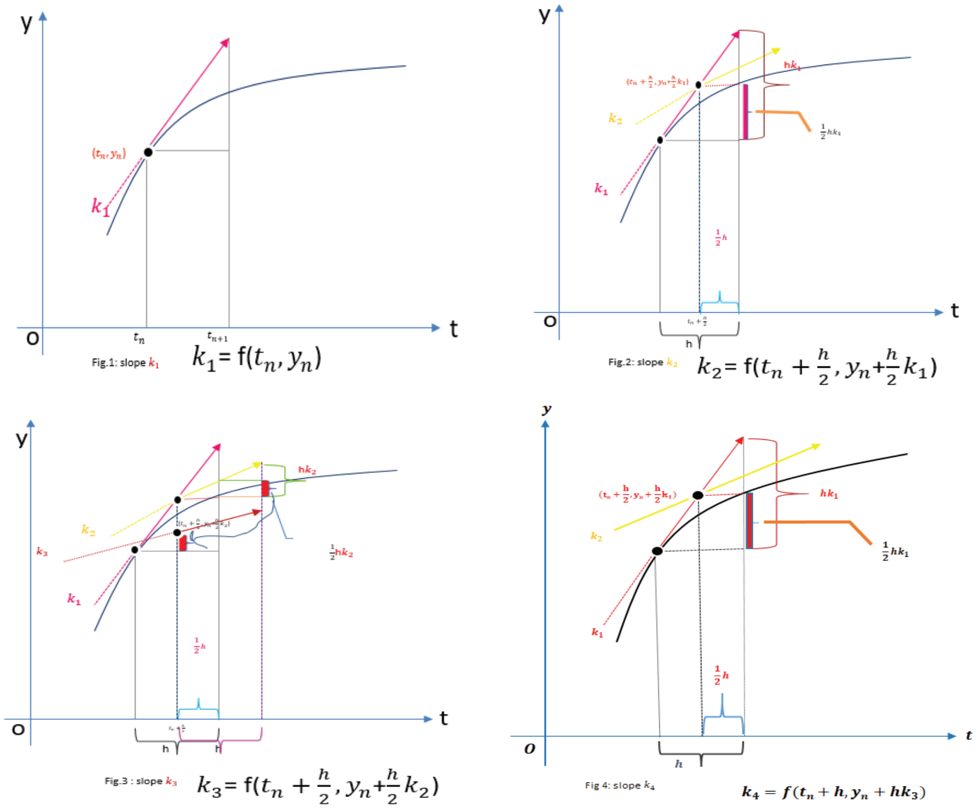

Figure 7: Visualization of RK4 method [7, 26].

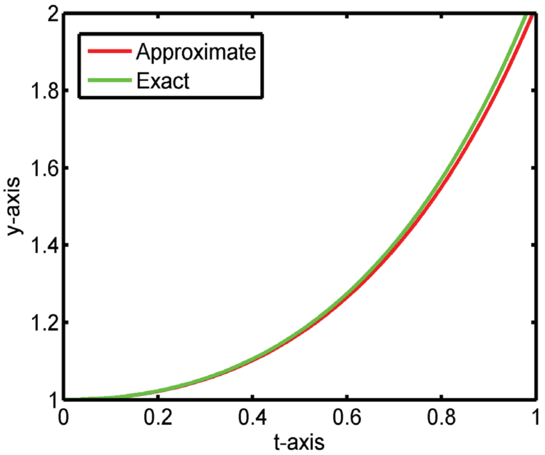

Figure 8: RK4 method with exact.

t_n	RK4 Method		
	Approximate solution	Exact solution	Error
0.0	0.000000	0.000000	0.000000
0.1	1.005346	1.005347	0.000001
0.2	1.022889	1.022889	0.000000
0.3	1.055192	1.055192	0.000000
0.4	1.105319	1.105319	0.000000
0.5	1.176975	1.176975	0.000000
0.6	1.274679	1.274679	0.000000
0.7	1.403988	1.403988	0.000000
0.8	1.571787	1.571788	0.000001
0.9	1.786665	1.786666	0.000001
1.0	2.059407	2.059407	0.000000

Table 5: RK4 with error.

2.6 Butchers Fifth Order Runge-Kutta (BRK5) Method

This method is distinguished by their order in the sense that agrees with Taylor's series solution up to terms of h^r where r is the order of the method [7, 13]. In this method, we consider the equation (4).

$$y_{n+1} = y_n + \frac{h}{90}\left(7k_1 + 32k_2 + 12k_4 + 32k_5 + 7k_6\right) \tag{10}$$

$$\text{where} \quad t_{n+1} = t_n + h, \quad k_1 = f(t_n, y_n), \quad k_2 = f\left(t_n + \frac{h}{4}, y_n + \frac{h}{4}k_1\right),$$

$$k_3 = f\left(t_n + \frac{h}{4}, y_n + \frac{1}{8}hk_1 + \frac{1}{8}hk_2\right), \qquad k_4 = f\left(t_n + \frac{h}{2}, y_n - \frac{1}{2}hk_2 + hk_3\right),$$

$$k_5 = f\left(t_n + \frac{3h}{4}, y_n + \frac{3}{16}hk_1 + \frac{9}{16}hk_4\right),$$

$$k_6 = \left(t_n + h, y_n - \frac{3}{7}hk_1 + \frac{2}{7}hk_2 + \frac{12}{7}hk_3 - \frac{12}{7}hk_4 + \frac{8}{7}hk_5\right).$$

The simulated result for the IVP (1) and exact solution given by equation (2) by using the Butcher Fifth Order Runge-Kutta (BRK5) method is shown below.

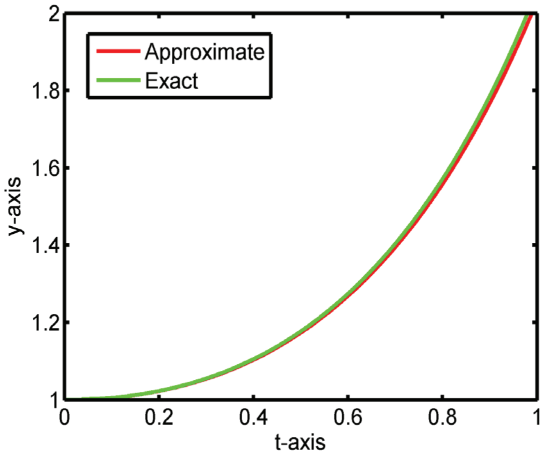

Figure 9: BRK5 method with exact.

t_n	BRK5 Method		Error
	Approximate solution	Exact solution	
0.0	0.000000	0.000000	0.000000
0.1	1.005347	1.005347	0.000000
0.2	1.022889	1.022889	0.000000
0.3	1.055192	1.055192	0.000000
0.4	1.105319	1.105319	0.000000
0.5	1.176975	1.176975	0.000000
0.6	1.274679	1.274679	0.000000
0.7	1.403988	1.403988	0.000000
0.8	1.571788	1.571788	0.000000
0.9	1.786666	1.786666	0.000000
1.0	2.059407	2.059407	0.000000

Table 6: BRK5 with error.

Table 6 shows the result of the BRK5 method with the exact solution and approximate solutions for the step size $h = 0.1$. The last column in Table 6 contains the data of errors which is the modulus of difference between the exact solution and approximate solutions. From Fig. 9, we observed that the approximate solution curve represented by the red line is approaching the exact solution curve represented by the green line. In this method, the approximate solution curve is converging faster towards the exact solution curve than the Explicit Euler method Implicit Euler method, RK2 method, RK3 method, and RK4 method. The error is also less than that of the other above-mentioned methods.

2.7 Comparison of numerical methods

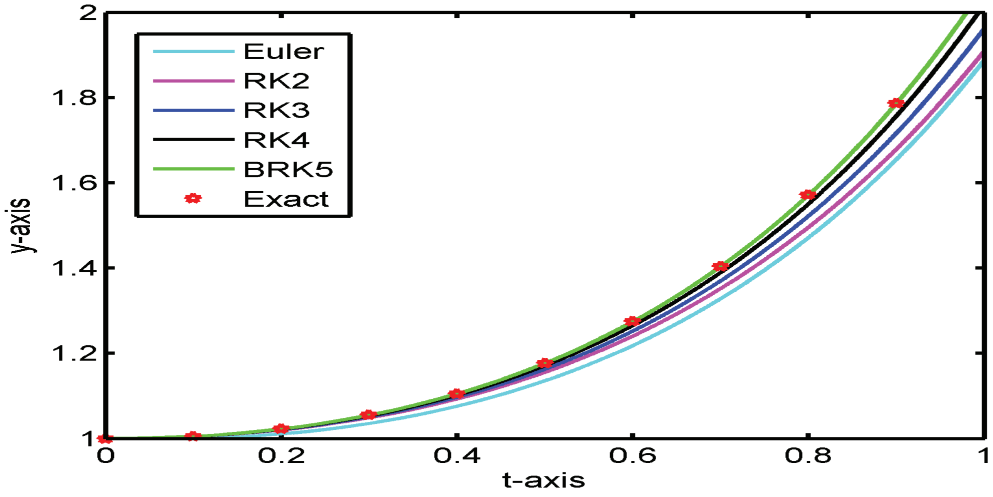

Figure 10: Comparison of Euler, RK2, RK3, RK4 and BRK5 method with exact solution.

From the above results, we see that the approximation results converge to the exact solution and if the step size is decreased then the error also decreased. This shows that the small step size gives an improved estimation. The Butcher fifth-order Runge-Kutta method requires five evaluations per step, so it should give more accurate results in comparison with others methods. Other methods were found to be less accurate due to the inaccurate numerical results. From the above study the Butcher fifth-order Runge-Kutta method was found to be more accurate and also the approximate solution converged faster to the exact solution when compared to the other methods. It may be concluded that the Butcher fifth-order Runge-Kutta method is powerful and more efficient in finding numerical solutions of initial value problems (IVP).

3 Conclusion

As we see in the above simulation results that numerical solution produced more error than any other method like analytical method. The implicit Euler method gives more approximate results than the Euler method. Besides Euler and Implicit Euler, the RK3 method gives less error in approximating the solution. Runge-Kutta's fourth-order method (RK4) is pretty accurate and steady so it gives a more accurate solution other than Euler, Implicit Euler, and RK3 method. It exhibits less error to the exact solution. Among all these methods Butcher fifth-order Runge-Kutta method (BRK5) is the best for approximating the solution of the non-linear equation as it gives the least error with the highest approximation. From the above discussion, we observed that the approximate results converge to the exact solution and if the step size is decreased then the error is also reduced. The small step size gives improved estimation. The Butcher fifth-order Runge-Kutta method requires five evaluations per step, so it should give more accurate results compared with other methods. The Butcher fifth-order Runge-Kutta method was found to be highly accurate and also the approximate solution converged faster to the exact solution compared to the other methods. We conclude that to obtain more accurate result higher order numerical methods is appropriate than lower-order methods.

References

[1] Aggarwal, S. and Singh, G.P. **(2019)** Kamal transform of error function. *Journal of Applied Science and Computations*, **6(5)**: 2223-2235.

[2] Ahamad, N. and Charan, S.**(2019)** Study of numerical solution of fourth order ordinary differential equations (ODEs) by fifth order Runge-Kutta method. *International Journal of Science, Engineering and Technology*, **6**: 230-237.

[3] Ascher, U.M. and Petzold, L.R. **(1998)** *Computer Methods for Ordinary Differential Equations and Differential Algebraic Equations.* Volume **61**: Siam.

[4] Atkinson, K., Han, W., and Stewart, D.E. **(2011)** *Numerical Solution of Ordinary Differential Equations.* Volume **108**: John Wiley & Sons.

[5] Baglan, I., Kanca, F. and Mishra, V.N. **(2018)** Determination of an Unknown Heat Source from Integral Overdetermination Condition, *Iran J Sci Technol Trans Sci* , **42**: 1373-1382.

[6] Boyce, W.E. and DiPrima, R.C. **(2012)** *Elementary Differential Equations and Boundary Value Problems.* John Wiley & Sons, Inc.

[7] Butcher, J.C. **(1933)** *Numerical Methods for Ordinary Differential Equations, second edition.* Volume **2**: New York, John Wiley & Sons, Inc.

[8] Hossain, M.J., Alam, M.S. & Hossain, M.B. **(2017)** A study on numerical solutions of second order initial value problems (IVP) for ordinary differential equations with fourth order and Butchers fifth order Runge- Kutta methods. *American Journal of Computational and Applied Mathematics*, **7(5)**: 129-137.

[9] Hossain, M.B., Hossain, M.J., Miah, M.M. & Alam, M.S. **(2017)**, A comparative study on fourth order and Butcher's fifth order Runge-Kutta methods with third order initial value problem. *Applied and Computational Mathematics*, **6(6)**: 243-253.

[10] Hussain, K.A., Ismail, F., Senu, N., & Rabiei, F. **(2017)**. Fourth order improved Runge–Kutta method for directly solving special third-order ordinary differential equations. *Iranian Journal of Science and Technology, Transactions A: Science*, **41(2)**: 429-437.

[11] Islam, M.A. **(2015)**. A comparative study on numerical solutions of initial value problems (IVP) for ordinary differential equations (ODE) with Euler and Runge-Kutta Methods. *American Journal of computational mathematics*, **5(3)**: 393-404.

[12] Ismail, M.M. **(2011)** Goeken-Johnson sixth-order Runge-Kutta method. *Journal of Education and Science*, **24(1)**: 119-128.

[13] Kafle, J., Bagale, L. & K.C., D. **(2020)** Numerical solution of parabolic partial differential equation by using finite difference method. *Journal of Nepal Physical Society*, **6(2)**: 57-65.

[14] Kafle, J., Thakur, B.K. & Bhandari, I.B. **(2021)** Visualization, formulation and intuitive explanation of iterative methods for transient analysis of series RLC circuit. *Journal of Physical Sciences, BIBECHANA*, **18(2)**: 9-17.

[15] Kafle, J., Thakur, B.K. & Bhandari, I.B. **(2021)** Application of numerical methods for the analysis of damped parallel RLC circuit. *Journal of Institute of Science and Technology*, **26(1)**: 28-34.

[16] Kamruzzaman, M. and Nath, M.C. **(2018)** A Comparative study on numerical solution of initial value problem by using Euler's method, modified Euler's method and Runge-Kutta method. *Journal of Computer and Mathematical Sciences*, **9(5)**: 493-500.

[17] Kendall A., Weimin H. & David, S. **(2009)** *Numerical Solution of Ordinary Differential Equations.* John Wiley & Sons Inc., Iowa City.

[18] Mishra, V.N., Khatri, K. and Mishra, L.N. **2013** Statistical approximation by Kantorovich type Discrete $q-$Beta operators, *Advances in Difference Equations*, 345.

[19] Ogunrinde, R.B., Fadugba, S.E. & Okunlola, J.T. **(2012)**. On some numerical methods for solving initial value problems in ordinary differential equations. *On Some Numerical Methods for Solving Initial Value Problems in Ordinary Differential Equations*, **1(3)**: 25-31.

[20] Paudel, K., Bhandari, P. & Kafle, J. **(2021)** Analytical solution for advection dispersion equation of the pollutant concentration using laplace transformation. *Journal of Nepal Mathematical Society*, **4(1)**: 33-40.

[21] Pokhrel, P.R., Kafle, J., Kattel, P. & Gaire, H.P. **(2020)** Analysis of blood flow through artery with mild stenosis. *Journal of Institute of Science and Technology*, **25(2)**: 33-38.

[22] Pokhrel, P.R., Lamsal, B., Kafle, J. & Kattel, P. **(2020)** Analysis of displacement of vibrating of mass-spring due to opposition force. *Tribhuvan University Journal*, **35(1)**: 21-32.

[23] Rabiei, F. and Ismail, F. **(2012)** Fifth order improved Runge-Kutta methods with reduced number of function evaluations. *Australian Journal of Basic and Applied Sciences*, **6(3)**: 97-105.

[24] Senthilnathan, C. **(2018)** A numerical solutions of initial value problems for ordinary differential equations with Euler and higher order of Runge-Kutta methods using matlab. *International Journal of Engineering Science Invention*, **7(4)**: 25-31.

[25] https://nm.mathforcollege.com/.

[26] https://mat.iitm.ac.in/home/sryedida/public-html/caimna/ode/rk/rk.html.

□□