Journal of Applied Science and Education, 2025,
Vol. 05, Iss. 01, S. No. 115, pp. 1-10
ISSN (Online): 2583-1372

Mathematical Modelling Based on Runge-Kutta Method

Nidhi Gargav[1], Chitra Singh[2], Deepak Gupta[3]
[1]Associate Professor, Barkatullah University Bhopal,
[2]Professor, Bansal Institute of Research Technology & Science
[3]Assistant Professor, FoET, University of Lucknow
nidhigargav8@gmail.com ,drchitrasingh123@gmail.com, dg61279@gmail.com

How to cite this paper: N Gargav, C Singh, D Gupta, "Mathematical Modelling Based on Runge-Kutta Method," *Journal of Applied Science and Education (JASE)*, Vol. 05, Iss. 01, S. No. 115, pp 1-10, 2025.

https://doi.org/10.54060/a2zjournals.jase.115

Received: 12/02/2025
Accepted: 18/03/2025
Online First: 20/04/2025
Published: 25/04/2025

Abstract

In this research paper, the Runge-Kutta method is used to minimize the error estimation in solving the problem of ordinary differential equations. By using the Runge-Kutta method, we can construct a higher- order accurate functions without having to calculate higher-order derivatives. The results show that the minimum error is obtained by using the Runge-Kutta second, third, and fourth order methods with step doubling method. It is important to note that there exists a straightforward technique for adaptive step size control in fourth-order Runge-Kutta, known as the step-doubling method (also referred to as the Local Error method). The method estimates the error by taking two steps of half the size and comparing the results. The computational simplicity of the step-doubling technique is an advantage, but in practice, algorithms based on embedded Runge-Kutta formulas are found to be more efficient.

Keywords: *Ordinary differential equation, Runge-Kutta method, error of approximation, first order differential equation.*

1. Introduction

The first order ordinary differential equation

$$\frac{dy}{dx} = f(x, y),\ y(x_0) = y_0 \qquad \text{--------- (1)}$$

There are numerous analytical methods available for solving these kinds of equations, but they are limited to solving a particular class of differential equations and their physical problems cannot be solved analytically. Thus, it becomes all important to take about their solution by numerical methods. When using numerical approaches, we obtain the numerical values of the dependent variable for certain values of the independent variable rather than trying to find a relationship between the variables. It must be noted that even the differential equations which are solvable by analytical methods can be solved numerically as well. Separation of variables is a technique commonly used to solve ordinary differential equations. If ϕ (x) is a solution of c_1(x), where c_1 is an arbitrary (non-zero) constant, then it is considered as a linear differential equation and homogenous.

It is often used to solve differential equation but is also used inside multivariable calculus when multiplying by an integrating factor allowing a non-exact differential to be made into an exact differential. The techniques take a long time and might not provide an exact answer and may contain a lot of errors. Thus, we need a method which gives us the solution as quick as possible with negligible error i.e. numerical method. Problems in which all the conditions are specified at the initial point only are called initial-value problems. For example, the problem given by equation (1) is an initial value problem. In order to derive a unique solution for an n^{th} order ordinary differential equation, n values of the dependent variable and/or its derivative must be specified for independent variable. In numerical analysis, the Runge-Kutta methods are an important family of implicit and explicit iterative methods. These techniques were developed around 1895 by the German mathematicians Carl David Runge and Martin Wilhelm Kutta. C.D. Runge and M.W. Kutta developed explicit and implicit Runge-Kutta methods, and in 1895, first and second-order methods were developed. The Runge-Kutta technique of third order was created in 1901. In 1905, Martin Kutta introduced the fourth and fifth order methods. In 1964, Butcher introduced the sixth-order Runge-Kutta method, and the seventh and eighth-order methods were later introduced by Curtis in 1996. This method helps to minimize the error in the equation.

2. BACKGROUND OF THE STUDY

Many mathematicians had ordinary differential equations (ODEs) as a point of interest to develop and solve numerical solutions. With the development of digital computers, scientists and mathematicians encounter a lot of challenging issues while solving ODEs that cannot be solved analytically. This prompted a need of solving such problems numerically. The Runge-Kutta method is a widely used numerical method for solving ODEs. Using an equation including the derivative of the answer at that point, the Runge-Kutta method is an iterative technique that determines the approximate solution at a given point based on the solution at the previous point. The Runge- Kutta method is known for its simplicity and efficiency, and it is widely used in a variety of scientific and engineering applications. However, these techniques are used to make predictions by representing continuous variables as discrete values and using mathematical operations to find approximate solutions, which rely on precise mathematical formulas to solve problems.

The adaptability of numerical approaches is one of their primary benefits. They can be used to solve a wide range of problems, including linear and nonlinear equations, differential equations, and optimization problems. Numerical methods also offer a dependable and effective solution to address complicated problems because they can be used on computers.

3. PROBLEM STATEMENT

In research and engineering, mathematical issues are frequently solved using numerical techniques. While studying literature, we come across numerous works that apply Runge- Kutta methods of first, second, third, and fourth orders to solve initial value problems. Several authors have attempted to obtain good precision by using these methods. Runge-Kutta methods have been employed in research for several years, and researchers have investigated and developed lower and higher order Runge-Kutta methods in terms of accuracy, stability, and efficiency. Adaptive Runge-Kutta methods use embedded integration formulas, which are pairs of Runge-Kutta formulas of different orders. The higher-order formula is used to estimate the error of the lower-order formula, and the step-size is adjusted accordingly. Step-doubling is a mechanism for adaptive step-size control in fourth-order Runge-Kutta method. A criterion for changing the step-size on the following step or rejecting the present step can be found by comparing the accuracy of the big step with the two tiny steps.

The above background motivates our research to explore the computational techniques of Runge-Kutta methods, with a focus on the first-order method. In this project, we have considered two types of ordinary differential equations which are stiff and non-stiff. Our research will contribute to the development of numerical methods for solving initial value problems and provide insights into the importance of choosing an appropriate step-size. In the current work, we have proposed a technique of solving nonlinear ordinary differential equations using numerical methods. We provide a study on the numerical solutions of nonlinear ordinary differential equations using the popular adaptive Runge-Kutta-Fehlberg (RKF) method.

4. LITERATURE REVIEW

Maximum work has been done on Runge-Kutta method by different researchers since its inception (James, 2013). In mathematical science, the Runge-Kutta method serves as a great tool in solving complicated ordinary differential equation and it was said that this method was established for the development of "differential equation" from Calculus, which itself was independently invented by English physicist Isaac Newton and German mathematician Gottfried Wilhelm Leibniz (Rajesh, 2019). A numerical technique for solving ordinary differential equations (ODEs) is the Runge-Kutta method. It has become one of the most widely used methods for solving ODEs in engineering, physics, and other scientific fields. In Runge-Kutta method, the fundamental notion is to approximate an ODE's solution at discrete times. The method uses a Taylor series expansion to approximate the change in the solution over a small time-step and then use this approximation to estimate the solution at the next time-step (Press, et al., 2007). The process is repeated to produce a sequence of approximations that converge to the true solution of the ODE. There are several variations of the Runge-Kutta method, including the classical fourth-order Runge-Kutta method which is the most used version of the method. Other variants include the second-order Runge-Kutta method and the high order Runge-Kutta methods, which use more terms in the Taylor series expansion to achieve higher accuracy. The Runge-Kutta method has several advantages over other methods for solving ODEs (Deuflhard, 2004), including its simplicity and ease of implementation, along with its ability to handle a wide range of ODEs, and the ability to preserve stability and accuracy even in the presence of large time-steps or nonlinearity. However, the method also has some limitations, including its sensitivity to the choice of time-step and its requirement for accurate initial conditions (Iserles, 2009). In recent years, there has been growing interest in using the Runge-Kutta method for solving ODEs in large-scale scientific simulations, such as climate models and astrophysical simulations. The method has also been applied to the study of chaotic dynamics, where its ability to handle nonlinear ODEs and its ability to preserve stability and accuracy make it well suited to the study of complex, nonlinear systems (Stoer and Bulirsch, 2002). The first of these three methods is the mid-point rule adapted to ordinary differential equations, while the second and third methods are different versions of the Trapezoidal rule. The last of these methods suggests interactive computation of the stage values (Breen, 2018). The Runge-Kutta method extended the approximation method of earlier to a more elaborate scheme which was capable of greater accuracy. The plan was to apply an initial value problem's solution in a series of tiny increments. In each step, the rate of change of the solution is treated as constant and is found from the formula for the derivative evaluated at the beginning of the steps for the equation $y'(x) = f(x, y(x))$, with given initial value $y(x_0) = y_0$. The first step is from the initial x_0 to a slightly larger value x_1. The approximate solution at this point is taken to be:

$y_1 = y_0 + (x_1 - x_0) f(x_0, y_0)$.

In general, for a sequence of time values solution approximations $y_0, y_1, y_2 \ldots$ are given by:

$y_n = y_{n-1} + (x_n - x_{n-1}, y_{n-1})$.

Years after, more researchers had stood up to the course of Runge-Kutta method to make it better and more efficient. Some special Runge Kutta formula have been developed and implemented for boundary-value problems by Enright and Muir (1984–1986) (Shampine, 2000). Sharp (1987–1989) worked on the Runge-Kutta-Nystrom integrator for second order initial value problems, whereas Chan and Jackson (1986) used iterative linear equation solvers in coding for huge systems of stiff initial value problems and he analysed new low explicit Runge-Kutta pairs respectively. Investigations related to the assessment and comparison of numerical method have continued. Enright (1989-1991) analysed error control strategies (Ababneh 2009).

5. METHODOLOGY

A first-order Runge-Kutta method (RK1) is used for solving a first order differential equation (initial value problem) (Arqub, 2014). In this method, each successive value(s) y_{n+1} is obtained from information from the immediate proceeding value(s) y_n (Abbasbandy, 2002). There are many different numerical methods for solving first-order differential equations which are as below:

- Euler's method
- Runge-Kutta methods
- Adams-Bashforth methods
- Adams-Moulton methods

The choice of which method to use depends on the specific equation being solved, as well as the desired accuracy and stability of the solution. The numerical methods that mentioned, such as Runge-Kutta method, are very effective for solving first-order differential equations. They can achieve high accuracy with relatively few steps, and they are also relatively stable.

Therefore, they are often the preferred choice for solving first-order differential equations.

Let it be required to find an approximation solution of the differential equation:

$$dy/dx = y' = f(x, y) \qquad \text{----------- (2)}$$

satisfying the initial condition y (x_0) = y_0

This problem is called the Cauchy problem. Numerical solution of the Cauchy problem consists in calculating approximate values of x_1, x_2, x_3.......... x_n and the corresponding values of y_1, y_2, y_3 .... yn.

Generally, $x_i = x_0 + i\,h$ where i = 1, 2, 3…. n. (3)

The points x_i are called grid nodes and h is the grid step, and 0 < h< 1.

Among the numerical methods for solving ordinary differential equation the most famous are the methods of Euler and of Runge-Kutta. These methods belong to the group of one step methods in which to calculate the point information is required only about the last calculated point Y_i. The Runge-Kutta methods are an important family of predictor corrector methods for approximation of solutions of ODEs. Let us consider an initial value problem (IVP) $u'(t) = f(t, u(t))$ where $(u_1(t), u_2(t) \ldots.\ u_n(t))^T$, f ∈ $[a, b] \times R^n \rightarrow R^n$ to solve numerically approximate the continuously differentiable equations of the IVP. Over the time interval, we subdivide the interval into equal sub intervals and select the mesh points

$t_j = a + jh$, j = 0, 1, ...N and h = (b – a)/N where h is called step-size.

The family of explicit Runge-Kutta (RK) methods of the m^th stage is given by:

$$t_{n+1} = u_{n+1} = u_n + \sum_{i=1}^{m} c_i k_i \qquad \text{----------- (4)}$$

$k_1 = (t_n, u_n)$

$k_2 = (t_n + \alpha_2 h u_n + h\beta_{21} k_1(t_n, u_n)$

$k_3 = (t_n + \alpha_3 h u_n + h\{\beta_{31} k_1(t_n, u_n) + \beta_{32} k_2(t_n, u_n)\}$

and, in general $k_m = f(t_n + \alpha_m h + u_n + h \sum_{i=1}^{m-1} \beta_{mj} k_j)$

Since the exact solution of the Cauchy problem is often unknown, the Runge's rule or the double recalculation rule is used to estimate the error from the method: the calculation is repeated with a step h/2 and the absolute difference is calculated:

$\left| \dfrac{u_n^h - u_n^{h/2}}{2^p - 1} \right|$ where u_n^h is the value of the function at the point t and step h and $u_n^{h/2}$ is the value of the function at the point u_n and step h/2. p is the order of the method (the RK4 method is a fourth order method) meaning that the error per step is of the order of h^5 while the total accumulated error has order h^4.

The error of the method is estimated using the following formula:

$$\max \left| \frac{u_n^h - u_n^{h/2}}{2^p - 1} \right| \quad \text{for } i = 0,1,2\ldots\ldots\ldots$$

6. Description of the Method

The adaptive step-size control algorithm proceeds in the following manner: in each method we must find the value of u at the step-size h, then calculate at the step-size h/2.

The error estimation is max:

Next, at each step two attained approximation are compared, difference between these approximations gives the per step error ε. The numerical results are presented by depicting two numerical solution profiles (by Runge-Kutta method and Runge's rule) along with the profile of exact solution. The comparative error estimate for two numerical solutions have also been presented. Additionally, we have presented the progression of both numerical solution and the absolute error.

7. RESULT AND DISCUSSION

Here, we demonstrated the efficiency of first order Runge-Kutta (RK1) method in solving the given differential equation. This ODE is solved with three different step-sizes. The solutions thus obtained are compared with the exact solution of the equation to ascertain the level of accuracy regarding each step-size.

To obtain the approximate value of the solution (y) for the initial value problem, we first solve the given equation for the exact equation.

$u' = 0.2u + t$ ------------ (5)

u (0) = 1 [0, 0.2] in 0.1 increments

the equation is: $u' - 0.2u = t$

I.F. = $e^{\int p\, dt}$

On comparing the above equation with we get P = – 0.2 and Q = t

∴ I.F. $= e^{-\int 0.2 dt} = e^{-0.2t}$

The solution of the linear equation is

$u\, e^{-\int 0.2 dt} = c + \int t . e^{-0.2t} dt$

$u\, e^{-\int 0.2 dt} = c + 5(5 - t)\, e^{-0.2t}$

$u = c\, e^{0.2t} + 5(5 - t)$

Now, to find the value of c, we use u (0) = 1 in $u = c\, e^{0.2t} + 5(5 - t)$ and get, c =– 24.

∴ $u = -24e^{0.2t} + 5(5 - t)$

The exact solution for this problem is $u = -24e^{0.2t} + 5(5 - t)$, and we are interested in the value of u for $0 \le t \le 0.2$.

We first solve the problem using the second order Runge-Kutta method with h = 0.1 and from t = 0 to t = 0.2, with step size h = 0.1. It takes 2 steps: $t_0 = 0$, $t_1 = 0.1$, $t_2 = 0.2$

Step 0: $t_0 = 0$ $u_0 = 1$.

Step 1: $t_1 = 0.1$ $u_1 = 1.106$

Step 2: $t_2 = 0.2$ $u_2 = 1.22523$

Now let's compare what we got with the exact solution.

Consider the numerical solution of the Cauchy problem

u' = 0.2 u + t, u (0) = 1 [0, 0.2] in 0.1 increments

To solve this problem, we use the second order Runge-Kutta method. The estimation of errors the method will be carried out based on the Runge's rule.

Table 1: The comparison of second order Runge-Kutta method and Runge's rule.

Step Size	0.00	0.05	0.1	0.15	0.2
Value by RK2 Method	1.0000		1.106		1.22523
Value by Runge's Rule	1.0000	1.05375	1.11077	1.17121	1.176713
Exact Value	1.0000	1.051525	1.116205	1.18420	1.25568

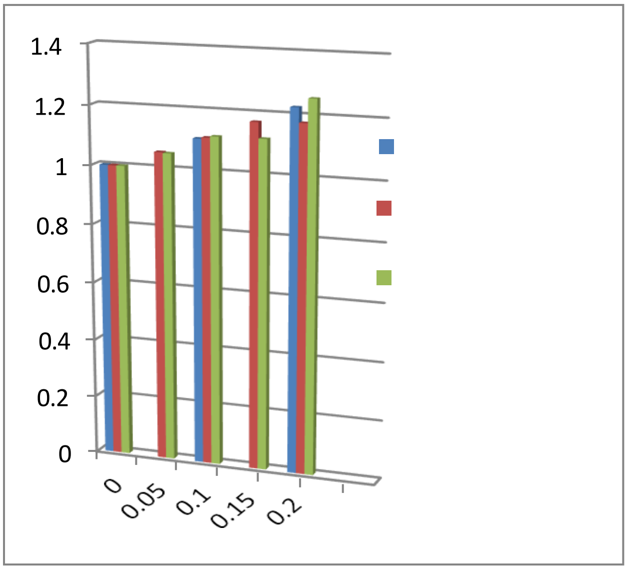


Fig 1(a) Behaviour of the exact and numerical solution

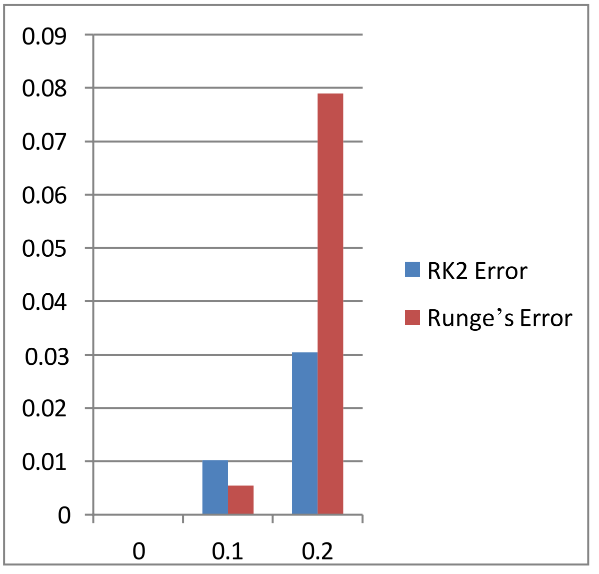


Fig. 1(b) Absolute error for the second order

The figure 1(a) depicts the behaviour of the exact and numerical solution of eq (5) and figure 1(b) shows the absolute error for the second order Runge-Kutta method and Runge's rule between numerical solutions and the exact solution.

The maximum absolute error by RK2 method is 0.03045 and 0.078967 by Runge's rule.

Table 2: Comparison for number of steps between third order Runge-Kutta method and Runge's rule.

Step Size	0.00	0.05	0.1	0.15	0.2
Value by RK3 Method	1.0000		1.1062		1.225134
Value by Runge's Rule	1.0000	1.053125	1.162845	1.225596	1.2694800
Exact Value	1.0000	1.051525	1.116205	1.18420	1.25568

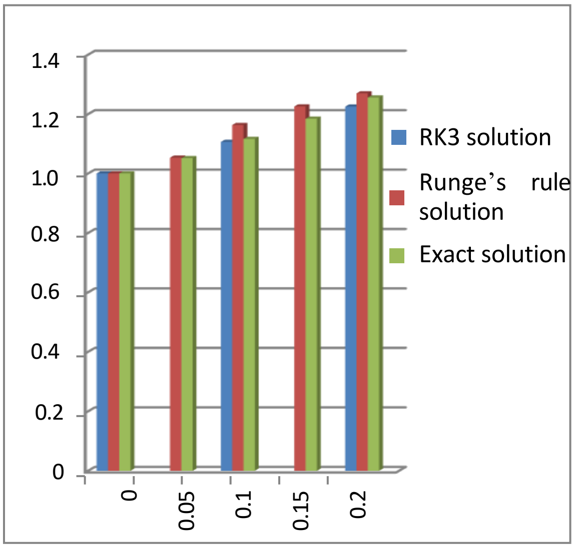


Fig 2(a) Behaviour of the exact and numerical solution

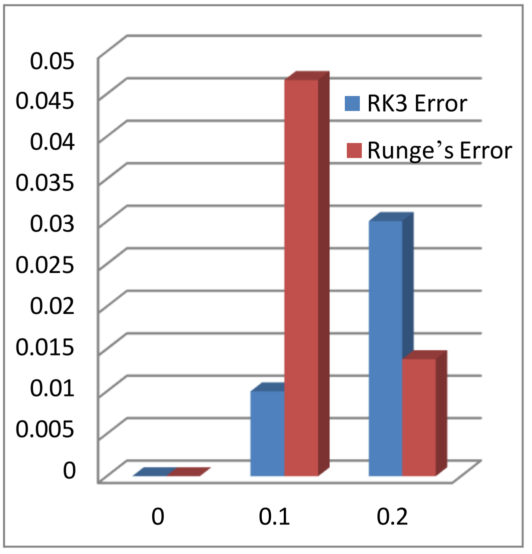


Fig 2(b) The absolute error for the fourth order

The figure 2(a) depicts the behaviour of the exact and numerical solution of eq (5) and figure 2(b) shows the absolute error for the fourth order Runge-Kutta method and Runge's rule between numerical solutions and the exact solution.
The maximum absolute error by third order Runge-Kutta method is 0.03045 and 0.078967 by Runge's rule.

Table 3: Comparison for number of steps between fourth order Runge-Kutta method and Runge's rule.

Step Size	0.00	0.05	0.1	0.15	0.2
Value by RK4 method	1.0000		1.1062		1.225134
Value by Runge's rule	1.0000	1.053125	1.162845	1.225596	1.2694800
Exact Value	1.0000	1.051525	1.116205	1.18420	1.25568

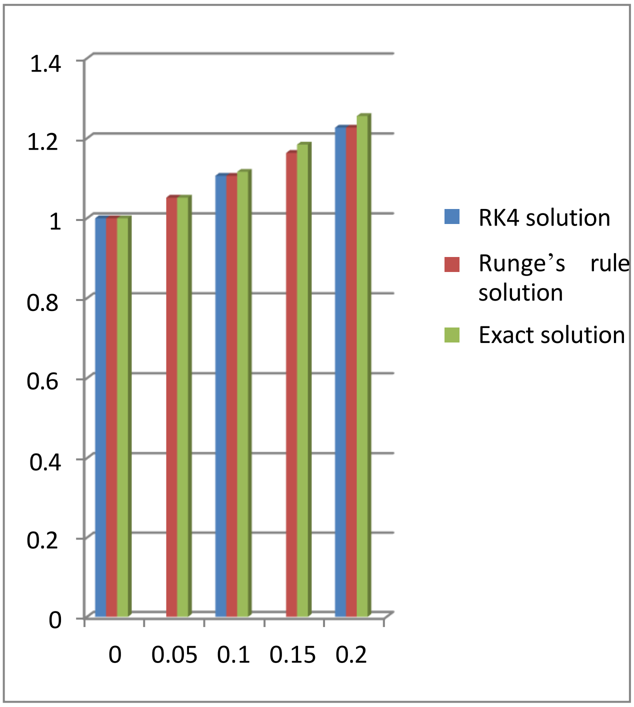


Fig 3(a) Behaviour of the exact and numerical solution

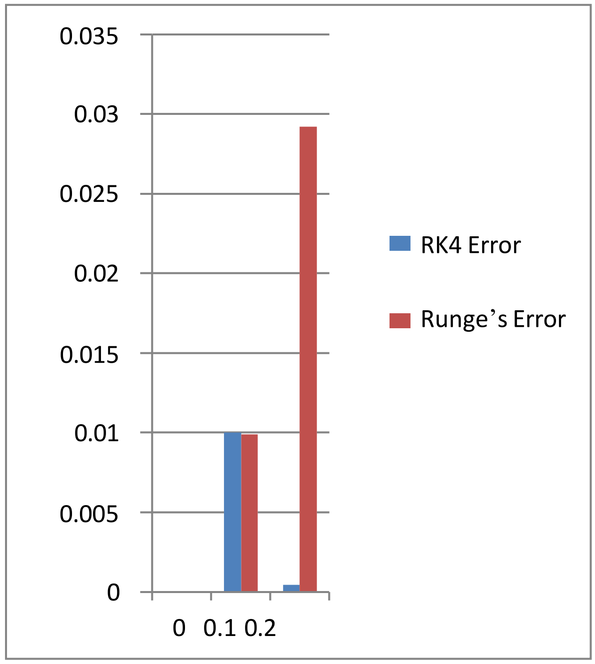
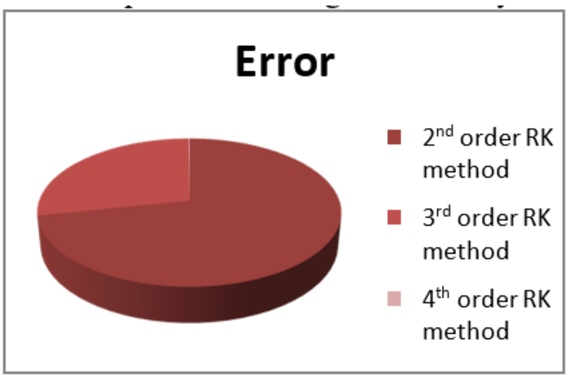


Fig 3 (b) Absolute error for the fourth order

The figure 3(a) depicts the behaviour of the exact and numerical solution of eq (5) and figure 3(b) shows the absolute error for the fourth order Runge-Kutta method and Runge's rule between numerical solutions and the exact solution.
The maximum absolute error by fourth order Runge-Kutta method is 0.000462 and 0.02892 by Runge's rule.

Comparison of Errors in Solving ODEs Using Runge-Kutta Method and Runge's Rule
From the pie chart and the adjacent table, the fourth order Runge-Kutta Method provides the highest accuracy



Method	Error
Second order Runge-Kutta method	0.0161723
Third order Runge-Kutta method	0.006335
Fourth order Runge-Kutta method	0.0000329

Fig 4 Comparison of Errors in Solving ODEs Using Runge-Kutta Method and Runge's Rule

References

[1]. O. A. Arqub and M. Al-Smadi, "Numerical Algorithm for Solving Two-Point, Second Order Periodic Boundary Value Problems for Mixed Integro-Differential Equations," Appl. Math. Comput., vol. 243, pp. 911–922, 2014.
[2]. Edeo, "Solution of Second Order Linear and Nonlinear Two Point Boundary Value Problems Using Legendre Operational Matrix of Differentiation," Amer. Sci. Res. J. Eng. Technol. Sci. (ASRJETS), vol. 51, no. 1, pp. 225–234, 2019.
[3]. Yu. N. Gorelov, Numerical Methods for Solving Ordinary Differential Equations (Runge-Kutta methods). Samara, Russia: Samara National Research University, 2006, p. 45.
[4]. J. F. Epperson, An Introduction to Numerical Methods and Analysis. Hoboken, NJ: Wiley, 2013.
[5]. Iserles, A First Course in the Numerical Analysis of Differential Equations, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2009.
[6]. S. Arumugan, I. A. Thangapandi, and A. Somasundaram, Numerical Solutions of Ordinary Differential Equations: Numerical Methods. Chennai, India: Scitech Publications Pvt. Ltd., 2005.
[7]. J. Stoer and R. Bulirsch, Introduction to Numerical Analysis, 2nd ed. New York, NY: Springer-Verlag, 2002.
[8]. L. F. Vakhlaeva and T. V. Molodenkovs, Numerical Methods for Solving Grid Equations. Saratov, Russia: Saratov National Research State University, 2003, p. 46.