Turkish Journal of Science & Technology
20(1), 141-158, 2025

Research Paper
https://doi.org/10.55525/tjst.1433935

An Investigative Comparison of Higher-Order Runge-Kutta Techniques for Resolving First-Order Differential Equations

Khadeejah James AUDU[1*], Victor James UDOH[2], Jamiu GARBA[3]
[1,2,3] Department of Mathematics, School of Physical Science, Federal University of Technology, Minna, Nigeria.
[*1.] k.james@futminna.edu.ng, [2] udoh.m1701594@st.futminna.edu.ng, [3]jamiu.garba@st.futminna.edu.ng

(Geliş/Received: 08/02/2024; Kabul/Accepted: 14/07/2024)

Abstract: In the context of solving first-order ordinary differential equations (ODEs), this paper thoroughly compares various higher-order Runge-Kutta methods. Reviewing the effectiveness, precision, and practicality of several Runge-Kutta schemes and highlighting their usage in numerical approximation is the main goal of the research. The study explores traditional approaches, including the fifth-order, six-stage Runge-Kutta (RK56), the sixth-order, seven-stage Runge-Kutta (RK67), and the seventh-order, nine-stage Runge-Kutta (RK79), with the goal of offering a comprehensive comprehension of their individual advantages and disadvantages. In order to help academics and practitioners choose the best approach based on the features of the problem, comparative benchmarks are constructed, utilizing both theoretical underpinnings and real-world implementations. Robustness evaluations and sensitivity analysis complement the comparison research by illuminating how flexible these techniques are in various context. The results of this study provide important new understandings of how higher-order Runge-Kutta methods function and provide a thorough manual for applying them to solve first-order differential problems in a variety of scientific and engineering fields. The study's examination of three higher order Runge-Kutta algorithms reveals that the RK56 is more effective at solving first order ODEs.

Keywords: Runge-Kutta technique, ordinary differential equations, numerical integration, error analysis, computational comparison.

Birinci Dereceden Diferansiyel Denklemlerin Çözümü için Yüksek Dereceli Runge-Kutta Yöntemlerinin Karşılaştırmalı Araştırması

Öz: Birinci dereceden adi diferansiyel denklemlerin (ODE'ler) çözümünde, bu makale çeşitli yüksek dereceli Runge-Kutta yöntemlerini kapsamlı bir şekilde karşılaştırmaktadır. Araştırmanın ana amacı, çeşitli Runge-Kutta şemalarının etkinliğini, doğruluğunu ve uygulanabilirliğini gözden geçirmek ve bunların sayısal yaklaşımlarda kullanımını vurgulamaktır. Çalışma, beşinci dereceli, altı aşamalı Runge-Kutta (RK56), altıncı dereceli, yedi aşamalı Runge-Kutta (RK67) ve yedinci dereceli, dokuz aşamalı Runge-Kutta (RK79) gibi geleneksel yaklaşımları araştırmakta olup, bu yöntemlerin bireysel avantaj ve dezavantajlarına dair kapsamlı bir anlayış sunmayı amaçlamaktadır. Akademisyenler ve uygulayıcıların, problemin özelliklerine göre en uygun yaklaşımı seçmelerine yardımcı olmak için teorik temeller ve gerçek dünya uygulamaları kullanılarak karşılaştırmalı ölçütler oluşturulmuştur. Dayanıklılık değerlendirmeleri ve hassasiyet analizleri, bu tekniklerin farklı bağlamlardaki esnekliğini aydınlatarak karşılaştırma araştırmasını tamamlamaktadır. Bu çalışmanın sonuçları, yüksek dereceli Runge-Kutta yöntemlerinin nasıl çalıştığına dair önemli yeni anlayışlar sunmakta ve bu yöntemlerin çeşitli bilim ve mühendislik alanlarında birinci dereceden diferansiyel problemleri çözmek için uygulanması konusunda kapsamlı bir kılavuz sağlamaktadır. Üç yüksek dereceli Runge-Kutta algoritmasının incelenmesi, RK56'nın birinci dereceden ODE'leri çözmede daha etkili olduğunu ortaya koymaktadır.

Anahtar kelimeler: Runge-Kutta tekniği, adi diferansiyel denklemler, sayısal entegrasyon, hata analizi, hesaplamalı karşılaştırma.

1. Introduction

This study aims to investigate the role those differential equations more specifically, ordinary differential equations, or ODEs have in determining the relationships between functions and their derivatives. When it comes to studying dynamic processes in phenomena like quantum mechanics, population modeling, and ecological

[*] Corresponding author: k.james@futminna.edu.ng. ORCID Number of authors: [1]0000-0002-6986-3491,[2]0009-0009-9319-6986 [3]0000-0002-2850-7730

interactions, ODEs are essential in many scientific domains, including physics, engineering, and the social sciences. The objective of the research is to categorize and contrast approaches to solving ODEs, highlighting their effectiveness and supporting mathematicians and researchers in making decisions when faced with these equations [1,2].

Higher-order Runge-Kutta techniques play a pivotal role in numerical analysis, particularly in resolving first-order differential equations with precision and efficiency. Despite their widespread use, there exists a need for a comprehensive comparative investigation to discern the performance discrepancies among these techniques. This research aims to bridge this gap by conducting an investigative comparison of various higher-order Runge-Kutta methods. The motivation stems from the critical importance of accurately solving first-order differential equations across diverse scientific and engineering domains. By systematically analyzing the strengths and weaknesses of different methods, this study seeks to identify the most effective approach for achieving accurate numerical solutions. The novelty lies in the comprehensive evaluation of multiple higher-order Runge-Kutta techniques within a unified framework, shedding light on their comparative performance and offering insights into their applicability in real-world scenarios. Ultimately, the contribution of this research lies in advancing the understanding of numerical techniques for solving first-order differential equations, thereby facilitating more informed decision-making in practical problem-solving contexts.

Higher-order Runge-Kutta methods, a particular class of numerical techniques are emphasized for their precision in solving ODEs, offering crucial instruments for comprehending and forecasting actual occurrences [3]. A thorough comparative analysis of higher-order Runge-Kutta techniques for first-order differential problem solving is covered in the study’s second section. Although differential equations are widely used in many different contexts, the study highlights the value of numerical solutions, particularly when dealing with complicated real-world systems [4,5]. By assessing and contrasting the performance of several higher-order Runge-Kutta procedures, the study seeks to advance existing knowledge by providing an understanding of their advantages and disadvantages. Important information for academics, practitioners, and educators is provided by highlighting the possible consequences for future numerical analytic applications and improvements. First-order differential problem solving is emphasized because it is important in many areas of science, engineering, and mathematics and because it provides the framework for simulating and interpreting real-world processes in many different domains [6].

Numerous investigations have been undertaken regarding Runge-Kutta methods. [7] conducted a Comparative Analysis of Runge-Kutta Methods for Solving Ordinary Differential Equations, providing insights into their numerical performance and computational efficiency. [8] reviewed Higher-Order Runge-Kutta Methods in Scientific Computing, addressing advancements and challenges in this field. [9] evaluated Runge-Kutta Techniques in Atmospheric Modeling, focusing on their accuracy in capturing atmospheric processes. [10] compared Runge-Kutta Methods for Solving Heat Transfer Equations in Engineering Applications, aiming to model heat transfer phenomena effectively. [11] surveyed Runge-Kutta Methods for Solving Chemical Reaction Kinetics, emphasizing their importance in chemical engineering applications. The authors in [12] introduced a novel approach aimed at tackling Ordinary Differential Equation (ODE) problems. Runge-Kutta methods are also used for solving partial differential equations (PDEs) by numerically integrating them over time. [13-17] show that these methods approximate the solution of the differential equations by iteratively advancing the solution from one time step to the next. By developing a sixth-stage fifth-order method, the solutions achieved notably enhanced accuracy and minimized error levels when dealing with initial value problems. [18] establishes and derives a Runge-Kutta method of the sixth order, employing seven stages to facilitate precise numerical approximation. Utilizing Butcher’s table, the researcher constructs a non-linear equation system, which is subsequently solved to determine the values of all relevant parameters. Finally, the reduction formula for the Runge-Kutta seventh order with nine steps method is derived [19]. These studies collectively contribute to understanding the theoretical foundations, numerical properties, and practical applications of Runge-Kutta methods across scientific and engineering disciplines. [20] studied different techniques on resolving linear differential equations.

The purpose of this study is to conduct a comprehensive comparative analysis of higher-order Runge-Kutta techniques for resolving first-order differential equations. This study aims to achieve several objectives in its investigation of higher-order Runge-Kutta techniques for resolving first-order differential equations. Firstly, it seeks to categorize and describe various higher-order Runge-Kutta methods to provide a comprehensive overview of available numerical techniques. Secondly, the study endeavors to conduct numerical experiments to compare the accuracy and stability of these techniques, thus facilitating a thorough assessment of their performance. Thirdly, it aims to evaluate the computational efficiency of each method, considering factors such as runtime and memory requirements. Finally, the study aims to synthesize its findings into actionable recommendations, guiding practitioners in selecting the most suitable higher-order Runge-Kutta technique for resolving first-order differential equations in diverse scientific and engineering applications. Through these objectives, the study endeavors to contribute to the advancement of numerical methods in the field of differential equations and facilitate informed decision-making among researchers and practitioners.

The study will provide clear recommendations for selecting the most effective higher-order Runge-Kutta technique based on its comparative analysis. Additionally, it aims to highlight innovative insights gleaned from the research process, particularly regarding the performance and applicability of different numerical methods in resolving first-order differential equations. The research study under review focuses on conducting a comparative analysis of higher-order Runge-Kutta techniques for resolving first-order differential equations. It emphasizes the precision of these numerical techniques in solving ordinary differential equations (ODEs) and their significance in understanding real-world phenomena. The study aims to advance existing knowledge by assessing and contrasting the performance of various higher-order Runge-Kutta procedures, providing insights into their advantages and disadvantages. It highlights the importance of numerical solutions, particularly in complex real-world systems, and offers valuable information for academics, practitioners, and educators regarding potential consequences and improvements for future numerical analytic applications.

Comparing this study with similar studies in the literature, it is evident that numerous investigations have been undertaken regarding Runge-Kutta methods in various scientific and engineering disciplines. For instance, previous studies have conducted comparative analyses of Runge-Kutta methods for solving ordinary differential equations, reviewed higher-order Runge-Kutta methods in scientific computing, evaluated their accuracy in capturing atmospheric processes, and compared their effectiveness in modeling heat transfer phenomena and chemical reaction kinetics. Some studies have also introduced novel approaches, such as sixth-stage fifth-order methods and sixth-order Runge-Kutta methods employing seven stages, to enhance accuracy and precision in solving ODE problems. Overall, while previous studies have contributed to understanding the theoretical foundations, numerical properties, and practical applications of Runge-Kutta methods across different domains, the research study under review provides a focused investigation specifically on higher-order Runge-Kutta techniques for resolving first-order differential equations. By synthesizing existing knowledge and conducting a comparative analysis within a unified framework, the study aims to offer valuable insights and recommendations for selecting the most suitable numerical technique for practical problem-solving contexts.

2. Methodology

2.1 The Fifth Order Six-Step Runge-Kutta (RK56) Technique

The fifth-order six-step Runge-Kutta (RK56) Scheme is a numerical method utilized for solving ordinary differential equations (ODEs). It progresses the solution through six stages, where computations are based on derivatives of the function being solved. Notably, RK56 achieves fifth-order accuracy, indicating a significant reduction in global error with each step, typically proportional to the fifth power of the step size. This scheme strikes a balance between accuracy and computational cost, making it suitable for a variety of practical applications. Implementation involves iteratively computing solution values at discrete points using weighted averages of function values at different stages. By carefully selecting weights and stage values, RK56 achieves a desirable balance between accuracy and computational efficiency, rendering it valuable for numerical simulations and analysis [21]. The outlined procedure for obtaining the fifth-order sixth-stage Runge-Kutta formula are;

i. Obtain a sixth-stage, fifth-order method from the general Runge-Kutta approach
ii. Obtain the Taylor series expansion of $k_{i's}$ about the point $(n_c), i = 1,2,\ 3,\ 4,\ 5,\ 6$
iii. Carry out substitution to ensure that all $k_{i's}$ are in terms of k_1 only
iv. Reducing all the $k_{i's}$ in terms of k_1 and substituting into the increment function, $\phi(n_c, d) = \sum_{i=1}^{6} e_i k_i$,
v. By comparing the coefficients of all partial derivatives of y with the fifth-order Taylor series expansion involving only partial derivatives concerning n.

According to [12], a new fifth-order sixth-stage explicit Runge-Kutta formula will be obtained after some simplification, as shown in Equation 1,2.

$$s_{c+1} - s_c = \frac{d}{144}(14k_1 + 48k_2 + 162k_3 + 33k_4 - 125k_5 + 12k_6) \tag{1}$$

Where;

$$k_1 = f(z_c, s_c)$$
$$k_2 = f\left(z_c + \frac{1}{3}d,\ s_c + \frac{1}{3}dk_1\right)$$
$$k_3 = f\left(z_c + \frac{2}{3}d, s_c + \frac{2}{3}dk_2\right)$$
$$k_4 = f\left(z_c + \frac{1}{3}d,\ s_c + d\left(-\frac{167765027}{45900120}k_1 + \frac{43549}{7217}k_2 - \frac{30361}{14840}k_3\right)\right)$$
$$k_5 = f\left(z_c + \frac{3}{5}d, s_c + d\left(-\frac{516388549921283}{28366716018615}k_1 + \frac{35525}{9169}k_2 - \frac{27646}{19955}k_3 - \frac{10643}{155037}k_4\right)\right)$$
$$k_6 = f\left(z_c + d, s_c + d\left(-\frac{9039268043}{1401332565}k_1 + \frac{736810}{53619}k_2 - \frac{28702}{5227}k_3 - \frac{7}{5}k_4 + \frac{3}{5}k_5\right)\right) \quad (2)$$

2.1.1 Implementation Procedure for RK56 Technique

Step 1: Express the function $f(z,s)$ such that $f(z,s) \in (p,q)$.
Step 2: Provide the initial estimate for z_0 and s_0.
Step 3: Choose the desired step size $h = \frac{q-p}{i}$, where i is number of steps.
Step 4: Input $p,\ q,\ z_0,\ s_0,\ i$.

Step 5: for c from 1 to i

Compute $k_1,\ k_2,\ k_3, k_4, k_5\ and\ k_6$ as denoted in the RK56 method.

Step 6: Set $z_{c+d} \rightarrow z_c$, then compute, using Equation 3.

$$s_{c+1} = s_c + \frac{d}{144}(14k_1 + 48k_2 + 162k_3 + 33k_4 - 125k_5 + 12k_6) \quad (3)$$

Step 7: Output z_0 and s_0.

Step 8: End the process if $z_c \geq h$ such that $\|s_{c+1} - s_c\| < \varepsilon$.

2.2 The Sixth Order Seven-Step Runge-Kutta (RK67) Technique

The Sixth Order Seven-Step Runge-Kutta (RK67) Scheme is a highly accurate numerical method for solving ordinary differential equations, progressing the solution through seven stages with sixth-order accuracy and a balance between precision and computational efficiency. Considering the first-order differential equation $z'(s) = \frac{dz}{ds} = f(z,s)$, we introduce the initial value $s(z_0) = s_0$, and prioritize our objectives in this case to finding the absolute solution of $s(z)$, the sixth order with seven stages Runge-Kutta method is employed to evaluate n_{c+1} as an approximation to $s(z_{c+1}) = s(z_c + d)$ without the loss of generality. If the function f does not depend on z but only of s, then by setting $z' = 1$, then, the equation above reduces to the relation shown in Equation 4.

$$s'(z) = f(s(z)), s(z_0) = s_0 \quad (4)$$

The suggested explicit Runge-Kutta method of sixth order with seven stages, denoted by $k_1, \ldots, k_7$ for one step, entails that, following this approach, the solution to equation (3) at the end of the first step can be calculated using Equation 5.

$$s_1 = s_0 + d\sum_{i=1}^{7} e_i k_i \quad (5)$$

And the exact solution of equation (4) can be calculated using Equation 6.

$$S_i = s(z_0 + d\sum_{j=1}^{7} a_{ij}) + O(d^2) \quad (6)$$

Thus, RK67 can be calculated using Equation 7,

$$s_{c+1} = s_c + \frac{d}{200}(13k_1 + 55k_3 + 55k_4 + 32k_5 + 32k_6 + 13k_7) \tag{7}$$

where ;

$$\begin{aligned}
k_1 &= f(z_c, s_c) \\
k_2 &= f\left(z_c + \frac{1}{3}d, s_c + \frac{1}{3}dk_1\right) \\
k_3 &= f\left((z_0 + \frac{2}{3}d, s_0 + \frac{2}{3}dk_2\right) \\
k_4 &= f\left((z_0 + \frac{1}{3}d, s_0 + d\left(\frac{1}{12}k_1 + \frac{1}{3}k_2\frac{1}{12}k_3\right)\right) \\
k_5 &= f\left((z_0 + \frac{5}{6}d, s_0 + d\left(\frac{25}{48}k_1 - \frac{55}{24}k_2 + \frac{35}{48}k_3 + \frac{15}{8}k_4\right)\right) \\
k_6 &= f\left((z_0 + \frac{1}{6}d, s_0 + d\left(\frac{3}{20}k_1 - \frac{11}{20}k_2 - \frac{1}{8}k_3 + \frac{1}{2}k_4 + \frac{1}{10}k_5\right)\right) \\
k_7 &= f\left((z_0 + d, s_0 + d\left(-\frac{261}{260}k_1 + \frac{33}{13}k_2 + \frac{43}{156}k_3 + \frac{118}{79}k_4 + \frac{32}{195}k_5 + \frac{80}{39}k_6\right)\right)
\end{aligned} \tag{8}$$

2.2.1 Implementation Procedure for RK67 Method

Step 1: Express the function $f(z, s)$ such that $f(z, s) \in (p, q)$.

Step 2: Provide the initial estimate for z_0 and s_0.
Step 3: Choose the desired step size $h = \frac{q-p}{i}$, where i is number of steps.
Step 4: Input p, q, z_0, s_0, i.

Step 5: for c from 1 to i, compute $k_1, k_2, k_3, k_4, k_5, k_6$ *and* k_7 as denoted in the RK67 method, shown in Equation (8,9).
Step 6: Set $z_{c+d} \rightarrow z_c$, then compute

$$s_{c+1} = s_c + \frac{d}{200}(13k_1 + 55k_3 + 55k_4 + 32k_5 + 32k_6 + 13k_7) \tag{9}$$

Step 7: Output z_0 and s_0

Step 8: End the process if $z_c \geq h$ such that $\|s_{c+1} - s_c\| < \varepsilon$

2.3 The Seventh Order Nine-Step Runge-Kutta (RK79) Technique

Considering the equation $z_{c+1} - z_c = +d\phi(z, s, d)$ and the relations shown in Equations 10, 11 and 12

$$d\phi(z_c, s_c, d) = \sum_{i=1}^{P} e_i k_i i = 1,2,3, \ldots 6 \tag{10}$$

$$k_1 = f(z_c, s_c), \quad k_i = f\left[z_c + g_i d, s_c + d\sum_{j=1}^{i-1} a_{ij} k_j\right], \; i = 2,3, \ldots .6 \tag{11}$$

$$g_i = \sum_{j=1}^{i-1} a_{ij}, i = 2,3, \ldots .6 \tag{12}$$

as the reduction formula of Runge-Kutta methods, the nine-step seventh order Runge-Kutta is shown in Equation 13,14.

$$s_{c+1} = s_c + \frac{41k_1 + 216k_4 + 27k_5 + 272k_6 + 27k_7 + 216k_8 + 41k_9}{840} \tag{13}$$

Where;

$$k_1 = f(z_c, s_c)$$
$$k_2 = f\left(z_c + \frac{1}{12}d, s_c + \frac{1}{12}dk_1\right)$$
$$k_3 = f\left(z_c + \frac{1}{12}d, s_c + d\left(\frac{-10k_1 + 11k_2}{12}\right)\right)$$
$$k_4 = f\left(z_c + \frac{2}{12}d, s_c + \frac{2k_3}{12}d\right)$$
$$k_5 = f\left(z_c + \frac{4}{12}d, s_c + d\left(\frac{157k_1 - 318k_2 + 4k_3 + 160k_4}{9}\right)\right)$$
$$k_6 = f\left(z_c + \frac{6}{12}d, s_c + d\left(\frac{-322k_1 + 199k_2 + 108k_3 - 131k_5}{30}\right)\right)$$
$$k_7 = f\left(z_c + \frac{8}{12}d, s_c + d\left(\frac{3158k_1}{45} - \frac{638k_2}{6} - \frac{23k_3}{2} + \frac{157k_4}{3} + \frac{157k_6}{45}\right)\right)$$
$$k_8 = f\left(z_c + \frac{10}{12}d, s_c + d\left(-\frac{53k_1}{14} + \frac{38k_2}{7} - \frac{3k_3}{14} - \frac{65k_5}{72} + \frac{29k_7}{90}\right)\right)$$
$$k_9 = f\left(z_c + d, s_c + d\left(\frac{56k_1}{25} + \frac{283k_2}{14} - \frac{119k_3}{6} - \frac{26k_4}{7} - \frac{13k_5}{15} + \frac{149k_6}{32} - \frac{25k_7}{9} + \frac{27k_8}{25}\right)\right) \tag{14}$$

2.3.1 Implementation Procedure for RK79 Technique

Step 1: Express the function $f(z,s)$ such that $f(z,s) \in (p,q)$.

Step 2: Provide the initial estimate for z_0 and s_0.

Step 3: Choose the desired step size $h = \frac{q-p}{i}$, where i is number of steps.

Step 4: Input $p,\ q,\ z_0,\ s_0, i$.

Step 5: for c from 1 to i,

Compute $k_1,\ k_2,\ k_3, k_4, k_5, k_6, k_7, k_8\ and\ k_9$ as denoted in the RK79 method.

Step 6: Set $z_{c+d} \rightarrow z_c$, then compute, using Equation (15)

$$s_{c+1} = s_c + \frac{41k_1 + 216k_4 + 27k_5 + 272k_6 + 27k_7 + 216k_8 + 41k_9}{840} \tag{15}$$

Step 7: Output z_0 and s_0

Step 8: End the process if $z_c \geq h\ such\ that\ \ \|s_{c+1} - s_c\| < \mathcal{E}$

2.4 Error Analysis

Numerical solutions of ordinary differential equations may encounter two types of errors: rounding errors and truncation errors. Rounding errors arise from the limited precision with which computers can represent integers, leading to discrepancies known as round-off errors. The computers fixed and restricted number of significant figures prevents the exact representation of certain numbers in memory. On the other hand, truncation errors in numerical analysis occur when approximations are employed to estimate a value, with the precision of the solution dependent on the chosen step size, denoted as 'h.' A numerical method is considered convergent when the solution approaches the exact solution as the step size (h) approaches zero [22].

In this investigation, a first-order initial value problem is examined to validate the accuracy of the proposed method. Subsequently, numerical approximations are sought for specific initial value problems using this approach [23]. The Maple software is employed to explore the estimated solutions for three selected numerical algorithms, each with varying step sizes.

The expression $t_c = |s(z_c) - s_c| < \varepsilon$ calculates the convergence of the initial value problems, where $s(z_c)$ signifies the approximate answer and s_c denotes the precise solution, is dependent on the problem and varies from 10^{-15}. These two formulas' faults are specified by the expression $errors = |n(m_c) - n_c|$.

3. Numerical Investigation

To validate the feasibility and performance of the three previously discussed algorithms, we introduce first-order ordinary differential problems for numerical exploration in this section. We compute the numerical solutions and absolute errors and provide a graphical representation of the computational results, which are obtained with the use of the MAPLE 2021 package. The particular numerical issues that are being looked into are detailed below:

Problem 1: Considering the numerical solutions of RK56, RK67 and RK79 for the first-order ordinary differential equation

$$\frac{ds}{dz} = z^2 + zs, \quad s(0) = 1$$

within interval $0 \le z \le 1, h = 0.1,$ Exact solution: $s(z) = \sqrt{\frac{\pi}{2}} e^{\frac{z^2}{2}} erf\left(\frac{z}{\sqrt{2}} + e^{\frac{z^2}{2}} - z\right)$

Problem 2: Considering the approximate solutions provided by RK56, RK67 and RK79 for a first-order ODE.

$s' = z(1+s), \quad z_0 = 0, \quad s_0 = 1, \quad h = 0.1,$ Exact solution: $s(z) = -1 + 2e^{\frac{z^2}{2}}$

Problem 3: We aim to solve the following non-linear first order ordinary differential equation

$\frac{ds}{dz} = -zs^2, \quad s_0 = 1,\ z_0 = 2,\ h = 0.1,$ Exact solution: $s(z) = e^{\frac{1}{4}z(z^3-6)}$

using RK56, RK67 and RK79.

Problem 4: We intend to address the given ODE by utilizing RK56, RK67 and RK79 techniques

$s' = z^2 - s, \quad s_0 = 1, \quad z_0 = 0, \quad h = 0.1,$ Exact solution: $s(z) = z^2 - 2z + 2 - e^{-z}$

Problem 5: $s' = -zs^2, s_0 = 1, z_0 = 2,\ h = 0.1,$ Exact solution: $s(z) = \frac{2}{z^2-2}$

Problem 6: $s' = -2zs^2, s_0 = 1,\ z_0 = 2,\ h = 0.1,$ Exact solution: $s(z) = \frac{1}{1+z^2}$

4. Results

Here, we apply the RK56, RK67, and RK79 methods to solve the starting value problems for a first-order ordinary differential equation. The first order ordinary differential equations discussed in the previous section are treated with these techniques. To demonstrate which of the numerical methods converges to the analytical solution more quickly, the three approaches are applied. The Maple progamming with its 2023 software version is used to compute the numerical solutions and errors. A Laptop of 8GB Ram, 2.7 GHZ processor and storage of 500GB, keyboard and mouse constitute the harware used for the computations.

Table 1. Computational solution for RK56, RK67, RK79 experiment 1.

x	Approx. Values for RK56 (s)	Approx. Values for RK67 (s)	Approx. Values for RK79 (s)	Precise Values(S)
0.0	1.000	1.000	1.000	1.000
0.1	1.005	1.005	1.005	1.005
0.2	1.023	1.023	1.023	1.023
0.3	1.055	1.055	1.054	1.055
0.4	1.105	1.105	1.102	1.105
0.5	1.177	1.177	1.168	1.177
0.6	1.275	1.275	1.258	1.275
0.7	1.404	1.405	1.374	1.404
0.8	1.572	1.574	1.520	1.572
0.9	1.787	1.791	1.703	1.787
1.0	2.059	2.066	1.928	2.059

Table 2. RK56, RK67, RK79 contrast errors for experiment 1.

x	Eror Values for RK56	Error Values for RK67	Error Values for RK79
0.0	0	0	0
0.1	2.092×10^{-6}	3.588×10^{-6}	0.116×10^{-5}
0.2	4.242×10^{-6}	7.026×10^{-6}	0.305×10^{-4}
0.3	6.478×10^{-6}	0.546×10^{-5}	0.134×10^{-3}
0.4	8.821×10^{-6}	0.170×10^{-4}	0.377×10^{-3}
0.5	0.113×10^{-5}	0.397×10^{-4}	0.849×10^{-3}
0.6	0.139×10^{-5}	0.794×10^{-4}	0.017×10^{-2}
0.7	0.166×10^{-5}	0.144×10^{-3}	0.302×10^{-2}
0.8	0.194×10^{-5}	0.246×10^{-3}	0.514×10^{-2}
0.9	0.223×10^{-5}	0.400×10^{-3}	0.835×10^{-2}
1.0	0.251×10^{-5}	0.628×10^{-3}	0.131

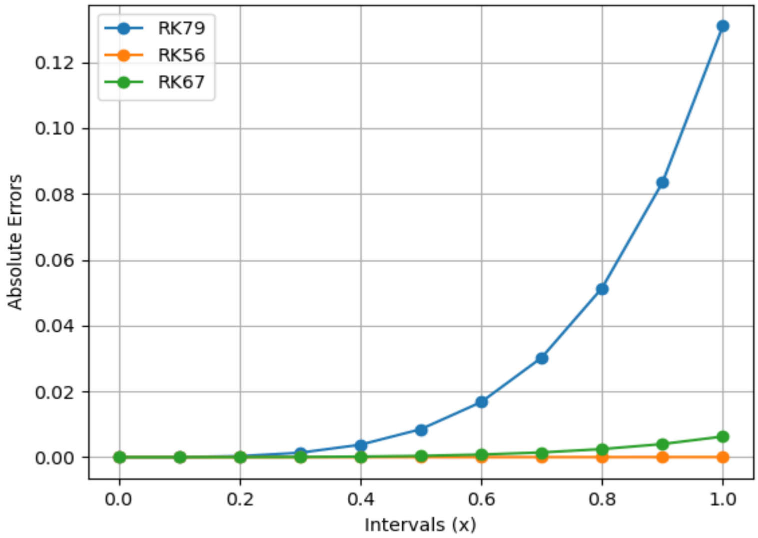

Figure 1. Plot showing contrast errors for experiment 1.

Figure 1 presents a comprehensive evaluation of the RK56, RK67, and RK79 methods as they are employed to solve experiment 1.

Table 3. Computational solution for RK56, RK67 and RK79 experiment 2.

x	Approx. Values for RK56 (s)	Approx. Values for RK67 (s)	Approx. Values for RK79 (s)	Precise Values (S)
0.0	1.0	1.0	1.0	1.0
0.1	1.010	1.010	1.010	1.010
0.2	1.040	1.040	1.040	1.040
0.3	1.092	1.092	1.090	1.092
0.4	1.167	1.167	1.161	1.167
0.5	1.266	1.267	1.254	1.266
0.6	1.394	1.396	1.371	1.394
0.7	1.555	1.557	1.514	1.555
0.8	1.754	1.758	1.687	1.754
0.9	1.999	2.004	1.894	1.999
1.0	2.297	2.305	2.139	2.297

Table 4. RK56, RK67, RK79 contrast errors for experiment 2.

x	Error Values for RK56	Error Values for RK67	Error Values for RK79
0.0	0	0	0
0.1	4.184×10^{-6}	6.097×10^{-6}	0.220×10^{-5}
0.2	8.471×10^{-6}	0.165×10^{-5}	0.548×10^{-4}
0.3	0.129×10^{-5}	0.103×10^{-4}	0.226×10^{-3}
0.4	0.177×10^{-5}	0.293×10^{-4}	0.597×10^{-3}
0.5	0.225×10^{-5}	0.636×10^{-4}	0.127×10^{-2}
0.6	0.277×10^{-5}	0.120×10^{-3}	0.248×10^{-2}
0.7	0.335×10^{-5}	0.207×10^{-3}	0.411×10^{-2}
0.8	0.394×10^{-5}	0.335×10^{-3}	0.669×10^{-2}
0.9	0.460×10^{-5}	0.522×10^{-3}	0.104
1.0	0.531×10^{-5}	0.789×10^{-3}	0.158

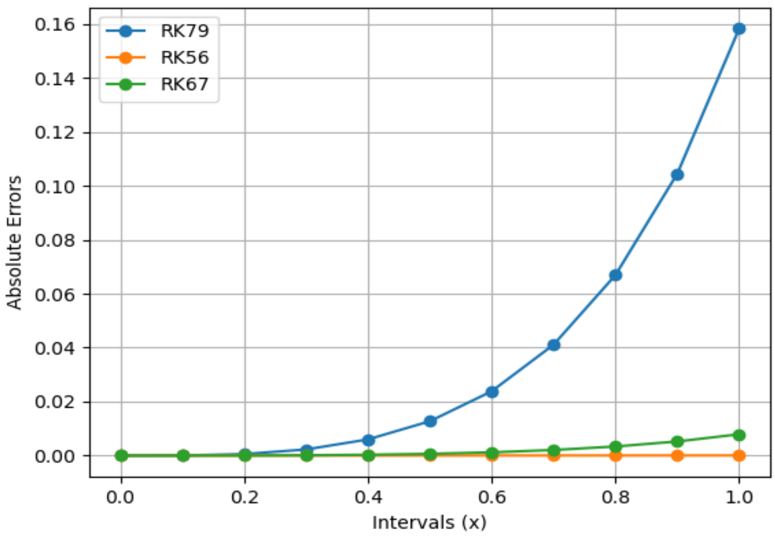

Figure 2. Plot showing contrast errors for experiment 2.

The graphical representation in Figure 2 offers a thorough evaluation of the RK56, RK67, and RK79 methods in their application to solve experiment 2.

Table 5. Computational solution for RK56, RK67 and RK79 experiment 3.

x	Approx. Values for RK56 (s)	Approx. Values for RK67 (s)	Approx. Values for RK79 (s)	Precise Values (S)
0.0	1.0	1.0	1.0	1.0
0.1	0.861	0.863	0.836	0.861
0.2	0.741	0.746	0.699	0.741
0.3	0.639	0.644	0.585	0.639
0.4	0.552	0.559	0.492	0.552
0.5	0.479	0.487	0.416	0.480
0.6	0.419	0.428	0.355	0.420
0.7	0.371	0.380	0.308	0.372
0.8	0.332	0.342	0.272	0.334
0.9	0.303	0.314	0.246	0.305
1.0	0.284	0.296	0.229	0.287

Table 6. RK56, RK67, RK79 contrast errors for experiment 3.

x	Error Values for RK56	Error Values for RK67	Error Values for RK79
0.0	0	0	0
0.1	1.341×10^{-6}	0.261×10^{-3}	0.249×10^{-2}
0.2	0.173×10^{-5}	0.449×10^{-3}	0.423×10^{-2}
0.3	0.710×10^{-5}	0.580×10^{-3}	0.537×10^{-2}
0.4	0.186×10^{-4}	0.668×10^{-3}	0.605×10^{-2}
0.5	0.382×10^{-4}	0.725×10^{-3}	0.639×10^{-2}
0.6	0.668×10^{-4}	0.763×10^{-3}	0.647×10^{-2}
0.7	0.105×10^{-3}	0.794×10^{-3}	0.639×10^{-2}
0.8	0.150×10^{-3}	0.827×10^{-3}	0.621×10^{-2}
0.9	0.203×10^{-3}	0.875×10^{-3}	0.598×10^{-2}
1.0	0.261×10^{-3}	0.950×10^{-3}	0.577×10^{-2}

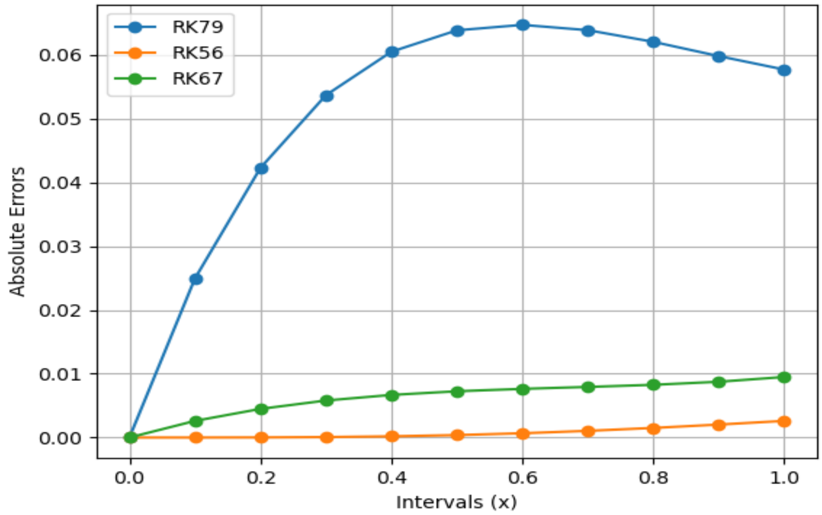

Figure 3. Plot showing contrast errors for experiment 3.

In the assessment of experiment 3, the RK56, RK67, and RK79 methods are comprehensively evaluated through the insights provided in Figure 3.

Table 7. Computational solutions for RK56, RK67 and RK79 experiment 4

x	Approx. Values for RK56 (s)	Approx. Values for RK67 (s)	Approx. Values for RK79 (s)	Precise Values (S)
0.0	1.0	1.0	1.0	1.0
0.1	0.905	0.906	0.893	0.905
0.2	0.821	0.823	0.799	0.821
0.3	0.749	0.751	0.719	0.749
0.4	0.690	0.692	0.655	0.690
0.5	0.643	0.646	0.606	0.643
0.6	0.611	0.614	0.573	0.611
0.7	0.593	0.596	0.557	0.593
0.8	0.591	0.593	0.557	0.591
0.9	0.603	0.606	0.574	0.603
1.0	0.632	0.634	0.609	0.632

Table 8: RK56, RK67, RK79 contrast errors for experiment 4.

x	Error Values for RK56	Error Values for RK67	Error Values for RK79
0.0	0	0	0
0.1	4.083×10^{-6}	0.823×10^{-4}	0.124×10^{-2}
0.2	7.776×10^{-6}	0.149×10^{-3}	0.221×10^{-2}
0.3	0.111×10^{-5}	0.201×10^{-3}	0.294×10^{-2}
0.4	0.141×10^{-5}	0.239×10^{-3}	0.343×10^{-2}
0.5	0.169×10^{-5}	0.264×10^{-3}	0.370×10^{-2}
0.6	0.194×10^{-5}	0.276×10^{-3}	0.377×10^{-2}
0.7	0.216×10^{-5}	0.276×10^{-3}	0.365×10^{-2}
0.8	0.236×10^{-5}	0.264×10^{-3}	0.336×10^{-2}
0.9	0.255×10^{-5}	0.242×10^{-3}	0.291×10^{-2}
1.0	0.271×10^{-5}	0.210×10^{-3}	0.230×10^{-2}

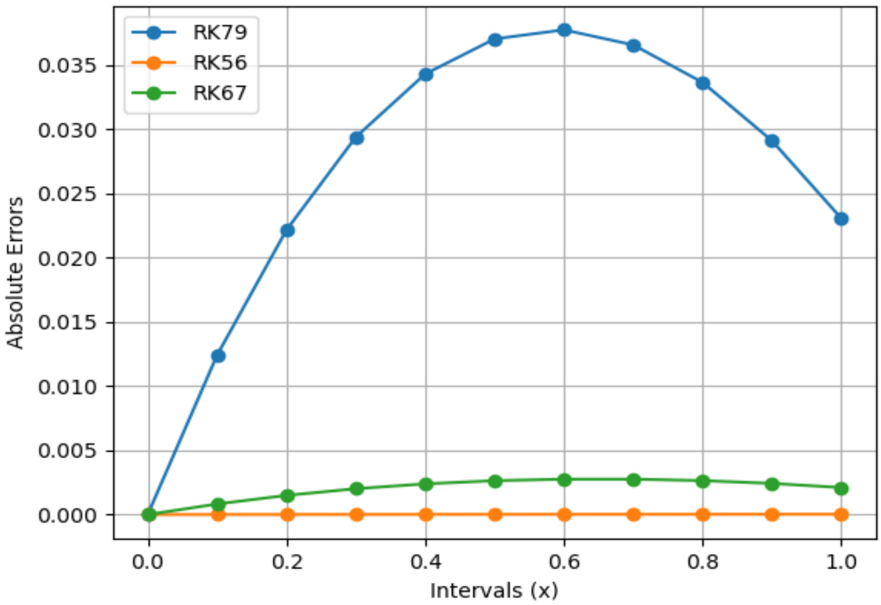

Figure 4. Plot showing contrast errors for experiment 4.

Figure 4 provides a detailed analysis of how the errors of the RK56, RK67, and RK79 methods perform in solving experiment 4.

Table 9. Computational solutions for RK56, RK67 and RK79 experiment 5.

x	Approx. Values for RK56 (s)	Approx. Values for RK67 (s)	Approx. Values for RK79 (s)	Precise Values (S)
2.0	1.0	1.0	1.0	1.0
2.1	0.815	0.818	0.776	0.830
2.2	0.651	0.691	0.634	0.704
2.3	0.511	0.597	0.535	0.608
2.4	0.393	0.524	0.462	0.533
2.5	0.297	0.466	0.405	0.471
2.6	0.221	0.418	0.360	0.420
2.7	0.161	0.378	0.323	0.378
2.8	0.115	0.345	0.291	0.342
2.9	0.081	0.316	0.266	0.312
3.0	0.056	0.291	0.244	0.286

Table 10. RK56, RK67, RK79 contrast errors for experiment 5.

x	Error Values for RK56	Error Values for RK67	Error Values for RK79
2.0	0	0	0
2.1	0.158×10^{-4}	0.489×10^{-3}	0.230
2.2	0.171×10^{-4}	0.583×10^{-3}	0.211
2.3	0.152×10^{-4}	0.549×10^{-3}	0.186
2.4	0.128×10^{-4}	0.479×10^{-3}	0.162
2.5	0.107×10^{-4}	0.405×10^{-3}	0.142
2.6	0.889×10^{-5}	0.336×10^{-3}	0.124
2.7	0.743×10^{-5}	0.277×10^{-3}	0.110
2.8	0.624×10^{-5}	0.227×10^{-3}	0.097
2.9	0.528×10^{-5}	0.186×10^{-3}	0.087
3.0	0.450×10^{-5}	0.152×10^{-3}	0.077

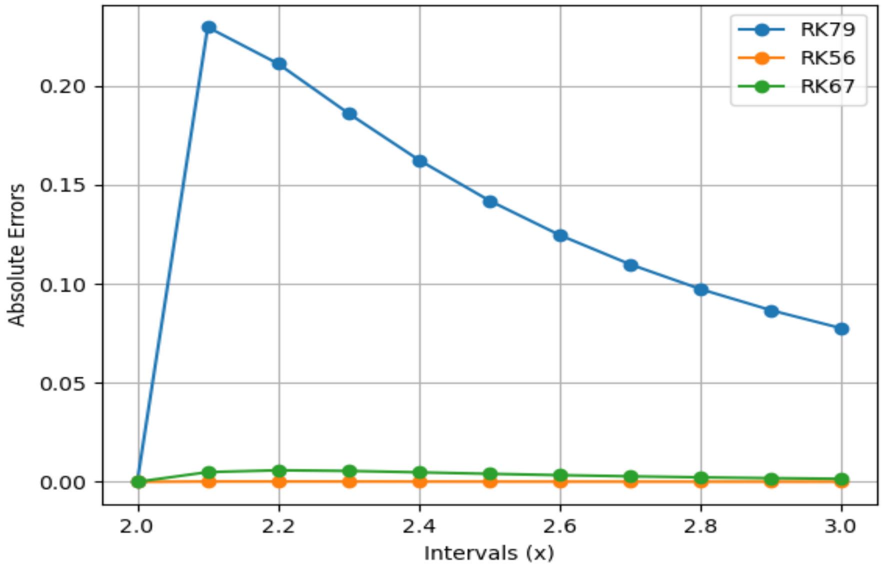

Figure 5. Plot showing contrast errors for experiment 5.

Table 11. Computational solutions for RK56, RK67 and RK79 experiment 6.

x	Approx. Values for RK56 (s)	Approx. Values for RK67 (s)	Approx. Values for RK79 (s)	Precise Values (S)
0.0	1.0	1.0	1.0	1.0
0.1	0.990	0.990	0.990	0.990
0.2	0.962	0.962	0.960	0.962
0.3	0.917	0.918	0.910	0.917
0.4	0.862	0.862	0.846	0.862
0.5	0.780	0.800	0.772	0.800
0.6	0.735	0.736	0.694	0.735
0.7	0.671	0.671	0.619	0.671
0.8	0.610	0.610	0.549	0.610
0.9	0.552	0.553	0.487	0.552
1.0	0.500	0.500	0.432	0.500

Table 12. RK56, RK67, RK79 contrast errors for experiment 6.

x	RK56 Errors	RK67 Errors	RK79 Errors
0.0	0	0	0
0.1	0.165×10^{-5}	0.258×10^{-5}	0.827×10^{-5}
0.2	0.314×10^{-5}	0.175×10^{-5}	0.195×10^{-3}
0.3	0.439×10^{-5}	0.129×10^{-4}	0.725×10^{-3}
0.4	0.530×10^{-5}	0.251×10^{-4}	0.164×10^{-2}
0.5	0.585×10^{-5}	0.329×10^{-4}	0.283×10^{-2}
0.6	0.606×10^{-5}	0.339×10^{-4}	0.409×10^{-2}
0.7	0.598×10^{-5}	0.293×10^{-4}	0.520×10^{-2}
0.8	0.572×10^{-5}	0.215×10^{-4}	0.604×10^{-2}
0.9	0.534×10^{-5}	0.134×10^{-4}	0.656×10^{-2}
1.0	0.489×10^{-5}	0.650×10^{-5}	0.680×10^{-2}

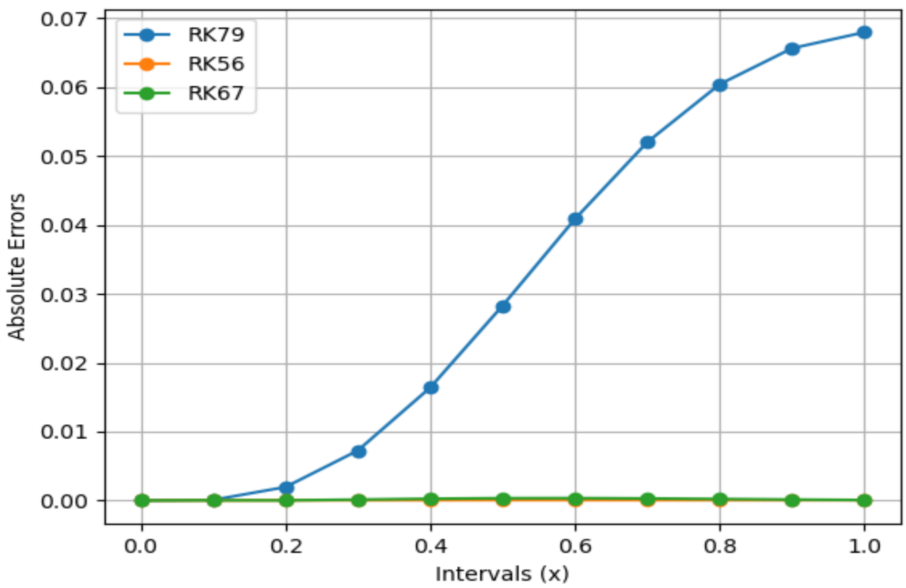

Figure 6. Plot showing contrast errors for experiment 6.

4. 1 Discussion

According to Poornima and Nirmala (2020), a numerical solution is said to be convergent if $\lim_{l\to\infty}|s(z_c)-s_c|=0$, where c varies from 1 to S and the error is depicted by the relation $rrors=|s(z_c)-s_c|$, where $s(z_c)$ represent the numerical solution and s_c represent the analytical solution. In light of this, a thorough examination of the data from Tables 1, 2, 3, and 4 and a comparison of the numerical solutions obtained by the RK56, RK67, and RK79 methods with the analytical solution for Problems 1-4 provide significant new information regarding the effectiveness of these numerical techniques. Furthermore, Figures 1-4 offer additional clarification by evaluating the mistakes related to every approach.

Numerical Approximations Comparison (Tables 1, 3, 5, 7, 9 and 11)

i. The tables demonstrate that by employing a consistent step size of 0.1, the RK56, RK67, and RK79 approaches produce numerical solutions for all three problems that closely approximates the analytical answer.
ii. The RK56, RK67, and RK79 solutions deviate somewhat from one another, indicating that these approaches approximate the real answer about equally well.
iii. The outcomes highlight how well the three numerical approaches performed in resolving the given problems.

Error Comparison (Tables 2, 4, 6, 8, 10, 12 and Figures 1-6)

i. A more thorough understanding of the effectiveness of each technique is offered by the tables and graphical representations of errors.
ii. It is obvious that the RK56 approach consistently provides more accurate findings and exhibits lower error values compared to both the RK67 and RK79 procedures.
iii. The RK56 method's error curves graphically show in Figures 1-3 that they approach zero, showing a convergence to the exact solution as long as the step size stays constant.
iv. The RK67 and RK79 approaches, on the other hand, show somewhat larger error numbers, indicating a considerably less precise approximation of the actual solution.

Computational Time (seconds)

Table 13. Computer Simulation Speed of the Problems.

Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6
RK56 - 0.5s RK67 - 0.92s RK79 - 1.26s	RK56 - 0.29s RK67 - 0.79s RK79 - 1.28s	RK56 - 0.20s RK67 - 0.78s RK79 - 1.14s	RK56 - 0.28s RK67 - 0.73s RK79 - 1.04s	RK56 - 0.28s RK67 - 0.71s RK79 -1.01s	RK56 - 0.52s RK67 - 0.71s RK79 - 1.15s

Discoveries

i The comparative analysis unequivocally demonstrates the RK56 method's advantage over the other three approaches in solving Problems 1-6.
ii The RK56 technique is noteworthy for its excellent accuracy and efficiency as it constantly converges to the analytical answer with low error.
iii Although the RK67 and RK79 procedures yield dependable outcomes as well, the RK56 method is the most accurate numerical solution for these issues.
iv The RK79 procedure gives higher errors than the other approaches. RK79 is a powerful numerical technique known for its high accuracy and stability in solving differential equations. However, it may yield higher errors compared to other approaches due to its intricate complexity, which can amplify round-off errors and numerical instability. RK79's sensitivity to step size selection and stiffness-induced errors further contribute to this phenomenon. While RK79 offers superior accuracy theoretically, its practical implementation may be challenged by computational overhead and difficulty in balancing accuracy and efficiency. Understanding these limitations is crucial for selecting appropriate numerical methods in scientific and engineering applications.

5. Conclusion

This study thoroughly explored the resolution of first-order ordinary differential equations using RK56, RK67, and RK79 techniques. In the course of this study, extensive numerical exploration has revealed compelling results showcasing the outstanding accuracy achieved by all three methodologies. Based on its smaller error values, RK56 is the most proficient model in the graphical representations (Figures 1–6). Tables 1, 3, 5, 7, 9 and 11's analysis provide more evidence for RK56's improved performance. Regarding the computer simulation speed, we observe variations in the performance of the RK methods across the tested problems. Generally, the RK56 method demonstrates the fastest computational speed among the tested problems, with execution times ranging from 0.20s to 0.52s. On the other hand, the RK79 method consistently exhibits slower computational speeds, with execution times ranging from 1.01s to 1.28s. These findings suggest that the RK56 method offers superior computational efficiency compared to RK67 and RK79 for the given simulation tasks.

According to the research, RK56 is the best option because of its dependability, efficacy, and efficiency when solving these kinds of problems. By adding insightful new information to the body of knowledge, this contribution helps academics and practitioners choose efficient numerical solutions for related mathematical issues.

References

[1] Lee KC, Senu N, Ahmadian A, Ibrahim SI & Baleanu D. Numerical Study of Third-Order Ordinary Differential Equations Using a New Class of Two Derivative Runge-Kutta Type Methods. Alex Eng J 2020; 59, 2449–2467.

[2] Poornima S, and Nirmala T. Comparative Study of Runge-Kutta Methods of Solving Ordinary Differential Equations. Int J Res in Eng, Sci and Mgt 2020; .3: 557-559.

[3] Jamali N. Analysis and Comparative Study of Numerical Methods to Solve Ordinary Differential Equation with Initial Value Problem. Int J Adv Res 2020; 7(5): 117-128.

[4] Okeke AA, Hambagda BM, & Tumba P. Accuracy Study on Numerical Solutions of Initial Value Problems (IVP) in Ordinary Differential Equations. Int J Math and Stat Invention 2019. 7(2), 2321-4759.

[5] Soliu AA. Comparative Study on Some Numerical Algorithms for First Order Ordinary Differential Equations. B. Tech, Federal University of Technology, Minna, Nigeria. 2023.

[6] Mesa F, Devia-Narvaez DM, Correa-Velez G. Numerical Comparison by Different Methods (Second Order Runge Kutta Methods, Heun Method, fixed Point Method and Ralston Method) to Differential Equations with Initial Condition. Scientia et Technica 2020; 25(2): 299-305

[7] Smith J, & Johnson A. Comparative Analysis of Runge-Kutta Methods for Solving Ordinary Differential Equations. J Comput Math 2019. 45(2), 210-225.

[8] Wang L, & Li HA. Review of Higher-Order Runge-Kutta Methods in Scientific Computing. Applied Numerical Analysis 2020; 35(4): 567-582.

[9] Jones R, Brown M. Performance Evaluation of Runge-Kutta Techniques in Atmospheric Modeling. J Atmosph Sci 2018; 25(3): 410-425.

[10] Garcia P, & Martinez E. Comparative Study of Runge-Kutta Methods for Solving Heat Transfer Equations in Engineering Applications. Heat Trans Eng 2017;, 33(1): 89-104.

[11] Chen Y, & Zhang Q. A Survey of Runge-Kutta Methods for Solving Chemical Reaction Kinetics. Chem Eng. J. 2016; 40(2): 315-330.

[12] Agbeboh GU, Adoghe LO, Ehiemua ME, Ononogbo BC. On the Derivation of a Sixth-Stage-Fifth-Order Runge-Kutta Method for Solving Initial Value Problems in Ordinary Differential Equations. American J Sci Eng Res. 2020; 3(5): 29-41.

[13] Başhan A, Battal S, Karakoç G, and Geyikli T. Approximation of the KdVB Equation by the Quintic B-spline Differential Quadrature method. Kuwait J.Sci 2015; 42(2)**:** 67-92.

[14] Bashan A, Ucar Y, Yagmurlu NM, Esen A. An effective approach to numerical soliton solutions for the Schrodinger equation via modified cubic B-spline differential quadrature method. Article *in* Chaos Solitons & Fractals 2017; 100, 45–56.

[15] Bashan A. An effective application of differential quadrature method based on modified cubic B-splines to numerical solutions of the kdV equation. Turk J Math 2018; 42: 373 – 394.

[16] Bashan A, Ucar Y, Yagmurlu NM, Esen A. Numerical Solutions for the Fourth Order Extended Fisher-Kolmogorov Equation with High Accuracy by Differential Quadrature Method. Sigma J Eng & Nat Sci 2018; 9(3): 273-284.

[17] Ucar Y, Yagmurlu NM, Bashan A. Numerical Solutions and Stability Analysis of Modified Burgers Equation via Modified Cubic B-Spline Differential Quadrature Methods. Sigma J Eng & Nat Sci 2019; 37 (1): 129-142.

[18] Al-Shimmary AF. Solving initial value problem using Runge-Kutta 6th order method. ARPN J Eng Appl Sci 2017; 12(13): 3953-3961.

[19] Trikkaliotis GD & Gousidou-Koutita MCh. Production of the Reduction Formula of Seventh Order Runge-Kutta Method with Step Size Control of an Ordinary Differential Equation. Appl Math 2022; 13, 325-337.

[20] Tuba G. Some Approaches For Solving Mulplicative Second Order Linear Differential Equations with Variable Exponentials and Multiplicative Airy's Equation. Turk J Sci & Tech 2023; 18(2): 301-309.

[21] Audu KJ, Taiwo AR, Soliu AA. Assessment of Numerical Performance of Some Runge-Kutta Methods and New Iteration Method on First Order Differential Problems. Dutse J Pure & Appl Sci 2023; 9(4a): 58-70.

[22] Arora G, Joshi V & Garki I. Developments in Runge–Kutta Method to Solve Ordinary Differential. Recent Advan Math Eng 2020; pp 193-202..

[23] Hetmaniok E, Pleszczynski M. Comparison of the Selected Methods Used for Solving the Ordinary Differential Equations and Their Systems. Mathematics 2022; 10(3): 1-15.