algorithms

MDPI

Article

An Efficient Third-Order Scheme Based on Runge–Kutta and Taylor Series Expansion for Solving Initial Value Problems

Noori Y. Abdul-Hassan [1], Zainab J. Kadum [2] and Ali Hasan Ali [1,3,*]

[1] Department of Mathematics, College of Education for Pure Sciences, University of Basrah, Basrah 61001, Iraq
[2] Basrah Education Directorate, Ministry of Education, Basrah 61001, Iraq
[3] Institute of Mathematics, University of Debrecen, Pf. 400, H-4002 Debrecen, Hungary
* Correspondence: ali.hasan@science.unideb.hu

Abstract: In this paper, we propose a new numerical scheme based on a variation of the standard formulation of the Runge–Kutta method using Taylor series expansion for solving initial value problems (IVPs) in ordinary differential equations. Analytically, the accuracy, consistency, and absolute stability of the new method are discussed. It is established that the new method is consistent and stable and has third-order convergence. Numerically, we present two models involving applications from physics and engineering to illustrate the efficiency and accuracy of our new method and compare it with further pertinent techniques carried out in the same order.

Keywords: numerical methods; initial value problem; autonomous equation; local truncation error; consistency; stability

MSC: 65L05; 65L07; 65L09; 65L12; 65L20

Citation: Abdul-Hassan, N.Y.; Kadum, Z.J.; Ali, A.H. An Efficient Third-Order Scheme Based on Runge–Kutta and Taylor Series Expansion for Solving Initial Value Problems. *Algorithms* **2024**, *17*, 123. https://doi.org/10.3390/a17030123

Academic Editors: Alicia Cordero and Juan Ramón Torregrosa Sánchez

Received: 20 November 2023
Revised: 22 February 2024
Accepted: 14 March 2024
Published: 16 March 2024

1. Introduction

Numerical analysis is the area of mathematics that deals with computational techniques for studying and finding solutions to math problems. It is an offshoot of computer science and mathematics that develops, analyzes, and generates algorithms for numerically solving mathematical problems. Numerical methods are typically centered on the implementation of numerical algorithms. The goal of these numerical methods is to provide systematic procedures for numerically solving mathematical problems. The ordinary differential equation (ODE) is a mathematical equation that is used in natural physical processes, chemistry, engineering, and other sciences. Ordinary differential equations are notoriously difficult to approximate analytically, so the numerical treatment is crucial because it offers a powerful alternate solution method for resolving the differential equation.

We frequently use initial value problems (IVPs), such that

$$y' = f(x, y(x)),\ y(x_0) = y_0 \tag{1}$$

where x is the independent variable, which might also indicate time in physical problems, and $y(x)$ is the solution. Furthermore, because $y(x)$ can be a vector-valued function with N-dimensions, the domain and range of f and the solution y are given by

$$f : \mathbb{R} \times \mathbb{R}^N \to \mathbb{R}$$

$$y : \mathbb{R} \to \mathbb{R}^N$$

Furthermore, Equation (1), in which f is a function both of x and y, is known as a "non-autonomous" equation. However, by simply adding an extra variable, which is

always equal to x, it can be easily rewritten in an equivalent "autonomous" form below, in which f is a function of y only:

$$y' = f(y(x)),\ y(x_0) = y_0 \tag{2}$$

There are numerous well-known numerical methods that can be used to solve the IVP in Equation (1). One category of these is third-order methods, like Ralston's third-order Runge–Kutta method (Ralston's scheme) [1], the third-order Runge–Kutta method (RK3 scheme) [2], the third-order Runge–Kutta method based on the arithmetic mean (ARK3 scheme) [3], the third-order Heun's method (Heun's scheme) [4], and several other methods. These numerical methods have been constructed to solve Equations (1) and (2) using different techniques such as the Taylor series expansion technique, homotopy analysis technique, quadrature formulas technique, and Adomian decomposition technique; for more details, see [5–7].

In this study, we construct a new numerical method based on a variation of the standard formulation of the Runge–Kutta method using Taylor series expansion. The rest of this paper is divided into the following sections. In Section 2, we recall some definitions and auxiliary results that we will use in our work. The derivation of the new method is described in Section 3. Section 4 provides details on the local truncation error. Section 5 discusses the stability analysis of the suggested technique. The consistency of the new method and its convergence are detailed in Section 6. Several numerical models are shown in Section 7 to show the efficiency of this method and compare it with other relevant methods. Finally, Section 8 offers the discussion and the conclusions.

2. Preliminaries

Theorem 1 ([8]). *(Existence and Uniqueness Theorem).*

Let $f(x,\ y)$ and $\frac{\partial f}{\partial y}$ be continuous functions of x and y at all points $(x,\ y)$ in some neighborhood of the initial point $(x_0,\ y_0)$. Then, there is a unique function $y(x)$ defined on some interval $[x_0 - \varepsilon,\ x_0 + \varepsilon]$ satisfying

$$y' = f(x, y(x)),\ y(x_0) = y_0,\ x\ \in\ [x_0 - \varepsilon,\ x_0 + \varepsilon],\ \varepsilon > 0 \tag{3}$$

Definition 1 ([9]). *A Taylor sum or Taylor series is a mathematical function representation in the form of a series consisting of terms calculated using the values of the function's derivation at a specific point and given by*

$$\psi_T(x_n, y_n; h) = \sum_{r=0}^{\infty} \frac{h^r}{(r+1)!} \left(\frac{\partial}{\partial x} + f \frac{\partial}{\partial y} \right)^r f(x, y) \tag{4}$$

Definition 2 ([10]). *Let $I \subseteq \mathbb{R}$ be an open interval and $f : I \to \mathbb{R}$ be a n-times differentiable function at $\alpha \in I$. The Taylor polynomial $T_n^{\alpha} f$ of degree n at a point α of f is the polynomial*

$$T_n^{\alpha} f(\alpha) = f(\alpha) + (x - \alpha) f\prime(\alpha) + \frac{(x-\alpha)^2}{2!} f''(\alpha) + \frac{(x-\alpha)^3}{3!} f'''(\alpha) + \cdots + \frac{(x-\alpha)^n}{n!} f^{(n)}(\alpha) \tag{5}$$

Definition 3 ([11]). *The difference between the numerical solution*

$$y_{n+1} = y_n + h\psi(x_n, y_n; h) \tag{6}$$

and the exact solution $y(x_{n+1})$ is called the local truncation error (L.T.E.) for a one-step method x_n with step size h is given by

$$L.T.E. = y(x_{n+1}) - y_{n+1} \tag{7}$$

Definition 4 ([12]). *The numerical method is said to be stable if there exists $h > 0$ for each differential equation such that changing the initial values by a fixed amount produces a bounded change in the numerical solution for all h*

Definition 5 ([13]). *A numerical method is said to be consistent when all step sizes $h \to 0$, as it will converge to the differential equation. In other words, we say the method is consistent if and only if*

$$\underset{h\to 0}{Lim}\frac{L.T.E.}{h} = 0 \; for \; h = \frac{b-a}{N} \tag{8}$$

3. Construction of the New Scheme

In order to construct new single-step methods to solve the IVP Equation (1), we rely on a variant of the standard form of the Runge–Kutta method given by

$$y_{n+1} = y_n + h\psi(x_n, y_n; h)$$

where

$$\psi(x_n, y_n; h) = \sum_{i=1}^{2m} w_i k_{i+1}$$

and

$$k_1 = f(x_n, y_n),\; k_i = f\left(x_n + c_i h, y_n + h\sum_{\substack{j=1\\ j\neq i}}^{i-1} a_{ij}k_j\right),\; i = 2,\; 3, \ldots,\; 2m+1$$

and

$$\sum_{j=1}^{i-1} a_{ij} = c_i,\; i = 2,\; 3,\; \ldots,\; 2m+1$$

By using the Taylor series expansion of any arbitrary function $\psi(x_n, y_n; h)$ for the non-autonomous Equation (1), we have

$$\psi_T(x_n, y_n; h) = \sum_{r=0}^{\infty} \frac{1}{(r+1)!}\left(\frac{\partial}{\partial x} + f\frac{\partial}{\partial y}\right)^r f(x, y)$$

and in the autonomous case Equation (2), $\psi_T(x_n, y_n; h)$ becomes

$$\psi_T(y_n; h) = \sum_{r=0}^{\infty} \frac{1}{(r+1)!}\left(f\frac{\partial}{\partial y}\right)^r f(y)$$

We now start with the numerical method by using the family of explicit Runge–Kutta methods listed below to solve the mentioned problem in (2).

$$y_{n+1} = y_n + h(w_1 k_2 + w_2 k_3) \tag{9}$$

with

$$\left.\begin{array}{c} k_1 = f(x_n, y_n) \\ k_2 = f(x_n + c_2 h, y_n + a_1 h k_1) \\ k_3 = f(x_n + c_3 h, y_n + h(a_2 k_1 + a_3 k_2)) \end{array}\right\} \tag{10}$$

where $a_1 = a_{21}$, $a_2 = a_{31}$, and $a_3 = a_{32}$. We must first calculate the unknowns a_1, a_2, a_3, w_2 and w_3. Using Taylor series expansion around $f(y)$, we obtain

$$k_1 = f(y_n) \tag{11}$$

$$k_2 = \sum_{r=0}^{\infty} \frac{1}{r!} \left(h a_1 k_1 \frac{d}{dy} \right)^r f(y_n) \tag{12}$$

$$k_3 = \sum_{r=0}^{\infty} \frac{1}{r!} \left(h(a_2 k_1 + a_3 k_2) \frac{d}{dy} \right)^r f(y_n) \tag{13}$$

By expanding k_1, k_2, and k_3 in Equations (11)–(13), we have

$$\left.\begin{array}{c} k_1 = f \\ k_2 = f + h a_1 f f_y + \frac{h^2}{2!}\left(f^2 {a_1}^2 f_{yy}\right) + \frac{h^2}{3!}\left({a_1}^3 f^3 f_{yyy}\right) + \cdots \\ k_3 = f + h(a_2 + a_3) f f_y + \frac{h^2}{2!}\left(\left({a_2}^2 + 2a_2 a_3 + {a_3}^2\right) f^2 f_{yy} + 2 a_1 a_3 f {f_y}^2\right) \\ + h^3\left(\left(\frac{1}{2} a_2 {a_3}^3 + {a_2}^2 a_3 + \frac{1}{6}{a_3}^3 + \frac{1}{6}{a_2}^3\right) f^3 f_{yyy} + a_1 a_2 a_3 f^2 f_y f_{yy} + \frac{1}{2}{a_1}^2 a_3 f^2 f_y f_{yy}\right) + \cdots \end{array}\right\} \tag{14}$$

Substituting Equation (14) into Equation (9) yields

$$y_{n+1} = y_n + (w_1 + w_2) f\, h + ((a_2 + a_3)\, w_2 + a_1\, w_1) f f_y\, h^2 + \frac{1}{2}\left(\left((a_2 + a_3)^2\, w_2 + {a_1}^2\, w_1\right) f f_{yy} + 2\, w_2 a_1 a_3\, {f_y}^2\right) f\, h^3 + \cdots \tag{15}$$

The Taylor series expansion of an exact solution $y(x_{n+1})$ is given by

$$y(x_{n+1}) = y(x_n) + fh + \left(\frac{1}{2} f f_y\right) h^2 + \frac{1}{6}\left(f^2 f_{yy} + f {f_y}^2\right) h^3 + \frac{1}{24}\left(f^3 f_{yy} + 4 f^2 f_y f_{yy} + f {f_y}^3\right) h^4 + \ldots \tag{16}$$

The following system of equations is obtained by expressing k_1, k_2, and k_3 in the Taylor expansion, ignoring terms with powers of h higher than 3, and then substituting them into formula (15) and comparing them to Equation (16):

$$hf:\ w_1 + w_2 = 1$$

$$h^2 f f_y : (a_2 + a_3) w_2 + w_1 a_1 = \frac{1}{2}$$

$$h^3 f^2 f_{yy} :\ (a_2 + a_3)^2 w_2 + {a_1}^2 w_1 = \frac{1}{3}$$

$$h^3 f {f_y}^2 :\ w_2 a_1 a_3 = \frac{1}{6}$$

This is a system with an infinite number of solutions comprising four equations and five unknowns. Assuming that $w_1 = \frac{3}{7}$, we obtain the optimal solution listed below:

$$w_1 = \frac{3}{7},\ w_2 = \frac{4}{7},\ a_1 = \frac{1}{6},\ a_2 = -1,\ a_3 = \frac{7}{4}$$

Additionally, from Equation (6), we obtain $c_2 = \frac{1}{6}$ and $c_3 = \frac{3}{4}$.

Thus, substituting the above results in Equations (9) and (10), we present the new method, and we call it the variation Runge–Kutta method of order three (VRK3), given as follows:

$$y_{n+1} = y_n + \frac{h}{7}(3k_2 + 4k_3) \tag{17}$$

with

$$\left.\begin{array}{c} k_1 = f(x_n, y_n) \\ k_2 = f\left(x_n + \frac{h}{6}, y_n + \frac{h}{6}k_1\right) \\ k_3 = f\left(x_n + \frac{3h}{4}, y_n + h(-k_1 + \frac{7}{4}k_2)\right) \end{array}\right\} \tag{18}$$

4. Accuracy of the New Scheme

Here, the local truncation error of the proposed scheme is investigated as follows. The set of Equation (18), when expanded using Taylor expansion, yields

$$k_1 = f \tag{19}$$

$$k_2 = f + \frac{h}{6}ff_y + \frac{h^2}{72}f^2 f_{yy} + \frac{h^3}{1296}f^3 f_{yyy} + \cdots \tag{20}$$

$$k_3 = f + \frac{3h}{4}ff_y + h^2\left(\frac{9}{32}f^2 f_{yy} + \frac{7}{24}f{f_y}^2\right) + h^3\left(\frac{9}{128}f^3 f_{yyy} - \frac{77}{288}f^2 f_y f_{yy}\right) + \cdots \tag{21}$$

Now, substituting Equations (19)–(21) into Equation (17) yields

$$y_{n+1} = yn + hf + \frac{1}{2}h^2 ff_y + \left(\frac{1}{6}f^2 f_{yy} + \frac{1}{6}f{f_y}^2\right)h^3 + \left(\frac{35}{864}f^3 f_{yyy} - \frac{11}{72}f^2 f_y f_{yy}\right)h^4 + \ldots$$

Hence, from Definition 3, we have

$$L.T.E. = \left(\frac{1}{864}f(f^2 f_{yyy} + 276ff_y f_{yy} + 36{f_y}^3)\right)h^4 + O(h^5) \tag{22}$$

As per Equation (22), our proposed method (VRK3) is of third order, with an $L.T.E.$ of fourth order.

5. Stability Analysis of New Scheme

To test the absolute stability of the presented scheme (VRK3), we use the set of Equations (18) to derive the following:

$$k_1 = \lambda y_n \tag{23}$$

$$k_2 = \lambda y_n\left(1 + \frac{h\lambda}{6}\right) \tag{24}$$

$$k_3 = \lambda y_n\left(1 + \frac{11h\lambda}{4} + \frac{7h^2\lambda^2}{24}\right) \tag{25}$$

By substituting Equations (23)–(25) into (17) and allowing $z = h\lambda$, we obtain

$$y_{n+1} = y_n\left[1 + z + \frac{Z^2}{2} + \frac{z^3}{6}\right] \tag{26}$$

Then, from Equation (26), the stability polynomial is

$$R(z) = \frac{y_{n+1}}{y_n} = \left[1 + z + \frac{Z^2}{2} + \frac{z^3}{6}\right] + o\left(z^4\right) \tag{27}$$

Utilizing the MATLAB program, Figure 1 below graphically illustrates the absolute stability region of the Formula (27):

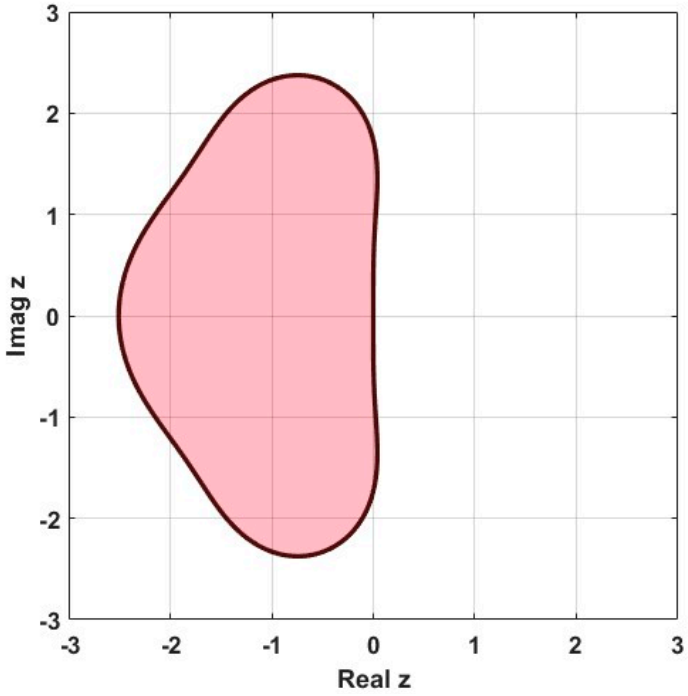

Figure 1. The absolute stability region of the VRK3 method.

6. Consistency of the New Scheme

To explain the consistency property of the newly proposed scheme, we adopt Definition 5. Therefore, by substituting Equation (22) into Equation (8), we obtain

$$\lim_{h\to 0}\frac{L.T.E.}{h} = \lim_{h\to 0}\frac{(\frac{1}{864}f(f^2 f_{yyy} + 276 f f_y f_{yy} + 36 {f_y}^3))h^4 + O(h^5)}{h} = 0 \tag{28}$$

According to Lambert [13], a numerical method is consistent if the order is bigger than one. Therefore, our new method is consistent since it is of order three.

Lambert also defines a numerical method as convergent if it is consistent and stable. Following from Equations (27) and (28), this method is consistent and stable. We conclude that the new method (VRK3) is convergent because it satisfies the consistency and stability properties.

7. Numerical Examples

In this section, we introduce two models of IVPs with varying step sizes h to compare the efficiency and the accuracy of the proposed new method (VRK3 scheme) with other third-order methods, like Ralston's scheme, RK3 scheme, ARK3 scheme, and Heun's scheme. Here, all calculations and figures are performed using MATLAB (R2022a) software. The numerical results are presented in Tables 1–8, and the error analysis is illustrated in Figures 2–7.

7.1. Problem 1 [14]

Take into consideration the first order IVP $y' = y - x^2 + 1, y(0) = 0.5$, with the exact solution $y = x^2 + 2x - 0.5e^x + 1, 0 \le x \le 1$. Tables 1–3 show the results that were obtained. The graphs of absolute errors are shown in Figures 2–4. The comparison of CPU time between the VRK3 scheme and other relevant third-order schemes is shown in Table 7.

Table 1. Comparison of the absolute errors among third-order schemes in Problem 1, for $h = 10^{-1}$.

x_i	Ralston's Scheme	RK3 Scheme	ARK3 Scheme	Heun's Scheme	VRK3 Scheme
zero	zero	zero	zero	zero	zero
0.1	6.2076×10^{-6}	6.2076×10^{-6}	8.9854×10^{-6}	3.4299×10^{-6}	6.5207×10^{-7}
0.2	1.2845×10^{-5}	1.2845×10^{-5}	1.8692×10^{-5}	6.9968×10^{-6}	1.1492×10^{-6}
0.3	1.9932×10^{-5}	1.9932×10^{-5}	2.9173×10^{-5}	1.0692×10^{-5}	1.4515×10^{-6}
0.4	2.7492×10^{-5}	2.7492×10^{-5}	4.0482×10^{-5}	1.4502×10^{-5}	1.5125×10^{-6}
0.5	3.5546×10^{-5}	3.5546×10^{-5}	5.2680×10^{-5}	1.8412×10^{-5}	1.2781×10^{-6}
0.6	4.4113×10^{-5}	4.4113×10^{-5}	6.5826×10^{-5}	2.2399×10^{-5}	6.8563×10^{-7}
0.7	5.3212×10^{-5}	5.3212×10^{-5}	7.9987×10^{-5}	2.6437×10^{-5}	3.3777×10^{-7}
0.8	6.2861×10^{-5}	6.2861×10^{-5}	9.5229×10^{-5}	3.0492×10^{-5}	1.8762×10^{-6}
0.9	7.3074×10^{-5}	7.3074×10^{-5}	1.1162×10^{-4}	3.4524×10^{-5}	4.0265×10^{-6}
1	8.3864×10^{-5}	8.3864×10^{-5}	1.2925×10^{-4}	3.8482×10^{-5}	6.9006×10^{-6}

Table 2. Comparison of the absolute errors among third-order schemes in Problem 1, for $h = 5 \times 10^{-2}$.

x_i	Ralston's Scheme	RK3 Scheme	ARK3 Scheme	Heun's Scheme	VRK3 Scheme
zero	zero	zero	zero	zero	zero
0.1	7.9184×10^{-7}	7.9184×10^{-7}	1.1480×10^{-6}	4.3572×10^{-7}	7.9594×10^{-8}
0.2	1.6379×10^{-6}	1.6379×10^{-6}	2.3876×10^{-6}	8.8818×10^{-7}	1.3848×10^{-7}
0.3	2.5408×10^{-6}	2.5408×10^{-6}	3.7254×10^{-6}	1.3561×10^{-6}	1.7141×10^{-7}
0.4	3.5031×10^{-6}	3.5031×10^{-6}	5.1684×10^{-6}	1.8377×10^{-6}	1.7229×10^{-7}
0.5	4.5273×10^{-6}	4.5273×10^{-6}	6.7240×10^{-6}	2.3307×10^{-6}	1.3400×10^{-7}
0.6	5.6159×10^{-6}	5.6159×10^{-6}	8.3997×10^{-6}	2.8321×10^{-6}	4.8292×10^{-8}
0.7	6.7710×10^{-6}	6.7710×10^{-6}	1.0204×10^{-5}	3.3383×10^{-6}	9.4375×10^{-8}
0.8	7.9947×10^{-6}	7.9947×10^{-6}	1.2144×10^{-5}	3.8448×10^{-6}	3.0504×10^{-7}
0.9	9.2884×10^{-6}	9.2884×10^{-6}	1.4231×10^{-5}	4.3460×10^{-6}	5.9642×10^{-7}
1	1.0653×10^{-5}	1.0653×10^{-5}	1.6472×10^{-5}	4.8351×10^{-6}	9.8318×10^{-7}

Table 3. Comparison of the absolute errors among third-order schemes in Problem 1, for $h = 25 \times 10^{-3}$.

x_i	Ralston's Scheme	RK3 Scheme	ARK3 Scheme	Heun's Scheme	VRK3 Scheme
zero	zero	zero	zero	zero	zero
0.1	9.9973×10^{-8}	9.9973×10^{-8}	1.4505×10^{-7}	5.4894×10^{-8}	9.8153×10^{-9}
0.2	2.0675×10^{-7}	2.0675×10^{-7}	3.0165×10^{-7}	1.1185×10^{-7}	1.6954×10^{-8}
0.3	3.2066×10^{-7}	3.2066×10^{-7}	4.7062×10^{-7}	1.7070×10^{-7}	2.0745×10^{-8}
0.4	4.4202×10^{-7}	4.4202×10^{-7}	6.5283×10^{-7}	2.3121×10^{-7}	2.0405×10^{-8}
0.5	5.7114×10^{-7}	5.7114×10^{-7}	8.4920×10^{-7}	2.9308×10^{-7}	1.5023×10^{-8}
0.6	7.0830×10^{-7}	7.0830×10^{-7}	1.0607×10^{-6}	3.5592×10^{-7}	3.5422×10^{-9}
0.7	8.5378×10^{-7}	8.5378×10^{-7}	1.2883×10^{-6}	4.1926×10^{-7}	1.5261×10^{-8}
0.8	1.0078×10^{-6}	1.0078×10^{-6}	1.5331×10^{-6}	4.8250×10^{-7}	4.2799×10^{-8}
0.9	1.1705×10^{-6}	1.1705×10^{-6}	1.7962×10^{-6}	5.4492×10^{-7}	8.0702×10^{-8}
1	1.3422×10^{-6}	1.3422×10^{-6}	2.0787×10^{-6}	6.0565×10^{-7}	1.3084×10^{-7}

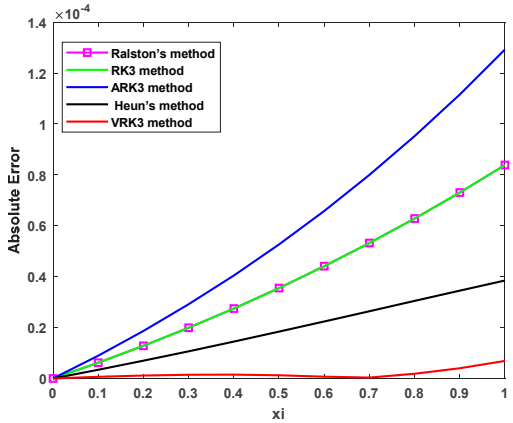

Figure 2. The absolute errors for numerical results in Table 1.

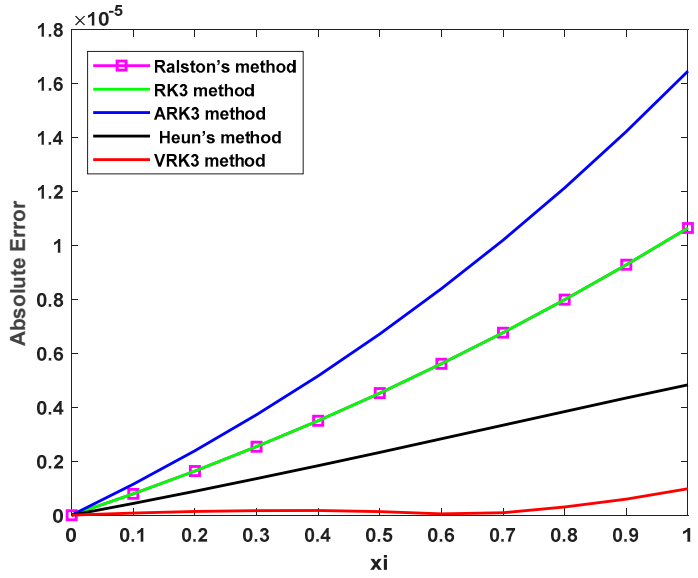

Figure 3. The absolute errors for numerical results in Table 2.

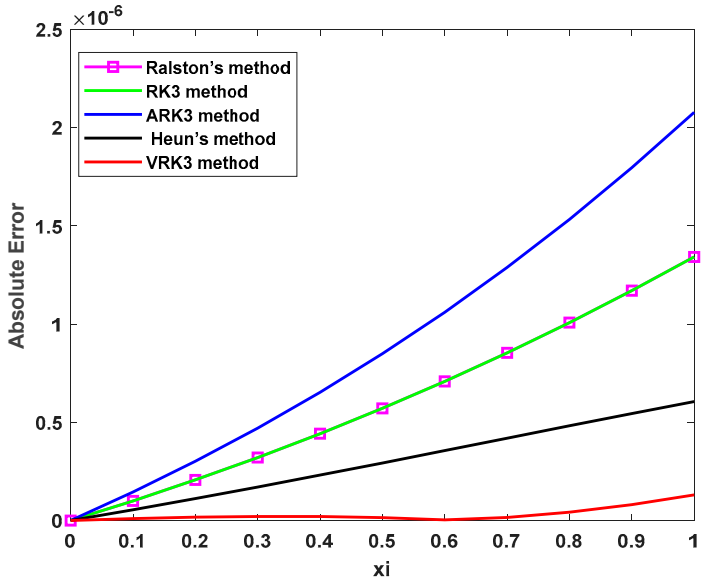

Figure 4. The absolute errors for numerical results in Table 3.

7.2. Problem 2 (Mixture Model)

We consider here the IVP proposed in [15], which was a model of a storage tank in an oil refinery that holds 2000 gal of gasoline with 100 lb of an additive mixed within it. To prepare for winter weather, 40 gal/min of gasoline that contains 2 lb of additive per gallon is pumped into the storage tank. The well-mixed solution is pumped out at a rate of 45 gal/min. Let y be the amount of additive (in pounds) in the tank at time x. When $x = 0$, we know that $y = 100$. The mixture process is modeled by the IVP, $y' = 80 - \frac{45}{(2000-5x)}\, y$, $y(0) = 100$, and the analytic solution, $y(x) = (2000 - 5x)\{2 - \frac{3900}{(2000)^9}(2000 - 5x)^8\}$, $0 \leq x \leq 1$.

Tables 4–6 show the absolute errors among third-order methods and the VRK3 scheme, with different step sizes of $h = 0.1$, $h = 0.05$, and $h = 0.025$. Figures 5–7 depict the graphical analysis used to support the numerical results in Tables 4–6. A comparison of CPU time between the new method and other third-order schemes is shown in Table 8.

Table 4. Comparison of the absolute errors among third-order schemes in Problem 2, for $h = 10^{-1}$.

x_i	Ralston's Scheme	RK3 Scheme	ARK3 Scheme	Heun's Scheme	VRK3 Scheme
zero	zero	zero	zero	zero	zero
0.1	3.7246×10^{-9}	4.1133×10^{-9}	3.8694×10^{-9}	3.3207×10^{-9}	3.1607×10^{-9}
0.2	7.4366×10^{-9}	8.2127×10^{-9}	7.7257×10^{-9}	6.6303×10^{-9}	6.3109×10^{-9}
0.3	1.1136×10^{-8}	1.2298×10^{-8}	1.1568×10^{-8}	9.9282×10^{-9}	9.4499×10^{-9}
0.4	1.4821×10^{-8}	1.6367×10^{-8}	1.5397×10^{-8}	1.3214×10^{-8}	1.2577×10^{-8}
0.5	1.8494×10^{-8}	2.0424×10^{-8}	1.9213×10^{-8}	1.6488×10^{-8}	1.5694×10^{-8}
0.6	2.2154×10^{-8}	2.4466×10^{-8}	2.3015×10^{-8}	1.9752×10^{-8}	1.8800×10^{-8}
0.7	2.5801×10^{-8}	2.8493×10^{-8}	2.6804×10^{-8}	2.3003×10^{-8}	2.1895×10^{-8}
0.8	2.9435×10^{-8}	3.2507×10^{-8}	3.0579×10^{-8}	2.6243×10^{-8}	2.4979×10^{-8}
0.9	3.3057×10^{-8}	3.6506×10^{-8}	3.4342×10^{-8}	2.9472×10^{-8}	2.8053×10^{-8}
1	3.6665×10^{-8}	4.0491×10^{-8}	3.8090×10^{-8}	3.2689×10^{-8}	3.1114×10^{-8}

Table 5. Comparison of the absolute errors among third-order schemes in Problem 2, for $h = 5 \times 10^{-2}$.

x_i	Ralston's Scheme	RK3 Scheme	ARK3 Scheme	Heun's Scheme	VRK3 Scheme
zero	zero	zero	zero	zero	zero
0.1	4.6501×10^{-10}	5.1354×10^{-10}	4.8308×10^{-10}	4.1457×10^{-10}	3.9459×10^{-10}
0.2	9.2889×10^{-10}	1.0258×10^{-9}	9.6499×10^{-10}	8.2821×10^{-10}	7.8832×10^{-10}
0.3	1.3910×10^{-9}	1.5361×10^{-9}	1.4451×10^{-9}	1.2403×10^{-9}	1.1805×10^{-9}
0.4	1.8506×10^{-9}	2.0437×10^{-9}	1.9225×10^{-9}	1.6499×10^{-9}	1.5704×10^{-9}
0.5	2.3096×10^{-9}	2.5506×10^{-9}	2.3994×10^{-9}	2.0592×10^{-9}	1.9600×10^{-9}
0.6	2.7670×10^{-9}	3.0556×10^{-9}	2.8745×10^{-9}	2.4671×10^{-9}	2.3482×10^{-9}
0.7	3.2222×10^{-9}	3.5584×10^{-9}	3.3474×10^{-9}	2.8729×10^{-9}	2.7345×10^{-9}
0.8	3.6763×10^{-9}	4.0598×10^{-9}	3.8191×10^{-9}	3.2778×10^{-9}	3.1199×10^{-9}
0.9	4.1288×10^{-9}	4.5595×10^{-9}	4.2892×10^{-9}	3.6813×10^{-9}	3.5040×10^{-9}
1	4.5789×10^{-9}	5.0566×10^{-9}	4.7568×10^{-9}	4.0826×10^{-9}	3.8859×10^{-9}

Table 6. Comparison of the absolute errors among third-order schemes in Problem 2, for $h = 25 \times 10^{-3}$.

x_i	Ralston's Scheme	RK3 Scheme	ARK3 Scheme	Heun's Scheme	VRK3 Scheme
zero	zero	zero	zero	zero	zero
0.1	5.7938×10^{-11}	6.4006×10^{-11}	6.0197×10^{-11}	5.1628×10^{-11}	4.9127×10^{-11}
0.2	1.1617×10^{-10}	1.2828×10^{-10}	1.2069×10^{-10}	1.0360×10^{-10}	9.8595×10^{-11}
0.3	1.7408×10^{-10}	1.9220×10^{-10}	1.8083×10^{-10}	1.5525×10^{-10}	1.4776×10^{-10}
0.4	2.3087×10^{-10}	2.5497×10^{-10}	2.3985×10^{-10}	2.0580×10^{-10}	1.9583×10^{-10}
0.5	2.8851×10^{-10}	3.1858×10^{-10}	2.9971×10^{-10}	2.5724×10^{-10}	2.4477×10^{-10}
0.6	3.4589×10^{-10}	3.8193×10^{-10}	3.5934×10^{-10}	3.0846×10^{-10}	2.9351×10^{-10}
0.7	4.0254×10^{-10}	4.4454×10^{-10}	4.1820×10^{-10}	3.5894×10^{-10}	3.4157×10^{-10}
0.8	4.5941×10^{-10}	5.0736×10^{-10}	4.7731×10^{-10}	4.0967×10^{-10}	3.8989×10^{-10}
0.9	5.1622×10^{-10}	5.7003×10^{-10}	5.3629×10^{-10}	4.6035×10^{-10}	4.3809×10^{-10}
1	5.7196×10^{-10}	6.3162×10^{-10}	5.9421×10^{-10}	5.0994×10^{-10}	4.8530×10^{-10}

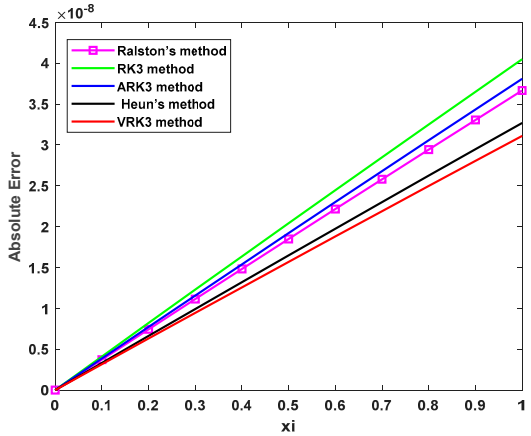

Figure 5. The absolute errors for numerical results in Table 4.

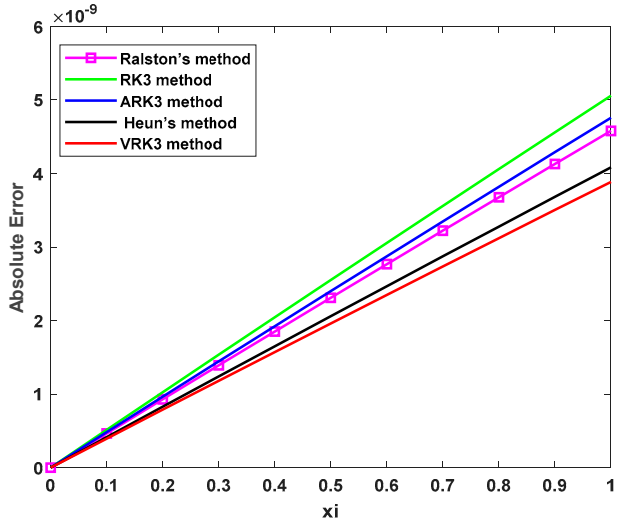

Figure 6. The absolute errors for numerical results in Table 5.

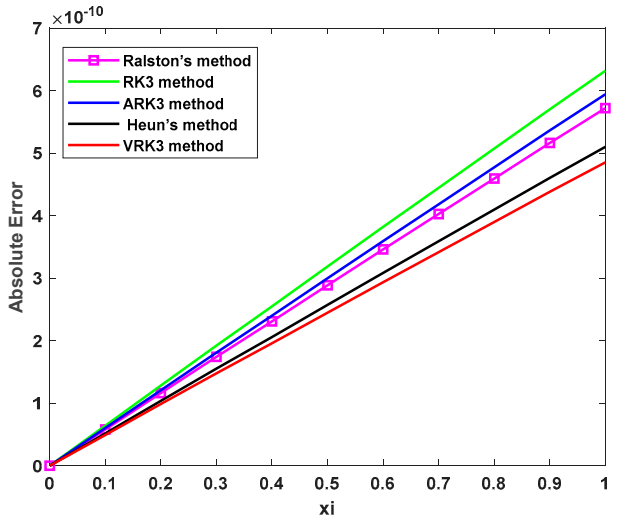

Figure 7. The absolute errors for numerical results in Table 6.

Table 7. Comparisons of CPU time in Problem 1, for different step sizes h.

Step Size	CPU Time				
	Ralston's Scheme	**RK3 Scheme**	**ARK3 Scheme**	**Heun's Scheme**	**VRK3 Scheme**
$h = 0.1$	0.003325	0.003508	0.004631	0.005653	0.001558
$h = 0.05$	0.003407	0.004827	0.003017	0.005066	0.001027
$h = 0.025$	0.003655	0.003144	0.005021	0.004364	0.001273

Table 8. Comparisons of CPU time in Problem 2, for different step sizes h.

Step Size	CPU Time				
	Ralston's Scheme	**RK3 Scheme**	**ARK3 Scheme**	**HEUN'S SCHEME**	**VRK3 Scheme**
$h = 0.1$	0.004160	0.003116	0.004698	0.004627	0.001358
$h = 0.05$	0.003328	0.004782	0.004879	0.003469	0.001003
$h = 0.025$	0.003506	0.004392	0.003230	0.004234	0.001296

8. Discussion and Conclusions

In this study, we introduced an innovative third-order method designed for solving initial value problems (IVPs). Our approach is rooted in a novel adaptation of the standard formulation employed in Runge–Kutta methods, incorporating Taylor series expansion. To validate the effectiveness of this new method, we employed two distinct numerical models, effectively showcasing its fundamental capabilities. It is important to underscore that all our numerical findings, including the accompanying tables and figures, were calculated using MATLAB (R2022a) software on a dedicated computer system operating with Windows 11 Pro. The system uses an 11th Generation Intel(R) Core (TM) i7-11800H processor running at 2.30 GHz, backed by 16.0 GB of RAM (15.7 GB usable).

A comprehensive numerical assessment was conducted using Tables 1–6, which present an intricate comparison of absolute errors across various step sizes, specifically $h = 10^{-1}$, $h = 5 \times 10^{-2}$, and $h = 25 \times 10^{-3}$. Through the graphical representations found in Figures 2–7, we were able to discern that our novel method, referred to as VRK3, consistently outperformed several benchmark techniques including Ralston's scheme, RK3 scheme, ARK3 scheme, and Heun's scheme. This superiority primarily stems from the reduced local truncation error of VRK3. Additionally, our investigation revealed a significant insight regarding the impact of step size on accuracy. As we decreased the step size, the error progressively approached zero, strongly indicating that precision increased with smaller step sizes. This observation reinforces the importance of carefully selecting step sizes to achieve higher levels of accuracy in numerical solutions. Turning our attention to computational efficiency, Tables 7 and 8 provided valuable insights. The VRK3 scheme consistently demonstrated reduced CPU time compared to its counterparts, further validating its utility in practical applications. Furthermore, Figure 1 depicts the stability region of our third-order VRK3 scheme, establishing its equivalence to similar methodologies. Importantly, we substantiated the convergence of our VRK3 scheme, as it satisfies both the consistency and stability criteria.

In conclusion, our newly proposed third-order method exhibits a commendable blend of efficiency and reliability. The method's stability and high accuracy render it particularly robust for a wide range of applications. This research contributes to the field of numerical methods for IVPs by presenting an innovative approach that holds promise for improving computational accuracy and efficiency. Future research directions might explore the extension of this method to more complex problems or its integration into broader computational frameworks.

Author Contributions: Conceptualization, N.Y.A.-H., Z.J.K. and A.H.A.; formal analysis, N.Y.A.-H., Z.J.K. and A.H.A.; investigation, N.Y.A.-H., Z.J.K. and A.H.A.; methodology, N.Y.A.-H., Z.J.K. and A.H.A.; software, A.H.A.; supervision, N.Y.A.-H.; writing—original draft, N.Y.A.-H. and Z.J.K.; writing—review and editing, A.H.A. All authors have read and agreed to the published version of the manuscript.

Funding: The author received no direct funding for this work.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The author would like to express gratitude to the anonymous referees for their valuable comments and suggestions.

Conflicts of Interest: The authors declare that they have no competing interests.

References

1. Ralston, A. Runge-Kutta Methods with Minimum Error Bounds. *Math. Comput.* **1962**, *16*, 431–437. [CrossRef]
2. Ralston, A.; Rabinowitz, P. *A First Course in Numerical Analysis*, 2nd ed.; McGraw-Hill: New York, NY, USA, 1978.

3. Wazwaz, A.M. A Comparison of Modified Runge-Kutta Formulas Based on a Variety of Means. *Int. J. Comput. Math.* **1994**, *50*, 105–112. [CrossRef]
4. Hatun, M.; Vatansever, F. Differential Equation Solver Simulator for Runge-Kutta Methods. *Uludağ Univ. J. Fac. Eng.* **2016**, *21*, 145–162.
5. Burden, R.L.; Faires, J.D. *Numerical Analysis*, 9th ed.; Brooks/Cole Publishing Company: Pacific Grove, CA, USA, 2011.
6. Ahmad, N.; Charan, S. A Comparative Study on Numerical Solution of Ordinary Differential Equation by Different Method with Initial Value Problem. *Int. J. Recent. Sci. Res.* **2017**, *8*, 21134–21139.
7. Nhawu, G.; Mafuta, P.; Mushanyu, J. The Adomian Decomposition Method for Numerical Solution of First-Order Differential Equations. *J. Math. Comput. Sci.* **2016**, *6*, 307–314.
8. Nagle, R.K.; Saff, E.B.; Snider, A.D. *Fundamentals of Differential Equations*, 9th ed.; Pearson: Boston, MA, USA, 2018.
9. Wusu, A.S.; Akanbi, M.A.; Okunuga, S.A. A Three-stage Multiderivative Explicit Runge-Kutta Method. *Am. J. Comput. Math.* **2013**, *3*, 121–126.
10. Ali, A.H.; Pales, Z.S. Taylor-type Expansions in Terms of Exponential Polynomials. *Math. Inequalities Appl.* **2022**, *25*, 1123–1141. [CrossRef]
11. Kadum, Z.J.; Abdul-Hassan, N.Y. New Numerical Methods for Solving the Initial Value Problem Based on a Symmetrical Quadrature Integration Formula Using Hybrid Functions. *Symmetry* **2023**, *15*, 631. [CrossRef]
12. Corless, R.M.; Kaya, C.Y.; Moir, R.H.C. Optimal Residuals and the Dahlquist Test Problem. *Numer. Algorithms* **2019**, *81*, 1253–1274. [CrossRef]
13. Lambert, J.D. *Computational Methods in Ordinary Differential Equations*; John Wiley & Sons Inc.: New York, NY, USA, 1973.
14. Ram, T.; Solangi, M.A.; Sanghah, A.A. A Hybrid Numerical Method with Greater Efficiency for Solving Initial Value Problems. *Math. Theory Model.* **2020**, *10*, 2224–5804.
15. Omar, Z.; Adeyeye, O. Numerical Solution of First Order Initial Value Problems Using a Self-Starting Implicit Two-Step Obrechkoff-Type Block Method. *J. Math. Stat.* **2016**, *12*, 127–134. [CrossRef]