IHJPAS. 37 (2) 2024
Ibn Al-Haitham Journal for Pure and Applied Sciences
Journal homepage: jih.uobaghdad.edu.iq
PISSN: 1609-4042, EISSN: 2521-3407

Efficient Embedded Diagonal Implicit Runge-Kutta Method for Directly Solving Third Order ODEs

Firas A. Fawzi*[1], Hadeer M. Globe[2] and Nizam G. Ghawadri[3]
[1,2] Department of Mathematics, Faculty of Computer Science and Mathematics, Tikrit University, Salah Al-Din, Iraq.
[3] Doctor of Mathematics, Ministry of Education, Jenin, Palestine.
*Corresponding Author.

Received: 13 April 2023 Accepted: 12 June 2023 Published: 20 April 2024
doi.org/10.30526/37.2.3409

Abstract

This paper presents two important contributions to the field of numerical analysis for third-order ordinary differential equations (ODEs). First, a new class of direct implicit Runge-Kutta (RK) processes, called RKTDIO, is introduced as solutions to third-order ODEs. Secondly, it develops the ERKTDIO method, which is an embedded pairwise diagonal implicit RK method. The study begins by introducing the theory of relevant-colored trees and B-series as fundamental concepts. By utilizing the order constraints, two RKTDIO methods are derived: a fifth-order method with three stages and a sixth-order method with four stages. In addition, an embedded method called ERKTDIO6(5) is derived, which has orders six and five. The derivation of the embedded method includes strategies to ensure that the higher-order method achieves high accuracy while the lower-order method provides optimal error estimates. To evaluate the effectiveness of the proposed methods, variable step-size codes are developed and applied to a set of specific third-order problems. The numerical evaluation involves converting the problems into a system of first-order ODEs and comparing the results with existing methods in terms of accuracy and function evaluations. The numerical demonstrations emphasise the superior performance and efficiency of the new methods in solving third-order ODEs. The comparative analysis shows the accuracy achieved by the higher-order method and the improved error estimation of the lower-order method. The results validate the efficacy of the proposed approaches and their potential for practical applications in various domains.

Keywords: Third-order ODEs, order conditions, B-series, Relevant-colored trees.

1. Introduction

Third-order differential equations can be found in various fields, such as applied sciences, neural network engineering [1,2], fluid dynamics [3] and thin film flow [4]. The aim of this paper is to develop and explain a computational method for solving initial value problems of third-order differential equations.

$$\alpha'''(x) = \mu(x, \alpha(x)), \qquad x \geq x_0, \tag{1}$$

With initial conditions

$$\alpha(x_n) = \alpha_n , \quad \beta(x_n) = \alpha'_n , \quad \gamma(x_n) = \alpha''_n .$$

Where $(x) \in R^d$, $f\colon R^d \times R^d \to R^d$ has a continuous value but lacks the first and second derivatives. Numerical solutions are often necessary for third-order differential equations since analytical solutions are often not available. Some researchers have used classical methods to solve higher-order differential equations by converting them into a system of first-order differential equations [5-16,27]. However, this method can be computationally intensive and time consuming. Direct numerical approaches have been proposed to reduce the computation time, but they require a second method to obtain initial values for the numerical solutions. Implicit methods are useful because they can achieve high accuracy with fewer steps, making it easier to find solutions to complex problems. Several researchers have developed embedded Runge-Kutta methods with high algebraic orders for solving third-order differential equations [17-18]. Ismail et. al [19] proposed the Singly Embedded Diagonally Implicit Runge-Kutta (SDIRK) method to combine delay differential equations (DDEs) and compared the computational results. The researchers in [20,21,26] developed a novel embedded explicit and implicit Runge-Kutta method for solving special third and fourth-order problems.

The main objective of this work is to develop a new method called RKTDIO, an implicit one-step Runge-Kutta method designed for directly solving specific third-order differential equations. This method is developed using the theory of relevant-colored trees theory and also involves the derivation of embedded diagonally implicit Runge-Kutta methods for the direct integration of certain third-order differential equations.

The structure of this paper is as follows: Section 2 provides the formulation concept for the RKTDIO approach for the direct integration of certain third-order ODEs. In Section 3, we develop the novel theory of relevant-colored trees theory and the corresponding B-series theory. Section 4 contains the derivation of the ordering criteria of the RKTDIO method. Section 5 presents the construction of a three-stage RKTDIO approach for order five and a four-stage method for order six. Section 6 describes the derivation of the embedded ERKTDIO6(5) method. To demonstrate the efficiency and effectiveness of the RKTDIO and ERKTDIO6(5) methods compared to the methods currently used in the scientific literature, we give numerical findings in Section 7. Section 8 provides conclusions.

2. The formulation of RKTDIO method

By turning it into a system of first-order ODEs, the special third-order IVP (1) can be solved as follows:

$$\begin{pmatrix} \alpha(x) \\ \beta(x) \\ \gamma(x) \end{pmatrix}' = \begin{pmatrix} \beta(x) \\ \gamma(x) \\ \mu(x, \alpha(x)) \end{pmatrix} \tag{2}$$

With initial conditions

$\alpha(x_n) = \alpha_n$, $\beta(x_n) = \alpha'_n$, $\gamma(x_n) = \alpha''_n$.

The Runge-Kutta method first-order is used to obtain the following system of equations

$$\alpha_{i=} \alpha_n + h \sum_{j=1}^{s} a_{ij}\alpha'_j , \tag{3}$$

$$\alpha'_i = \alpha'_n + h \sum_{j=1}^{s} a_{ij}\alpha''_j , \tag{4}$$

$$\alpha''_i = \alpha''_n + h \sum_{j=1}^{s} a_{ij}\, \mu\left(x_n + c_j h, \alpha_j\right) , \tag{5}$$

$$\alpha_{n+1} = \alpha_n + h \sum_{i=1}^{s} b_i \alpha'_i \quad , \tag{6}$$

$$\alpha'_{n+1} = \alpha'_n + h \sum_{i=1}^{s} b_i \alpha''_i \;, \tag{7}$$
$$\alpha''_{n+1} = \alpha''_n + h \sum_{i=1}^{s} b_i \mu\left(x_n + c_i h, \alpha_i\right) . \tag{8}$$

When we ignore α'_i, α''_i and α'''_i, from (3) – (8) we conclude

$$\alpha_i = \alpha_n + h \sum_{j=1}^{s} a_{ij}\, \alpha'_n + h^2 \sum_{j,k=1}^{s} a_{ij}\, \alpha_{jk} \alpha''_n$$
$$+ \; h^3 \sum_{j,k,l}^{s} a_{ij} a_{jk} a_{kl}\, \mu(x_n + c_i h, \alpha_l), \qquad i = 1, \dots, s. \tag{9}$$

$$\alpha_{n+1} = \alpha_n + h \sum_{i=1}^{s} b_i \alpha'_n + h^2 \sum_{i,j=1}^{s} b_i\, a_{ij} \alpha''_n$$
$$+ h^3 \sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk}\, \mu(x_n + c_k h, \alpha_k), \tag{10}$$

$$\alpha'_{n+1} = \alpha'_n + h \sum_{i=1}^{s} b_i \alpha''_n + h^2 \sum_{i.j=1}^{s} b_i a_{ij}\, \mu\left(x_n + c_j h\,, \alpha_j\right), \tag{11}$$

$$\alpha''_{n+1} = \alpha''_n + h \sum_{i=1}^{s} b_i\, \mu(x_n + c_i h,\ \alpha_i) \;. \tag{12}$$

We assume that

$$\sum_{j=1}^{s} a_{ij} = c_i, \qquad \sum_{j,k=1}^{s} a_{ij} a_{jk} = \frac{1}{2}\, {c_i}^2, \qquad \sum_{i=1}^{s} b_i = 1, \qquad \sum_{i,j=1}^{s} b_i a_{ij} = \frac{1}{2},$$
$$\sum_{j=1}^{s} b_i a_{ij} = b''_i \;, \quad \sum_{j,k=1}^{s} b_j a_{jk} a_{kl} = b''_i \;, \sum_{k,l,r=1}^{s} a_{ik}\, a_{kl} a_{lr} = \hat{a}_{ij} \;, \qquad i = 1, \dots, s.$$

As a result, we are able to solve the specific third-order IVP (1), indicated by the RKTDIO approach, using the following direct integration method. Thus, the following formula is used to represent the s-stage RKTDIO approach for the numerical solution of the IVP (1):

$$\alpha_{n+1} = \alpha_n + h\alpha'_n + \frac{1}{2}\, h^2 \alpha''_n + h^3 \sum_{i=1}^{s} b_i\;\; \mu(x_n + c_i h, \alpha_i), \tag{13}$$

$$\alpha'_{n+1} = \alpha'_n + h\alpha''_n + h^2 \sum_{i=1}^{s} b'_i\, \mu(x_n + c_i h,\ \alpha'_i) \,, \tag{14}$$

$$\alpha''_{n+1} = \alpha''_n + h \sum_{i=1}^{s} b''_i\;\; \mu\left(x_n + c_i h, \alpha'_i\right) , \tag{15}$$

$$\alpha_i = \alpha_n + c_i h \alpha'_n + \frac{1}{2} {c_i}^2 h^2 \alpha''_n + h^3 + \sum_{i,j=1}^{s} \hat{a}_{ij}\;\; \mu(x_n + c_i h,\ \alpha'_j). \tag{16}$$

All RKTDIO parameters a_{ij}, b_i, b'_i, b''_i, and c_i are real numbers and i, j = 1, 2, . . . , s.

The RKTDIO method (13)–(16) can be expressed in Butcher tableau as follows

$$
\begin{array}{c|ccc}
c_1 & \hat{a}_{11} & \cdots & \hat{a}_{1s} \\
\vdots & \vdots & \ddots & \vdots \\
c_s & \hat{a}_{s1} & \cdots & \hat{a}_{ss} \\
\hline
 & b_1 & \cdots & b_s \\
 & b'_1 & \cdots & b'_s \\
 & b''_1 & \cdots & b''_s
\end{array}
\quad :
$$

3. B-series and associated relevant-colored trees

The essential definitions of lemmas and associated theorems that are utilized throughout this article will be covered in this part.

Definition 3.1 [22,23]: **Order of the RKTDIO Method for Third-Order ODEs**

The RKTDIO method (13)-(16) has order p when the third-order ODE (1) with the assumption $\alpha(x_n) = \alpha_n,\ \alpha'(x_n) = \alpha'_n,\ \alpha''(x_n) = \alpha''_n$ hence the local truncation error norms of the exact solution and the first, and second derivatives of the solution must satisfied.

$$\alpha(x_n + h) - \alpha_{n+1} = O(h^{p+1}),$$
$$\alpha'(x_n + h) - \alpha'_{n+1} = O(h^{p+1}),$$
$$\alpha''(x_n + h) - \alpha''_{n+1} = O(h^{p+1}), \tag{17}$$

The following autonomous form of third-order IVP must be used in order to derive the algebraic order conditions for the RKTDIO technique (13)–(16).

$$\alpha^{(3)}(x) = \rho\big(\alpha(x)\big), \tag{18}$$

With initial conditions

$$\alpha(x_n) = \alpha_n, \quad \alpha'(x_n) = \alpha'_n, \quad \alpha''(x_n) = \alpha''_n.$$

By extending IVP (1) with a one-dimensional vector $w = x$, the autonomous problem can be expressed equivalently to the third-order initial value problem (1) as follows:

$$w^3 = 0, \tag{19}$$
$$\alpha^{(3)} = \rho(w, \alpha), \tag{20}$$
$$w(x_n) = w_n \quad w'(x_n) = w'_n = 1, \quad w''(x_n) = w''_n = 0, \tag{21}$$
$$\alpha(x_n) = \alpha_n, \quad \alpha'(x_n) = \alpha'_n, \quad \alpha''(x_n) = \alpha''_n. \tag{22}$$

Applying RKTDIO method (13)–(16) to the scheme (19)–(22), we obtain

$$W_i = w_n + c_i h\, w'_n + \frac{1}{2} c_i^2 h^2\, w''_n\,, \tag{23}$$
$$\alpha_i = \alpha_n + c_i h \alpha'_n + \frac{1}{2} c_i^2 h^2\, \alpha''_n + h^3 \sum_{i,j=1}^{s} a_{ij}\, \mu\big(W_j, \alpha_j\big)\,, \tag{24}$$
$$w_{n+1} = w_n + h\, w'_n + \frac{1}{2} h^2 w''_n\,, \tag{25}$$
$$w'_{n+1} = w'_n + h\, w''_n\,, \tag{26}$$
$$w''_{n+1} = w''_n\,, \tag{27}$$
$$\alpha_{n+1} = \alpha_n + h\, \alpha'_n + \frac{1}{2} h^2 \alpha''_n + h^3 \sum_{i=1}^{s} b_i\, \mu(W_i, \alpha_i), \tag{28}$$
$$\alpha'_{n+1} = \alpha'_n + h\, \alpha''_n + h^2 \sum_{i=1}^{s} b'_i\, \mu(W_i, \alpha_i)\,, \tag{29}$$

$$\alpha''_{n+1} = \alpha''_n + h\sum_{i=1}^{s} b''_i \ \mu(W_i, \alpha_i). \tag{30}$$

Substituting Eq. (21) into system of Eqs. (23)–(30), we get

$$W_i = x_n + c_i h, \tag{31}$$
$$w_{n+1} = x_n + h, \tag{32}$$
$$w'_{n+1} = 1, \tag{33}$$
$$w''_{n+1} = 0, \tag{34}$$
$$\alpha_{n+1} = \alpha_n + h\alpha'_n + \frac{1}{2} h^2\alpha''_n + h^3 \textstyle\sum_{i=1}^{s} b_i \ \mu(x_n + c_i h, \alpha_i), \tag{35}$$
$$\alpha'_{n+1} = \alpha'_n + h\alpha''_n + h^2 \textstyle\sum_{i=1}^{s} b'_i \ \mu(x_n + c_i h, \alpha_i), \tag{36}$$
$$\alpha''_{n+1} = \alpha''_n + h \textstyle\sum_{i=1}^{s} b''_i \ \mu(x_n + c_i h, \alpha_i), \tag{37}$$
$$\alpha_i = \alpha_n + c_i h\alpha'_n + \frac{1}{2} c_i^2 h^2 \alpha''_n + h^3 \textstyle\sum_{i,j=1}^{s} \hat{a}_{ij} \ \mu(x_n + c_i h\,, \alpha_j). \tag{38}$$

We find that the system of equations (13)–(16), which is generated by using the RKTDIO approach on the non-autonomous problem (1), is totally similar to Eqs. (35)– (38). Hence, discussing the numerical solutions of autonomous form is sufficient (18). Hence, the RKTDIO method (13)–(16) can be implemented as follows.

$$\alpha_{n+1} = \alpha_n + h\,\alpha'_n + \frac{1}{2} h^2\alpha''_n + h^3 \sum_{i=1}^{s} b_i \ \mu(\alpha_i),$$

$$\alpha'_{n+1} = \alpha'_n + h\alpha''_n + h^2 \sum_{i=1}^{s} b'_i \ \mu(\alpha_i),$$

$$\alpha''_{n+1} = \alpha''_n + h \sum_{i=1}^{s} b''_i \ \mu(\alpha_i),$$

$$\alpha_i = \alpha_n + c_i h\alpha'_n + \frac{1}{2} c_i^2 h^2 \alpha''_n + h^3 \textstyle\sum_{i,j=1}^{s} \hat{a}_{ij} \ \mu(\alpha_j). \tag{39}$$

The following elementary differentials are obtained by using the elementary differential notation on the analytical solution $\alpha(x)$. [15]

$$\alpha^{(1)} = \alpha', \qquad \alpha^{(2)} = \alpha'', \qquad \alpha^{(3)} = \mu,$$
$$\alpha^{(4)} = \mu'\alpha', \qquad \alpha^{(5)} = \mu''(\alpha', \alpha') + \mu'\alpha''(\alpha', \alpha'),$$
$$\alpha^{(6)} = 3\alpha''(\alpha', \alpha'') + \mu'''(\alpha', \alpha', \alpha') + \mu'\alpha''. \tag{40}$$

These processes very quickly get more difficult as the order increases. The optimum method for overcoming this challenge, according to [25], will be to use a graphical representation with a few modifications for third-order ODEs, denoted by relevant-colored trees. The three sorts of nodes in the relevant-colored trees are "meager," "black ball", and "white ball", and they are connected by arcs. In these trees, we specifically use the end meager node to denote each α', the end black ball node to denote each α'', the end white ball to denote each μ, and each arc to denote each arc, leaving this node to represent the $m-th$ derivative of μ with respect to α. In addition, the symbols t_1 and t_2 denote the first-order and the second-order tree and $t3$ the third-order tree, respectively (see **Figure 1**)

$\tau_1 =$ 	$\tau_2 =$ 	$\tau_3 =$ 

Figure 1. The relevant-colored trees

Here, we'll go through some essential definitions for the relevant-colored trees and related B-series that are necessary for this work.

Definition 3.2 [22,23]: **Relevant-Colored Trees (RT) and Meager Node**

The following definitions are repeated for the set of relevant-colored trees (RT):

1- The trees t_1, t_2 and t_3 above are all in RT, and the tree t_1 includes only one meager node (known as the root).

2- If $\tau_1, \tau_2, \dots, \tau_m \in RT$, then $\tau = [\tau_1, \tau_2, \dots, \tau_m]_3$ is the tree obtained by linking the roots $\tau_1, \tau_2, \dots, \tau_m$ and the root is the ‘‘meager node’’ t_1 is at the bottom. The subscript 3 is to mention that the trees of the roots of $\tau_1, \tau_2, \dots, \tau_m$ onto the tree t_3 contain a chain of three nodes.

We will employ the following principles to create the appropriate colored trees:

1- The ‘‘meager’’ node is always the root.

2- A ‘‘meager’’ node has only single child and that child should be ‘‘black ball’’.

3- A ‘‘black ball’’ node has only single child and that child should be ‘‘white ball’’.

Definition 3.3 [13,14]: **Order Function $\boldsymbol{\rho(\tau)}$ for Relevant-Colored Trees (RT)**

The order $\rho(\tau)$ and symmetry $\sigma(\tau)$ functions are defined recursively as follows:

1- $\rho(t_1) = 1,\ \rho(t_2) = 2,\ \rho(t_3) = 3,$

2- $\sigma(t_1) = 1,\ \sigma(t_2) = 1,\ \sigma(t_3) = 1,$

3- If $\tau = [\tau_1, \tau_2, \dots, \tau_m]_3$ for each $\tau \in RT$, then $\rho(\tau) = 3 + \sum_{i=1}^{m} \rho(\tau_i)$ and $\sigma(\tau) = \prod_{i=1}^{m} \sigma(\tau_i)(v_1!\ v_2! \dots)$ where $\rho(\tau)$ is the number of nodes of τ, $\forall \tau \in$ RT and $\nu 1!\ \nu 2! \dots$ count equal trees among $\tau_1, \tau_2, \dots, \tau_m$. Then we can define the set Sr which consist of every trees RT of order r.

Definition 3.4 [22,23]: **Elementary Differential and B-Series on Relevant-Colored Trees (RT) for the RKTDIO Approach**

The elementary differential for every tree $\tau \in RT$ is a function $F(\tau): R^d \times R^d \times R^d \to R^d$, recursively defined on RT as follows

1- $\mathbb{U}(t_1)(\alpha, \alpha', \alpha'') = \alpha',\ \mathbb{U}(t_2)(\alpha, \alpha', \alpha'') = \alpha'', \mathbb{U}(t_3)\ (\alpha, \alpha', \alpha'') = \mu(\alpha),$

2- $\mathbb{U}(\tau) = \mu^{(m)}(\alpha)\ (\mathbb{U}(\tau_1)(\alpha, \alpha', \alpha''), \dots, \mathbb{U}(\tau_m)(\alpha, \alpha', \alpha''))$ for $\tau = [\tau_1, \tau_2, \dots, \tau_m]_3$.

We expanded these definitions to provide the definition of B-series on the set RT of the relevant-colored trees for the RKTDIO approach, which was motivated by the definitions of B-series on the root trees in [23] and the tri-colored trees in [24].

Definition 3.5 [22,23]: **B-Series Representation**

Let $\delta: RT \cup \{\emptyset\} \to R^d$ be a mapping, then we can give the form of a formal series as follows:

$$B(\delta, y) = \delta(\emptyset) + \sum_{\tau \in RT} \frac{h^{\rho(\tau)}}{\sigma(\tau)} \delta(\tau)\ \mathbb{U}(\tau)(\alpha, \alpha'\alpha''), \tag{41}$$

Which is called the B-series.

We present the following crucial lemma that is important to this derivation in order to accomplish the main goal of this research, which is the derivation of the order conditions of the RKTDIO technique.

Lemma 3.1 [22,23]: Let δ be a function $\delta : RT \cup \{\emptyset\} \to R^d$ with $\delta(\emptyset) = 1$. Thus $h^3\mu(B(\delta,\alpha))$ is also a B-series $h^3\mu\big(B(\delta,\alpha)\big) = B(\delta',\alpha)$ where $\delta'(\emptyset) = 0$, $\delta'(t_1) = 0$, $\delta'(t_2) = 0$, $\delta'(t_3) = 1$, and for $\tau = [\tau_1,\tau_2,\dots,\tau_m]_3 \in RT$, $\delta'(\tau) = \delta(\tau_1),\dots\delta(\tau_m)$.

Lemma 3.2 [22,23]: If we suppose that the analytic solution of (18) is a B-series $B(\vartheta,\alpha_0)$ with a real function ϑ which is defined on $RT \cup \{\emptyset\}$, then

$\vartheta(\emptyset) = 1$, $\vartheta(t_1) = 1$, $\vartheta(t_2) = 1$, $\vartheta(t_3) = 1$,

And $\tau = [\tau_1,\tau_2,\dots,\tau_m]_3 \in RT$, we have

$$\vartheta(\tau) = \frac{1}{\rho(\tau)(\rho(\tau)-1)(\rho(\tau)-2)}\big(\vartheta(\tau_1),\dots\vartheta(\tau_m)\big).$$

Proposition 3.2.1 [22,23]: The density $\sigma(\tau)$ is the non negative integer factors defined on trees RT, $\forall\tau \in RT$ satisfy

1- $\sigma(t_1) = 1$, $\sigma(t_2) = 2$, $\sigma(t_3) = 6$,
2- with $\tau = [\tau_1,\tau_2,\dots,\tau_m]_3$, we have $\sigma(\tau) = \rho(\tau)(\rho(\tau)-1)(\rho(\tau)-2)\big(\sigma(\tau_1),\dots,\sigma(\tau_m)\big)$.

Proposition 3.2.2 [22,23]: The non-negative integer $\varepsilon(\tau)$, $\forall\tau \in RT$ satisfy

1- $\epsilon(t_1) = 1$, $\varepsilon(t_2) = 1$, $\epsilon(t_3) = 1$,
2- For the tree $\tau = \left[\tau_1^{v_1},\dots,\tau_m^{v_m}\right]_3 \in RT$, with distinct τ_i we have $\varepsilon(\tau) = (\rho(\tau)-3)!\,\prod_{i=1}^{m}\frac{1}{v_i}\left(\frac{\varepsilon(\tau_i)}{\rho(\tau_i)!}\right)^{v_i}$, where v_i count similar tree of τ_i, $i = 1,\dots,m$.

Therefore we can represent B-series (41) as follows:

$$B(\delta,\alpha) = \delta(\emptyset) + \textstyle\sum_{\tau\in RT}\frac{h^{\rho(\tau)}}{\rho(\tau)!}\delta(\tau)\,\varepsilon(\tau)\,\sigma(\tau)\,\mu(\tau)(\alpha,\alpha',\alpha''). \tag{42}$$

The previous analysis results in the following theorem.

Theorem 3.1 [22,23]: The exact solution of (18) is a B-series

$$\alpha(x_0+h) = \textstyle\sum_{\tau\in RT}\frac{h^{\rho(\tau)}}{\rho(\tau)!}\varepsilon(\tau)\mathbb{U}(\tau)(\alpha_0,\alpha_0',\alpha_0''), \tag{43}$$

And the first and second derivatives have the following B-series respectively,

$$\alpha'(x_0+h) = \textstyle\sum_{\tau\in RT}\frac{h^{\rho(\tau)-1}}{(\rho(\tau)-1)!}\varepsilon(\tau)\mathbb{U}(\tau)(\alpha_0,\alpha_0',\alpha_0''), \tag{44}$$

$$\alpha''(x_0+h) = \textstyle\sum_{\tau\in RT/[t_1]}\frac{h^{\rho(\tau)-2}}{(\rho(\tau)-2)!}\varepsilon(\tau)\mathbb{U}(\tau)(\alpha_0,\alpha_0',\alpha_0''). \tag{45}$$

Lemma 3.3 [22,23]: We can calculate the function $\eta_i(\tau)$ on $\in RT/[t_1,t_2]$ recursively as

1- $\eta_i(t_3) = 1$,
2- For the tree $\tau = [{\tau_1}^{v_1},{t_2}^{v_2},{t_3}^{v_3},\dots,{\tau_m}^{v_m}]_3 \in RT$, with distinct τ_i, $i = 1,\dots,m$ and different t_1, and t_2, $\eta_j = \frac{1}{2^{v_2}}{c_j}^{v_1+2v_2}\prod_{i=3}^{m}(\sum_{k=1}^{s}\hat{a}_{jk}\eta_k(\tau_i))^{v_i}$.

Now, we denote the vector $\eta(\tau) = (\eta_1(\tau)(\tau),\dots,\eta_s(\tau)(\tau))^T, \forall\alpha \in RT\backslash\{t_1,t_2\}$. the initial weight associated to α_{n+1} is indicated by $\partial(\tau)$ and is defined as follows:

$\partial(\tau) = \sum_{i=1}^{s} s_i\,\eta_i(\tau) = s^T\,\eta(\tau)$,

$\partial'(\tau)$ is indicated to the initial weight associated with ${\alpha'}_{n+1}$ and is defined as follows:

$\partial'(\tau) = \sum_{i=1}^{s} s_{i'}\,\eta_i(\tau) = {s'}^T\,\eta(\tau)$,

and the initial weight associated with ${\alpha''}_{n+1}$ indicated by $\partial''(\tau)$ and is defined as follows:

$\partial''(\tau) = \sum_{i=1}^{s} s_{i''}\, \eta_i(\tau) = s''^{T}\, \eta(\tau)$.

As a result, we obtain the following fundamental theorem for the numerical solution and RKTDIO technique numerical derivatives.

Theorem 3.2 [22,23]: When we apply the RKTDIO method (39) on the autonomous problem (18) yields the numerical solution α_{n+1} and numerical derivatives $\alpha'_{n+1}, \alpha''_{n+1}$ which have the B-series as follows:

$$\alpha_{n+1} = \alpha_n + h\alpha'_n + \frac{1}{2}h^2\alpha''_n + \sum_{\tau\in\frac{RT}{\{t_1,t_2\}}} \frac{h^{\rho(\tau)}}{\rho(\tau)!}\varepsilon(\tau)\sigma(\tau)\theta(\tau).\,\mathbb{U}(\tau)(\alpha_0, \alpha'_0, \alpha''_0), \tag{45}$$

$$\alpha'_{n+1} = \alpha'_n + h\alpha''_n + \sum_{\tau\in RT/\{t_1,t_2\}} \frac{h^{\rho(\tau)-1)}}{\rho(\tau)!}\varepsilon(\tau)\sigma(\tau)\theta'^{(\tau)}.\,\mathbb{U}(\tau)(\alpha_0, \alpha'_0, \alpha''_0), \tag{46}$$

$$\alpha''_{n+1} = \alpha''_n + \sum_{\tau\in RT/\{t_1,t_2\}} \frac{h^{\rho(\tau)-2)}}{\rho(\tau)!}\varepsilon(\tau)\sigma(\tau)\theta''(\tau).\,\mathbb{U}(\tau)(\alpha_0, \alpha'_0, \alpha''_0)\ . \tag{47}$$

4. Algebraic order conditions

We arrive at this paper's main contribution—the order conditions of the RKTDIO method-through Theorems 3.1 and 3.2. In **Table 1**, the relevant-colored trees of orders up to six are listed together with the accompanying function values.

Theorem 4.1: The RKTDIO method has order p $(p \geq 3)$ if and only if it satisfies the following conditions.

1- $\theta(\tau) = \frac{1}{\sigma(\tau)}, \tau \in \bigcup_{r=4}^{p} s_r,$ (48)

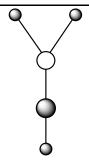
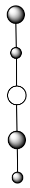
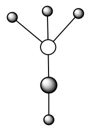
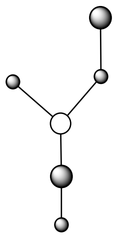
2- $\theta'(\tau) = \frac{\rho(\tau)}{\sigma(\tau)}, \tau \in \bigcup_{r=4}^{p+1} s_r,$ (49)

3- $\theta''^{(\tau)} = \frac{\rho(\tau)(\rho(\tau)-1)}{\sigma(\tau)}, \tau \in \bigcup_{r=4}^{p+2} s_r.$ (50)

Even though some of the trees in the set RT provide the same order criteria and pertain to various elementary differentials, it is still unnecessary. In general, the following corollary, which may be derived from the definition of density and order, and from Lemma 3.3, can be used to overcome the similarity between the order criteria.

Table 1. lists elementary differentials, relevant-colored trees of up to six orders, and related functions.

Order	τ	Tree	$\alpha(\tau)$	Density	$n(\tau)$	Elementary
0	$\emptyset$	$\emptyset$	1	1		α
1	t_1		1	1		α'
2	t_2		1	2		α''
3	t_3		1	6		μ
4	τ_{41}		1	24	c	$\mu'\alpha'$

5	τ_{51}		1	60	c^2	$\mu'\alpha''$
5	τ_{52}		1	120	$\frac{1}{2}c^2$	$\mu''(\alpha',\alpha')$
6	τ_{61}		3	120	c^3	$\mu''(\alpha',\alpha'')$
6	τ_{62}		1	240	$\frac{1}{2}c^3$	$\mu'''(\alpha',\alpha',\alpha')$
6	τ_{63}		1	720	$\frac{1}{6}c^3$	$\mu'\alpha''$

The order criteria for the RKTDIO technique up to the sixth order can be expressed as follows using Theorem 4.1 as a foundation:

Order condition for $\boldsymbol{\alpha}$

order 3

$$\sum k_i = \frac{1}{6} \tag{51}$$

Order 4

$$\sum k_i\, s_i = \frac{1}{24} \tag{52}$$

Order 5

$$\sum k_i {s_i}^2 = \frac{1}{60} \tag{53}$$

Order 6

$$\sum k_i {s_i}^3 = \frac{1}{120}, \qquad \sum k_i a_{ij} = \frac{1}{720} \tag{54}$$

Order condition for $\boldsymbol{\alpha'}$

Order 2

$$\sum k'_i = \frac{1}{2} \tag{55}$$

Order 3

$$\sum k_i'\, s_i = \frac{1}{6} \tag{56}$$

Order 4

$\sum k_i' s_i^2 = \frac{1}{12}$ (57)

Order 5

$\sum k_i' s_i^3 = \frac{1}{20}$, $\sum k_i' a_{ij} = \frac{1}{120}$ (58)

Order 6

$\sum k_i' s_i^4 = \frac{1}{30}$, $\sum k_i' a_{ij} s_j = \frac{1}{720}$, $\sum k_i' s_j a_{ij} = \frac{1}{180}$ (59)

Order condition for α''

Order 1

$\sum k_i'' = 1$ (60)

Order 2

$\sum k_i'' s_i = \frac{1}{2}$ (61)

Order 3

$\sum k_i'' s_i^2 = \frac{1}{3}$ (62)

Order 4

$\sum k_i'' s_i^3 = \frac{1}{4}$, $\sum k_i'' a_{ij} = \frac{1}{24}$ (63)

Order 5

$\sum k_i'' s_i^4 = \frac{1}{5}$, $\sum k_i'' a_{ij} s_j = \frac{1}{120}$, $\sum k_i'' s_j a_{ij} = \frac{1}{30}$ (64)

Order 6

$\sum k_i'' s_i^2 a_{ij} = \frac{1}{36}$, $\sum k_i'' s_i^5 = \frac{1}{6}$, $\sum k_i'' a_{ij} s_j^2 = \frac{1}{360}$, $\sum k_i'' s_i a_{ij} s_j = \frac{1}{144}$ (65)

5. The Construction of RKTDIO Method

When creating implicit RKTDIO methods, the order conditions listed in Section 3.2 must be met. For the q-order RKTDIO method, the local truncated error is defined as follows:

$$\left\|L_g^{(q+1)}\right\|2 = (\sum_{i=1}^{n_{q+1}} (L_i^{(q+1)})^2 + (\sum_{i=1}^{n'_{q+1}} (L_{i'}^{(q+1)})^2 + (\sum_{i=1}^{n''_{q+1}} (L_{i''}^{(q+1)})^2 \quad (66)$$

Where $L^{(q+1)}, L'^{(q+1)}, L''^{(q+1)}$ the local truncation error are terms respectively, $L_g^{(q+1)}$ is the global local truncation error.

5.1 A Three-Stage Fifth-Order RKTDIO Method

In this subsection, the derivation of the three-stage RKTDIO technique of order five by using the algebraic order conditions up to order five will be considered. The resulting system consists of 16 nonlinear equations with 16 unknown variables, solving the system simultaneously, and assuming $a_{11} = a_{22}$ and $a_{22} = a_{33}$

$a_{21} = a_{21}, a_{31} = a_{21}, a_{32} = \frac{3}{20}\, Root\ Of\ (5 - z^2 - 3), a_{33} = -\frac{5}{9}\, a_{21} - \frac{1}{24}\, Root\ Of(5 - z^2 - 3) + \frac{1}{24}$, $b_1 = \frac{2}{9}$, $b_2 = \frac{1}{36}\, \frac{200\, a_{21} Root\ Of(5-z^2-3)+15 Root\ Of(5-z^2-3)+200\, a_{21}+9}{40\, a_{21}+3\, Root\ Of(\,5-z^2-3)}, b_3 = -\frac{1}{36}\, \frac{200\, a_{21} Root\ Of(5-z^2-3)-15 Root\ Of(5-z^2-3)+200\, a_{21}+9}{40\, a_{21}+3\, Root\ Of(\,5-z^2-3)}, c_1 = \frac{1}{2}, c_2 = \frac{1}{2} - \frac{1}{2}\, Root\ Of(\,5 - z^2 - 3), c_3 = \frac{1}{2}\, Root\ Of(\,5 - z^2 - 3) + \frac{1}{2}, d_1 = \frac{1}{18}, d_2 = \frac{5}{72}\, Root\ Of(\,5 - z^2 - 3) + \frac{1}{18},\quad d_3 = -\frac{5}{72}\, Root\ Of(\,5 - z^2 - 3) + \frac{1}{18}, g_1 = \frac{4}{9},\ g_2 = \frac{5}{18}, g_3 = \frac{5}{18}$.

Next, we minimize the truncation error term by using minimize command in Maple. Thus, for the optimized value of coefficients in fractional we chose $a_{21} = -\frac{1}{125}$ with this value $\left\|\pi_g^{(5)}\right\|_2 =$

0.001366343866. Finally, all the parameters of three-stage fifth-order RKTDIO approach that will be denoted as RKTDIO5 can be written as follows (see **Table 2**):

Table 2. The RKTDIO5 Method

$\frac{1}{2}$	$\frac{83}{1800}-\frac{\sqrt{15}}{120}$		
$\frac{1}{2}-\frac{\sqrt{15}}{10}$	$-\frac{1}{125}$	$\frac{83}{1800}-\frac{\sqrt{15}}{120}$	
$\frac{1}{2}+\frac{\sqrt{15}}{10}$	$-\frac{1}{125}$	$\frac{3\sqrt{15}}{100}$	$\frac{83}{1800}-\frac{\sqrt{15}}{120}$
	$\frac{1}{18}$	$\frac{1}{18}+\frac{\sqrt{15}}{72}$	$\frac{1}{18}-\frac{\sqrt{15}}{72}$
	$\frac{2}{9}$	$\frac{5}{36}+\frac{\sqrt{15}}{36}$	$\frac{5}{36}-\frac{\sqrt{15}}{36}$
	$\frac{4}{9}$	$\frac{5}{18}$	$\frac{5}{18}$

5.2 A Four-Stage RDTDIO Method of Order Six

For the four-stage RKTDIO technique of order six, the algebraic conditions up to order six will be solved. The resulting system consists of 25 nonlinear equations with 23 unknown variables, and using simplifying assumption $b_i' = b_i''(1-c_i), i = 1, \dots, s,$ and supposing $b_2'' = 0$, and $b_2''' = 0$

$a_{21} = \frac{3}{20}\ Root\ Of\ (10 - z^2 - 10 - z + 1) - \frac{3}{80}, a_{31} = -\frac{1}{40},\ a_{32} = \frac{11}{80} - \frac{3}{20}\ Root\ Of(10 - z^2 - 10 - z + 1),\ a_{41} = \frac{1}{40}, a_{42} = -\frac{1}{40}, a_{43} = \frac{3}{20}\ Root\ Of(10 - z^2 - 10 - z + 1) - \frac{3}{80}, a_{44} = \frac{1}{48}, b_1 = \frac{2}{9}, b_2 = 0, b_3 = \frac{5}{18}\ Root\ Of(10 - z^2 - 10 - z + 1), b_4 = -\frac{5}{18}\ Root\ Of(10 - z^2 - 10 - z + 1) + \frac{5}{36}, c_1 = \frac{1}{2}, c_2 = Root\ Of(10 - z^2 - 10 - z + 1), c_3 = 1 - Root\ Of(10 - z^2 - 10 - z + 1), c_4 = Root\ Of(10 - z^2 - 10 - z + 1), d_1 = \frac{1}{18}, d_2 = d_2, d_3 = \frac{5}{36} Root\ Of(10 - z^2 - 10 - z + 1) - \frac{1}{72}, d_4 = -d_2 - \frac{5}{36} Root\ Of(10 - z^2 - 10 - z + 1) - \frac{1}{8}, g_1 = \frac{4}{9}, g_2 = 0, g_3 = \frac{5}{18}, g_4 = \frac{5}{18}$

Lastly, all the parameters of four-stage sixth-order RKTDIO method indicated by RKTDIO6 can be written as follows:

Table 3. The RKTDIO6 Method.

$\frac{1}{2}$	$\frac{1}{48}$			
$\frac{1}{2}-\frac{\sqrt{15}}{10}$	$\frac{3}{80}-\frac{3\sqrt{15}}{200}$	$\frac{1}{48}$		
$\frac{1}{2}+\frac{\sqrt{15}}{10}$	$-\frac{1}{40}$	$\frac{1}{16}+\frac{3\sqrt{15}}{200}$	$\frac{1}{48}$	
$\frac{1}{2}-\frac{\sqrt{15}}{10}$	$\frac{1}{40}$	$-\frac{1}{40}$	$\frac{3}{80}-\frac{3\sqrt{15}}{200}$	$\frac{1}{18}+\frac{\sqrt{15}}{72}$

	$\frac{1}{18}$	0	$\frac{1}{18}-\frac{\sqrt{15}}{72}$	$\frac{1}{18}-\frac{\sqrt{15}}{72}$
	$\frac{2}{9}$	0	$\frac{5}{36}-\frac{\sqrt{15}}{36}$	$\frac{5}{36}+\frac{\sqrt{15}}{36}$
	$\frac{4}{9}$	0	$\frac{5}{18}$	$\frac{5}{18}$

6. Derivation Embedded ERKTDIO6(5) Method

The RKTDIO method with $\mathfrak{s}$-stages for solving equation (1) is presented in its general form. Subsequently, the development of the embedded pair RK approach is discussed, which is an area of active research aimed at improving existing codes. To estimate the minimum error, implicit RKTDIO techniques are used, employing pairs of $\mathcal{P}(\mathcal{Q})$ orders in the values of step size codes. These methods are based on the $\mathcal{P}$-order method (C, A, b, b', b'') and the $\mathcal{Q}$-order method $(C, A, \ddot{b}, \ddot{b}', \ddot{b}'')$, and can be represented using the Butcher Tabular notation. The embedded pair can be initialized in the following manner:

C	A
	$b^{\,T}$ $b'^{\,T}$ $b''^{\,T}$
	$\ddot{b}^{\,T}$ $\ddot{b}'^{\,T}$ $\ddot{b}''^{\,T}$

The main objective of constructing the embedded pair of implicit RKTDIO techniques is to obtain a single error estimate that can be used in step-size approaches. This is achieved by improving the existing pairs and local error estimates and then restricting the step size $\hbar$.

$$\hbar_{n+1} = 0.9\hbar_n\left(\frac{Tol}{LTE}\right)^{\frac{1}{\mathcal{Q}+1}} \tag{67}$$

A safety factor of 0.9 is used to determine the local error estimate at each step, and Tol represents the maximum allowable local error that ensures the necessary accuracy. If the local truncation error (LTE) is less than or equal to Tol, the step is accepted and the higher order method (or local extrapolation) is used, where a more accurate approximation is applied to drive the integration and update $\hbar$ using Equation (67). On the other hand, if LTE is greater than Tol, the step is rejected and the step size $\hbar$ is reduced by half. The RKTDIO method is an embedded Runge-Kutta method developed for solving third-order ODEs. To ensure high accuracy for the higher-order method and the most accurate error estimates for the lower-order methods, fractions were used to develop orders 6 and 5, respectively. The step size $\hbar$ plays a crucial role in obtaining accurate results and can be doubled to achieve this goal. The embedded RKTDIO6(5) method is derived in Table 4 for this study.

In RKTDIO6(5), the A and C values is computed from the 6$^{\text{th}}$-order solution then derived the four-stage 5$^{\text{rd}}$-order embedded equation. Solving of the eqs. (51-53), (55-58), and (60-64) simultaneously then the solution for $\ddot{b}$ and $\ddot{b}'$ while $\ddot{b}''$ have the same values as the 6th -order. The solutions are obtained as

$\ddot{b}_1' = \frac{2}{9},\ \ddot{b}_2' = 0,\ \ddot{b}_3' = \frac{5}{36} - \frac{\sqrt{15}}{36},\ \ddot{b}_4' = \frac{5}{36} + \frac{\sqrt{15}}{36}, \ddot{b}_1 = \frac{1}{18}, \ddot{b}_2 = \frac{1}{18} + \frac{\sqrt{15}}{72}, \ddot{b}_3 = \frac{1}{18} - \frac{\sqrt{15}}{72},\ \ddot{b}_4 = \ddot{b}_4,\ \ddot{b}_1'' = \frac{4}{9}, \ddot{b}_2'' = 0,\ \ddot{b}_3'' = \frac{5}{18}, \ddot{b}_4'' = \frac{5}{18}$.

The following simplifying assumption is used in order to reduce the number of equations to be solved:

$$b_i' = b_i''(1 - c_i),\ i = 1, \dots, \mathfrak{s}. \tag{68}$$

Initially, we have 16 nonlinear equations with 12 unknown variables that we need to find a solution for. However, as the number of equations is greater than the number of unknowns, there is no solution. To tackle this issue, we make the simplifying assumption (68) which reduces the number of equations to 12 with 11 unknowns, enabling us to solve the system. We choose $\ddot{b} = \frac{1}{10}$ as the free parameter, and as a result, we can express the coefficients of the 4-stage embedded ERKTDIO6(5) technique.

Table 4: Table of ERKTDIO6(5) method.

$\frac{1}{2}$	$\frac{1}{48}$			
$\frac{1}{2} - \frac{\sqrt{15}}{10}$	$\frac{3}{80} - \frac{3\sqrt{15}}{200}$	$\frac{1}{48}$		
$\frac{1}{2} + \frac{\sqrt{15}}{10}$	$-\frac{1}{40}$	$\frac{1}{16} + \frac{3\sqrt{15}}{200}$	$\frac{1}{48}$	
$\frac{1}{2} - \frac{\sqrt{15}}{10}$	$\frac{1}{40}$	$-\frac{1}{40}$	$\frac{3}{80} - \frac{3\sqrt{15}}{200}$	$\frac{1}{48}$
	$\frac{1}{18}$	0	$\frac{1}{18} - \frac{\sqrt{15}}{72}$	$\frac{1}{18} + \frac{\sqrt{15}}{72}$
	$\frac{2}{9}$	0	$\frac{5}{36} - \frac{\sqrt{15}}{36}$	$\frac{5}{36} + \frac{\sqrt{15}}{36}$
	$\frac{4}{9}$	0	$\frac{5}{18}$	$\frac{5}{18}$
	$\frac{1}{18}$	0	$-\frac{2}{45} - \frac{\sqrt{15}}{72}$	$\frac{1}{18} - \frac{\sqrt{15}}{72}$
	$\frac{2}{9}$	0	$\frac{5}{36} - \frac{\sqrt{15}}{36}$	$\frac{5}{36} + \frac{\sqrt{15}}{36}$
	$\frac{4}{9}$	0	$\frac{5}{18}$	$\frac{5}{18}$

7. Numerical Experiments

7.1 Constant Methods

In order to evaluate the performance of the new RKTDIO methods to the established RK methods in the scientific literature, a series of test problems are addressed in this part. The following methods have been selected for comparison:

- **RKTDIO5:** the implicit RKTDIO method of order five with three-stage derived in this paper.
- **RKTDIO6:** the implicit RKTDIO method of order six with four-stage derived in this paper.

- **DITRKM5:** three-stage fifth-order implicit RK derived by [25].
- **Radau I:** three-stage fifth-order implicit RK method presented in [28].
- **Radau IA:** three-stage fifth-order implicit RK method presented in [24].
- **Lobatto III:** The sixth-order four-stage implicit Runge–Kutta method as given by [30].
- **Lobatto IIIB:** The sixth-order four-stage implicit Runge–Kutta method as given by [29].

Problem (1): Consider a nonhomogeneous linear ODE given in [22]

$$\alpha'''(x) = \alpha(x) + \cos(x),$$

With

$$\alpha(0) = 0\,, \alpha'(0) = 0, \alpha''(0) = 1$$

Where $x \in [0,1]$,

and analytic solution $\alpha(x) = \frac{(e^x - \cos(x) - \sin(x))}{2}$.

Problem (2): Consider the nonhomogeneous nonlinear ODE

$$\alpha'''(x) = (\alpha(x))^2 + \cos^2(x) - \cos(x) - 1\,,$$

With

$\alpha(0) = 0,\ \alpha'(0) = 1, \alpha''(0) = 1\ \ where\ 0 \le x \le 2$,

and analytic solution $\alpha(x) = \sin(x)$.

Problem (3): The nonhomogeneous nonlinear ODEs is considered as

$$\alpha'''(x) = 8(\frac{\alpha^2(x)}{e^{2x}})$$

With

$\alpha(0) = 1,\ \alpha'(0) = 2\,, \alpha''(0) = 4\ \ \ where\ 0 \le x \le 1$,

and analytic solution $\alpha(x) = e^{2x}$.

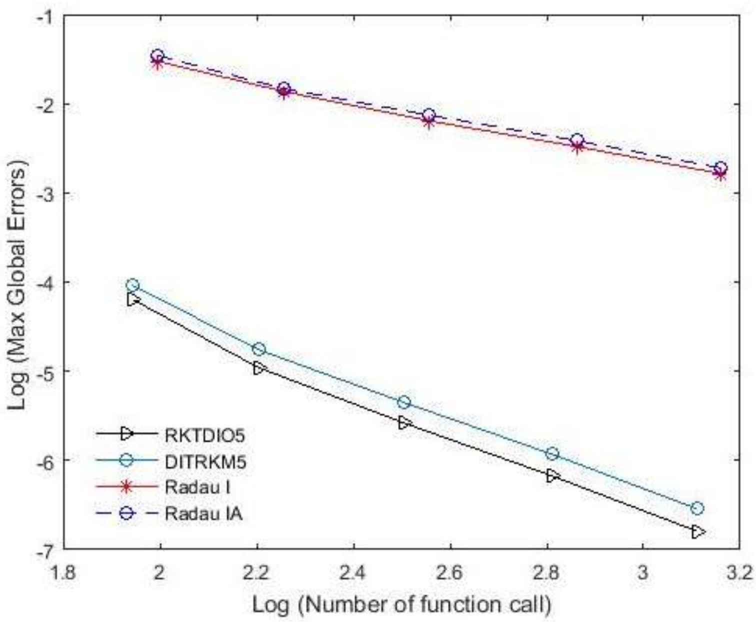


Figure 2. Accuracy curve for RKTDIO5, DITRKM5, Radau I, Radau IA with $h = 0.1, 0.05, 0.025, 0.00125, 0.00625$ for Problem1.

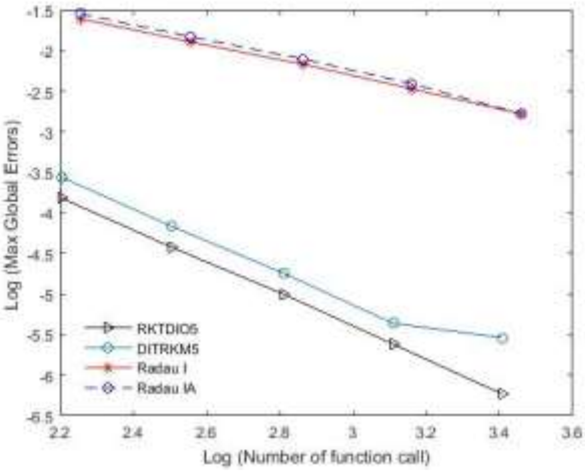


Figure3. Accuracy curve for RKTDIO5, DITRKM5, Radau I, Radau IA with $h = 0.1, 0.05, 0.025, 0.00125, 0.00625$ for Problem2.

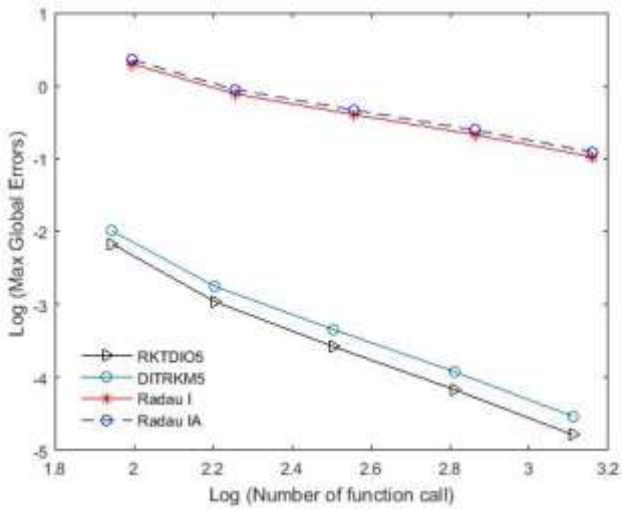


Figure4. Accuracy curve for RKTDIO5, DITRKM5, Radau I, Radau IA with $h = 0.1, 0.05, 0.025, 0.00125, 0.00625$ for Problem3.

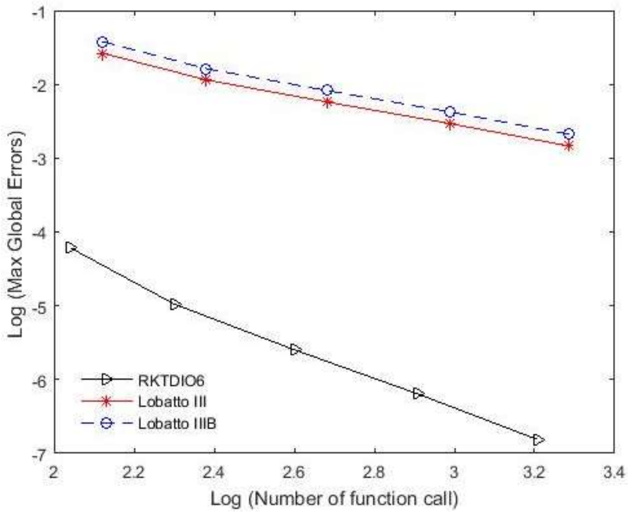


Figure 5. Accuracy curve for RKTDIO6, Lobattoo III, Lobattoo IIIB with $h = 0.1, 0.05, 0.025, 0.00125, 0.00625$ for Problem1.

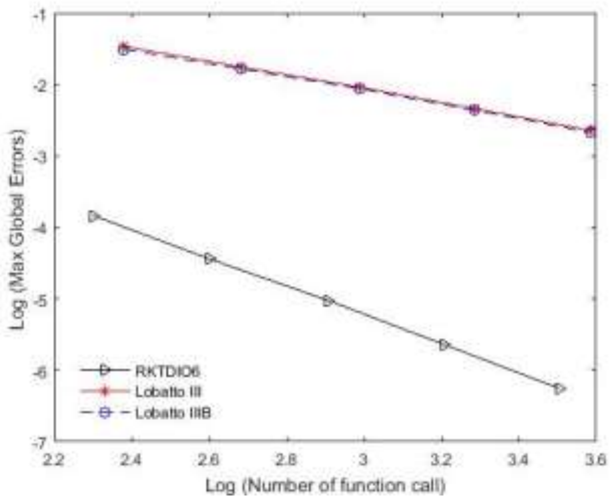


Figure 6. Accuracy curve for RKTDIO6, Lobattoo III, Lobattoo IIIB with $h = 0.1, 0.05, 0.025, 0.00125, 0.00625$ for Problem2.

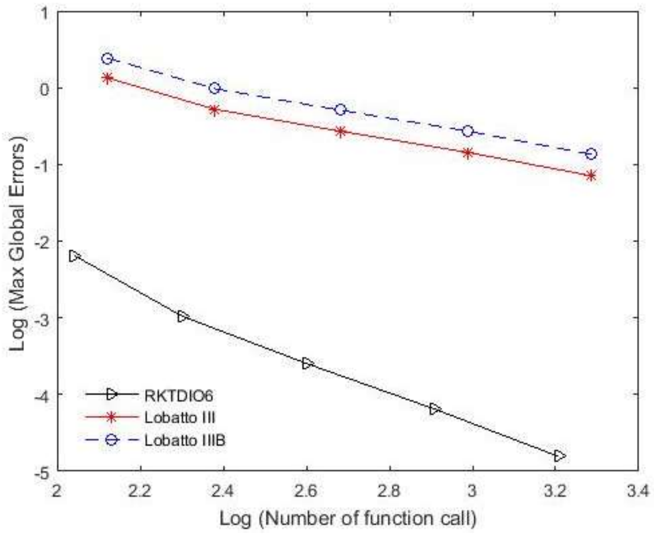


Figure 7. Accuracy curve for RKTDIO6, Lobattoo III, Lobattoo IIIB with $h = 0.1, 0.05, 0.025, 0.00125, 0.00625$ for Problem3.

7.2 Variable Method

This subsection will apply the new embedded method to the same third-order differential equation problems in the previous subsection. The following implicit diagonally RK method is selected for the numerical comparisons. The approximation results are illustrated in the tables below for solving problems. The following abbreviations will be used in the tables:

- $\boldsymbol{Tol}$: Tolerance.
- **Method**: method employed step sizes between two points or positions.
- **F. N**: number of the function call.
- **STEP**: The number of successful steps.
- **FSTEP**: The number of failed steps.
- **Time**: execution time.
- **ERKTDIO6(5)**: The novel embedded 6(5) derived in this study.
- **EDITRK5(4)**: The embedded diagonally implicit 5(4) RK derived in [7].

Table 4. Comparisons of number of function call and Time of ERKTDIO6(5) and EDITRK 5(4) with $h = 10^{-6}, 10^{-8}, 10^{-10}$ for the problem 1.

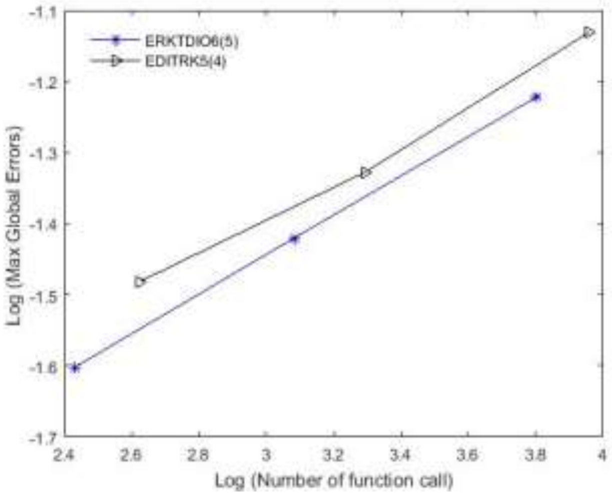
$TOL(\mathscr{k})$	Method	No. of Function Call	Time	Step	FSTEP
10^{-6}	ERKTDIO6(5)	270	0.025	5	2
	EDITRK 5(4)	421	0.033	13	0
10^{-8}	ERKTDIO6(5)	1210	0.038	16	2
	EDITRK 5(4)	1962	0.047	40	0
10^{-10}	ERKTDIO6(5)	6353	0.060	72	2
	EDITRK 5(4)	9105	0.074	128	1

Table 5. Comparisons of number of function call and Time of EDITRKM 4(3) and EDITRKM 5(4) with $h = 10^{-2}, 10^{-4}, 10^{-6}$ for the problem 2.

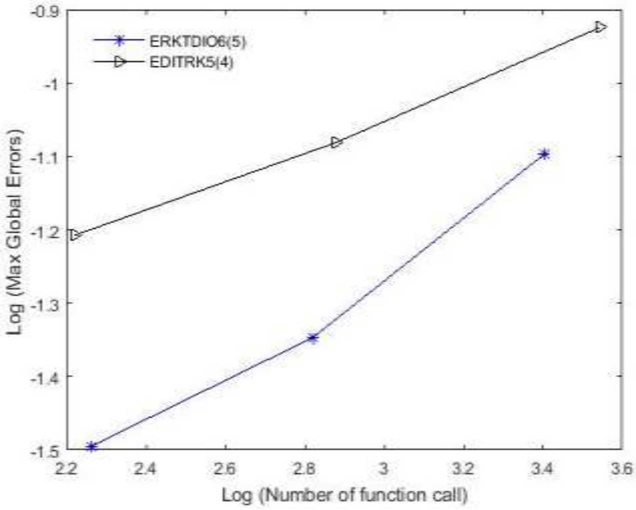
$TOL(\mathscr{k})$	Method	No. of Function Call	Time	Step	FSTEP
10^{-6}	ERKTDIO6(5)	183	0.032	3	0
	EDITRK 5(4)	166	0.062	7	0
10^{-8}	ERKTDIO6(5)	662	0.045	8	0
	EDITRK 5(4)	759	0.083	22	0
10^{-10}	ERKTDIO6(5)	2533	0.080	29	0
	EDITRK 5(4)	3495	0.119	69	1

Table 6. Comparisons of number of function call and Time of EDITRKM 4(3) and EDITRKM 5(4) with $h = 10^{-2}, 10^{-4}, 10^{-6}$ for the problem 3.

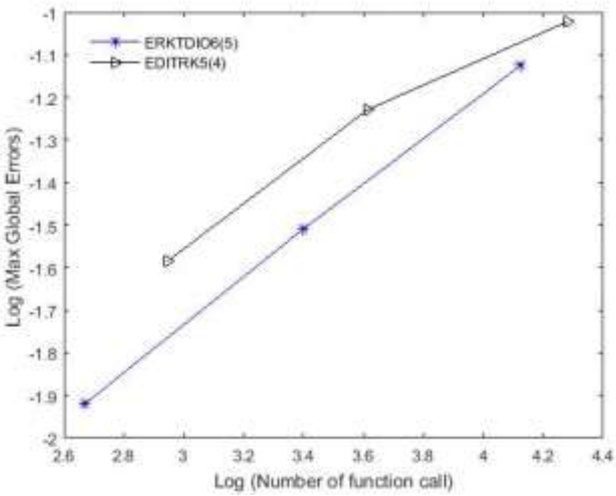
$TOL(\mathscr{k})$	Method	No. of Function Call	Time	Step	FSTEP
10^{-6}	ERKTDIO6(5)	466	0.012	97	1
	EDITRK 5(4)	882	0.035	29	0
10^{-8}	ERKTDIO6(5)	2511	0.031	315	1
	EDITRK 5(4)	4101	0.059	152	1
10^{-10}	ERKTDIO6(5)	13325	0.075	1260	1
	EDITRK 5(4)	19044	0.095	725	2



Problem 1



Problem 2



Problem 3

Figure 8: Accuracy curve for ERKTDIO6(5) and EDITRK5(4) with $h = 10^{-6}, 10^{-8}, 10^{-10}$.

8. Discussions

The objective of our research was to directly solve third-order ordinary differential equations (ODEs) using the RKTDIO approach, which is based on the algebraic theory of order conditions. Previous research has primarily concentrated on defining algebraic order conditions and B-series theory to solve first- and second-order ordinary differential equations (ODEs). However, we were inspired to expand upon these ideas and create the RKTDIO formula specifically designed for third-order ODEs. As a result, the RKTDIO5 and RKTDIO6 methods were developed, offering improved computing efficiency and accuracy for solving certain third-order ODEs as compared to other current approaches.

In addition, we shared the results of our study on the embedded diagonal implicit type Runge-Kutta technique (ERKTDIO). We assessed the performance of our method (ERKTDIO6(5)) in comparison to other approaches by examining the decimal logarithm of the highest time curve and the logarithm of the function call estimates obtained from Tables 4-6. We utilised three separate test problems and calculated the logarithm of the time curve utilising various tolerance values. Figure 10 was generated using the numerical data obtained from Tables 4-6. It presents the number of successful and failed steps that occurred during the calculations. Our work has enhanced and refined the method by transitioning from an explicit to an implicit approach and from a direct to a diagonal scheme.

9. Conclusions

Ultimately, our research has made a significant contribution to the progress of numerical methods for solving third-order ordinary differential equations. The introduction of the RKTDIO5 and RKTDIO6 techniques provides improved computational efficiency and precision for particular third-order ordinary differential equations (ODEs). Furthermore, our examination of the ERKTDIO technique showcases its efficacy in comparison to alternative methods, especially in terms of the number of successful steps in computations. Our methods provide the potential for enhanced solutions of third-order ODEs by adopting implicit and diagonal schemes. In summary, these discoveries expand the current understanding of numerical analysis and computational mathematics, offering useful insights for future investigations in this field.

Acknowledgment

The authors have greatly appreciated the referees for their valuable comments and suggestions for improving the paper**.**

Conflict of Interest

The authors declare that they have no conflicts of interest.

Funding

There is no financial support in preparation for the publication**.**

References

1. Boutayeb, A.; Chetouani A. A mini-review of numerical methods for high-order problems. *International Journal of Computer Mathematics*. **2007**, *84(4*), 563-579, https://doi.org/10.1080/00207160701242250.

2. Wu, Xiong-Jian; Yigong Wang; W. G. Price. Multiple resonances, responses, and parametric instabilities in offshore structures. *Journal of ship research*. **1988**, *32(04)*, 285-296, https://doi.org/10.5957/jsr.1988.32.4.285
3. Cortell, R. Application of the fourth-order Runge-Kutta method for the solution of high-order general initial value problems. *Computers & structures*. **1993**, *49(5)*, 897-900, https://doi.org/10.1016/0045-7949(93)90036-D
4. Malek, A.; R. Shekari Beidokhti. Numerical solution for high order differential equations using a hybrid neural network-optimization method. *Applied Mathematics and Computation*. **2006**, *183(1*), 260-271, https://doi.org/10.1016/j.amc.2006.05.068.
5. Awoyemi, D. O. Algorithmic collocation approach for direct solution of fourth-order initial-value problems of ordinary differential equations. *International Journal of Computer Mathematics*. **2005**, *82(3)*, 321-329, https://doi.org/10.1080/00207160412331296634.
6. Adel, F.; Senu, N.; Ismail, F.; Majid, Z. A. New efficient phase-fitted and amplification-fitted Runge-Kutta method for oscillatory problems. *International J Pure Appl Math***,** **2016**, *107*(1), 69-86, https://doi.org/10.12732/ijpam.v107i1.7
7. Hussain, K.; Ismail, F., ; Senu, N. A new optimized Runge-Kutta-Nyström method to solve oscillation problems. *World Applied Sciences Journal*, **2015**, *33(10)*, 1614-1622, https://doi.org/10.5829/idosi.wasj.2015.33.10.300.
8. Hussain, K. A.; Abdulnaby, Z. E.. A new two derivative FSAL Runge-Kutta method of order five in four stages. *Order*, **2020,** *17(1)*. https://doi.org/10.21123/bsj.2020.17.1.0166
9. Lee, K. C.; Senu, N., Ahmadian, A., Ibrahim, S. N. I., & Baleanu, D. Numerical study of third-order ordinary differential equations using a new class of two derivative Runge-Kutta type methods. *Alexandria Engineering Journal*, **2020**, *59(4)*, 2449-2467, https://doi.org/10.1016/j.aej.2020.03.008.
10. Ahmad, N. A.; Senu, N.; Ismail, F. Phase-fitted and amplification-fitted higher order two-derivative Runge-Kutta method for the numerical solution of orbital and related periodical IVPs. *Mathematical Problems in Engineering*, ***2017*****.** 10.1109/ICREM.2015.7357020.
11. Senu, N.; Ahmad, N. A., ; zarina, B. I. B. I. Numerical Study on Phase-Fitted and Amplification-Fitted Diagonally Implicit Two Derivative Runge-Kutta Method for Periodic IVPs. *Sains Malaysiana*, **2021,** *50*(6), 1799-1814, 10.17576/jsm-2021-5006-25.
12. Fawzi, F. A. An Embedded 5 (4) Pair of Optimized Runge-Kutta Method for the Numerical Solution of Periodic Initial Value Problems. *Iraqi Journal of Science*, **2022,** 3889-3901, https://doi.org/10.24996/ijs.2022.63.9.21.
13. SENU, N.; AHMAD, N.; ISMAIL, F. Embedded 5 (4) Pair Trigonometrically-Fitted Two Derivative Runge-Kutta Method with FSAL Property for Numerical Solution of Oscillatory Problems, WSEAS TRANSACTIONS on COMPUTERS, **2017**,*16*, 155-162.
14. Jikantoro, Y. D.; Ismail, F., Senu, N., ; Ibrahim, Z. B., Hybrid methods for direct integration of special third order ordinary differential equations. *Applied Mathematics and Computation*, **2018**, *320*, 452-463, https://doi.org/10.1016/j.amc.2017.10.003.
15. Shokri, A.; Mehdizadeh Khalsaraei, M. A new efficient high order four-step multiderivative method for the numerical solution of second-order IVPs with oscillating solutions. *Mathematics Interdisciplinary Research*, **2020**, *5(2*), 157-172, 10.22052/mir.2020.211603.1185.
16. Mechee, M.; Senu, N.; Ismail, F.; Nikouravan, B.; Siri, Z. A three-stage fifth-order Runge-Kutta method for directly solving special third-order differential equation with application to thin film flow problem. *Mathematical Problems in Engineering*. **2013**, 2013, https://doi.org/10.1155/2013/795397.
17. Fawzi, F. A.; Jumaa M, H. The Implementations Special Third-Order Ordinary Differential Equations (ODE) for 5th-order 3rd-stage Diagonally Implicit Type Runge-Kutta Method (DITRKM). *Ibn AL-*

Haitham Journal For Pure and Applied Sciences. **2022**, *35(1)*, 92-101, https://doi.org/10.30526/35.1.2803.

18. ISMAIL F.; Al-Khassawneh, R. A. Solving Delay Differential Equations Using embedded singly diagonally Implicit runge-kutta methods. *Acta Mathematica Vietnamica*. **2008**, *33(2)*, 95-105.
19. Jikantoro, Y. D., Ismail, F., Senu, N., & Ibrahim, Z. B,. A new integrator for special third order differential equations with application to thin film flow problem. *Indian Journal of Pure and Applied Mathematics*, **2018**, *49*, 151-167, 10.1007/s13226-018-0259-6.
20. Lee, K. C.; Senu, ; Ahmadian, A. ; Ibrahim, S. N. I., On two-derivative Runge–Kutta type methods for solving u‴= f (x, u (x)) with application to thin film flow problem. *Symmetry*, **2020**, *12(6)*, 924, https://doi.org/10.3390/sym12060924.
21. Hussain, K.; Ismail, F.; Senu, N. Solving directly special fourth-order ordinary differential equations using Runge–Kutta type method. *Journal of Computational and Applied Mathematics*. **2016**, *306*, 179-199, https://doi.org/10.1016/j.cam.2016.04.002.
22. Ghawadri, N.; Senu, N.; Adel Fawzi, F.; Ismail, F.; Ibrahim, Z. B. Explicit Integrator of Runge-Kutta Type for Direct Solution of $u^{(4)} = f\,(x, u, u', u'')$. *Symmetry*. **2019**, *11(2)*, 246, https://doi.org/10.3390/sym11020246.
23. Dormand, J. R. *Numerical methods for differential equations: a computational approach.* CRC press, **1996**, https://doi.org/10.1201/9781351075107 .
24. Fawzi, F. A.; Jaleel, N. W. Exponentially Fitted Diagonally Implicit EDITRK Method for Solving ODEs. *Ibn AL-Haitham Journal For Pure and Applied Sciences*. **2023**, *36(1)*, 389-399, https://doi.org/10.30526/36.1.2883 .
25. Saleh, O. M.; Fawzi, F. A.; Hussain, K. A.; Ghawadri, N. G. Derivation of Embedded Diagonally Implicit Methods for Directly Solving Fourth-order ODEs. *Ibn AL-Haitham Journal For Pure and Applied Sciences*, **2024,** *37(1)*, 375-385, https://doi.org/10.30526/37.1.3288.
26. Senu, N.; Suleiman, M.; Ismail, F. An embedded explicit Runge–Kutta–Nyström method for solving oscillatory problems. *Physica Scripta*. **2009**, *80(1)*, 015005. https://doi.org/10.1088/0031-8949/80/01/015005
27. Hairer E.; Hochbruck, M.; Iserles, A.; Lubich, C. Geometric numerical integration; *Oberwolfach Reports*. **2006**, *3(1)*, 805-882. https://doi.org/10.1017/S0962492902000144.
28. Hairer E, Norsett SP, Wanner G. Solving Ordinary Differential Equations I: Nonstiff Problems, II: Stiff Problems. *Springer*, **1987**. https://doi.org/10.1007/978-3-540-78862-1.
29. Lambert, J. D. Numerical methods for ordinary differential systems. *New York: Wiley*, **1991.** https://doi.org/10.1002/9781119121534