IHJPAS. 37 (1) 2024
Ibn Al-Haitham Journal for Pure and Applied Sciences
Journal homepage: jih.uobaghdad.edu.iq
PISSN: 1609-4042, EISSN: 2521-3407

Derivation of Embedded Diagonally Implicit Methods for Directly Solving Fourth-order ODEs

[1]Omar Mohammed Saleh **[2]Firas Adel Fawzi** **[3]Kasim Abbas Hussain***
[4]Nizam G. Ghawadri

[1,2]Department of Mathematics, Faculty of Computer Science and Mathematics, University of Tikrit, Tikrit, Iraq.
[3]Department of Mathematics, College of Science, Mustansiriyah University, Baghdad, Iraq.
[4]Doctor of Mathematics, Ministry of Education, Jenin, Palestine.

*Corresponding Author. kasimabbas@uomustansiriyah.edu.iq

Received 17 February 2023, Received 15 April 2023, Accepted 10 May 2023, Published 20 January 2024

doi.org/ 10.30526/37.1.3288

Abstract

EDIRKTO is an Implicit Type Runge-Kutta Method of Diagonally Embedded pairs, is a novel approach presented in the paper that may be used to solve 4th-order ordinary differential equations of the form $q^{(4)}(t) = f(t, q(t), q'(t))$. There are two pairs of EDIRKTO, with three stages each: EDIRKTO4(3) and EDIRKTO5(4). The derivation techniques of the method indicate that the higher-order pair is more accurate, while the lower-order pair provides superior error estimates. Next, using these pairs as a basis, we developed variable step codes and applied them to a series of 4^{th}-order ODE problems. The numerical outcomes demonstrated how much more effective their approach is in reducing the quantity of function evaluations needed to resolve fourth-order ODE issues.

Keywords: Fourth-order ODEs; Runge-Kutta methods; diagonally implicit technique; embedded method.

1. Introduction

Differential equations (DEs) are important tools in mathematical modeling, particularly in explaining physical phenomena like heat and fluid movement, object motion, and nuclear reactions. ODEs of different orders are used in various applied research fields such as mechanics, electrical and control engineering, fluid dynamics, ship dynamics, neural networks, and beam theory [1-3]. Numerical and approximated methods have been developed to solve specific types of DEs.

In this study, we discuss a class of quasi-linear, fourth-order (ODEs) and their numerical integration in the following form:

$$q^{(4)}(t) = f(t, q(t), q'(t)), \quad t \geq t_0, \tag{1}$$

with initial conditions

$$\sigma^i(\tau) = \zeta^i, i = 0, 1, \dots, 3.$$

Where $f : \mathcal{R} \times \mathcal{R}^N \longrightarrow \mathcal{R}^N$, $\sigma(\tau) = [\sigma_1(\tau), \sigma_2(\tau), \dots, \sigma_N(\tau)]$, $t(\tau, \sigma) = [t_1(\tau, \sigma), t_2(\tau, \sigma), \dots, t_N(\tau, \sigma)]$, and $\zeta^i = [\zeta_1^{\mathrm{i}}, \zeta_2^{\mathrm{i}}, \dots, \zeta_{\mathrm{N}}^{\mathrm{i}}]$, where $\mathrm{i} = 0,1,2, \dots ,4$.

Previously, researchers transformed Equation (1) into a system of first-order ODEs with four additional dimensions to solve it. However, it would be more efficient if numerical methods could be used to solve the problem accurately and quickly. In [4-12] contain such works. Multistep strategies for solving ODEs require initial values. However, according to several researchers (see [1, 13, 14]) this technique has a drawback as it consumes a lot of computing time and human effort. Consequently, several researchers have turned their attention to direct integration methods for solving higher-order ODEs, as these methods have demonstrated features of accuracy and speed [16-30]. Nevertheless, all of the methods mentioned above are above are multistep.

The primary objective of this paper is to introduce a new technique named EDIRKTO, which is a one-step implicit Runge-Kutta method created to directly solve general 4^{th}-order differential equations. The method involves deriving diagonally embedded implicit Runge-Kutta methods for the direct integration of specific 4^{th}-order differential equations. To address the IVPs problem in (1), the special version of the EDIRKTO method with m stages is as follows:

$$q_{n+1} = q_n + h\, q_n' + \frac{h^2}{2}\, q_n'' + \frac{h^3}{6}\, q_n''' + h^4 \sum_{i=1}^{s} b_i f(t_n + c_j h\,, Q_i, Q_i'), \tag{2}$$

$$q_{n+1}' = q_n' + h\, q_n'' + \frac{h^2}{2}\, q_n''' + h^3 \sum_{i=1}^{s} b_i'\, f\big(t_n + c_j h, Q_i, Q_i'\big), \tag{3}$$

$$q_{n+1}'' = q_n'' + h\, q_n''' + h^2 \sum_{i=1}^{s} b''_i\, f\big(t_n + c_j h, Q_i, Q_i'\big), \tag{4}$$

$$q_{n+1}''' = q_n''' + h \sum_{i=1}^{s} b'''_i\, f\big(t_n + c_j h, Q_i, Q_i'\big), \tag{5}$$

$$Q_j = q_n + h\, c_i q_n' + \frac{h^2}{2}\, c_i^2 q_n'' + \frac{h^3}{6}\, c_i^3 q_n''' + h^4 \sum_{j=1}^{s} a_{ij} f\big(t_n + c_j h, Q_j, Q_j'\big), \tag{6}$$

$$Q_j' = q_n' + h\, c_i q_n'' + \frac{h^2}{2}\, c_i^2 q_n''' + h^3 \sum_{j=1}^{s} \bar{a}_{ij} f\big(t_n + c_j h, Q_j, Q_j'\big). \tag{7}$$

The coefficients $b_i, b_i', b_i'', b_i''', a_{i,j}, \bar{a}_{ij}$ and c_i of diagonal implicit RK type (EDIRKTO) methods are real numbers. The method is diagonally implicit when $a_{i,j} \neq 0$ for $i \leq j$. Using the Butcher tableau, the EDIRKT approach is illustrated. See **Table 1**.

Table 1. The Butcher tableau EDIRKTO method

c_1	a_{11}	a_{12}	...	a_{1s}	$\bar{a}_{11}$	$\bar{a}_{12}$	...	$\bar{a}_{1s}$
c_2	a_{21}	a_{22}	...	a_{2s}	$\bar{a}_{21}$	$\bar{a}_{22}$	...	$\bar{a}_{2s}$
c_3	a_{31}	a_{32}	...	a_{3s}	$\bar{a}_{31}$	$\bar{a}_{32}$	...	$\bar{a}_{3s}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
c_s	a_{s1}	a_{s2}	...	a_{ss}	$\bar{a}_{s1}$	$\bar{a}_{s2}$	...	$\bar{a}_{ss}$
	b_1	b_2	...	b_s				
	b'_1	b'_2	...	b'_s				
	b''_1	b''_2	...	b''_s				
	b'''_1	b'''_2	...	b'''_s				

3. Order Conditions of the EDIRKTO Method

According to Ghawadri [15], the algebraic order conditions for the EDIRKTO formula up to order seven are as follows:

order 1: $\sum b_i''' = 1,$

order 2: $\sum b_i''' c_i = \frac{1}{2}, \quad \sum b_i'' = \frac{1}{2},$

order 3: $\sum b_i''' c_i^2 = \frac{1}{3}, \quad \sum b_i'' c_i = \frac{1}{6}, \quad \sum b_i' = \frac{1}{6},$

order 4: $\sum b_i''' c_i^3 = \frac{1}{4}, \quad \sum b_i''' \overline{a_{ij}} = \frac{1}{24}, \ \sum b_i'' c_i^2 = \frac{1}{12}, \quad \sum b_i' c_i = \frac{1}{24}, \quad \sum b_i = \frac{1}{24},$

order 5: $\sum b_i''' c_i^4 = \frac{1}{5}, \ \sum b_i''' a_{ij} = \frac{1}{120}, \ \sum b_i''' \bar{a}_{ij} c_j = \frac{1}{120}, \ \sum b_i''' c_i \bar{a}_{ij} = \frac{1}{30},$

$\sum b_i'' c_i^3 = \frac{1}{20}, \quad \sum b_i'' a_{ij} = \frac{1}{120}, \ \sum b_i' c_i^2 = \frac{1}{60}, \quad \sum b_i c_i = \frac{1}{120},$

order 6: $\sum b_i''' c_i^5 = \frac{1}{6}, \ \sum b_i''' a_{ij} c_j = \frac{1}{720}, \ \sum b_i''' \bar{a}_{ij} c_i^2 = \frac{1}{360}, \sum b_i''' c_i^2 \bar{a}_{ij} = \frac{1}{36},$

$\sum b_i''' c_i \bar{a}_{ij} c_j = \frac{1}{144}, \ \sum b_i''' c_i a_{ij} = \frac{1}{144}, \sum b_i'' c_i^4 = \frac{1}{30}, \sum b_i'' \bar{a}_{ij} c_j = \frac{1}{720},$

$\sum b_i'' c_i \bar{a}_{ij} = \frac{1}{180}, \ \sum b_i'' \bar{a}_{ij} = \frac{1}{720}, \ \sum b_i' c_i^3 = \frac{1}{120}, \quad \sum b_i c_i^2 = \frac{1}{360},$

$\sum b_i' \bar{a}_{ij} = \frac{1}{720}.$

4. Derivation Embedded EDIRKTO Methods

To solve Equation (1) numerically, the general form of the EDIRKTO technique with m-stage is given. The embedded pair RK method, a current research area that is constantly enhancing existing programs, is then developed. The derivation of $p(q)$ pairs of implicit EDIRKTO approaches is used to provide minimum error estimation for step size codes. Both the order p method $(C, A, b, b'^T, b''^T, b'''^T)$ and the order q method $(C, \bar{A}, b_1^T, b'^T_2, b''^T_3, b'''^T_4)$ constitute the foundation for them. The embedded pair can be launched in Butcher Tabular as follows:

Table 2. The Embedded Pair EDIRKTO Method

C	A	$\bar{A}$
	b^T b'^T b''^T b'''^T	
	b_1^T $b_2'^T$ $b_3''^T$ $b_4'''^T$	

The main idea is to obtain single cost error estimation for usage in step size approach values before generating the embedded pair of implicit EDIRKTO approaches. The methods are illustrated by increasing the significant pairs and local error estimates using the values step size h, as follows [8],

$$h_{n+1} = 0.9h_n \left(\frac{Tol}{LTE}\right)^{\frac{1}{q+1}}, \quad (8)$$

Where Tol refers to the required level of accuracy, and local truncation error (LTE) is performed at each stage. Therefore, the step will be admitted if $LTE \leq Tol$, uses the local extrapolation method, which indicates employing more precise calculations to boost the integration and h can be improved by using Equation (8). If $LTE > Tol$, the step will be denied and the step size h will be cut in half. Fourth-order ODEs can be solved using the EDIRKTO technique, an embedded RK-type method. The first pair has orders 3 and 4, whereas the second pair contains orders 4 and 5. These methods were built utilizing sections, which confirm that the lower-order methods produced the most accurate error estimates while the higher-order methods were extremely accurate. Therefore, doubling the step size h has an impact on getting the correct results. The two derivations for the embedded EDIRKTO4(3) and embedded EDIRKTO5(4) methods used in this work are shown in **Tables 3** and **4**.

The A and C values in EDIRKTO4(3) are calculated from the 4^{th}-order solution and then obtained from the three-stage embedded 3^{rd}-order equation. The simultaneous solution of equations, followed by the solution of b_1^T, b'^T_2, and b''^T_3 while b'''^T_4 has the same values as the 4^{th}-order. The outcomes are as follows:

$b_1^T = \frac{1}{6} - b'^T_2 - b'^T_3, b'^T_2 = b'^T_2, b'^T_3 = b'^T_3, b'''^T_1 = 0, b'''^T_2 = \frac{1}{2}, b'''^T_3 = \frac{1}{2},\ b''^T_1 = -\frac{1}{2} + b''^T_3 + b''^T_3\sqrt{3} + \frac{1}{3}\sqrt{3}, b''^T_2 = -2b''^T_3 + b''^T_3\sqrt{3} - \frac{\sqrt{3}}{3} + 1, b''^T_3 = b''^T_3.$

According to [12], using the minimized command in Maple to minimize the LTE, we can calculate the free parameters as follows:

b'^T_2= -0.0142606637044632, b'^T_3=0.644727870311524 and b''^T_3= 0.105662412164532. If the value is optimized in fractional form, then $b'^T_2 = -\frac{1}{100}$, $b'^T_3 = \frac{6}{10}$, and $b''^T_3 = \frac{1}{10}$ are chosen.

Table 3. The Embedded Pair EDIRKTO4 (3) Method.

1	$-\frac{1}{4}$			$-\frac{2}{1000}$		
$\frac{1}{2}-\frac{\sqrt{3}}{6}$	$\frac{3}{10}$	$-\frac{1}{4}$	0	0	$-\frac{2}{1000}$	
$\frac{1}{2}+\frac{\sqrt{3}}{6}$	$\frac{3}{10}$	$\frac{3}{10}$	$-\frac{1}{4}$	0	$\frac{9}{100}$	$-\frac{2}{1000}$
	$-\frac{2}{10}$	$\frac{7}{10}$	$\frac{6}{100}$			
	$\frac{2}{1000}$	$\frac{2}{10}$	$\frac{2}{100}$			
	0	$\frac{1}{4}+\frac{\sqrt{3}}{12}$	$\frac{1}{4}-\frac{\sqrt{3}}{12}$			
	0	$\frac{1}{2}$	$\frac{1}{2}$			
	$-\frac{2}{10}$	$\frac{7}{10}$	$\frac{6}{100}$			
	$-\frac{127}{300}$	$-\frac{1}{100}$	$\frac{3}{5}$			
	$-\frac{2}{5}+\frac{7\sqrt{3}}{30}$	$\frac{2}{5}+\frac{7\sqrt{3}}{30}$	$\frac{1}{10}$			
	0	$\frac{1}{2}$	$\frac{1}{2}$			

Moreover, from the 5^{th}-order method, the coefficients of A and C are computed, and a three-stage 4^{th}-order embedded approach is then derived. The solution for b_1^T, b'^T_2 and b''^T_3 is obtained by simultaneously solving the equations up to order 5, while b'''^T_3 has the same result as the fifth order. The answers are attained as

$b'^T_1=-\frac{1}{12}+\frac{2}{5}b'^T_3-\frac{2\sqrt{6}}{5}b'^T_3+\frac{\sqrt{6}}{24}, b'^T_2=-\frac{7}{5}b'^T_3+\frac{2\sqrt{6}}{5}b'^T_3-\frac{\sqrt{6}}{24}+\frac{1}{4}, b'^T_3=b'^T_3, b_1^T=\frac{1}{24}-b_2^T-b_3^T, b_2^T=b_2^T, b_3^T=b_3^T, b'''^T_1=\frac{1}{9}, b'''^T_2=\frac{4}{9}-\frac{\sqrt{6}}{36}, b'''^T_3=\frac{4}{9}+\frac{\sqrt{6}}{36}, b''^T_1=0, b''^T_2=\frac{1}{4}+\frac{\sqrt{6}}{36}, b''^T_3=\frac{1}{4}-\frac{\sqrt{6}}{36}$.

We determine the free parameters according to [12] by minimizing the LTE using Maple's minimized command as follows:

b'^T_3= 0.0323023077253058, b_2^T= -0.148646585401452 and b_3^T= 0.447631755184810. If the value is optimized in fractional form, then $b'^T_3=\frac{3}{100}$, $b_2^T=-\frac{1}{10}$, and $b_3^T=\frac{4}{10}$ are chosen.

Table 4. The Embedded Pair EDIRKTO5(4) Method.

1	$-\frac{3}{10}$			$-\frac{2}{100}$		
$\frac{2}{5}-\frac{\sqrt{6}}{10}$	$\frac{4}{10}$	$-\frac{3}{10}$		$\frac{279}{25000}-\frac{381\sqrt{6}}{25000}$	$-\frac{2}{100}$	
$\frac{2}{5}+\frac{\sqrt{6}}{10}$	$-\frac{1}{10}$	$\frac{5}{10}$	$-\frac{3}{10}$	$-\frac{63}{1000}+\frac{39\sqrt{6}}{1000}$	$\frac{9}{100}$	$-\frac{2}{100}$
	$\frac{9}{10000}$	$\frac{1223}{6000}+\frac{1331\sqrt{6}}{180000}$	$\frac{1223}{6000}-\frac{1331\sqrt{6}}{180000}$			
	0	$\frac{1}{12}+\frac{\sqrt{6}}{48}$	$\frac{1}{12}-\frac{\sqrt{6}}{48}$			
	0	$\frac{1}{4}+\frac{\sqrt{6}}{36}$	$\frac{1}{4}-\frac{\sqrt{6}}{36}$			
	$\frac{1}{9}$	$\frac{4}{9}-\frac{\sqrt{6}}{36}$	$\frac{4}{9}+\frac{\sqrt{6}}{36}$			
	$-\frac{31}{120}$	$-\frac{1}{10}$	$\frac{2}{5}$			
	$-\frac{107}{1500}+\frac{89\sqrt{6}}{3000}$	$\frac{26}{125}-\frac{89\sqrt{6}}{3000}$	$\frac{3}{100}$			
	0	$\frac{1}{4}+\frac{\sqrt{6}}{36}$	$\frac{1}{4}-\frac{\sqrt{6}}{36}$			
	$\frac{1}{9}$	$\frac{4}{9}-\frac{\sqrt{6}}{36}$	$\frac{4}{9}+\frac{\sqrt{6}}{36}$			

5. Problems Test

The methods presented in section 4 were tested with four different problems in this section. The following techniques were used to carry out the numerical experiments:

Problem 1. [15] (Inhomogeneous linear problem)

$$q^{(4)}(t) = q'(t) - \cos(t),\ q(0) = -\frac{1}{2}, q'(0) = \frac{1}{2}, q''(0) = \frac{1}{2}, q'''(0) = -\frac{1}{2},$$

where t ∈ [0,2],

Exact solution is $q(t) = \frac{1}{2}\sin(t) - \frac{1}{2}\cos(t)$.

Problem 2. [15] (Homogeneous linear problem)

$$q^{(4)}(t) = q^2(t) + (q'(t))^2 + \sin(t) - 1,\ \ q(0) = 0,\ q'(0) = 1,\ q''(0) = 0\ ,\ q'''(0) = -1,$$

Where t ∈ [0,1],

The exact solution is $q(t) = \sin(\mathrm{t})$.

Problem 3. [15] (Homogeneous linear system)

$$q_1^{(4)}(t) = -\frac{e^{3t}}{4}\ q_4'(t),\ q_1(0) = 1, q_1'(0) = -1, q_1''(0) = 1, q_1'''(0) = -1,$$
$$q_2^{(4)}(t) = -16\, e^{-t}\ q_1'(t),\ q_2(0) = 1, q_2'(0) = -2, q_2''(0) = 4, q_2'''(0) = -8,$$
$$q_3^{(4)}(t) = -\frac{81\, e^{-t}}{2}\ q_2'(t),\ q_3(0) = 1, q_3'(0) = -3, q_3''(0) = 9, q_3'''(0) = -27,$$

$$q_4^{(4)}(t) = -\frac{356\, e^{-t}}{4}\, q_3'(t),\ q_4(0) = 1, q_4'(0) = -4, q_4''(0) = 16, q_4'''(0) = -64,$$

where t ∈ [0,1],

The exact solution is $q_1(t) = e^{-t},\ q_2(t) = e^{-2t},\ q_3(t) = e^{-3t},\ q_4(t) = e^{-4t}$.

Problem 4. [15] (Inhomogeneous nonlinear problem)

$$q^{(4)}(t) = -\frac{15\, q'(t)}{8\, q^6(t)},\ q(0) = 1, q'(0) = \frac{1}{2}, q''(0) = -\frac{1}{4}, q'''(0) = \frac{3}{8},$$

Where t ∈ $\left[0, \frac{\pi}{4}\right]$,

Exact solution is $q(t) = \sqrt{t+1}$.

6. Numerical Results

The tables below show the approximation outcomes for resolving issues 1-4. In the tables, the following acronyms will be used:

- **Tol**: Tolerance.
- **F. N**: the function call's number
- **STEP**: succeeded strides number
- **FSTEP**: failure strides number
- **TIME**: enforcement time
- **EDIRKTO4(3)**: the embedded 4(3) method constructed in this paper
- **EDIRKTO5(4)**: the embedded 5(4) method constructed in this paper

Table 5. Problem 1-related numerical comparisons

TOL(h)	METHOD	F.N	TIME	STEP	FSTEP
10^{-2}	EDIRKTO4(3)	21	0.068	7	0
	EDIRKTO5(4)	15	0.029	5	0
10^{-4}	EDIRKTO4(3)	57	0.081	19	0
	EDIRKTO5(4)	46	0.036	14	2
10^{-6}	EDIRKTO4(3)	212	0.102	70	1
	EDIRKTO5(4)	133	0.045	43	2

Table 6. Problem 2-related numerical comparisons

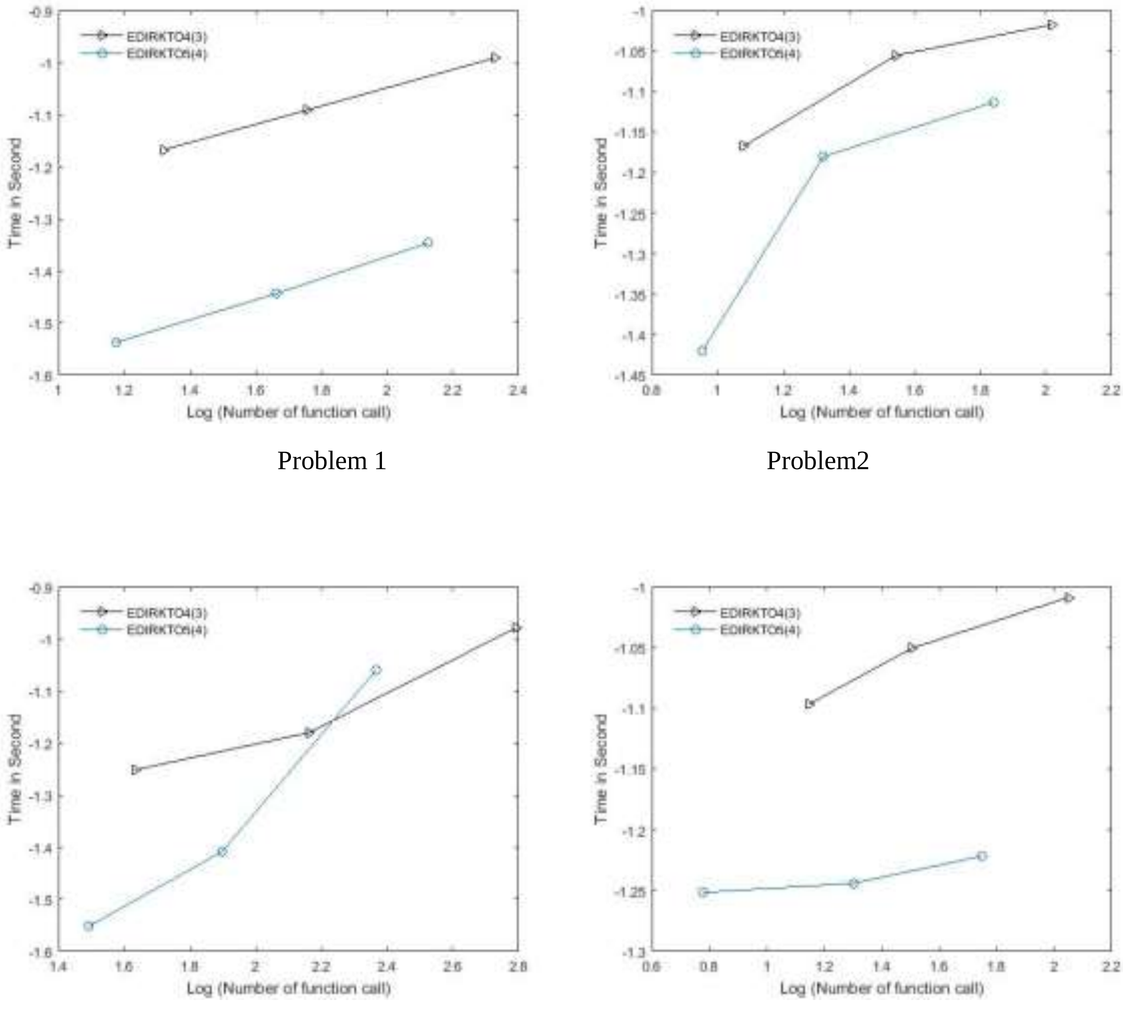
TOL(h)	METHOD	F.N	TIME	STEP	FSTEP
10^{-2}	EDIRKTO4(3)	12	0.068	4	0
	EDIRKTO5(4)	9	0.038	3	0
10^{-4}	EDIRKTO4(3)	35	0.088	11	1
	EDIRKTO5(4)	21	0.068	7	0
10^{-6}	EDIRKTO4(3)	104	0.096	34	1
	EDIRKTO5(4)	69	0.077	23	0

Table 7. Problem 3-related numerical comparisons

TOL(h)	METHOD	F.N	TIME	STEP	FSTEP
10^{-2}	EDIRKTO4(3)	43	0.056	13	2
	EDIRKTO5(4)	31	0.028	9	2
10^{-4}	EDIRKTO4(3)	145	0.066	47	2
	EDIRKTO5(4)	79	0.039	25	2
10^{-6}	EDIRKTO4(3)	621	0.105	205	3
	EDIRKTO5(4)	232	0.087	76	2

Table 8. Problem 4-related numerical comparisons

TOL(h)	METHOD	F.N	TIME	STEP	FSTEP
10^{-2}	EDIRKTO4(3)	14	0.080	4	1
	EDIRKTO5(4)	6	0.056	2	0
10^{-4}	EDIRKTO4(3)	32	0.089	10	1
	EDIRKTO5(4)	20	0.057	6	1
10^{-6}	EDIRKTO4(3)	112	0.098	36	2
	EDIRKTO5(4)	56	0.060	18	1



Problem 1 Problem2

Problem 3 Problem 4

Figure 1. Competence curves for methods

7. Discussion

In this paper, **Figure 1** illustrates the enhancements made to the Implicit Type Runge-Kutta Method of Diagonally Embedded Pair (EDIRKTO). This was done by plotting the decimal logarithm for the time curve with the highest value against the logarithm of some function call estimates that were taken from **Tables 5-8**. The current study comparison of EDIRKTO4(3) and EDIRKTO5(4) with the four different problems. Furthermore, **Figure 1** was made using the corresponding numerical results from **Tables 5–8**, respectively. In addition, as shown in **Tables 5-8,** computations of the succeeded stride number (Step) and the failure stride number (FSTEP). **Figure 1** illustrates the numerical comparison between the EDIRKTO4(3) and EDIRKTO5(4) outcomes from the current study. The current study is based on the Runge-Kutta Method, which has been previously studied by [4, 5, 8]. However, the research at hand broadened and enhanced the method from explicit to implicit and from direct to diagonal.

8. Conclusion

The solution of fourth ODEs using diagonally embedded implicit Runge-Kutta methods has been discussed in this paper. Numerical findings demonstrate that the suggested approaches are much more effective in terms of the number of function evaluations while solving the general 4th-order ODEs.

Acknowledgment

The authors are 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. Alomari, A.; Anakira, N.R.; Bataineh, A.S.; Hashim, I. Approximate solution of nonlinear system of BVP arising in fluid flow problem. *Mathematical Problems in Engineering*, **2013**, 2013,1-7.
2. Jator, S.N. Numerical integrators for fourth order initial and boundary value Problems. *International Journal of Pure and Applied Mathematics*, **2008**, *47(4)*, 563–576.
3. Kelesoglu, O. The solution of fourth order boundary value problem arising out of the beam-column theory using adomian decomposition method. *Mathematical Problems in Engineering*, **2014**, 2014,1-6.
4. Fawzi, F. A.; Abdullah, Z.M.; Hussein, N. K. Two Embedded Pairs for Solve Directly Third-Order Ordinary Differential Equation by Using Runge-Kutta Type Method (RKTGD). *Journal of Physics: Conference Series*, **2021**, *1879(2)*,1-11.
5. Hussain, K.; Ismail, F.; Senu, N. Two embedded pairs of Runge-Kutta type methods for direct solution of special fourth-order ordinary differential equations. *Mathematical Problems in Engineering*, ***2015***, 2015,1-12.
6. Ghawadri, N.; Senu, N.; Fawzi, F.A.; 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(246)*, 1-30.
7. Ghawadri, N.; Senu, N.; Fawzi, F.A.; Ismail, F.; Ibrahim, Z. B. Diagonally implicit Runge–Kutta type method for directly solving special fourth-order ordinary differential equations with Ill- Posed problem of a beam on elastic foundation,. *Algorithms*, **2018**,*12(10)*, 1-10.
8. Jumaa, M.H.; Fawzi, F.A. The Implementations of the Embedded Diagonally Implicit Type Runge-Kutta Method (EDITRKM) For Special Third Order of the Ordinary Differential Equations, *Tikrit Journal of Pure Science*, **2022**, *27(3)*, 85-91.
9. Mechee, M . Senu, N. Ismail, F . Nikouravan, Bijan ; 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, 1-9.
10. Demba, M.A.; Senu, N.; Ramos, H.; Watthayu, W. Development of an Efficient Diagonally Implicit Runge–Kutta–Nyström 5 (4) Pair for Special Second Order IVPs, *Axioms*, **2022**, *11(565)*, 1-14.
11. Demba, M.A.; Ramos, H.; Kumam, P.; Watthayu, W.; Senu, N.; Ahmed, I. A trigonometrically adapted 6 (4) explicit Runge–Kutta–Nyström pair to solve oscillating systems. *Mathematical Methods in the Applied Sciences*, **2023,** *46(1)*, 560-678.
12. 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.
13. Dormand, J. R. *Numerical methods for differential equations: a computational approach*, CRC press, **2018**, ISBN 9781315896007.
14. Gander, W. and Dominik G. Derivation of numerical methods using computer algebra. *SIAM review*, **1999**, *41(3)*, 577-593.

15. Ghawadri, N.G.F. Runge-Kutta type methods for solving high-order ordinary differential equations, PDF thesis, UPM, **2018.**
16. 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.
17. Hussain, K. A. Solving oscillation problems using optimized integrator method. *Italian Journal of Pure and Applied Mathematics*, **2022**, *47*, 578-587.
18. Ismail, F.; Senu, N.; Hussain, K. A sixth-order RKFD method with four-stage for directly solving special fourth-order ODEs. *Sains Malaysiana*, **2016**, *45(11),* 1747-1754.
19. Allogmany, R.; Ismail, F.; Majid, Z. A.; Ibrahim, Z. B. Implicit two-point block method for solving fourth-order initial value problem directly with application. *Mathematical Problems in Engineering*, **2020**, 2020, 1-13.
20. Asnor, A. I.; Yatim, S. A. M.; Ibrahim, Z. B. Numerical solution of fourth-order stiff ordinary differential equations by implicit high order block backward differentiation formula. In *AIP Conference Proceedings* , **2021**, *2423(1).* AIP Publishing.
21. Asnor, A. I.; Yatim, S. A.; M., Ibrahim, Z. B.; Zainuddin, N. High Order Block Method for Third Order ODEs. *Computers, Materials & Continua*, **2021**, *67(1)*,1253-1267.
22. 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.
23. Abolarin, O. E.; Akinola, L. S.; Adeyefa, E. O.; Ogunware, B. G. Implicit hybrid block methods for solving second, third and fourth orders ordinary differential equations directly. *Italian Journal of Pure and Applied Mathematics*, **2022**, *48*, 1-21.
24. Adeyeye, O.; Omar, Z. Implicit five-step block method with generalised equidistant points for solving fourth order linear and non-linear initial value problems. *Ain Shams Engineering Journal*, **2019**, *10(4),* 881-889.
25. Kuboye, J.; Elusakin, O. R.; Quadri, O. F. Numerical algorithms for direct solution of fourth order ordinary differential equations. *Journal of the Nigerian Society of Physical Sciences*, **2020**, *2*, 218-227.
26. Osa, A. L.; Olaoluwa, O. E. A fifth-fourth continuous block implicit hybrid method for the solution of third order initial value problems in ordinary differential equations. *Applied and Computational Mathematics,* **2019**, *8(3),* 50-57.
27. Adeleke, J. M.; Momoh, A. L.; Duromola, M. K. One-step Hybrid Block Method for Directly Solving Fifth-order Initial Value Problems of Ordinary Differential Equations. *Asian Research Journal of Mathematics*, **2022**, *18(1),* 53-64.
28. Jena, S. R.; Gebremedhin, G. S. Approximate solution of a fifth order ordinary differential equations with block method. *International Journal of Computing Science and Mathematics,* **2020**, *12(4),* 413-426.
29. Ramos, H.; Momoh, A. L. Development and implementation of a tenth-order hybrid block method for solving fifth-order boundary value problems. *Mathematical Modelling and Analysis*, **2021**, *26(2),* 267-286.
30. Adeyeye, O.; Omar, Z. Solving third order ordinary differential equations using one-step block method with four equidistant generalized hybrid points. *IAENG International Journal of Applied Mathematics,* **2019**, *49(2),*1-9.