International Journal of Mathematics Trends and Technology
ISSN: 2231-5373/ https://doi.org/10.14445/22315373/IJMTT-V70I3P101
Volume 70 Issue 3, 1-12, March 2024

Original Article

Second-Derivative Two-Step Mono-Implicit Runge-Kutta Method for Stiff ODEs

Imuwahen B. Aihie[1], Robert I. Okuonghae[2]

[1,2]*Department of Mathematics, University of Benin, Nigeria.*

Corresponding Author : decsin2002@yahoo.co.uk

Received: 11 January 2024 Revised: 23 February 2024 Accepted: 12 March 2024 Published: 30 March 2024

Abstract - *The study explores a specific class of Second Derivative Two-step mono-implicit Runge-Kutta (SDTSMIRKs) methods within a fixed step-size environment. This method is implemented as one step method in high dimension, addressing the numerical solution of stiff initial value problems (IVPs) in ordinary differential equations (ODEs). The p and q denote the order of the input and output methods respectively. Numerical results from linear and non-linear stiff systems demonstrate that the newly proposed methods surpass certain existing methods in the literature.*

Keywords - *Second-Derivative Two-step Runge-Kutta, Order condition, A−stability, A(α)− stability, stiff IVPs.*

1. Introduction

Consider a first order ordinary differential equations (ODE) of the form

$$w' = f(x, w),\ x \in [x_0, X],\ w(x_0) = w_0 \tag{1}$$

Where $f : R^s \rightarrow R^s$ and $g : R^s \rightarrow R^s$. Over the years much interest has been on the study of the Implicit Runge-Kutta (IRK) methods, (see [8]) and its subclasses, (see [4], [14] and [26]). Cash and Singhal [14] introduced a promising class of Mono Implicit Runge-Kutta (MIRK) method which is a subclass of method in [8] for the numerical solution of ordinary differential equations. These methods in [14] were further investigated by Muir and Owren [24], Burrage et al [7], De Meyer et al [17], Muir and Adams [23] and Dow [18] among others. A major advantage of MIRK methods over other IRK method is that it is very cheap to implement in term of the number of non-linear equations to be solved. while the pessimistic nature of MIRK method is the order reduction which gives rooms for the generalization of MIRK method as documented in Dow [18]. The general form of the method considered in [14] is given as

$$W_i = (1 - v_i)w_n + v_i w_{n+1} + h\sum_{j=1}^{i-1} X_{ij}\, f(x_n + c_j h, W_j), i = 1,2,\ldots,s;$$
$$w_{n+1} = w_n + h\sum_{i=1}^{s} b_i\, f(x_n + c_i h, W_i). \tag{2}$$

Where $c_i = (c_1,\ldots,c_s)^T, v_i = (v_1,\ldots,v_s)^T, b_i = (b_1,\ldots,b_s)^T$, X is the s by s matrix. The abscissa

$c_i = \sum_{j=1}^{i-1} x_{ij} + v_r$, (i.e, $c = Xe + v$) . However, the method in (2) requires starter during implementation on stiff problem. To obtain higher order scheme with good accuracy, authors including Chan and Tsai [16], Okuonghae [35], Turaci and Ozis [36], Aiguobasimwin and Okuonghae [1] and Okuonghae and Aiguobasimwin [31], Enright [20], Butcher and Hojjati in [11], Okuonghae and Ikhile in [32], Okuonghae and Ikhile [33], Olatunji and Ikhile in [28], Ogunfeyitimi and Ikhile in [30] incorporates the second derivative term for non-stiff and stiff problems. Moreso, to avoid introducing starting values during implementation, self-starting methods were considered, see Fatunla (1990, 1992), Brugnano and Trigiante,Ikhile (1994) (2000), Jator (2010) and Ikhile and Muka (2015). Our aim is to modify the method in (2) by incorporating second derivative function $w^{00} = f_x + f_w f = g(x,w)$ which are self-starting during implementation on stiff problem. The methods are of higher order compared to the conventional Runge-Kutta method, especially when implemented as one step method in higher dimensions. The proposed scheme is of the form:

$$W_m = (1 - v_m)w_n + v_m w_{n+1} + h\sum_{j=1}^{m-1} X_{mj}\, f(x_n + c_j h, W_j) + h^2 \sum_{j=1}^{m-1} \overline{X}_{mj}\, g(x_n + c_j h, W_j), \quad c_m \in (0,1), m = 1,2\ldots,s; \quad (3)$$

and

$$w_{n+1} = w_n + h\sum_{m=1}^{s-1} b_m(1) f(x_n + c_m h, W_m) + h^2 \sum_{m=1}^{s} \overline{b}_m(1) g(x_n + c_m h, W_m), \quad \theta = 1. \quad (4)$$

The $g(x,w)$ is the second derivative form of ODEs in (1), $c_m = (c_1,\ldots,c_s)^T$, is the abscissa value and $W_m = w(x_n + c_m h)$, the coefficients, $\{v_m\}_{m=1}^{s}, \{x_{mj}\}_{j=1,m=1}^{m-1,s}, \{\bar{x}_{mj}\}_{j=1,m=1}^{m-1,s}$, defined the stages, $\{b_m(\theta)\}_{m=1}^{s}$ and $\{\overline{b}_m(\theta)\}_{m=1}^{s}$, are the weight polynomials. We shall require $c_m = \sum_{j=1}^{m-1} x_{mj} + \sum_{j=1}^{m-1} \overline{x}_{mj} + v_m$ and $\theta = 1$, i.e $b_m(1) = b_m$ and $\bar{b}_m(1) = \bar{b}_m$. In equation (4) the derivative side on the left-hand side has $s-1$ stage, this algorithm is designed in such a way that stage order q equal the output order p.

The paper is organized as follows. In section 2 and 3, the order condition and stability analysis of the SDTSMIRK method are stated. Section 4 is devoted to the derivation of the SDTSMIRK method and section 5, numerical results are presented. The Butcher's tableaux of the method in (3) and (4) is

$$\begin{array}{c|c|c|c} c & v & X & \bar{X} \\ \hline & & b^T & \bar{b}^T \end{array} = \begin{array}{c|c|c|c} c_1 & v_1 & x_{11}\ldots x_{1s} & \bar{x}_{11}\ldots \bar{x}_{1s} \\ \vdots & \vdots & \vdots & \vdots \\ c_s & v_s & x_{s1}\ldots x_{ss} & \bar{x}_{1s}\ldots \bar{x}_{ss} \\ \hline & & b_1\ldots b_s & \bar{b}_1\ldots \bar{b}_s \end{array} \quad (5)$$

Where $c = (c_1,\ldots,c_s)^T$, $v = (v_1,\ldots,v_s)^T$, $b = (b_1,\ldots,b_s)^T$, $\bar{b} = (\bar{b}_1,\ldots,\bar{b}_s)^T$, X and $\bar{X}$ are the s matrix whose $(m,j)th$ components are x_{mj} and $\bar{x}_{mj}$ respectively.

2. The Order Condition of the SDTSMIRK Methods

The order condition of the method in (3) and (4) are obtained by Taylor's series expansion approach about x_n and equating the power of h to zero gives stage order q

$$c = Xe + v;\ \tau = 1,\ e = (1,1,\ldots,1);$$
$$\frac{c^\tau}{\tau!} = \frac{Xc^{\tau-1}}{(\tau-1)!} + \frac{\overline{X}c^{\tau-2}}{(\tau-2)!} + \frac{v}{\tau!}; \qquad \tau = 2(1)q, \quad (6)$$

and the method of order p

$$b^T e = e;$$
$$\frac{1}{\tau!} = \frac{b^T c^{\tau-1}}{(\tau-1)!} + \frac{\overline{b^T} c^{\tau-2}}{(\tau-2)!}; \qquad \tau = 2(1)p. \quad (7)$$

3. Stability Analysis

In this section, our interest is on the analysis of the stability of the method in (3) and (4). In what follows is the derivation of the stability function of the method in (3) and (4).

Theorem 3.1. *Let $R(z)$ denote the stability function for a SDTSMIRK method. Then for a linear differential equation $w'(x) = \lambda w(x)$, the methods in* (3) *and* (4) *has the stability function*

$$R(z) = \frac{\det[I - z\mathrm{x} - z^2\bar{\mathrm{x}} + ze b^T + z^2\bar{b}^T - zvb^T - z^2 v\bar{b}^T]}{\det[I - z\mathrm{x} - z^2\bar{\mathrm{x}} - zvb^T - z^2 v\bar{b}^T]}, \qquad z = \lambda h \quad (8)$$

Proof: For the special problem defined by $w' = \lambda w$, the stage derivatives f and $w^{00} = g$ is related to the stage values W by $f = \lambda w$ and $g = \lambda^2 w$. For convenience, we take $e = (1,...,1)^T$ and $v = (v_1,...,v_s)^T$, Hence, (3) reduces to the form

$$(I - zX - z^2\bar{X})W - vw_{n+1} = (e - v)w_n ,z = \lambda h \quad (9)$$

and

$$(-zbT - z2\bar{}bT)W + wn+1 = wn \quad (10)$$

From (9) we have,

$$W = ((e - v)w_n + vw_{n+1})(I - zX - z^2\bar{X})^{-1} \quad (11)$$

Inserting (11) into (10) gives

$$\left(-zb^T - z^2\bar{b}^T\right)\left((e-v)w_n + vw_{n+1}\right)(I - z\text{x} - z^2\bar{\text{x}}))^{-1} + w_{n+1} = w_n \quad (12)$$

Simplifying (12) and collecting like terms yields

$$[v(-zb^T - z^2\bar{b}^T) + (I - zX - z^2\bar{X})]w_{n+1} = [(I - zX - z^2\bar{X})(e - v)(-zb^T - z^2\bar{b}^T)]w_n. \quad (13)$$

From (14) we obtain $w_{n+1} = R(z)w_n$. Thus, the stability function is

$$R(z) = \frac{\det[\, I - z\text{x} - z^2\bar{\text{x}} + ze b^T + z^2 e\bar{b}^T - zvb^T - z^2 v\bar{b}^T]}{\det\,[I - z\text{x} - z^2\bar{\text{x}} - zvb^T - z^2 v\bar{b}^T]} \quad (14)$$

Definition 3.1. *A numerical method in* (4) *is said to be A-stable if* | $R(z)$ |6 1 ∀ $Re(z)$ 6 0

Definition 3.2. *A numerical method in* (4) *is said to be* $A(\alpha)$−*stable for some* $\alpha \in \left[0, \frac{\pi}{2}\right]$ *if the wedge* $S_\alpha =: |Arg(-z)| \leq \alpha,\ z \neq 0$ *is contained in its region of absolute stability.*

4. Construction of the SDTSMIRK Method

In this section, we will derive method (3) and (4) that has order p and stage order $q=p$. We consider such methods because there are some strong theoretical and numerical evidences that methods with $p=q$ has the greatest potential for practical use, (see [9], [10], [33], [34]). Thus, we will restrict our discussion and investigation to such schemes. The approach adopted here in the derivation of the method in (3) and (4) is similar to that used in [33], and [34].

4.1. SDTSMIRK method of order p = 1, s = 1

For example, fixing $m = 1$, and $v_1 = 0$ in (3) gives

$$W_1 = w_n \quad (15)$$

Similarly, we obtain the output method of order $p = 1$ in (4) for $s = 1$. That is

$$w_{n+1} = w_n + hf(x_n,W_1) \quad (16)$$

The tableau for (16) is

$$\begin{array}{c|c|c|c} c & v & X & \bar{X} \\ \hline & & b^T & \bar{b}^T \end{array} = \begin{array}{c|c|c|c} 0 & 0 & 0 & 0 \\ \hline & & 0 & 1 \end{array} \quad (17)$$

The method in (15) and (16) is an explicit Euler's method, which is not of interest in this paper but such schemes are suitable for non-stiff ODEs. The Euler's scheme has an interval of absolute stability of [-2,0].

4.2. SDTSMIRK method of order p = 3, s = 2

Taking $r = 2$ in (3) and fix $v_1 = 1$ gives

$$W1 = wn; \qquad W2 = wn+1$$

$$w_{n+1} = w_n + f(x_n, W_1) + \frac{h^2}{3} g(x_n, W_1) + \frac{h^2}{6} g(x_{n+1}, W_2) \quad (18)$$

The tableau of the scheme in (18) is given as

$$\begin{array}{c|c|c|c} c & v & X & \bar{X} \\ \hline & & b^T & \bar{b}^T \end{array} = \begin{array}{c|c|cc|cc} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ \hline & & 1 & 0 & \frac{1}{3} & \frac{1}{6} \end{array} \qquad (19)$$

The algorithm in (18) is of order $p = 3$, the interval of absolute stability of the method is [-2,0] and such scheme is good for the numerical solution of non-stiff ODEs (1). Our interest in this study is implicit Runge-Kutta method. Therefore, we give below some suitable methods emanating from (3) and (4) for stiff problems (1).

4.3. SDTSMIRK method of order p = q = 5, s = 3

Fixing $s = 3$, p=q=5 in (6) and (7) and solving the resulting system of linear equations in terms of $\{c_m\}_{m=1}^{3}$ such that c_1 6= c_2 6= c_3. The resulting tableau of the method of order $p = 5$ is

$$\begin{array}{c|c|c|c} c & v & X & \bar{X} \\ \hline & & b^T & \bar{b}^T \end{array} = \begin{array}{c|c|ccc|ccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3} & \frac{192}{243} & \frac{18}{243} & -\frac{48}{243} & 0 & \frac{2}{243} & \frac{4}{243} & 0 \\ \hline & & \frac{24}{120} & \frac{96}{120} & 0 & \frac{1}{120} & -\frac{10}{120} & -\frac{27}{120} \end{array} \qquad (20)$$

The stability function of the method in (20) $R(z) = \frac{388800+77760z-15120z^2-6480z^3-720z^4}{388800-311040z+101520z^2-17280z^3+1440z^4}$ is and plotting the stability function of (20) in boundaries locus sense shows that the scheme in (20) is A-stable. The *SDTSMIRK* method of order p = q = 5, s = 3 is represnted by *SDTSMIRK*5.

4.4. SDTSMIRK method of order p = q = 7, s = 4

Similarly, setting $p = q = 7$, $s = 4$ in (6) and (7) and solving the resulting system of linear equations in terms of $\{c_m\}^4{}_{m=1}$ such that c_1 6= c_2 6= c_3 =6 c_4. The resulting tableau of the method of order $p = 7$ is;

$$\begin{array}{c|c|c|c} c & v & X & \bar{X} \\ \hline & & b^T & \bar{b}^T \end{array} = \begin{array}{c|c|cccc|cccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{96}{192} & \frac{18}{192} & -\frac{18}{192} & 0 & 0 & \frac{1}{192} & \frac{1}{192} & -\frac{8}{192} & 0 \\ \frac{2}{3} & \frac{1214}{2187} & \frac{178}{2187} & -\frac{224}{2187} & \frac{256}{2187} & 0 & \frac{10}{2187} & \frac{12}{2187} & -\frac{64}{2187} & 0 \\ \hline & & \frac{160}{840} & \frac{40}{840} & \frac{640}{840} & 0 & \frac{9}{840} & \frac{2}{840} & -\frac{32}{840} & \frac{81}{840} \end{array} \qquad (21)$$

The stability function of the method in (21) $R(z) = \frac{272160+155520z+39780z^2+5940z^3+561z^4+33z^5+z^6}{272160-116640z+20340z^2-1440z^3-69z^4+24z^5-2z^6}$ is and the method in (21) is A-stable as shown in the stability plot in Figure 1. Again, the *SDTSMIRK* method of order $p = q = 7$, $s = 4$ will be reference as *SDTSMIRK*7.

4.5. SDTSMIRK method of order p = q = 9, s = 5

Setting $s = 5, c = (0,1,\frac{1}{2},\frac{2}{3},\frac{3}{4})^T$ in (6) and (7) yield the *SDTSMIRK* methods of order 9

with the modified Butcher tableaux of the resulting coefficients given below.

$$\begin{array}{c|c|c|c} c & v & X & \bar{X} \\ \hline & & b^T & \bar{b}^T \end{array} \qquad (22)$$

Where,

$$v = (0,1,\frac{11184}{12288},\frac{17952}{19683},\frac{959472}{1048576})^T,$$

$$X = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{165}{12288} & \frac{-1248}{12288} & \frac{-6144}{12288} & \frac{2185}{12288} & 0 \\ \frac{259}{19683} & \frac{-2000}{19683} & \frac{-8192}{19683} & \frac{5103}{19683} & 0 \\ \frac{13341}{1048576} & \frac{-106272}{1048576} & \frac{-414720}{1048576} & \frac{334611}{1048576} & 0 \end{pmatrix},$$

$$\overline{X} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{7}{12288} & \frac{40}{12288} & \frac{-512}{12288} & \frac{-729}{12288} & 0 \\ \frac{11}{19683} & \frac{64}{19683} & \frac{-768}{19683} & \frac{-1215}{19683} & 0 \\ \frac{567}{1048576} & \frac{3344}{1048576} & \frac{-39168}{1048576} & \frac{-59049}{1048576} & 0 \end{pmatrix},$$

$$b(1)^T = (\frac{3735}{25200},\frac{-4320}{25200},\frac{-138240}{25200},\frac{164025}{25200},0)^T,$$

$$\overline{b}(1)^T = (\frac{157}{25200},\frac{312}{25200},\frac{-12288}{25200},\frac{-19683}{25200},\frac{8192}{25200})^T.$$

The stability function is

$$N(z) = (2090188800 + 979292160z + 212365440z^2 + 929694720z^3 + 2531640z^4 + 161088z^5 + 7224z^6 + 212z^7 + 3z^8)$$

$$D(z) = (2090188800 - 1110896640z + 278167680z^2 - 42936480z^3 + 4442520z^4 - 305352z^5 + 11508z^6 + 108z^7 - 36z^8). \quad (23)$$

The stability plot for the method of order $p = 9$ in Figure 1 is $A(\alpha) - stable$, similarly the method of order $p = q = 9$, $s = 5$ implies *SDTSMIRK*9.

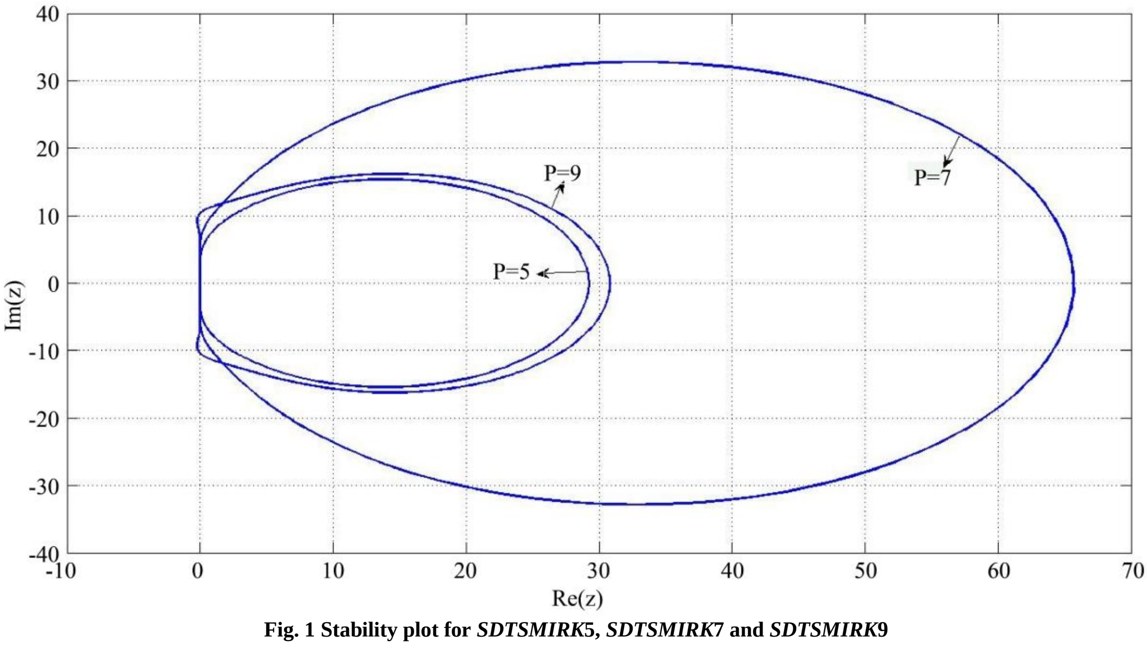

Fig. 1 Stability plot for *SDTSMIRK*5, *SDTSMIRK*7 and *SDTSMIRK*9

The Fig.1 shows the stability plot for second derivative two-step mono-implicit Runge-Kutta methods for *p*=5, 7 and 9, the exterior of the closed curve is the stability region. It is shown from Table 1 that the SDTSMIRK method has relatively small error constant and by Definition 3.1 the method in (3) and (4) are A-stable for order 5,7 and 9. Figure 2. shows that the SDTSMIRK scheme in (4) has small error constant compare to the methods in Cash [13], Ogunfeyitimi and Ikhile [29], Okor and Nwachukwu [37] for the same order *p*=5, 7 and $A(\alpha)$−*stable* for 9. This confirm theoretically the possibility of having more accurate solution on stiff problem in (1).

Table 1. Properties of SDTSMIRK

S	Error Constant	Order	Zero Stability	Stability Properties of SDTSMIRK Method
3	$\frac{1}{21600}$	5	Zero Stable	A-stable
4	$\frac{-1}{15240960}$	7	Zero Stable	A-stable
5	$\frac{-43}{219469824000}$	9	Zero Stable	A(α)-stable

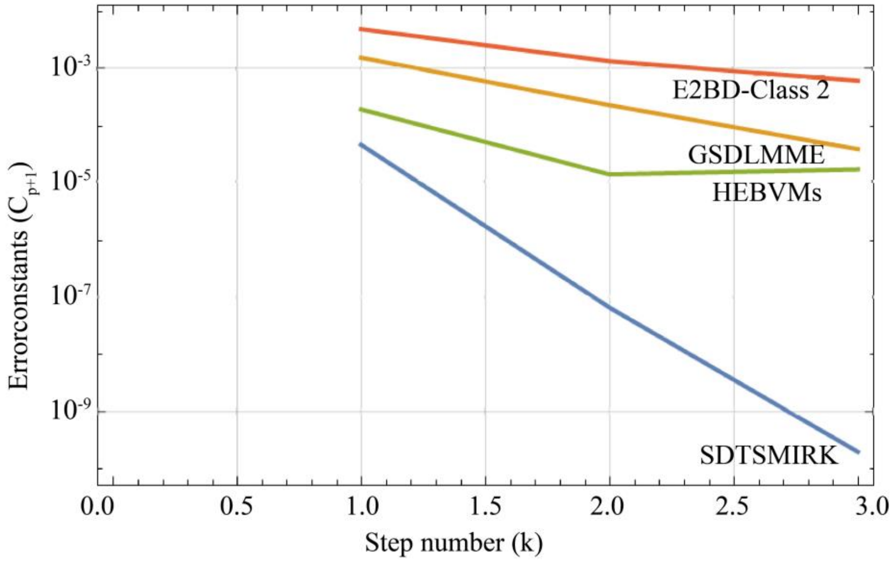

Fig. 2 Graph of the absolute values of the error constants against order *P* = 5,7 and 9 of the SDTSMIRK, HEBVMs [37], GSDLMME [29] and E2BD [13]

5. Implementation Procedures

This section presents an illustration for the implementation of SDTSMIRK method for s = 3, 4 and 5 with order p = 5, p = 7 and 9 respectively, as one step method in higher dimension by following the procedure in Jator [22], Akinfenwa [2] and Okor and Nwachukwu [37]. The 5*th*, 7*th* and 9*th* order are denoted by SDTSMIRK5, SDTSMIRK7 and SDTSMIRK9 respectively.

The 5*th* order of SDTSMIRK

$$w_{n+2/3} = \frac{51}{243}w_n + \frac{192}{243}w_{n+1} + \frac{18}{243}hf_n - \frac{48}{243}hf_{n+1} + \frac{2}{243}h^2g_n + \frac{4}{243}h^2g_{n+1}$$

$$w_{n+1} = w_n + \frac{24}{120}hf_n + \frac{96}{120}hf_{n+1} + \frac{1}{120}h^2g_n - \frac{10}{120}h^2g_{n+1} - \frac{27}{120}h^2g_{n+\frac{2}{3}},$$

The 7*th* order of SDTSMIRK

$$w_{n+1/2} = \frac{96}{192}w_n + \frac{96}{192}w_{n+1} + \frac{18}{192}hf_n - \frac{18}{192}hf_{n+1} + \frac{1}{192}h^2g_n + \frac{1}{192}h^2g_{n+1} - \frac{8}{192}h^2g_{n+\frac{1}{2}}$$

$$w_{n+\frac{2}{3}} = \frac{939}{2187}w_n + \frac{1248}{2187}w_{n+1} + \frac{178}{2187}hf_n - \frac{224}{2187}hf_{n+1} + \frac{256}{2187}hf_{n+\frac{1}{2}} + \frac{10}{2187}h^2g_n + \frac{12}{2187}h^2g_{n+1} - \frac{64}{2187}h^2g_{n+\frac{1}{2}}$$

$$w_{n+1} = w_n + \frac{160}{840}hf_n + \frac{40}{840}hf_{n+1} + \frac{640}{840}hf_{n+\frac{1}{2}} + \frac{9}{840}h^2g_n + \frac{2}{840}h^2g_{n+1} - \frac{32}{840}h^2g_{n+\frac{1}{2}} + \frac{81}{840}h^2g_{n+\frac{2}{3}}$$

and the 9*th* order of SDTSMIRK

$$w_{n+\frac{1}{2}} = \frac{1104}{192}w_n + \frac{11184}{12288}w_{n+1} + \frac{165}{12288}hf_n - \frac{1248}{12288}hf_{n+1} - \frac{6144}{12288}hf_{n+\frac{1}{2}} + \frac{2187}{12288}hf_{n+\frac{2}{3}} + \frac{7}{12288}h^2g_n + \frac{40}{12288}h^2g_{n+1} - \frac{512}{12288}h^2g_{n+\frac{1}{2}} - \frac{729}{12288}h^2g_{n+\frac{2}{3}}$$

$$w_{n+\frac{2}{3}} = \frac{1731}{2187}w_n + \frac{17952}{19683}w_{n+1} + \frac{259}{19683}hf_n - \frac{2000}{19683}hf_{n+1} - \frac{8192}{19683}hf_{n+\frac{1}{2}} + \frac{5103}{19683}hf_{n+\frac{2}{3}} + \frac{11}{19683}h^2g_n + \frac{64}{19683}h^2g_{n+1} - \frac{768}{19683}h^2g_{n+\frac{1}{2}} - \frac{1215}{19683}h^2g_{n+\frac{2}{3}}$$

$$w_{n+\frac{3}{4}} = \frac{89104}{1048576}w_n + \frac{959472}{1048576}w_{n+1} + \frac{13341}{1048576}hf_n - \frac{106272}{1048576}hf_{n+1} - \frac{414720}{1048576}hf_{n+\frac{1}{2}} + \frac{334611}{1048576}hf_{n+\frac{2}{3}} + \frac{567}{1048576}h^2g_n + \frac{3384}{1048576}h^2g_{n+1} - \frac{39168}{1048576}h^2g_{n+\frac{1}{2}} - \frac{59049}{1048576}$$

$$w_{n+1} = w_n + \frac{3735}{25200}hf_n - \frac{4320}{25200}hf_{n+1} - \frac{138240}{25200}hf_{n+\frac{1}{2}} + \frac{164025}{25200}hf_{n+\frac{2}{3}} + \frac{157}{25200}h^2g_n + \frac{312}{25200}h^2g_{n+1} - \frac{12288}{25200}h^2g_{n+\frac{1}{2}} - \frac{19683}{25200}h^2g_{n+\frac{2}{3}} + \frac{8192}{25200}h^2g_{n+\frac{3}{4}}$$

The main output obtained from (4) and input method derive from (3) give a block form of the same order, by this the main output and input method are combined as a one-step method in higher dimension given as

$$A1W\varphi+1 + A0W\varphi = h(B0F\varphi + B1F\varphi+1) + h2(C0G\varphi + C1G\varphi+1). \quad (24)$$

Where $p_i = \frac{i}{s},\ \ i = 1,2,\ldots,s$
$W_{\phi+1} = [w_{n+p_1}, w_{n+p_2}, \ldots, w_{n+p_{2-1}}, w_{n+p_s}]^T$

$$W_\phi = [w_{n-p_{s-1}}, w_{n-p_{s-2}}, \ldots, w_{n-p_1}, w_n]^T$$
$$F_{\phi+1} = [f_{n+p_1}, f_{n+p_2}, \ldots, f_{n+p_{2-1}}, f_{n+p_s}]^T$$
$$F_\phi = [f_{n-p_{s-1}}, f_{n-p_{s-2}}, \ldots, f_{n-p_1}, f_n]^T$$
$$G_{\phi+1} = [g_{n+p_1}, g_{n+p_2}, \ldots, g_{n+p_{2-1}}, g_{n+p_s}]^T$$
$$G_\phi = [g_{n-p_{s-1}}, g_{n-p_{s-2}}, \ldots, g_{n-p_1}, g_n]^T$$

$$A_1 = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}, A_0 = \begin{pmatrix} 0 & 0 & \cdots & a_{10} \\ 0 & 0 & \cdots & a_{20} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{s0} \end{pmatrix}, B_0 = \begin{pmatrix} 0 & 0 & \cdots & b_{10} \\ 0 & 0 & \cdots & b_{20} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & b_{s0} \end{pmatrix}, B_1 = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1s} \\ b_{21} & b_{22} & \cdots & b_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ b_{s1} & b_{s2} & \cdots & b_{ss} \end{pmatrix}, C_0 = \begin{pmatrix} 0 & 0 & \cdots & c_{10} \\ 0 & 0 & \cdots & c_{20} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_{s0} \end{pmatrix}, C_1 = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1s} \\ c_{21} & c_{22} & \cdots & c_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ c_{s1} & c_{s2} & \cdots & c_{ss} \end{pmatrix} \quad (25)$$

Which simultaneously generate block solution values. The summary of the implementation procedure for order $7,\ s = 4$ is as follows:

let the partition $\Pi_N : a = x_0 < x_1 < \ldots < x_n < x_{n+1} = b, h = x_{n+1} - x_n, n = 0(1)N - 1$

Step1: input value of N, for $s = 4,\ h = \frac{b-a}{N}$, the number of block $l = \frac{N}{3}$. using (25) $n = 0$, $\varphi = 0$, the solution value of $(w_{\frac{1}{2}}, w_{\frac{2}{3}}, w_1)^T$ are generated simultaneously over the

sub-interval $[x_0,x_1]$, where w_0 is provided by the problem (1).
Step2: $n = 1$, $\varphi = 1$, the solution $(w_{\frac{3}{2}}, w_{\frac{5}{3}}, w_5)^T$ are obtained over the sub-interval $[x_1,x_2]$,

since w_1 is generated from the previous block.
Step3: the iteration is continued for $n = 2,...N-2$ and $\varphi = 2,...,\tau$ to generate solution of
(1) on sub-intervals $[x_2,x_3]...[x_{N-1},x_N]$.
By this, the accumulation error is in significant in the numerical solution, since the solution are generated concurrently, see ([19], [27]). In the case of non-linear problems, a modified Newton-Raphson method such as

$$W_{\emptyset+1}^{[i+1]} = W_{\emptyset+1}^{[i]} - \frac{\partial M\left(W_{\emptyset+1}^{[1]}\right)}{\partial w_{n+1}})^{-1} \quad M(W_{\emptyset+1}^{[1]}); i = 0(1)q \quad q \geq 1,$$

where

$$\frac{\partial F(W_{\emptyset+1})}{\partial W_{\emptyset+1}} = \frac{\partial f_{n+1},\dots\dots,f_{n+s}}{\partial w_{n+1},\dots\dots,w_{n+s}} == \begin{pmatrix} \frac{\partial f_{n+1}}{\partial w_{n+1}} & \frac{\partial f_{n+1}}{\partial w_{n+2}} & \cdots & \frac{\partial f_{n+1}}{\partial w_{n+s}} \\ \frac{\partial f_{n+2}}{\partial w_{n+1}} & \frac{\partial f_{n+2}}{\partial w_{n+2}} & \cdots & \frac{\partial f_{n+2}}{\partial w_{n+s}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_{n+s}}{\partial w_{n+1}} & \frac{\partial f_{n+s}}{\partial w_{n+2}} & \cdots & \frac{\partial f_{n+s}}{\partial w_{n+s}} \end{pmatrix},$$

and

$$\frac{\partial F(W_{\emptyset+1})}{\partial W_{\emptyset+1}} = \frac{\partial g_{n+1},\dots\dots,g_{n+s}}{\partial w_{n+1},\dots\dots,w_{n+s}}$$

$M(W\varphi+1) = A1W\varphi+1 + A0W\varphi - hB0F\varphi - hB1F\varphi+1 - h2C0G\varphi - h2C1G\varphi+1 = 0.$

6. Numerical Experiment

We present numerical results showing the implementation and accuracy of the constructed SDTSMIRK5, SDTSMIRK7 and SDTSMIRK9 in (20), (21) and (22) respectively. The order of SDTSMIRK5, SDTSMIRK7 and SDTSMIRK9 are $p = 5$, $p = 7$ and $p = 9$ respectively,
see Section 5 of this article. The implementation is done in fixed step size mode for accuracy purpose. Our interest here is to compare the results of our methods with the results obtained from some existing methods. Computational experiments are done by applying the
SDTSMIRK5, SDTSMIRK7 and SDTSMIRK9 methods to the problems below:
Problem 1: Consider the system of differential equations [29],

$$\begin{cases} w_1'(x) = -21w_1 + 19w_2 - 20w_3, \quad w_1(x) = \frac{1}{2}(e^{-2x} + e^{-40x}(\cos(40x) + \sin(40x))) \\ w_2'(x) = -19w_1 - 21w_2 - 20w_3, \quad w_2(x) = \frac{1}{2}(e^{-2x} + e^{-40x}(\cos(40x) + \sin(40x))) \\ w_3'(x) = 40w_1 - 40w_2 - 40w_3, \quad w_2(x) = \left(-e^{-40x}(\cos(40x) + \sin(40x))\right) \\ x \in [0,1], w(0) = [1,0-1]^T \end{cases}$$

See Table 2 for computed result of problem 1
Table 2 show that the new methods *SDTSMIRK*5 performs better in terms of accuracy
than the existing schemes in [29], [30], [5] and they are suitable for integrating stiff system in ordinary differential equations (ODEs).
Problem 2: Non-linear stiff system [29],

$$\begin{cases} w_1' = -(\epsilon^{-1} + 2)w_1 + \epsilon^{-1} w_2^2, \quad w_1(0) = 1 \\ w_2' = w_1 - w_2(1 + w_2), \quad w_2(0) = 1 \\ \textit{The Exact Solution} \\ w_1(x) = w_2^2, w_2(x) = e^{-x} \end{cases}$$

Table 2. Numerical results for Problem 1

h	SDTSMIRK5 (Rate) p=5	GSDLMM3[29] (Rate) p=5	IMEXSDLMM[30] (Rate) p=6	AMODIO6[5] (Rate) p=6
0.05	1.86e - 003 (-)	3.0e - 002 (-)	6.20e - 002 (-)	5.70e - 002 (-)
0.025	1.23e - 004 (3.95)	3.55e - 003 (3.07)	9.20e - 002 (2.75)	8.70e - 003 (2.70)
0.0125	4.33e - 006 (5.28)	2.26e - 004 (3.97)	5.61e - 004 (4.03)	4.9e - 004 (4.20)
0.00625	8.87e - 008 (5.77)	5.86e - 006 (5.27)	1.09e - 005 (5.68)	1.20e - 005 (5.80)

Table 3. Comparison of results at t = 1 and maximum absolute Error, for Problem 2

Methods	Order	N	h	w_1 $Max \parallel w_i - (w(x_i) \parallel$	w_2 $Max \parallel w_i - (w(x_i) \parallel$
SDTSMIRK5	5	125	0.008	2.48e - 016	1.11e - 016
GSDLMM3[29]	5	125	0.008	6.88e - 015	3.33e - 015
SDGBDF5[27]	5	125	0.008	1.80e - 015	6.11 - 016
Ehigieetal(BV M3)[19]	5	125	0.008	3.88e - 014	3.10e - 014

Table 4. Comparison of results for Problem 2, Errory_i=(| $w_i - w(x_i)$ |), i=1,2

Methods	x	h	N	Errorw_1	Errory_2
SDTSMIRK7 P=7	1	0.05	20	2.8441e - 016	4.5775e - 016
HEBV M3[37] P=7	1	0.05	20	1.1675e - 013	1.9218e - 013s
BBDFs [3] P=8	1	0.05	20	4.5602e - 013	6.263e - 013

From Table 3, it can be seen that our method *SDTSMIRK*5 of order 5 performs better than the methods of [29], [27] and [19]. In like manner, Table 4 show that the new method *SDTSMIRK*7 of order 7 outperformed the methods of [37] and [3]

Problem 3: Let consider another stiff system which has also been solved by Cash [13],

$$\begin{cases} w_1' = -\alpha w_1 - \beta w_2 + (\alpha + \beta - 1)e^{-x}, & w_1(0) = 1, \\ w_2' = \beta w_1 - \alpha w_2 + (\alpha - \beta - 1)e^{-x}, & w_2(0) = 1, \\ The\ stiffness\ ratio\ of\ this\ problem\ is\ 1:200\ and\ the\ exact\ solution\ is \\ w_1(x) = e^{-x},\ w_2(x) = e^{-x} \end{cases}$$

The problem 3 is integrated with $h = 0.01$, $h = 0.09$, $h = 0.2$ and $h = 0.25$ for the purpose of comparison. Thus, the results for $h = 0.01$, $h = 0.09$, $h = 0.2$ and $h = 0.25$ are tabulated at different values of x to show the performance of the method. Table 5, 6 and 7, reveals that the newly derived schemes in (20), (21) and (22) are better in terms of accuracy than the *ECBBDF*5 [2], *E*2*BD* [13] and *HEBV M*5 [37]

Table 5. Numerical results for Problem 3, $\alpha = 1$, $\beta = 30$, $h = 0.01$. *error* w_i =| $w_i - w(x_i)$ |), $i = 1,2$.

x	w_i	Error in SDTSMIRK5 p=5	Error in ECBBDF[2] p=5
1.0	w_1	2.00e - 016	1.28e - 015
	w_2	7.85e - 017	1.17e - 014
10.0	w_1	3.45e - 020	1.08e - 019
	w_2	1.48e - 019	1.62e - 018
20.0	w_1	7.31e - 024	7.25e - 023
	w_2	1.16e - 024	5.29e - 023

Table 6. Absolute Error for Problem 3, $h = 0.09$

x	w_i	*SDTSMIRK7* p=7	*SDTSMIRK9* p=9	*E2BD* Class 2 [13] p=8	*HEBV M5*[37] p=9
4.5	w_1	0.2e - 18	0.2e - 18	0.1e - 010	0.4e - 016
	w_2	0.1e - 18	0.3e - 18	0.1e - 010	0.4e - 016
9.0	w_1	0.4e - 20	0.2e - 19	0.1e - 012	0.7e - 018
	w_2	0.2e - 20	0.4e - 20	0.1e - 012	0.5e - 018
13.5	w_1	0.9e - 21	0.3e - 021	0.8e - 011	0.9e - 020
	w_2	0.8e - 21	0.2e - 021	0.6e - 011	0.6e - 020
18	w_1	0.8e - 23	0.6e - 023	0.1e - 011	0.1e - 021
	w_2	0.1e - 22	0.1e - 023	0.1e - 011	0.1e - 021

Table 7. Maximum Absolute Error $max|w_i - w(x_i)|$, $i = 1,2$ for for Problem 3

N	α	β	h	SDTSMIRK7 p=7	SDTSMIRK9 p=9	HEBV M5[35] P=9
50	1	30	0.2	0.6e - 18	5.6e - 018	1.4e - 015
80	1	30	0.25	0.1e - 21	9.1e - 021	6.9e - 019
50	1	10^4	0.2	0.00	0.00	5.e - 018
80	1	10^4	0.25	0.2e - 20	4.1e - 025	2.2e -021
50	1	10^5	0.2	0.3e - 021	6.7e - 021	5.6e - 019
80	1	10^5	0.25	0.8e - 21	4.1e - 025	2.2e - 022

7. Conclusion

We have proposed a family of Second Derivative Two-step Mono-Implicit Runge-Kutta method for the numerical solution of stiff IVPs in ODEs. The stability analysis is documented in Section 4. Figure 2 contain Error constant of SDTSMIRK method, GSDLMME [29], E2BD [13] and HEBVM5 [37], where the SDTSMIRK method possess smaller error constant than the compared method in [29], [13] and [37] to stiff system theoretically. The algorithms are self-starting and deliver high accuracy. The numerical example employing the suggested methods demonstrated the method's accuracy, as evident in Table 2-7 above.

Authors' Contribution

Both authors prepared the manuscript.

Acknowledgment

The authors will like to express their gratitude to the unknown referees for carefully reading the paper and for their helpful comments.

References

[1] I.B. Aiguobasimwin, and R.I. Okuonghae, "A Class of Two-Derivative Two-Step Runge-Kutta methods for Non-stiff ODEs," *Hindawi Journal of Applied Mathematics*, vol. 2019, pp. 1-10, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[2] O.A. Akinfenwa, and S.N. Jator, "Extended Continuous Block Backward Differentiation Formula for Stiff Systems," *Fasciculi Mathematici*, no. 55, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[3] Olusheye Akinfenwa, Samuel Jator, and Nianmin Yoa, "An Eight Order Backward Differentiation Formula with Continuous Coefficients for Stiff Ordinary Differential Equations," *International Journal of Mathematical and Computational Sciences*, vol. 5, no. 2, pp. 160-165, 2011. [Google Scholar] [Publisher Link]

[4] Roger Alexander, "Diagonally Implicit Runge-Kutta Methods for Stiff O.D.Es," *SIAM Journal of Numerical Analysis*, vol 14, no. 6, pp. 1006-1021, 1977. [CrossRef] [Google Scholar] [Publisher Link]

[5] P. Amodio, and F. Mazzia, "Boundary Value Methods Based on Adams-Type Method," *Applied Numerical Mathematics*, vol. 18, no. 1-3, pp. 23–25, 1995. [CrossRef] [Google Scholar] [Publisher Link]

[6] W.M.G. Bokhoven van, Implicit End-Point Quadrature Formulae, BIT 3, pp. 87-89, 1980. [Google Scholar]

[7] K. Burrage, F.H. Chipman, and P.H. Muir, "Order Results for Mono-Implicit Runge-Kutta Methods," *SIAM Journal of Numerical Analysis*, vol. 31, no. 3, pp. 867-891, 1994. [CrossRef] [Google Scholar] [Publisher Link]

[8] J.C. Butcher, "Implicit Runge-Kutta Processes," *Mathematics of Computation*, vol. 18, pp. 50-64, 1964. [Google Scholar] [Publisher Link]

[9] J.C. Butcher, P. Chartier, and Z. Jackiewicz, "Nordsieck Representation of DIMSIMs," *Numerical Algorithms*, vol. 16, pp. 209–230, 1997. [CrossRef] [Google Scholar] [Publisher Link]

[10] J.C. Butcher, and Z. Jackiewicz, "Implementation of Diagonally Implicit Multistage Integration Methods for Ordinary Differential Equations," *SIAM Journal of Numerical Analysis,* vol. 34, pp. 2119–2141, 1997. [CrossRef] [Google Scholar] [Publisher Link]
[11] J.C. Butcher, and G. Hojjati, "Second Derivative Methods with RK Stability," *Numerical Algorithms,* vol. 40, pp. 415–429, 2005. [CrossRef] [Google Scholar] [Publisher Link]
[12] J.R. Cash, "A Class of Implicit Runge-Kutta Methods for Numerical Integration of Stiff Differential Systems," *Journal of the Association for Computing Machinery,* vol. 22, no. 4, pp. 504-511, 1975. [Google Scholar] [Publisher Link]
[13] J.R. Cash, "Second Derivative Extended Backward Differentiation Formulas for the Numerical Integration of Stiff Systems," *SIAM Journal of Numerical Analysis,* vol. 18, no. 1, pp. 21–36, 1981. [CrossRef] [Google Scholar] [Publisher Link]
[14] J.R. Cash, and A. Singhal, "Mono-Implicit Runge-Kutta Formulae for Numerical Integration of Stiff Differential Systems," *IMA Journal of Numerical Analysis,* vol. 2, no. 2, pp. 211-227, 1982. [CrossRef] [Google Scholar] [Publisher Link]
[15] J.R. Cash, "Efficient Numerical Methods for the Solution of Stiff Initial-Value Problems and Differential Algebraic Equations," *Proceedings of Mathematical Physical and Engineering Science,* vol. 459, no. 2032, pp. 797-815, 2003. [CrossRef] [Google Scholar] [Publisher Link]
[16] Robert P.K. Chan, and Angela Y.J. Tsai, "On Explicit Two-Derivative Runge-Kutta Methods," *Journal of Numerical Algorithms,* vol. 53, pp. 171-194, 2010. [CrossRef] [Google Scholar] [Publisher Link]
[17] H. De Meyer et al., "On the Generation of Mono-Implicit Runge-Kutta-Nystrom Methods by Mono-Implicit Runge-Kutta Methods," *Journal of Computational and Applied Mathematics*, vol. 111, pp. 37–47, 1999. [CrossRef] [Google Scholar] [Publisher Link]
[18] Fatima Dow, "*Generalized Mono-Implicit Runge-Kutta Methods for Stiff Ordinary Differential Equations,*" Saint Marys University, Halifax, Nova Scotia, MSc Thesis, 2017. [Google Scholar] [Publisher Link]
[19] J.O. Ehigie et al., "Boundary Value Technique for Initial Value Problems with Continuous Second Derivative Multistep Method of Enright," *Computational and Applied Mathematics*, vol. 33, no. 1, pp. 81–93, 2014. [CrossRef] [Google Scholar] [Publisher Link]
[20] W.H. Enright, "Second Derivative Multistep Methods for Stiff ODEs," *SIAM Journal of Numerical Analysis,* vol. 11, no. 2, pp. 321-331, 1974. [CrossRef] [Publisher Link]
[21] Ernst Hairer, and Gerhard Wanner, *Solving Ordinary Differential Equations II: Stiff and Differential Algebraic Problems,* Second Revised Edition, Springer Verlag, Germany, 1996. [CrossRef] [Google Scholar] [Publisher Link]
[22] S.N. Jator, and R.K. Sahi, "Boundary Value Technique for Initial Value Problems Based on Adams-Type Second Derivative Methods," *International Journal of Mathematical Education in Science and Technology,* vol. 41, no. 6, pp. 819-826, 2010. [CrossRef] [Google Scholar] [Publisher Link]
[23] P.H. Muir, and M. Adams, "Mono-Implicit Runge-Kutta-Nystrom Methods with Application to Boundary Value Ordinary Differential Equations," *BIT Numerical Mathematics*, vol. 41, no. 4, pp. 776-799, 2001. [CrossRef] [Google Scholar] [Publisher Link]
[24] Paul Muir, and Brynjulf Owren, "Order Barriers and Characterizations for Continuous Mono-Implicit Runge-Kutta Schemes," *Mathematics of Computation,* vol. 61, no. 204, pp. 675-699, 1993. [Google Scholar] [Publisher Link]
[25] Zdzislaw Jackiewicz, Rosemary Anne Renaut, and Marino Zennaro, "Explicit Two-Step Runge-Kutta Methods," *Applications of Mathematics,* vol. 40, no. 6, pp. 433-456, 1995. [CrossRef] [Google Scholar] [Publisher Link]
[26] G.J. Cooper, and A. Sayfy, "Semiexplicit A-stable Runge-Kutta Methods," *Mathematics and Computation,* vol. 33, pp. 541-556, 1979. [Google Scholar] [Publisher Link]
[27] G.C. Nwachukwu, and T. Okor, "Second Derivative Generalized Backward Differentiation Formulae for Solving Stiff Problems," *IAENG International Journal of Applied Mathematics,* vol. 48, no. 1, pp. 1-15, 2018. [Google Scholar] [Publisher Link]
[28] P.O. Olatunji, and M.N.O. Ikhile, "FSAL Mono-Implicit Nordsieck General Linear Methods with Inherent Runge-Kutta Stability for DAEs," *Journal of the Korean Society for Industrial and Applied Mathematics,* vol. 25, no. 4, pp. 262–295, 2021. [CrossRef] [Google Scholar] [Publisher Link]
[29] S.E. Ogunfeyitimi, and M.N.O. Ikhile, "Generalized Second Derivative Linear Multistep Methods Based on the Methods of Enright," *International Journal of Applied and Computational Mathematics,* vol. 6, 2020. [CrossRef] [Google Scholar] [Publisher Link]
[30] S.E. Ogunfeyitimi, and M.N.O. Ikhile, "Implicit-explicit Second Derivative LMM for Stiff Ordinary Differential Equations," *Journal of the Korean Society for Industrial and Applied Mathematics,* vol. 25, no. 4, pp. 224-261, 2021. [CrossRef] [Google Scholar] [Publisher Link]
[31] R.I. Okuonghae, and I.B. Aiguobasimwin, "High-Order Hybrid Obreshkov Multistep Methods," *IAENG Journal of Applied Mathematics,* vol. 48, no. 1, pp. 73-83, 2018. [Google Scholar] [Publisher Link]
[32] R.I. Okuonghae, and M.N.O. Ikhile, "L(A)-Stable Variable-Order Implicit Second-Derivative Runge-Kutta Methods," *Numerical Analysis and Applications,* vol. 7, no. 4, pp. 314-327, 2014. [CrossRef] [Google Scholar] [Publisher Link]
[33] R.I. Okuonghae, and M.N.O. Ikhile, "Second Derivative General Linear Methods," *Numerical Algorithms,* vol. 67, no. 3, pp. 637-654, 2014. [CrossRef] [Google Scholar] [Publisher Link]
[34] R.I. Okuonghae, and M.N.O. Ikhile, "L(α)-Stable Multi-derivative GLM," *Journal of Algorithms and Computational Technology*, vol. 9, no. 4, pp. 339-376, 2015. [CrossRef] [Google Scholar] [Publisher Link]
[35] R.I. Okuonghae, "Variable Order Explicit Second Derivative General Linear Methods," Comp.Applied Maths, vol. 33, no. 1, pp. 243–255, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[36] Mukaddes Ökten Turaci, and Turgut Öziş, "On Explicit Two-Derivative Two-Step Runge-Kutta Methods," *Journal of Computational and Applied Mathematics*, vol. 37, pp. 6920-6954, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[37] T. Okor, and G.C. Nwachukwu, "High Order Extended Boundary Value Methods for Solution of Stiff System of ODEs," *Journal of Computational Applied Mathematics*, vol. 400, 2022. [CrossRef] [Google Scholar] [Publisher Link]