Available online at www.worldscientificnews.com

World Scientific News

An International Scientific Journal

WSN 195 (2024) 221-236 EISSN 2392-2192

Transforming A-stable Linear Multistep Formulas into Implicit Symmetric Runge-Kuta integration schemes for solving chaotic and stiff differential systems.

B. T. Olabode[1], A. L. Momoh[2]

Department of Mathematical Sciences, Federal University of Technology, Akure, Ondo State, Nigeria.

btolabode@futa.edu.ng ; almomoh@futa.edu.ng

ABSTRACT

There are countable number of governing equations that modelled either chaotic or stiff differential systems with high level of nonlinearity before scientists, engineers, and experts in chemistry. These type of equations are often solved by using explicit or diagonally implicit methods which has been found to be deficient. Due to this, implicit symmetric Runge-Kutta integration schemes are proposed in this work because of its A-stability advantage. Our further attention was devoted to the solution of Lorenz system in order to establish some numerical anomaly associated with the real life problems. The studies have led to the better understanding of the chaotic behavior of the system through the phase space, trajectories and attractors generated by illustrating the state of the system at time (t) with a single point in space.

Keywords: Implicit, Chaotic behavior, Symmetric, Trajectory, Attractor, Runge-Kutta

(Received 26 June 2024; Accepted 15 July 2024; Date of Publication 31 July 2024)

1. INTRODUCTION

In this paper, two numerical methods of Runge-Kutta type are derived for the numerical solution of chaotic system often described as:

$$y' = f\big(x, y(x)\big), \qquad y(a) = \mu \tag{1}$$

where the solution y(x) is assumed to be differentiable function on the closed interval $[a, b],\ b < \infty$. The differential equations often faced in many applications are not single, but a system of u simultaneous first-order equations in u dependent variables $y_1,\ y_2,\ y_3, \dots y_n$ If each of these variables satisfies given condition at the same value a of x then an initial value problem for a first-order system is obtained, which may be written as:

$$\left.\begin{array}{l} U_1' = f_1 \quad (x, U_1, U_2, \dots, U_U);\ U_1\ (a) = \mu_1, \\ U_2' = f_2 \quad (x, U_1, U_2, \dots, U_U);\ U_1\ (a) = \mu_2 \\ \quad \vdots \\ U_u' = f_u \quad (x, U_1, U_2, \dots, U_U);\ U_u\ (a) = \mu_u \end{array}\right\} \tag{2}$$

In vector notations $\boldsymbol{y} = [U_1',\ U_2', \dots, U_n',]^{\boldsymbol{T}}, \boldsymbol{f}(x, y) = [f_1,\ f_2, \dots, f_m\ ]^T$, $\mu=[\mu_1,\ \mu_2, \dots, \mu_m\ ]^T$,
First order system (2) may now be written in the form:

$$\boldsymbol{y}' = \boldsymbol{f}(x, y)\,, \qquad \boldsymbol{y}(a) = \boldsymbol{\mu} \tag{3}$$

where, $\boldsymbol{f}(x, y)$ is defined and continuous for all points $(x, \boldsymbol{y})$y) in the region D defined by $a\ \leq\ x\ \leq b,\ -\infty\ y_i < \infty,\ i\ =\ 1,\ 2,\ \cdots,\ u$ (see [13], [7]). There exists an extensive literature on numerical methods for solving (3) (see for example, [2], [18], [6], [1], [4] , [9], [13], [10] [17], [16] and [12]). Moreover, Implicit Runge-Kutta (IRK) methods with A-stability are suitable for solving stiff differential equations. Some of these methods are either fully implicit (Gauss, Radau and Lobatto methods) [21, 22], Diagonally/singly implicit [2, 19, 20] and linearly/semi implicit [9]. However, the fully implicit Runge-Kutta method is very expensive in solving large system problems. Although some IRK methods can reduce the cost of computation, their accuracy and stability are also adversely affected ([23]). On the other hand diagonally and semi implicit Runge-Kutta methods are deficient [2, 19, 20]. In particular, the methods presented in [20] are effective only when the problem to be solved is not exceedingly stiff. Furthermore, some methods that are on entire left hand complex plane which is precisely the set of eigenvalues where the exact solution of (1) is stable too are said to be A-stable [9]. A-stability of IRK methods are discussed by many authors (see [1, 2, 6, 9] to mention but a few). There are two approaches that are usually followed in the construction followed in the construction of the IRK methods. First is the generalization of Euler methods via multistep method while the other is by increasing the complexity of one-step methods as in Runge-Kutta method. Both approaches lead to methods decked with setbacks, which can be circumvented. Therefore, this paper proposes two ISRK integration schemes implemented in block like version [11] which retains the stability property of Runge-Kutta stability with the general nature of linear multistep method but overcome some of the handicaps involved in those two conventional methods.

Definition:1.1 (See [9])

A numerical method whose stability domain satisfies $S \supset \mathbb{C}^{-}\{z; |\arg(-z)| < \propto, z \neq 0\}$ is called A-stable.

Definition 1.2 (See [9])

A solution $y(x)$ is said to be stable if given any $\varepsilon > 0$ *there exist* $\delta > 0$ such that any other solution $\overline{y(x)}$ of (3) satisfies $|y(a) - \bar{y}\,(a)| \leq \delta and\; |y(x) - \bar{y}(x)| \leq \varepsilon, \forall x > a.$ The solution $y(x)$ is said to be stable asymptotically if in addition to the two conditions above$|y(x) - \bar{y}(x)| \to 0\; as\; x \to \infty.$

2. MATERIALS AND METHODS

In this section the construction of implicit Symmetric Runge-Kutta (ISRK) integration schemes are presented.

Definition 2.1. (see [9])

Let $b_i\, a_{ij},\;\; (i,j\; = \;1,\cdots, s)$ be real numbers and let c_i be defined by

$$C_i = \sum_{j=1}^{i-1} a_{ij,} \qquad The\;\; formula \qquad y_{n+1} = y_n + h\, C_{1=} \sum_{i=1}^{s} b_i\, k_i \tag{4}$$

where,

$$k_i = f\left(x_n + c_i h, \qquad y_n + h \sum_{i=1}^{s} a_{ij}\, k_j\right), i = 1, \ldots, s$$

is called s-stage Runge-Kutta method. It is convenient to represent a Runge-Kutta integration schemes eq.(4) by a partitioned table of the form:

$$\begin{array}{c|c} c_i & A_{ij} \\ \hline & b_i^T \end{array} = \begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array}$$

where "c" indicates the positions within the step of the stage values, the matrix A indicates the dependence of the stages on the derivatives found at other stages, and "b" is a quadrature weights showing how the final result depends on the derivatives computed at the various stages. Let the approximate solution of differential system of equation eq.(3) , be given as polynomial interpolant in variable x of the form:

$$y(x) = \sum_{r=0}^{(n+m)-1} \tau_r x^r \tag{5}$$

where x^r are known as the polynomial basis function and $\tau_r's$ are the real parameters to be determined. Here, m and n represent the number of collocation and interpolation points respectively. Next is to impose the interpolating function of eq.(5) that coincides with the analytical solution at the point x_n and also demand that the function eq.(5) satisfies the differential system y(x) , at the points $x_n + u_i, i = 1(1)k=$ to obtain:

$$y(x_{n+j}) = y_{n+j}, (j = 0, 1, 2, \,,\,,\, r-1), \tag{6}$$

$$y'(x_{n+j}) = f_{n+j}, (j = 0, 1, 2, \dots s-1), \tag{7}$$

$$y'(x_{n+u_i}) = f_{n+u_i}, (u_i = 0, 1, 2, \,,\,,\, s-1), \tag{8}$$

Equations (6-8) can be expressed in matrix vector form:

$$AX = B \tag{9}$$

where A is $(n \times n)$ matrix vector X and B are defined as follows:

$$A = \begin{bmatrix} 1 & x_n & x_n^2 & x_n^3 & x_n^4 & \cdots & x_n^{q-1} \\ 0 & 1 & 2x_n & 3x_n^2 & 4x_n^3 & \cdots & Mx_n^{q-2} \\ 0 & 1 & 2x_{n+u_1} & 3x_{n+u_1}^2 & 4x_{n+u_1}^3 \cdots & \cdots & Mx_{n+u_1}^{q-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & 2x_{n+k} & 3x_{n+k}^2 & 4x_{n+k}^3 \cdots & \cdots & Mx_{n+k}^{q-2} \end{bmatrix}, \tag{10}$$

$[\tau_0, \tau_1, \tau_2, \cdots, \tau_k]^T$ and $[y_n, f_n, f_{n+u_1} \cdots f_{n+k}]^T$. Matrix A is invertible. Solving eq. (9) for parameters τ_r's . Substituting τ_r's into (5) leads to continuous schemes$with\ x = x_n + dh$. The resulting continuous scheme after some algebraic manipulations yields

$$y(x + dh) = \nu_0 y_n + h\left(\sum_{j=0}^{k} \phi_j(x) f_{n+j} + \sum_{u=u_1}^{k-u_1} \phi_u(x) f_{n+u}\right), \tag{11}$$

evaluated at different values of d to obtained a block of discrete methods. Thereafter, conversion of the discrete block method to Runge-Kutta integration scheme is sought for.

2.1 SPECIFICATION OF RESULTS

Let $x = x_{n+u_i}, u_i = 0\left(\frac{1}{3}\right)1$ be the collocation points and $x = x_n$ be considered as the interpolation point. The continuous scheme is obtained as:

$$y(x+dh) = \nu_0 y_n + h\left(\phi_0 f_n + \phi_2 f_{n+1} + \phi_{\frac{1}{3}} f_{n+\frac{1}{3}} + \phi_{\frac{2}{3}} f_{n+\frac{2}{3}}\right) \quad (12)$$

whose coefficient are difficult expression that be obtained by adopting any know mathematical software. Evaluating (12), at $d = \frac{1}{3}\left(\frac{1}{3}\right)1$ gives the discrete methods that can be represented in Butcher tableau as shown in Table 1 where

$$A_{ij} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{1}{8} & \frac{19}{72} & \frac{-5}{72} & \frac{1}{72} \\ \frac{1}{9} & \frac{4}{9} & \frac{1}{9} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix}, \quad b_i = \begin{bmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix}^T \text{ and } c_i = \begin{bmatrix} 0 & \frac{1}{3} & \frac{2}{3} & 1 \end{bmatrix}^T.$$

This can be written inform of (4), as given below

$$\left.\begin{aligned} y_{n+1} &= y_n + \frac{h}{8}\left(k_0 + 3k_{\frac{1}{3}} + 3k_{\frac{2}{3}} + k_1\right) \\ k_0 &= f(x_n, y_n) \\ k_{\frac{1}{3}} &= f\left(x_n + \frac{h}{3},\ y_n + \frac{h}{72}\left(9k_1 + 19k_{\frac{1}{3}} - 5k_{\frac{2}{3}} + k_1\right)\right) \\ k_{\frac{2}{3}} &= f\left(x_n + \frac{2h}{3},\ y_n + \frac{h}{9}\left(k_0 + 4k_2 + k_{\frac{2}{3}}\right)\right) \\ k_1 &= f\left(x_n + h, y_n + \frac{h}{8}\left(k_0 + 3k_{\frac{1}{3}} + 3k_{\frac{2}{3}} + k_1\right)\right) \end{aligned}\right\}. \quad (13)$$

In a similar manner, let $x = x_{n+u_i}, u_i = 0\left(\frac{1}{4}\right)1$ be the collocation points and $x = x_n$ be the interpolation point. The discrete methods are presented in Butcher tableau described in Table 1 where

$$A_{ij} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{251}{2880} & \frac{323}{1440} & -\frac{11}{120} & \frac{53}{1440} & -\frac{19}{2880} \\ \frac{29}{360} & \frac{31}{90} & \frac{1}{15} & \frac{1}{90} & -\frac{1}{360} \\ \frac{27}{320} & \frac{51}{160} & \frac{9}{40} & \frac{21}{160} & -\frac{3}{320} \\ \frac{7}{90} & \frac{16}{45} & \frac{2}{15} & \frac{16}{45} & \frac{7}{90} \end{bmatrix}, \quad b_i = \begin{bmatrix} \frac{7}{90} & \frac{16}{45} & \frac{2}{15} & \frac{16}{45} & \frac{7}{90} \end{bmatrix}, \text{ and } c_i = \begin{bmatrix} 0 & \frac{1}{4} & \frac{1}{2} & \frac{3}{4} & 1 \end{bmatrix}.$$

The method represented in Table 1 in Runge-Kutta form is given as

$$\left.\begin{aligned} y_{n+1} &= y_n + \frac{h}{90}\left(7\,k_0 + 32\,k_{\frac{1}{4}} + 12\,k_{\frac{1}{2}} + 32\,k_{\frac{3}{4}} + 7\,k_1\right) \\ k_0 &= f\left(x_n, y_n\right) \\ k_{\frac{1}{4}} &= f\left(x_n + \tfrac{1}{4}\,h, y_n + \frac{h}{2880}\left(251\,k_0 + 646\,k_{\frac{1}{4}} - 264\,k_{\frac{1}{2}} + 106\,k_{\frac{3}{4}} - 19\,k_1\right)\right) \\ k_{\frac{1}{2}} &= f\left(x_n + \tfrac{1}{2}\,h, y_n + \frac{h}{360}\left(29\,k_0 + 124\,k_{\frac{1}{4}} + 24\,k_{\frac{1}{2}} + 4\,k_{\frac{3}{4}} - k_1\right)\right) \\ k_{\frac{3}{4}} &= f\left(x_n + \tfrac{3}{4}\,h, y_n + \frac{3\,h}{320}\left(9\,k_0 + 34\,k_{\frac{1}{4}} + 24\,k_{\frac{1}{2}} + 14\,k_{\frac{3}{4}} - k_1\right)\right) \\ k_1 &= f\left(x_n + h, y_n + \frac{h}{90}\left(7\,k_0 + 32\,k_{\frac{1}{4}} + 12\,k_{\frac{1}{2}} + 32\,k_{\frac{3}{4}} + 7\,k_1\right)\right) \end{aligned}\right\}. \qquad (14)$$

2.2 Stability and order of accuracy of the proposed methods

The stability of the proposed methods are obtained by applying them to the test equation

$$y' = z\,y, \quad z \in C \quad and \qquad R, z < 0$$

which resulted in

$$y_{n+1} = R(z\,h)y_n.$$

By replacing $z\,h$ with H, the stability function $R(H) : C \to C$ are obtained as

$$R(H) = I + H\,bT(I - HA)^{-1}\,e \qquad (15)$$

where e is the vector of s components given by $e = (1,\cdots,1)^T$. This way, according to the findings of [9], [17], [16], [25] and many others, the methods are called A-stable if the left-half complex plane are

included in the stability domain, that is

$$C- = \{H;\ Re\,H < 0\} \subset S = \{H; |R \leq 1|\qquad (16)$$

The rational functions

$$R_i(H) = \frac{N_i(H)}{D_i(H)}, \qquad i = 1,2 \qquad (17)$$

give the stability function of the proposed methods. This are obtained respectively for (13) and (14)

as

$$R_1(H) = \frac{-H^3 + 11H^2 + 54H + 108}{H^3 - 11H^2 + 54H - 108}, \qquad (18)$$

$$R_2(H) = \frac{3H^4 + 50H^3 + 420H^2 + 1920H + 3840}{3H^4 - 50H^3 + 420H^2 - 1920H + 3840}, \qquad (19)$$

The rational functions (18) and (19) satisfy (16) and thus are A-stable since their modulus are less than one on the left-half complex planes as shown in **Figure 1** below.

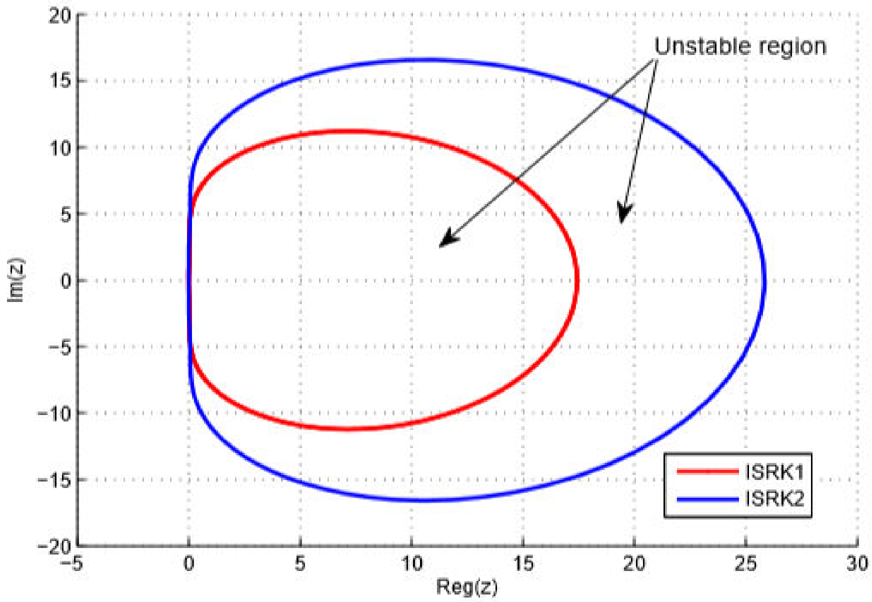


Figure 1: Region of Absolute Stability of ISRK1 and ISRK2

We verified the order condition of the Butcher theory on the proposed formulas with the aid of Maple system version 17. It was found that the formulas are of algebraic orders 4 and 6. The algebraic orders of the last members of the stages which are also the last equations that approximate the final values are obtained as respectively.

$$y(x_n+h)-\left(y(x_n)+h\sum_{i=0}^{3}\phi_i\, y'(x_n+s_ih)\right)=-\frac{h^5y^{(5)}(x_n)}{6480}+\mathcal{O}(h^6), \tag{20}$$

and

$$y(x_n+h)-\left(y(x_n)+h\sum_{i=0}^{4}\phi_i\, y'(x_n+s_ih)\right)=-\frac{h^7y^{(7)}(x_n)}{1935360}+\mathcal{O}(h^8) \tag{21}$$

3. Numerical Experiment and Results

In this section, six examples are considered to illustrate the efficiency and accuracy advantages of the proposed methods. The absolute error (Err) on the partition Ω_h is $Err = |y(x) - y_n(x)|$ where, the step-length $h = (b-a)/N$ and N is the number of iterations. Furthermore, we shall refer to the two methods (13) and (14) as Implicit Symmetric Runge-Kutta schemes ISRK1 and ISRK2 respectively.

Example 1: The nonlinear system of stiff first order ordinary differential equation

$$y_1' = -1002y_1 + 1000y_2, y_1(0) = 1,$$

$$y_2' = -y_1 - y_2(1 + y_2), \ y_2(0) = 1,$$

is considered where $x \in [0,10]$. The theoretical solution of the system is given as

$$y_1(x) = e - 2x,$$

$$y_2(x) = e - x,$$

The result of the numerical simulation is displaced in **Figure 2.**

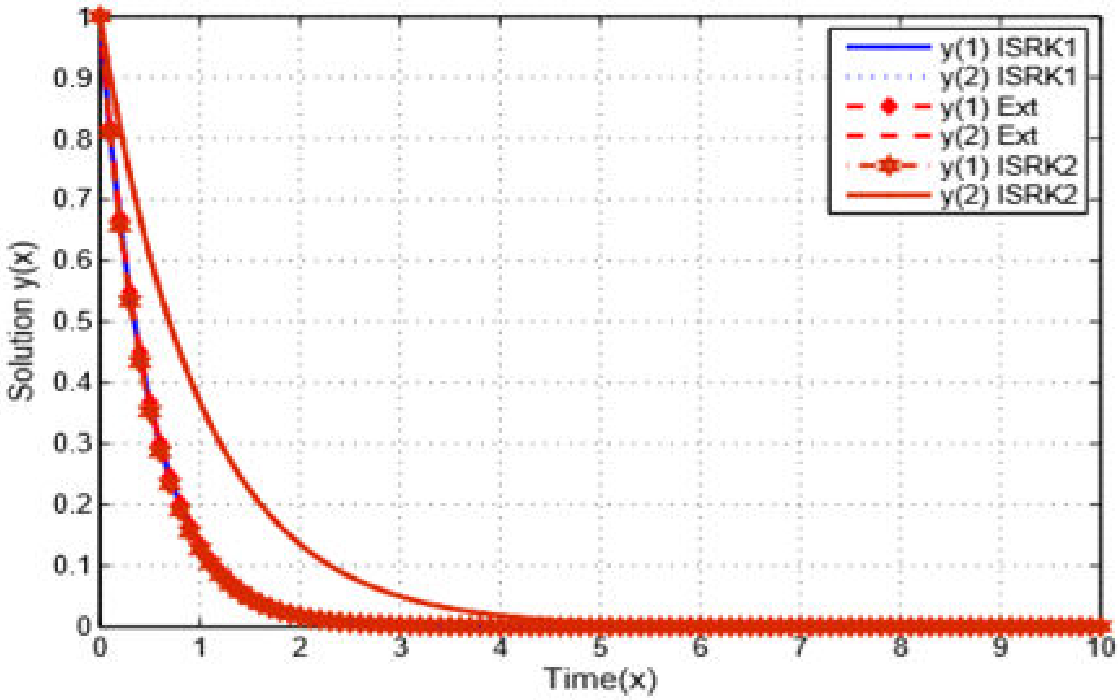


Figure 2: Solution curve of example 2 through the ISRK1, ISRK2. and ode45.

Example 2: The classical problem that models the kinetics of a chemical reaction proposed by Robertsion in [9] given as

$$y_1' = -0.04y_1 + 104y_2 y_3, \qquad y_1(0) = 1$$
$$y_2' = 0.04y_1 - 104y_2y - 3 \times 10^7 - y_2, \qquad y_2(0) = 0$$
$$y_3' = 3 \times 10^7 y_2^2, \qquad y_3(0) = 0$$

The problem is solved in the range [0,10] using the two methods. The solution curves is presented in **Figure 3.**

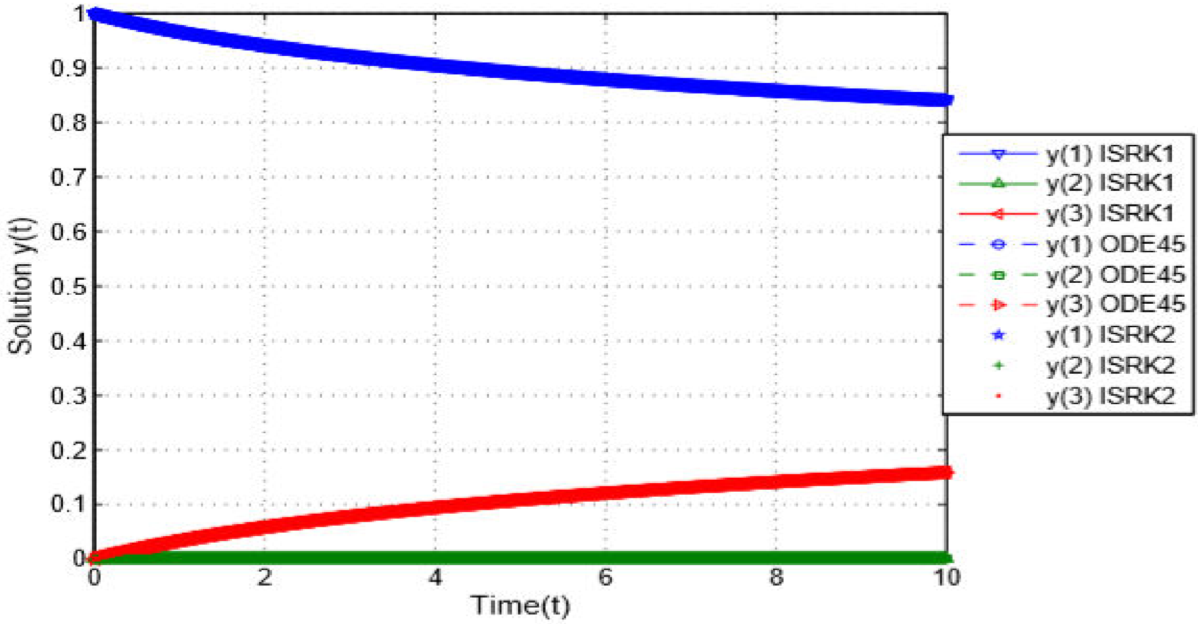


Figure 3: Solution curve of example 2 through the ISRK1, ISRK2. and ode45.

Example 3: The following problem which is commonly refers to as troublesome due to it eigenvalues lying close to imaginary axis was also considered

$$\begin{aligned}
y_1' &= -10y_1 + 100y_2, & y_1(0) &= 1 \\
y_2' &= -100y_1 - 108y_2, & y_1(0) &= 1 \\
y_3' &= -4y_3, & y_1(0) &= 1 \\
y_4' &= -y_4, & y_1(0) &= 1 \\
y_5' &= -0.5y_5, & y_1(0) &= 1 \\
y_6' &= -0.1y_6, & y_1(0) &= 1
\end{aligned}$$

The solution curves are as shown in **Figure 4.**

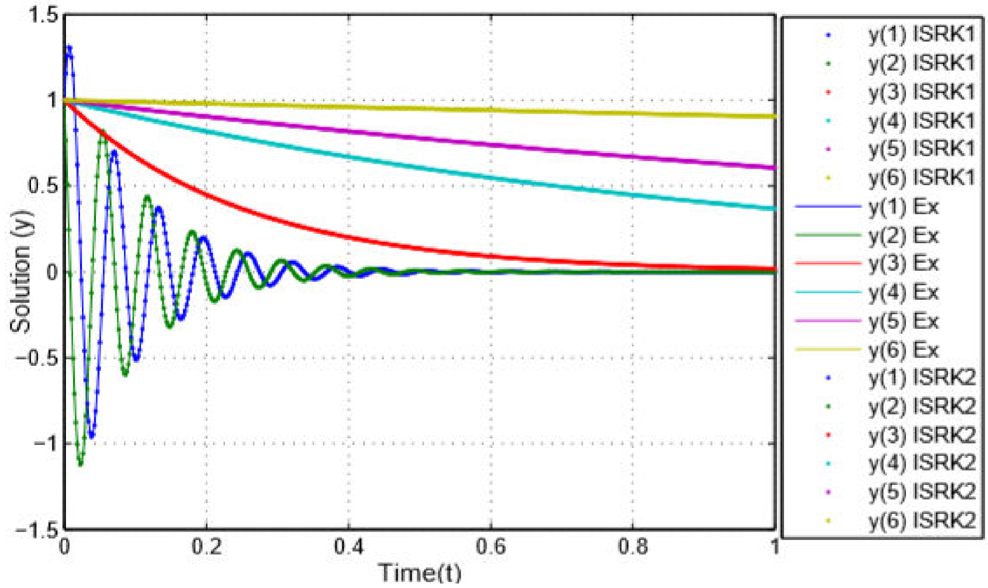


Figure 4: Solution curve of example 3 using the ISRK1 and ISRK2 methods with exact solution.

Example 4: The Jacobi ellipic function problem for the equation of rigid body without external forces

$$y'_1 = y_2 y_3, \qquad y_1(x) = 0,$$

$$y'_2 = -y_1 y_3, \qquad y_2(x) = 1,$$

$$y'_3 = -5.1 y_1 y_3, \qquad y_3(x) = 1.$$

The solutions are illustrated as **Figure 5** as compared with MATLAB ode45 solver.

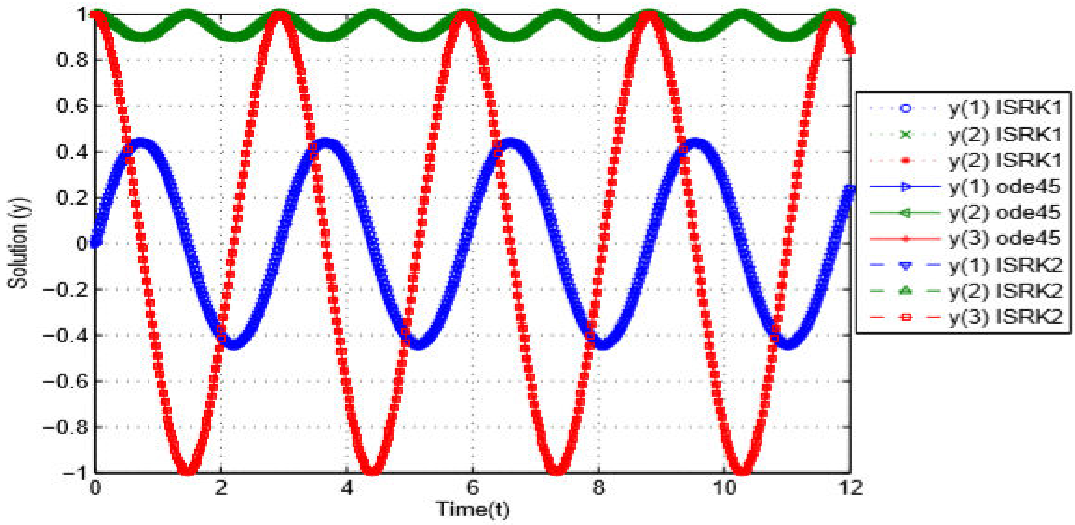


Figure 5: Solution curve of example 4 using the ISRK1 and ISRK2 methods with exact solution

Example 5: This is a nonlinear equation solved by [15] within the interval [0,1].

$$y_1 = -(2 + \epsilon^{-1})y_1 + \epsilon^{-1}y_2^2, \qquad y_1(x) = 1, \qquad y_1(x) = e^{-2x}$$
$$y_2' = y_1 - y_2 - y_2^2, \qquad y_2(x) = 1, \qquad y_2(x) = e^{x}$$
$$x \in [0,1], \quad \epsilon = 10^{-2}$$

Note that as $\epsilon \to 0$, the problem become very stiff.

Table 1: Comparison of accuracy of ISRK1 and SGLM [15] on example 5 for fixed step size $h = 10^{-4}$ in the interval of $x \in [0, 40]$.

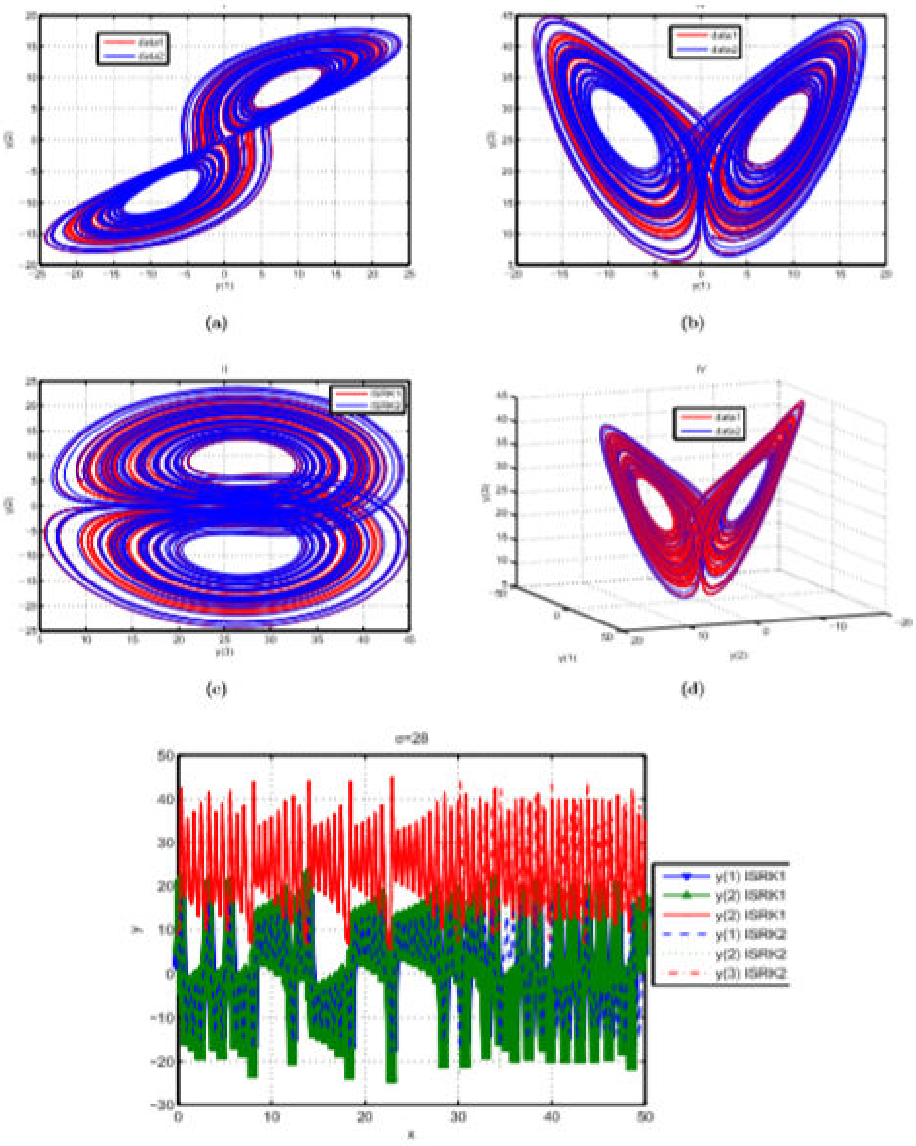
x	y_i	ISRK1	SGLM [15]	Exact
	y_1	4.494188374243970e − 01	4.49418835052771e − 01	4.494188388972235e − 01
0.4	y_2	6.703870814136931e − 01	6.703870807281706e − 01	6.703870813919548e − 01
	y_1	3.355297260382144e − 04	3.355297183504160e − 04	3.355297271377920e − 04
4	y_2	1.831747054479764e − 02	1.831747035941888e − 02	1.831747054420430e − 02
	y_1	1.805212388306414e − 35	1.805212021980264e − 35	1.805212394222405e − 35
40	y_2	4.248779112097137e − 18	4.248778686677310e − 18	4.248779111959581e − 18

Continuation of table 1 for example 5

x	Error in ISRK1	Error in SGLM [15]
0.4	1.4728e − 09	3.5919e − 09
4	5.9340e − 13	1.8478e − 10
40	1.3756e − 28	4.2528e − 25

Table 2 : The global error and the convergence order of ISRK1 and SGLM [15] on example 5, in $x \in [0, 1]$.

h	Error in ISRK1	$R(h)$	SGLM[15]	$R(h)$
1e − 03	3.0448e − 07	−	1.538e − 07	−
5e − 05	7.5141e − 08	4	3.794e − 08	4
2.5e − 04	1.8663e − 08	4	9.417e − 09	4
1.25e − 04	4.6506e − 09	4	2.345e − 09	4



(e) Time dependency of y_1, y_2 and y_3 using ISRK1 and ISRK2.

Figure 6: 6e is the solution **of Lorenz equation for** $\rho = 10, \beta = \frac{8}{3}, \sigma = 28$ respectively with initial conditions $y_1 = 1.18947681558376, y_2 = 4.983140519821429, y_3 = 9.596939585160811$ using ISRK1 and ISRK2 with time step $h = 0.005$ for 10000 steps. Figures (6a-6d) are the phase spaces of y_1, y_2 and y_3 corresponding to **Figure 5 e**

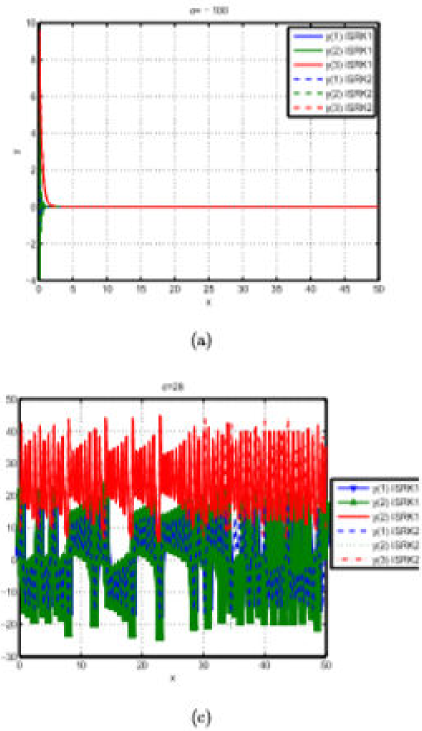


Figure 7: (7a-7d) are the corresponding attractors to the solution curves generated for $\rho = -1000, 15, 28$ and 32 respectively with initial condition $y_1 = 1.18947681558376, y_2 = 4.983140519821429, y_3 = 9.596939585160811$ using ISRK1 and ISRK2 with time step $h = 0.005$ for 1000 steps.

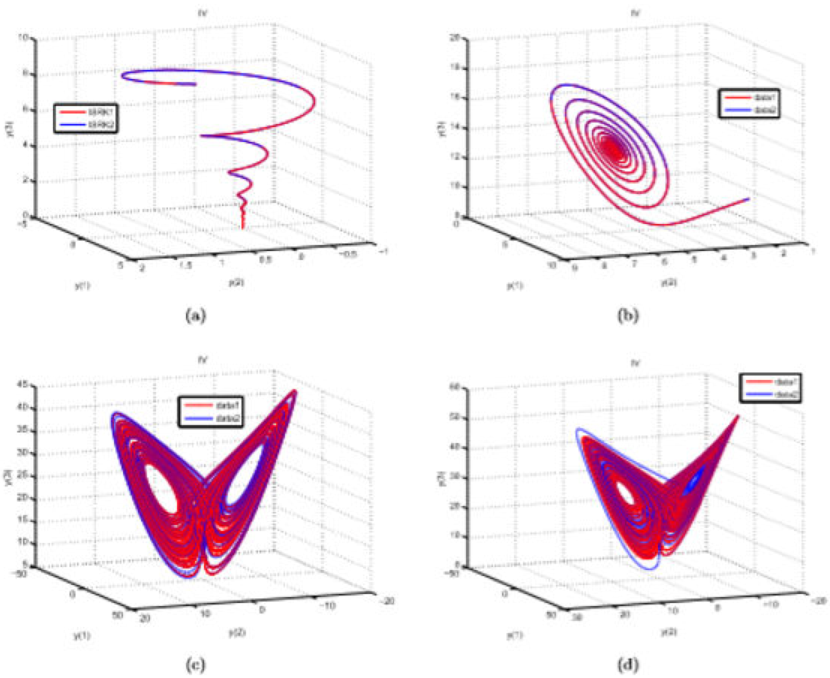


Figure 8: (8a-8d) are the corresponding attractors to the solution curves in figures (7a-7b) respectively

3.1 Application of ISRK1 and ISRK2 methods to chaotic system

The ISRK1 and ISRK2 are applied to solve one of the well-known chaotic systems in literature that is Lorenz equations to further demonstrate its usability. The equation is given as

$$\begin{aligned} y_1' &= \sigma(y_2 - y_1) \\ y_2' &= \rho y_1 - y_2 - y_1 y_3 \\ y_3' &= y_1 y_2 - \beta y_3 \end{aligned}$$

Here y_1, y_2 and y_3 do not refer to coordinates in space. In fact, y_1 represents the convective overturning on the plane, while y2 and y3 are the horizontal and vertical temperature variation respectively. The parameters of this model are σ which represents the Prandtl number, or the ratio between the fluid viscosity to its thermal conductivity, ρ, which represents the difference in temperature between thee top and bottom of the atmosphere plane, and β, which is the ratio of the width to the height of the plane ([14], [8]). Lorenz found the values of $\sigma = 10$ and $\beta = \frac{8}{3}$, and initial conditions of $(y_1, y_2, y_3) = (0, 1, 0)$ to be the best representation of the earth's atmosphere (Lorenz 136-137). For the purpose of this work, we considered the values for parameters ρ, β and then studied the effects of change of σ.

3.1.1 The Lorenz Attractor

The Lorenz system is simulated for the initial positions $(y_1, y_2, y_3) = (1.18947681558376,$

$4.983140519821429, 9.596939585160811)$ through the methods (13) and (14), by setting the parameter $\rho = 28$. The results are shown in Fig.6. It is observed that the attractors in figure 6 resemble a butterfly, and thus, the Lorenz Attractor and chaos in general are nicknamed, "The Butterfly Effect". Time dependency of y_1, y_2 and y_3 shown that after an initial transient, the y_1 and y_2 components oscillate irregularly alternating around a positive and negative values whereas the y_3 component oscillates always around positive values.

3.1.2 Effects of variation of the parameter σ

The effects of variation of the parameter σ on the shape of the trajectory. First, we set σ to a negative value that is −100 are illustrated and it is observed that the trajectory crashed more or less straight into the origin no matter the initial positions. Using $\sigma = 15 < 24$ both time dependency and then phase space plot in figures 8(b) and 9(b) showed overtime and over much iteration; no chaos is present. Figures 8(c)& (d), 9(c)& (d) of the Lorenz system demonstrate chaos.

4. Conclusions

In this study numerical methods for the solution of system of first order time-dependent ordinary differential equations that exhibit chaos and quadratic nonlinearities have been proposed. The stability plots of the methods show A-stability which is a desired property for solving stiff differential systems. To further demonstrate in this work the efficiency and usability of the ISRK methods, six numerical examples are considered with inclusion of Lorenz system. Our future work shall be focused on implementing ISRK type methods in a variable step size mode.

References

[1] Bellen, A. and Zennaro, M. (1988). Stability properties of interpolants for Runge-Kutta methods. SIAM Journal on Numerical Analysis, 25(2): 411-432.

[2] Burrage, K., & Butcher, J. C. (1979). Stability Criteria for Implicit Runge–Kutta Methods. SIAM Journal on Numerical Analysis, 16(1), 46–57. doi:10.1137/0716004

[3] Butcher, J. and Chartier, P. (1997). A generalization of singly-implicit runge-kutta methods. Applied Numerical Mathematics, 24(2-3):343-350.

[4] Butcher, J. C. and Cash, J. R. (1990). Towards efficient runge-kutta methods for stiff systems. SIAM Journal on Numerical Analysis, 27(3):753-761

[5] Chollom, J. and Jackiewicz, Z. (2003). Construction of two-step Runge-Kutta methods with large regions of absolute stability. Journal of Computational and Applied Mathematics, 157(1):125-137.

[6] Cooper, G. J. (1987). Stability of Runge-Kutta methods for trajectory problems. IMA Journal of Numerical Analysis, 7(1):1-13.

[7] Fatunla, S. O. (1988). PREFACE. Numerical Methods for Initial Value Problems in Ordinary Differential Equations, ix–xi. doi:10.1016/b978-0-12-249930-2.50004-7

[8] Feki, M. (2006). SYNCHRONIZATION OF GENERALIZED LORENZ SYSTEM USING ADAPTIVE CONTROLLER. IFAC Proceedings Volumes, 39(8):2-6

[9] Hairer, E. and Wanner, G. (1996). Solving Ordinary Differential Equations II. Springer Berlin Heidelberg

[10] Hojjati, G., Ardabili, M. R., and Hosseini, S. (2006). New second derivative multistep methods for stiff systems. Applied Mathematical Modelling, 30(5):466-476.

[11] Jator, S. N. (2012). A continuous two-step method of order 8 with a block extension for. Applied Mathematics and Computation, 219(3), 781–791. doi:10.1016/j.amc.2012.06.027

[12] Jator, S. N. and Oladejo, H. B. (2017). Block Nystrom method for singular differential equations of the lane-emden type and problems with highly oscillatory solutions. International Journal of Applied and Computational Mathematics, 3(S1):1385-1402.

[13] Lambert J.D., Computational Methods in Ordinary Differential Equations, John Wiley, New York, 1973

[14] Munmuangsaen, B. and Srisuchinwong, B. (2018). A hidden chaotic attractor in the classical Lorenz system. Chaos, Solitons & Fractals, 107:61-66.

[15] Okuonghae, R. I. and Ikhile, M. N. O. (2013). Second derivative general linear methods. Numerical Algorithms, 67(3):637-654.

[16] Ramos, H. and Singh, G. (2017). A note on variable step-size formulation of a simpson'stype second derivative block method for solving stiff systems. Applied Mathematics Letters, 64:101107.

[17] Yakubu, D. and Kwami, A. (2015). Implicit two-derivative Runge-Kutta collocation methods for systems of initial value problems. Journal of the Nigerian Mathematical Society, 34(2):128142.

[18] Zennaro, M. (1986). Natural continuous extensions of Runge-Kutta methods. Mathematics of Computation, 46(173):119-119.

[19] Alexander, R. (1977). Diagonally Implicit Runge–Kutta Methods for Stiff O.D.E.'s. SIAM Journal on Numerical Analysis, 14(6), 1006–1021. doi:10.1137/0714068

[20] Kennedy, C. A., & Carpenter, M. H. (2019). Diagonally implicit Runge–Kutta methods for stiff ODEs. Applied Numerical Mathematics, 146, 221–244. doi:10.1016/j.apnum.2019.07.008

[21] Butcher, J. C. (1964). Implicit Runge-Kutta processes. Mathematics of Computation, 18(85), 50–50. doi:10.1090/s0025-5718-1964-0159424-9

[22] Cong, N. H., & Thuy, N. T. (2011). Two-step-by-two-step PIRK-type PC methods based on Gauss–Legendre collocation points. Journal of Computational and Applied Mathematics, 236(2), 225–233. doi:10.1016/j.cam.2011.06.020

[23] Liu, M. Y., Zhang, L., & Zhang, C. F. (2019). Study on Banded Implicit Runge–Kutta Methods for Solving Stiff Differential Equations. Mathematical Problems in Engineering, 2019, 1–8. doi:10.1155/2019/4850872