Application of second derivative Runge-Kutta collocation methods to stiff systems of initial value problems

Samaila MARKUS[1] and Dauda Gulibur YAKUBU[2*]

[1]*Department of Mathematics and Statistics, University of Maiduguri, PMB 1069, Maiduguri, Borno State, Nigeria*
[2]*Department of Mathematical Sciences, Abubakar Tafawa Balewa University, PMB 0248, Bauchi, Bauchi State, Nigeria*
E-mails: daudagyakubu@gmail.com &smvalleh@gmail.com

*Corresponding author: Phone: 2348134168950 & 2348052168433

Abstract

In this paper we consider the construction of implicit second derivative Runge-Kutta collocation methods designed for the continuous numerical solution of stiff systems of first order initial value problems in ordinary differential equations. These methods are obtained based on the multistep collocation technique, which are shown to be convergent, with improved regions of absolute stability. Although the implementation of the second derivative Runge-Kutta collocation methods remains iterative due to the implicit nature of the methods, the advantage gained makes them suitable for solving stiff systems with eigenvalues of large modulus lying close to the imaginary axis. Some absolute stability characteristics and order of accuracy of the methods are studied. Finally, we show two possible ways of implementing the methods and compare them on some numerical examples found in the literature to demonstrate the high order of accuracy and reliability of the methods.

Keywords

Block hybrid schema; Continuous schema; Multistep collocation formula; Second-derivative Runge-Kutta method; Stiff system

Introduction

In the past few decades many authors proposed different numerical integration methods to obtain more accurate approximate solution to ordinary differential equations (ODEs), especially stiff systems of initial value problems in ordinary differential equations. However, extensive numerical experiments have shown that implicit integrators have the numerical superiority over the explicit methods when applied to solve stiff systems of ODEs [1-4]. In particular, implicit Runge-Kutta methods play an important role in the numerical solution of stiff systems of ODEs, because they have good stability properties and high order of convergence. In this paper we derive implicit second-derivative Runge-Kutta (SDRK) collocation methods designed for the continuous numerical solution of stiff systems of initial value problems in ODEs of the form

$$\frac{dy}{dx} = f(x, y(x)), \quad y(x_0) = y_0, \quad x \in [x_0, T] \tag{1.1}$$

Here the unknown function y is a mapping $y:[x_0,T] \to R^d$ and the right-hand side function f is $f:[x_0,T]_x R^d \to R^d$ which is assumed to be sufficiently smooth, $y_0 \to R^d$ is the given initial value and dy/dx is the differential operator. Let h>0 be a constant step size and we define the grid by $x_n = x_0 + nh$, n=0,1,2,..,N where $Nh = T - x_0$ and a set of uniformly spaced points on the integration interval is defined by $x_0 < x_1 < x_2 < \ldots < x_{n+1} = T$ The main reason for considering the second derivative terms is to derive a set of methods which are suitable for the continuous numerical solution of stiff differential systems with Jacobians having large eigenvalues lying close to the imaginary axis. However, for some important classes of problems it is necessary, for the sake of efficiency, to allow second-derivative integration methods to be used (see [5-8]) and in this case existing numerical methods tend to be much less satisfactory. Further, we will examine in detail the problem of implementing the second-derivative Runge-Kutta integration methods with fixed time steps. We show that, even though enormous gains in efficiency can be achieved if the methods are implemented in an appropriate way, there are still some important practical problems to be overcome, for example, the calculation of the second derivative terms in the methods which costs little higher than the first derivative terms [9].

There are several eminent authors who derived methods that can handle non-stiff, stiff, periodic and oscillatory problems handy [2-4,9], but little seems to have been done in deriving implicit second-derivative Runge-Kutta collocation methods with minimal function

evaluations, uniformly accurate orders and maximal gain in efficiency. Hence, these motivate us to consider the construction of the implicit second derivative Runge-Kutta collocation methods in this paper.

Definition 1.1: A numerical method is said to be A(α)-stable $\alpha \in (0, \pi/2)$ if its region of absolute stability contains the infinite wedge W_α where

$$W_\alpha = \{\lambda h : -\alpha < |\pi - \arg(\lambda h)| < \alpha\} \qquad \text{see [10].}$$

Theorem 1.1: If f satisfies Lipschitz condition with constant L then the initial value problem y'(x)=f(x,y(x)), y(x$_0$)=y$_0$ possesses a unique solution on the interval [x$_0$, T] see [11].

Definition 1.2: A solution y(x) of (1.1) is said to be stable if given any $\in > 0$ there is $\delta > 0$ such that any other solution y(x) of (1.1) which satisfies

$$|y(a) - \hat{y}(a)| \leq \delta \tag{1.2a}$$

also satisfies

$$|y(x) - \hat{y}(x)| \leq \in \tag{1.2b}$$

for all x>a.

The solution y(x) is asymptotically stable if in addition to (1.2b) as x→∞.

Material and method

The implicit second derivative Runge-Kutta collocation methods

In this section our objective is to describe the construction of the implicit second-derivative Runge-Kutta collocation methods based on the multistep collocation technique. In this regard we seek an approximate solution to the exact solution of (1.1) by the interpolant of the form

$$y(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \cdots + \alpha_{p-1} x^{p-1} = \sum_{i=0}^{p-1} \alpha_i x^i \tag{1.3}$$

which is twice continuously differentiable. We set the sum r+s+t to be equal to p so as to be able to determine $\{\alpha_i\}$ in (1.3). In this formulation r denotes the number of interpolation points used and s>0, t>0 are distinct collocation points. Interpolating (1.3) at the points $\{x_{n+j}\}$, and collocating y'(x) and y''(x) at the points $\{c_{n+j}\}$ we obtain the following system of equations

$$y(x_{n+j}) = y_{n+j}, \qquad (j = 0,1,2,\ldots,r-1) \tag{1.4}$$

$$y'(c_{n+j}) = f_{n+j}, \qquad (j = 0,1,2,\ldots,s-1) \tag{1.5}$$

$$y''(c_{n+j}) = g_{n+j}, \qquad (j = 0,1,2,\ldots,t-1). \tag{1.6}$$

In fact equations (1.4) to (1.6) can be expressed in the matrix-vector form as

$$D\alpha = y \tag{1.7}$$

where the p-square matrix D, p- vectors α and y are defined as follows:

$$D = \begin{pmatrix} 1 & x_n & x_n^2 & x_n^3 & x_n^4 & \cdots & x_n^{p-1} \\ 1 & x_{n+1} & x_{n+1}^2 & x_{n+1}^3 & x_{n+1}^4 & \cdots & x_{n+1}^{p-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n+r-1} & x_{n+r-1}^2 & x_{n+r-1}^3 & x_{n+r-1}^4 & \cdots & x_{n+r-1}^{p-1} \\ 0 & 1 & 2c_n & 3c_n^2 & 4c_n^3 & \cdots D'c_n^{p-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 2c_{n+s-1} & 3c_{n+s-1}^2 & 4c_{n+s-1}^3 & \cdots D'c_{n+s-1}^{p-2} \\ 0 & 0 & 2 & 6c_n & 12c_n^2 & \cdots D''c_n^{p-3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 2 & 6c_{n+t-1} & 12c_{n+t-1}^2 & \cdots D''c_{n+t-1}^{p-3} \end{pmatrix} \tag{1.8}$$

$$\alpha = \left(\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_{p-1}\right)^T, \quad y = \left(y_n, \ldots, y_{n+r-1},\ y'_n, \ldots, y'_{n+s-1}, y''_n, \ldots, y''_{n+t-1}\right)^T$$

where D'=(p-1) and D''=(p-1)(p-2) in (1.8) represent the first and second derivatives respectively and correspond to the differentiation with respect to x. Similar to the Vandermonde matrix, the matrix *D* in (1.7) is non-singular. Consequently, equation (1.7) has the unique solution given by

$$\alpha = C_y \text{ where } C = D\text{-}1 \tag{1.9}$$

Rearranging equations (1.7) to (1.9) we obtain the multistep collocation formula of the type in [12] which was a generalization of [13] and here we extend to second derivative as follows.

$$y(x) = \sum_{j=0}^{r-1} \alpha_j(x) y_{n+j} + h \sum_{j=0}^{s-1} \beta_j(x) f_{n+j} + h^2 \sum_{j=0}^{t-1} \omega_j(x) g_{n+j} \tag{1.10}$$

where:

$$y_{n+j} \approx y(x_n + jh)$$

$$f_{n+j} \equiv f(x_n + jh, y(x_n + jh))$$

$$g_{n+j} \equiv \frac{df(x, y(x))}{dx}\bigg|_{\substack{x = x_{n+j} \\ y = y_{n+j}}}$$

Here $\alpha_j(x)$, $\beta_j(x)$, $\omega_j(x)$ are continuous coefficients of the method which are to be determined. They are assumed polynomials of the form

$$\alpha_j(x) = \sum_{i=0}^{p-1} \alpha_{j,i+1} x^i$$

$$h\beta_j(x) = \sum_{i=0}^{p-1} h\beta_{j,i+1} x^i \qquad (1.11)$$

$$h^2\omega_j(x) = \sum_{i=0}^{p-1} h^2\omega_{j,i+1} x^i$$

The numerical constant coefficients $\alpha_{j,i+1}$, $\beta_{j,i+1}$, $\omega_{j,i+1}$ in (1.11) are to be determined.

In fact, the above coefficients can be obtained from the components of the matrix D^{-1}. That is, if the identity (1.12) holds. Actual evaluations of the matrices C and D are carried out with a computer algebra system, for example, Maple, to determine the constant coefficients $\alpha_{j,i+1}$, $\beta_{j,i+1}$, and $\omega_{j,i+1}$ in (1.11).

$$C = \begin{pmatrix} \alpha_0^{(0)}\alpha_0^{(1)}\cdots\alpha_0^{(r-1)} & h\beta_0^{(0)}\cdots h\beta_0^{(s-1)} & h^2\omega_0^{(0)}\cdots h^2\omega_0^{(t-1)} \\ \alpha_1^{(0)}\alpha_1^{(1)}\cdots\alpha_1^{(r-1)} & h\beta_1^{(0)}\cdots h\beta_1^{(s-1)} & h^2\omega_1^{(0)}\cdots h^2\omega_1^{(t-1)} \\ \alpha_2^{(0)}\alpha_2^{(1)}\cdots\alpha_2^{(r-1)} & h\beta_2^{(0)}\cdots h\beta_2^{(s-1)} & h^2\omega_2^{(0)}\cdots h^2\omega_2^{(t-1)} \\ \alpha_3^{(0)}\alpha_3^{(1)}\cdots\alpha_3^{(r-1)} & h\beta_3^{(0)}\cdots h\beta_3^{(s-1)} & h^2\omega_3^{(0)}\cdots h^2\omega_3^{(t-1)} \\ \vdots \quad \vdots \quad \ddots \quad \vdots & \vdots \quad \ddots \quad \vdots & \vdots \quad \ddots \quad \vdots \\ \alpha_{p-1}^{(0)}\alpha_{p-1}^{(1)}\cdots\alpha_{p-1}^{(r-1)} & h\beta_{p-1}^{(0)}\cdots h\beta_{p-1}^{(s-1)} & h^2\omega_{p-1}^{(0)}\cdots h^2\omega_{p-1}^{(t-1)} \end{pmatrix} = D^{-1} \qquad (1.12)$$

To obtain the continuous scheme, we insert (1.11) into (1.10) to have

$$y(x) = \sum_{j=0}^{r-1}\sum_{i=0}^{p-1} \alpha_{j,i+1} x^i y_{n+j} + h\sum_{j=0}^{s-1}\sum_{i=0}^{p-1} \beta_{j,i+1} x^i f_{n+j} + h^2\sum_{j=0}^{t-1}\sum_{i=0}^{p-1} \omega_{j,i+1} x^i g_{n+j}$$

$$= \sum_{i=0}^{p-1}\left\{\sum_{j=0}^{r-1} \alpha_{j,i+1} y_{n+j} + h\sum_{j=0}^{s-1} \beta_{j,i+1} f_{n+j} + h^2\sum_{j=0}^{t-1} \omega_{j,i+1} g_{n+j}\right\} x^i .$$

$$= \sum_{i=0}^{p-1} \phi_i x^i \qquad (1.13)$$

where:

$$\phi_i = \left\{\sum_{j=0}^{r-1} \alpha_{j,i+1} y_{n+j} + h\sum_{j=0}^{s-1} \beta_{j,i+1} f_{n+j} + h^2\sum_{j=0}^{t-1} \omega_{j,i+1} g_{n+j}\right\}$$

From (1.13) we have

$$y(x) = \left\{ \sum_{j=0}^{r-1} \alpha_{j,1} y_{n+j} + \sum_{j=0}^{s-1} h\beta_{j,1} f_{n+j} + \sum_{j=0}^{t-1} h^2 \omega_{j,1} g_{n+j}, \right.$$

$$\sum_{j=0}^{r-1} \alpha_{j,2} y_{n+j} + \sum_{j=0}^{s-1} h\beta_{j,2} f_{n+j} + \sum_{j=0}^{t-1} h^2 \omega_{j,2} g_{n+j}, \tag{1.14}$$

$$\left. \sum_{j=0}^{r-1} \alpha_{j,p-1} y_{n+j} + \sum_{j=0}^{s-1} h\beta_{j,p-1} f_{n+j} + \sum_{j=0}^{t-1} h^2 \omega_{j,p-1} g_{n+j} \right\} \left(1, x, x^2, \dots, x^{p-1}\right)^T$$

Recall that p=r+s+t, such that (1.14) reduces to

$$y(x) = \left\{ \sum_{j=0}^{r-1} \alpha_{j,r+s+t-1} y_{n+j} + \sum_{j=0}^{s-1} h\beta_{j,r+s+t-1} f_{n+j} + \sum_{j=0}^{t-1} h^2 \omega_{j,r+s+t-1} g_{n+j} \right\} \left(1, x, x^2, \dots, x^{r+s+t-1}\right)^T \tag{1.15}$$

Thus, expanding (1.15) fully we obtain the proposed continuous scheme as follows:

$$y(x) = (y_n, \dots, y_{n+r-1}, f_n, \dots, f_{n+s-1}, g_n, \dots, g_{n+t-1}) C^T \left(1, x, x^2, \dots, x^{r+s+t-1}\right)^T \tag{1.16}$$

where T denotes transpose of the matrix C in (1.12) and the vector $(1, x, x^2, \dots, x^{r+s+t-1})$.

Remark 1.

We call D the multistep collocation and interpolation matrix which has a very simple structure. From (1.8), the columns of D which give the continuous coefficients $\alpha_j(x)$, $\beta_j(x)$, $\omega_j(x)$ can be obtain from the corresponding columns of C. As can be seen the entries of C are the constant coefficients of the polynomial given in (1.11) which are to be determined. The matrix C is the solution vector (output) and D is termed the data (input), which is assumed to be non-singular for the existence of the inverse matrix C.

In the second-derivative methods, we see that not only the function f(x,y) is evaluated at some intermediate points, but in addition the functions Df, D^2f where D is the differential operator (see [14]). Hence, in addition to the computation of the f-values at the internal stages in the standard Runge-Kutta methods, the modified methods involve computing g-values, where f and g are as defined in (1.10).

According to [15] these methods can be practical if the costs of evaluating g are comparable to those in evaluating f and can even be more efficient than the standard Runge-Kutta methods if the number of function evaluations is fewer. It is convenient to rewrite the coefficients of the defining formula (1.10) evaluated at some certain points in the block matrix form as,

$$Y = e \otimes y_n + h(A \otimes I_N)F(Y) + h^2(\hat{A} \otimes I_N)G(Y), \tag{1.17a}$$

$$y_{n+1} = y_n + h\left(b^T \otimes I_N\right)F(Y) + h^2\left(\hat{b}^T \otimes I_N\right)G(Y), \tag{1.17b}$$

where $A=[a_{ij}]_{sxs}$, $\hat{A}=[\hat{a}_{ij}]_{sxs}$ indicate the dependence of the stages on the derivatives found at the other stages and $b=[b_i]_{sx1}$, $\hat{b}=\left[\hat{b}_i\right]_{sx1}$ are vectors of quadrature weights showing how the final result depends on the derivatives computed at the various stages, I is the identity matrix of size equal to the differential equation system to be solved and N is the dimension of the system. Also $\otimes$ is the Kronecker product of two matrices. For simplicity, we write the method in (1.17) as follows:

$$Y = y_n + hAF(Y) + h^2\hat{A}G(Y), \tag{1.18a}$$

$$y_{n+1} = y_n + hb^T F(Y) + h^2\hat{b}^T G(Y), \tag{1.18b}$$

and the block vectors in R^{sN} are defined by

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_s \end{bmatrix} \qquad F(Y) = \begin{bmatrix} f(Y_1) \\ f(Y_2) \\ \vdots \\ f(Y_s) \end{bmatrix} \qquad G(Y) = \begin{bmatrix} g(Y_1) \\ g(Y_2) \\ \vdots \\ g(Y_s) \end{bmatrix} \tag{1.19}$$

The coefficients of the implicit second-derivative Runge-Kutta collocation methods can be conveniently represented more compactly in an extended partitioned Butcher Tableau, of the form

$$\begin{array}{c|c||c} c & A & \hat{A} \\ \hline & b^T & \hat{b}^T \end{array} \tag{1.20}$$

where $c=[1]_{sx1}$ is the abscissae vectors which indicates the position within the step of the stage values.

Specification of the implicit second derivative Runge-Kutta collocation methods

A fourth-order implicit second derivative Runge-Kutta collocation method

In this section we develop the general form of conditions for the coefficients of the implicit second-derivative Runge-Kutta collocation methods. To obtain the coefficients of the first method, the matrix D in (1.8) takes the form,

$$D = \begin{pmatrix} 1 & x_n & x_n^2 & x_n^3 & x_n^4 \\ 0 & 1 & 2c_{n+u} & 3c_{n+u}^2 & 4c_{n+u}^3 \\ 0 & 1 & 2c_{n+1} & 3c_{n+1}^2 & 4c_{n+1}^3 \\ 0 & 0 & 2 & 6c_{n+u} & 12c_{n+u}^2 \\ 0 & 0 & 2 & 6c_{n+1} & 12c_{n+1}^2 \end{pmatrix} \tag{1.21}$$

Inverting the matrix D in (1.21) once, we obtain the continuous scheme of the form in (1.16) as follows

$$y(x) = \alpha_0(x)y_n + h[\beta_0(x)f_{n+u} + \beta_1(x)f_{n+1}] + h^2[\omega_0(x)g_{n+u} + \omega_1(x)g_{n+1}] \tag{1.22}$$

where

$$\alpha_0(x) = 1$$

$$h\beta_0(x) = \left[\frac{27(x-x_n)^4 - 72h(x-x_n)^3 + 54h^2(x-x_n)^2}{16h^3} \right]$$

$$h\beta_1(x) = \left[\frac{-27(x-x_n)^4 + 72h(x-x_n)^3 - 54h^2(x-x_n)^2 + 16h^3(x-x_n)}{16h^3} \right]$$

$$h^2\omega_0(x) = \left[\frac{9(x-x_n)^4 - 28h(x-x_n)^3 + 30h^2(x-x_n)^2 - 12h^3(x-x_n)}{16h^2} \right]$$

$$h^2\omega_1(x) = \left[\frac{9(x-x_n)^4 - 20h(x-x_n)^3 + 14h^2(x-x_n)^2 - 4h^3(x-x_n)}{16h^3} \right]$$

To get the value of u in (1.21), we find the zero of the Nth degree Jacobi polynomial (see, [16]) defined by

$$P_N^{(\alpha,\beta)} = \sum_{i=1}^{N} (-1)^{N-1}\gamma_i x^i \tag{1.23a}$$

where

$$\gamma_i = \frac{N-1+1}{i} \frac{N+i+\alpha+\beta}{i+\beta}\gamma_{i-1}, \qquad i = 1,2,\dots,N \tag{1.23b}$$

with $\gamma_0 = 1$.

Next by substituting $u = \frac{1}{3}$ into the continuous scheme (1.22) and evaluate at $x = x_{n+1}$ and x_{n+u} we obtain the following block hybrid discrete scheme, which can be applied simultaneously as block method for dense output, if desired,

$$y_{n+1} = y_n + \frac{h}{16}[9f_{n+u} + 7f_{n+1}] + \frac{h^2}{16}[-g_{n+u} - g_{n+1}] \tag{1.24a}$$

$$y_{n+u} = y_n + \frac{h}{432}\left[99f_{n+u} + 45f_{n+1}\right] + \frac{h^2}{432}\left[-43g_{n+u} - 11g_{n+1}\right] \tag{1.24b}$$

If we convert the block hybrid discrete scheme (1.24(a-b)) to implicit second-derivative Runge-Kutta collocation method and write the method in the form of (1.18) we have the following

$$y_n = y_{n-1} + h\left(\frac{9}{16}\right)F_2 + h\left(\frac{7}{16}\right)F_3 - h^2\left(\frac{1}{16}\right)G_2 - h^2\left(\frac{1}{16}\right)G_3 \tag{1.25}$$

The internal stage values at the nth step are calculated as,

$$Y_1 = y_{n-1}$$

$$Y_2 = y_{n-1} + h\left(\frac{11}{48}\right)F_2 + h\left(\frac{5}{48}\right)F_3 - h^2\left(\frac{43}{432}\right)G_2 - h^2\left(\frac{11}{432}\right)G_3$$

$$Y_3 = y_{n-1} + h\left(\frac{9}{16}\right)F_2 + h\left(\frac{7}{16}\right)F_3 - h^2\left(\frac{1}{16}\right)G_2 - h^2\left(\frac{1}{16}\right)G_3$$

with the stage derivatives as follows:

$$F_1 = f\left(x_{n-1} + h(0), Y_1\right),$$

$$F_2 = f\left(x_{n-1} + h\left(\frac{1}{3}\right), Y_2\right),$$

$$F_3 = f\left(x_{n-1} + h(1), Y_3\right).$$

The coefficients of the implicit second-derivative Runge-Kutta collocation method (1.25) can be conveniently represented more compactly in an extended partitioned Butcher Tableau of the form in (1.20) as follows:

$$
\begin{array}{c|cc||cc}
\dfrac{1}{3} & \dfrac{11}{48} & \dfrac{5}{48} & \dfrac{-43}{432} & \dfrac{-11}{48} \\
1 & \dfrac{9}{16} & \dfrac{7}{16} & \dfrac{-1}{16} & \dfrac{-1}{16} \\
\hline
& \dfrac{9}{16} & \dfrac{7}{16} & \dfrac{-1}{16} & \dfrac{-1}{16}
\end{array}
\qquad \dfrac{c\ \left|\ A\ \right\|\ \widehat{A}}{\ b^T\ \left\|\ \widehat{b}^T\right.}
$$

A sixth-order implicit second derivative Runge-Kutta collocation method

Here we derive implicit second-derivative Runge-Kutta collocation method of higher order for the numerical solution of stiff systems. The matrix D in (1.8) takes the following form

$$D = \begin{pmatrix} 1 & x_n & x_n^2 & x_n^3 & x_n^4 & x_n^5 & x_n^6 \\ 0 & 1 & 2c_{n+u} & 3c_{n+u}^2 & 4c_{n+u}^3 & 5c_{n+u}^4 & 6c_{n+u}^5 \\ 0 & 1 & 2c_{n+v} & 3c_{n+v}^2 & 4c_{n+v}^3 & 5c_{n+v}^4 & 6c_{n+v}^5 \\ 0 & 1 & 2c_{n+1} & 3c_{n+1}^2 & 4c_{n+1}^3 & 5c_{n+1}^4 & 6c_{n+1}^5 \\ 0 & 0 & 2 & 6c_{n+u} & 12c_{n+u}^2 & 20c_{n+u}^3 & 30c_{n+u}^4 \\ 0 & 0 & 2 & 6c_{n+v} & 12c_{n+v}^2 & 20c_{n+v}^3 & 30c_{n+v}^4 \\ 0 & 0 & 2 & 6c_{n+1} & 12c_{n+1}^2 & 20c_{n+1}^3 & 30c_{n+1}^4 \end{pmatrix} \tag{1.26}$$

where u and v are the zeros of p2(x)=0. Jacobi polynomial in (1.23) of degree 2 which are valid in the interval $[x_n,x_{n+1}]$. Inverting the matrix in (1.26) once we obtain the continuous scheme as follows:

$$y(x)=\alpha_0(x)y_n + h\left[\beta_0(x)f_{n+u}+\beta_1(x)f_{n+v}+\beta_2(x)f_{n+1}\right]+h^2\left[\omega_0(x)g_{n+u}+\omega_1(x)g_{n+v}+\omega_2(x)g_{n+1}\right] \tag{1.27}$$

where

$$\alpha_0(x)=1,$$

$$h\beta_0(x)=\left[\frac{\begin{array}{c}(1600-100\sqrt{6})\zeta^6-(5232+48\sqrt{6})h\zeta^5+(6336+729\sqrt{6})h^2\zeta^4\\-(3536+904\sqrt{6})h^3\zeta^3+(912+318\sqrt{6})h^5\zeta\end{array}}{216h^5}\right]$$

$$h\beta_1(x)=\left[\frac{\begin{array}{c}(12400+3900\sqrt{6})\zeta^6-(36048+10128\sqrt{6})h\zeta^5+(35604+7569)h^2\zeta^4\\-(13904+744\sqrt{6})h^3\zeta^3+(2568-402\sqrt{6})h^5\zeta\end{array}}{216(7+2\sqrt{6})h^5}\right]$$

$$h\beta_2(x)=\left[\frac{-16000\zeta^6+52320h\zeta^5-63360h^2\zeta^4+35360h^3\zeta^3-9120h^4\zeta^2+1080h^5\zeta}{1080h^5}\right]$$

$$h^2\omega_0(x)=\left[\frac{\begin{array}{c}2500\zeta^6-(9600+300\sqrt{6})h\zeta^5+(14325+1050\sqrt{6})h^2\zeta^4-\\(10400+1350\sqrt{6})h^3\zeta^3+(3750+750\sqrt{6})h^4\zeta^2-(600+150\sqrt{6})h^5\zeta\end{array}}{(1512+432\sqrt{6})h^4}\right]$$

$$h^2\omega_1(x)=\left[\frac{\begin{array}{c}(4200+1200\sqrt{6})\zeta^6-(15264+4104\sqrt{6})h\zeta^5+(21042+5112\sqrt{6})h^2\zeta^4\\-(13584+2724\sqrt{6})h^3\zeta^3+(4140+540\sqrt{6})h^4\zeta^2-(576+36\sqrt{6})h^5\zeta\end{array}}{1296h^4}\right]$$

$$h^2\omega_2(x)=\left[\frac{4000\zeta^6-12480h\zeta^5+14640h^2\zeta^4-8000h^3\zeta^3+2040h^4\zeta^2-240h^5\zeta}{2160h^4}\right]$$

Evaluating the continuous scheme (1.27) at the points $\{x_{n+1},x_{n+u},x_{n+v}\}$ we obtain the block hybrid discrete scheme which can also be applied simultaneously as block method for dense output, if desired,

$$y_{n+1} = y_n + \frac{h}{216}\left[\left(80-5\sqrt{6}\right)f_{n+u} + \left(80+5\sqrt{6}\right)f_{n+v} + 56f_{n+1}\right] + \frac{h^2}{216}\left[-\left(7-2\sqrt{6}\right)g_{n+u} - \left(7+2\sqrt{6}\right)g_{n+v} - 4g_{n+1}\right]$$

$$y_{n+u} = y_n + \frac{h}{2700000}\left[\left(460000-68125\sqrt{6}\right)f_{n+u} + \left(418528-168467\sqrt{6}\right)f_{n+v} + \left(201472-33408\sqrt{6}\right)f_{n+1}\right]$$

$$+ \frac{h^2}{2700000}\left[\left(128675-27700\sqrt{6}\right)g_{n+u} - \left(45299+6964\sqrt{6}\right)g_{n+v} - \left(21704-3456\sqrt{6}\right)g_{n+1}\right] \qquad (1.28)$$

$$y_{n+v} = y_n + \frac{h}{2700000}\left[\left(418528+168467\sqrt{6}\right)f_{n+u} + \left(460000+68125\sqrt{6}\right)f_{n+v} + \left(201472+33408\sqrt{6}\right)f_{n+1}\right]$$

$$+ \frac{h^2}{2700000}\left[\left(45299-6964\sqrt{6}\right)g_{n+u} - \left(128675+27700\sqrt{6}\right)g_{n+v} - \left(21704+3456\sqrt{6}\right)g_{n+1}\right].$$

Writing the method in the formalism of (1.18) we have

$$y_n = y_{n-1} + h\left(\frac{10}{27}-\frac{5\sqrt{6}}{216}\right)F_2 + h\left(\frac{10}{27}+\frac{5\sqrt{6}}{216}\right)F_3 + h\left(\frac{7}{27}\right)F_4 - h^2\left(\frac{7}{216}-\frac{\sqrt{6}}{108}\right)G_2 - h^2\left(\frac{7}{216}+\frac{\sqrt{6}}{108}\right)G_3 - h^2\left(\frac{4}{216}\right)G_4 \qquad (1.29)$$

The internal stage values at the nth step are calculated as,

$$Y_1 = y_{n-1}$$

$$Y_2 = y_{n-1} + h\left(\frac{23}{135}-\frac{109\sqrt{6}}{4320}\right)F_2 + h\left(\frac{13079}{84375}-\frac{168467\sqrt{6}}{2700000}\right)F_3 + h\left(\frac{6296}{84375}-\frac{116\sqrt{6}}{9375}\right)F_4$$

$$- h^2\left(\frac{5147}{108000}-\frac{277\sqrt{6}}{27000}\right)G_2 - h^2\left(\frac{45299}{2700000}+\frac{1741\sqrt{6}}{675000}\right)G_3 - h^2\left(\frac{2713}{337500}-\frac{4\sqrt{6}}{3125}\right)G_4$$

$$Y_3 = y_{n-1} + h\left(\frac{13079}{84375}+\frac{16847\sqrt{6}}{2700000}\right)F_2 + h\left(\frac{23}{135}+\frac{109\sqrt{6}}{4320}\right)F_3 + h\left(\frac{6296}{84375}+\frac{116\sqrt{6}}{9375}\right)F_4$$

$$- h^2\left(\frac{45299}{2700000}-\frac{1741\sqrt{6}}{675000}\right)G_2 - h^2\left(\frac{5147}{108000}+\frac{277\sqrt{6}}{27000}\right)G_3 - h^2\left(\frac{2713}{337500}+\frac{4\sqrt{6}}{3125}\right)G_4$$

$$Y_4 = y_{n-1} + h\left(\frac{10}{27}-\frac{5\sqrt{6}}{216}\right)F_2 + h\left(\frac{10}{27}+\frac{5\sqrt{6}}{216}\right)F_3 + h\left(\frac{7}{27}\right)F_4 - h^2\left(\frac{7}{216}-\frac{\sqrt{6}}{108}\right)G_2 - h^2\left(\frac{7}{216}+\frac{\sqrt{6}}{108}\right)G_3 - h^2\left(\frac{4}{216}\right)G_4$$

where the stage derivatives are as follows:

$$F_1 = f\left(x_{n-1} + h(0), Y_1\right),$$

$$F_2 = f\left(x_{n-1} + h\left(\frac{2}{5}-\frac{\sqrt{6}}{10}\right), Y_2\right),$$

$$F_3 = f\left(x_{n-1} + h\left(\frac{2}{5}+\frac{\sqrt{6}}{10}\right), Y_3\right),$$

$$F_4 = f\left(x_{n-1} + h(1), Y_4\right)$$

We write the coefficients of the method in (1.29) in an extended partitioned Butcher Tableau (1.20) as follows:

$\dfrac{4-\sqrt{6}}{10}$	$\dfrac{736-109\sqrt{6}}{4320}$	$\dfrac{418528-168467\sqrt{6}}{2700000}$	$\dfrac{6296-1044\sqrt{6}}{84375}$	$\dfrac{-5147+1108\sqrt{6}}{108000}$	$\dfrac{-45299-6964\sqrt{6}}{2700000}$	$\dfrac{-21704+3456\sqrt{6}}{2700000}$
$\dfrac{4+\sqrt{6}}{10}$	$\dfrac{418528+16847\sqrt{6}}{2700000}$	$\dfrac{736+109\sqrt{6}}{4320}$	$\dfrac{6296+1044\sqrt{6}}{84375}$	$\dfrac{-45299+6964\sqrt{6}}{2700000}$	$\dfrac{-5147-1108\sqrt{6}}{108000}$	$\dfrac{-21704+3456\sqrt{6}}{2700000}$

| 1 | $\dfrac{80-5\sqrt{6}}{216}$ | $\dfrac{80+5\sqrt{6}}{216}$ | $\dfrac{56}{216}$ | $\dfrac{-7+2\sqrt{6}}{216}$ | $\dfrac{-7-2\sqrt{6}}{216}$ | $\dfrac{-4}{216}$ |
| | $\dfrac{80-5\sqrt{6}}{216}$ | $\dfrac{80+5\sqrt{6}}{216}$ | $\dfrac{56}{216}$ | $\dfrac{-7+2\sqrt{6}}{216}$ | $\dfrac{-7-2\sqrt{6}}{216}$ | $\dfrac{-4}{216}$ |

Analysis of the properties of the implicit SDRK collocation methods

Order, Consistency, Zero-Stability and Convergence of the Implicit SDRKCMs

With the multi-step collocation formula (1.10) we associate the linear difference operator ℓ defined by:

$$\ell[\,y(x);h\,] = \sum_{j=0}^{r}\alpha_j(x)y(x+jh) + h\sum_{j=0}^{s}\beta_j(x)y'(x+jh) + h^2\sum_{j=0}^{t}\omega_j(x)y''(x+jh) \qquad (1.30)$$

where y(x) is an arbitrary function, continuously differentiable on [x₀, T]. Following [17], we can write the terms in (1.30) as a Taylor series expansion about the point x to obtain the expression,

$$\ell[y(x);h] = C_0 y(x) + C_1 hy'(x) + C_2 h^2 y''(x) + \cdots + C_p h^p y^{(p)}(x) + \cdots, \qquad (1.31)$$

where the constant coefficients C_p, p = 0,1,2,… are given as follows:

$$C_0 = \sum_{j=0}^{r}\alpha_j$$

$$C_1 = \sum_{j=1}^{r} j\alpha_j - \sum_{j=0}^{s}\beta_j$$

$$C_2 = \frac{1}{2!}\left(\sum_{j=1}^{r} j^2\alpha_j - 2\sum_{j=1}^{s} j\beta_j - \sum_{j=0}^{t}\omega_j\right)$$

$$\vdots \qquad\qquad \vdots \qquad\qquad \vdots$$

$$C_p = \frac{1}{p!}\left(\sum_{j=1}^{r} j^p\alpha_j - \frac{1}{(p-1)!}\sum_{j=1}^{s} j^{p-1}\beta_j - \frac{1}{(p-2)!}\sum_{j=1}^{t} j^{p-2}\omega_j\right), \quad p = 3,4,\ldots$$

According to [17], the multistep collocation formula (1.10) has order p if

$$\ell[\,y(x);h\,] = O\!\left(h^{p+1}\right) \quad C_0 = C_1 = \cdots = C_p = 0, \quad C_{p+1} \neq 0 \qquad (1.32)$$

Therefore, C_{p+1} is the error constant and $C_{p+1}h^{p+1}y^{(p+1)}(x_n)$ is the principal local truncation error at the point x_n (see [18]). Hence from our calculation the order and error constants for the constructed methods are presented in Table 1. It is clear from the Table that the implicit second-derivative Runge-Kutta collocation methods are of high-order. They have smaller error constants and hence more accurate than the conventional Radau-Runge-Kutta methods of the same order of convergence.

Table 1. Order and error constants of the implicit SDRK collocation methods

Method	Order	Error constant
Method (1.25)	P=4	$C_5 = 9.8765 \times 10^{-3}$
Method (1.29)	P=6	$C_7 = 2.5714 \times 10^{-4}$

Definition 1.3: Consistency

A linear multistep method is said to be consistent if the order of the method is greater or equal to one, that is if p>1 (see, [17]).(i) $\rho(1)=0$ and(ii)$\rho'(1)=\sigma(1)$, where $\rho(z)$ and $\sigma(z)$ are respectively the 1st and 2nd characteristic polynomials.

From Table 1 and definition 1.3, we can attest that the implicit second-derivative Runge-Kutta collocation methods are consistent.

Definition 1.4: Zero-stability

A linear multistep method is said to be zero-stable if the roots:

$$\rho(\lambda) = \det\left[\sum_{i=0}^{k} A^{(i)} \lambda^{k-i} \right] = 0$$

$$|\lambda_j| \leq 1, \quad j = 1,\ldots,k$$

$$|\lambda_j| = 1$$

see [17].

Based on definition 1.4, the newly constructed implicit second-derivative Runge-Kutta collocation methods are zero-stable.

Definition 1.5: Convergence

The necessary and sufficient conditions for a linear multistep method to be convergent are that it must be consistent and zero-stable (see [10]). Hence, from definitions 1.3 and 1.4 the implicit second-derivative Runge-Kutta collocation methods are convergent.

Regions of absolute stability of the implicit SDRK collocation methods

To study the stability properties of the implicit second-derivative Runge-Kutta collocation methods we reformulate them as general linear methods (see [19]). Hence, we use the notations introduced by [20] which a general linear method is represented by the partitioned (s+r)×(s+r) matrix, (containing A, U, B, V),

$$\begin{bmatrix} Y^{[n]} \\ y^{[n-1]} \end{bmatrix} = \left[\begin{array}{c|c} A & U \\ \hline B & V \end{array} \right] \begin{bmatrix} hf(Y^{[n]}) \\ y^{[n]} \end{bmatrix} \quad n = 1, 2, \ldots,N, \qquad (1.33a)$$

where

$$
Y^{[n]} = \begin{bmatrix} Y_1^{[n]} \\ Y_2^{[n]} \\ \vdots \\ Y_s^{[n]} \end{bmatrix}
\qquad
y^{[n-1]} = \begin{bmatrix} y_1^{[n-1]} \\ y_2^{[n-1]} \\ \vdots \\ y_r^{[n-1]} \end{bmatrix}
\qquad
f\left(Y^{[n]}\right) = \begin{bmatrix} f(Y_1^{[n]}) \\ f(Y_2^{[n]}) \\ \vdots \\ f(Y_s^{[n]}) \end{bmatrix}
\qquad
y^{[n]} = \begin{bmatrix} y_1^{[n]} \\ y_2^{[n]} \\ \vdots \\ y_r^{[n]} \end{bmatrix}
$$

$$
A = \begin{bmatrix} 0 & 0 \\ A & B \end{bmatrix}
\qquad
U = \begin{bmatrix} I & 0 & 0 \\ 0 & \mu & e-\mu \end{bmatrix}
\qquad
B = \begin{bmatrix} A & B \\ 0 & 0 \\ v^T & \omega^T \end{bmatrix}
\qquad
V = \begin{bmatrix} I & \mu & e-\mu \\ 0 & 0 & I \\ 0 & 0 & I-\theta \end{bmatrix}
$$

and e=[1,…,1]$^t \in$R. Hence (1.33a) takes the form

$$
\begin{bmatrix} Y_1^{[n]} \\ Y_2^{[n]} \\ \vdots \\ Y_s^{[n]} \\ - \\ y_1^{[n]} \\ \vdots \\ y_r^{[n]} \end{bmatrix}
=
\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array} \right]
\begin{bmatrix} hf(Y_1^{[n]}) \\ hf(Y_2^{[n]}) \\ \vdots \\ hf(Y_s^{[n]}) \\ - \\ y_1^{[n-1]} \\ \vdots \\ y_r^{[n-1]} \end{bmatrix}
\tag{1.33b}
$$

where *r* denotes quantities as output from each step and input to the next step and *s* denotes stage values used in the computation of the step $y_1,…,y_s$. The coefficients of these matrices indicate the relationship between the various numerical quantities that arise in the computation of stability regions. The elements of the matrices A, U, B and V are substituted into the stability matrix. In the sense of [21] we apply (1.33) to the linear test equation y'=λy,x>0 and λ∈C but for the second-derivative high-order methods we use y''=λ2y which leads to the recurrence relation y$^{[n+1]}$=M(z)y$^{[n]}$, n=1,2,…,N-1, z=λh, where the stability matrix M(z) is defined by

$$
M(z) = V + zB(1 - zA)^{-1}U
\tag{1.34}
$$

We also define the stability polynomial p(η,z) by the relation

$$
\rho(\eta, z) = \det\left(\eta I - M(z)\right)
\tag{1.35}
$$

and the absolute stability region ℜ of the method is given by

$$
\Re = x \in C : \rho(\eta, z) = 1 \Rightarrow |\eta| \leq 1
$$

To compute the regions of absolute stability we substitute the elements of the matrices A, U, B and V into the stability function (1.34) and finally into the stability polynomial (1.35)

of the methods, which is plotted to produce the required graphs of the absolute stability regions of the methods as shown in Figure 1.
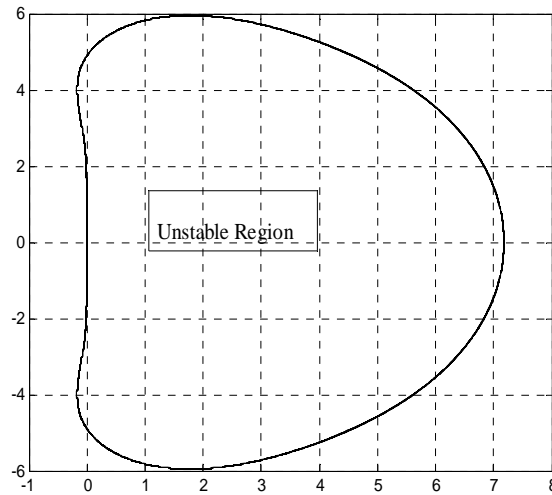
Method (1.25) is A(α)-stable
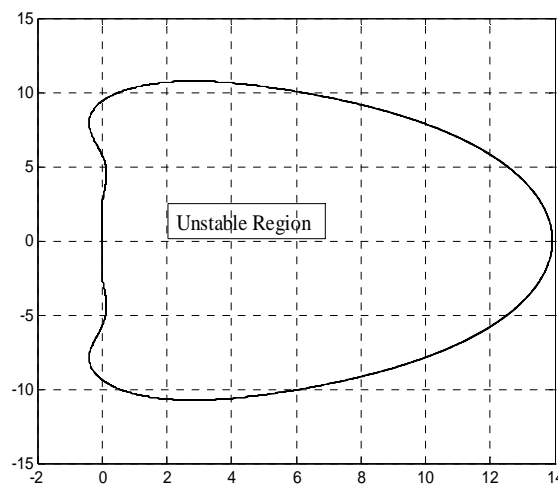
Method(1.29) is A(α)-stable

Figure 1. *Regions of absolute stability of the implicit SDRK collocation methods*

The methods constructed in this paper are A(α)-stable, since their regions consist of the complex plane outside the enclosed Figures, with only very small portion inside the left hand-half plane.

Numerical experiments

Our aim in this section is to compare the performance of the new methods with other methods of the conventional type (Radau methods) which are widely known to be among the

most efficient methods for the numerical solution of stiff ordinary differential equations. Therefore, we report the numerical results obtained from the applications of the new implicit second-derivative Runge-Kutta collocation methods. We present the computed results in the formalism of [22]. In the presentation we use *nfe* to denote the number of function evaluations and *Ext* to indicate the exact solutions.

Example 1. Consider the stiff system of first order differential equation,

$$y_1'(x) = -y_1(x) - 15y_2(x) + 15e^{-x}, \qquad y_1(0) = 1,$$

$$y_2'(x) = 15y_1(x) - y_2(x) - 15e^{-x}, \qquad y_2(0) = 1.$$

The exact solution is

$$y_1(x) = \exp(-x),$$

$$y_2(x) = \exp(-x).$$

This system has eigenvalues of large modulus lying close to the imaginary axis $-1 \pm 15i$. We solve the system in the range of [0, 50] with h = 0.1 and the results obtained are shown in Table 2. In this example we compare the results from the two new methods side by side in the Table. The solution curves obtained are compared with the exact (Ext) solutions in Figure 2. Thus, for fair comparison see the results from the third derivative methods of [23] Table 4 page 447 and results of [24] Table 2 page 166.

Table 2. Absolute errors in the numerical integration of example 1

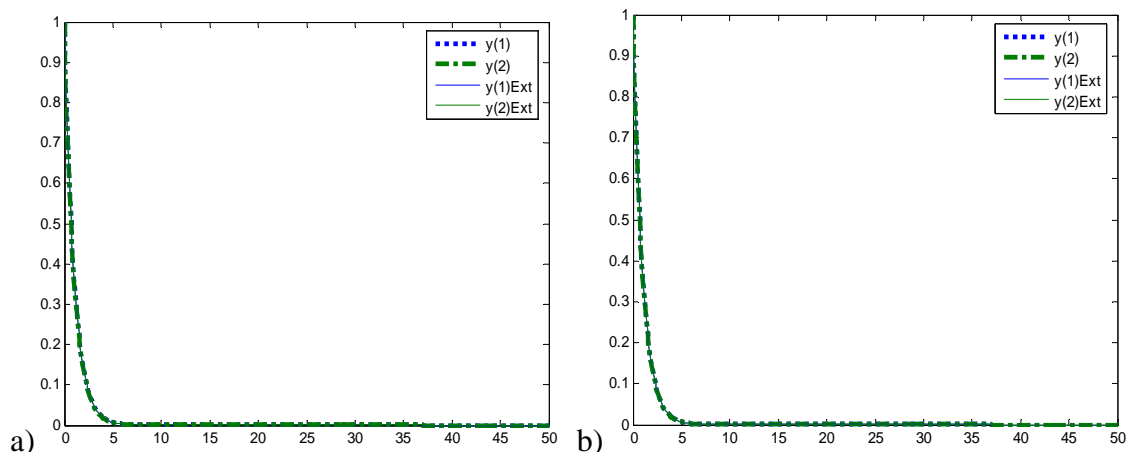| x	Y_i	Method (1.25)	Method (1.29)
5	Y_1	$3.46368813864673 \times 10^{-3}$	$2.16834805259769 \times 10^{-3}$
	Y_2	$3.94364780750056 \times 10^{-4}$	$6.23671456063679 \times 10^{-4}$
50	Y_1	$1.49495218771792 \times 10^{-4}$	$7.38442455972734 \times 10^{-5}$
	Y_2	$4.86742095474412 \times 10^{-9}$	$2.19934956374418 \times 10^{-9}$
150	Y_1	$1.85517639473028 \times 10^{-4}$	$1.20177414793520 \times 10^{-4}$
	Y_2	$1.45786198883123 \times 10^{-8}$	$7.02697714351372 \times 10^{-9}$
250	Y_1	$1.25223273382179 \times 10^{-13}$	$1.99479371074799 \times 10^{-14}$
	Y_2	$8.66743589934986 \times 10^{-13}$	$3.55424556071686 \times 10^{-13}$
500	Y_1	$1.64891543846432 \times 10^{-23}$	$2.32684011997371 \times 10^{-24}$
	Y_2	$4.31480861603385 \times 10^{-24}$	$3.70791674680754 \times 10^{-24}$

Figure 2. Graphical plots of example 1 using Second Derivative Runge-Kutta Collocation methods: a) Solution curve of example 1 using method (1.25), with nfe =500; b) Solution curve of example 1 using method (1.29), with nfe =500

Example 2: The second example is a stiff linear system of equation, see [25],

$$\begin{bmatrix} y_1'(x) \\ y_2'(x) \end{bmatrix} = \begin{bmatrix} -2 & 1 \\ 998 & -999 \end{bmatrix} \begin{bmatrix} y_1(x) \\ y_2(x) \end{bmatrix} + \begin{bmatrix} 2\sin(x) \\ 999(\cos(x) - \sin(x)) \end{bmatrix}, \quad \begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

The exact solution is given by

$$\begin{bmatrix} y_1(x) \\ y_2(x) \end{bmatrix} = \begin{bmatrix} 2\exp(-x) + \sin(x) \\ 2\exp(-x) + \cos(x) \end{bmatrix}$$

We solve this system in the range [0, 10]. In this example we also compare the results from the two new methods in Table 3 and the efficiency curves obtained are compared with the exact solutions in Figure 3.

Table 3. Absolute errors in the numerical integration of example 2

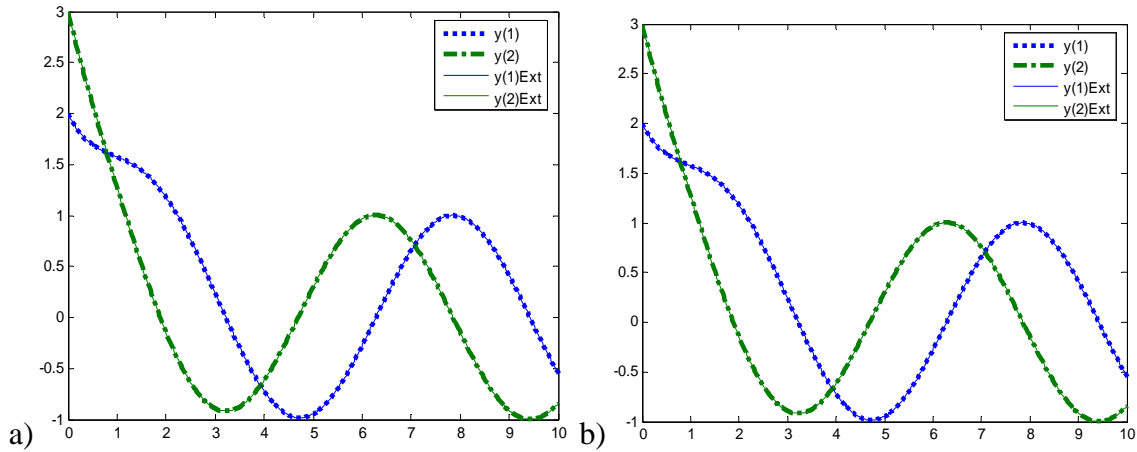| x	Y_i	Method (1.25)	Method (1.29)
5	Y_1	$7.00794977603891 \times 10^{-12}$	$2.22044604925031 \times 10^{-16}$
	Y_2	$7.04547531427124 \times 10^{-12}$	0
50	Y_1	$2.27042828981894 \times 10^{-11}$	$1.11022302462516 \times 10^{-15}$
	Y_2	$2.27533547558778 \times 10^{-11}$	$1.11022302462516 \times 10^{-15}$
150	Y_1	$7.35667637918880 \times 10^{-11}$	$4.44089209850063 \times 10^{-16}$
	Y_2	$7.35377314597940 \times 10^{-11}$	$3.33066907387547 \times 10^{-16}$
250	Y_1	$4.08534317486442 \times 10^{-11}$	$1.11022302462516 \times 10^{-16}$
	Y_2	$4.08287292863463 \times 10^{-11}$	$1.11022302462516 \times 10^{-16}$
500	Y_1	$6.84553524976650 \times 10^{-11}$	$5.55111512312578 \times 10^{-16}$
	Y_2	$6.84068357514889 \times 10^{-11}$	$5.55111512312578 \times 10^{-16}$

Figure 3. *Graphical plots of example 2 using second derivative Runge-Kutta collocation methods: a) Solution curve of example 2 using method (1.25), with nfe =500;b) Solution curve of example 2 using method (1.29), with nfe =500*

Example 3: The third test example is a stiff problem taken from [25].

$$
\begin{bmatrix} y_1'(x) \\ y_2'(x) \\ y_3'(x) \end{bmatrix} = \begin{bmatrix} 42.2 & 50.1 & -42.1 \\ -66.1 & -58 & 58.1 \\ 26.1 & 42.1 & -34 \end{bmatrix} \begin{bmatrix} y_1(x) \\ y_2(x) \\ y_3(x) \end{bmatrix}, \qquad \begin{bmatrix} y_1(0) \\ y_2(0) \\ y_3(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}
$$

The exact solution is,

$$
\begin{cases} y_1(x) = \exp(0.1x)\sin 8x + \exp(-50x) \\ y_2(x) = \exp(0.1x)\cos 8x - \exp(-50x) \\ y_3(x) = \exp(0.1x)(\cos 8x + \sin 8x) + \exp(-50x) \end{cases}
$$

The eigenvalues of the system are $\lambda_1=-50$ and $\lambda_{2,3} = 0.1\pm8i$. The computed solutions of the example within the interval of [0, 1] are presented in Table 4 and the graphical plots are shown in Figure 4.

Example 4: We consider another linear problem which is particularly referred to by [27] as a troublesome problem for some existing methods because the eigenvalues of this problem are lying close to the imaginary axis where some stiff integrators were known to be inefficient. The eigenvalues of the Jacobian are $\lambda_{1,2} = -10\pm100i$, $\lambda_3 = -4$, $\lambda_4 = -1$, $\lambda_5 = -0.5$ and $\lambda_6 = -0.1$.

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

Table 4. Absolute errors in the numerical integration of example 3

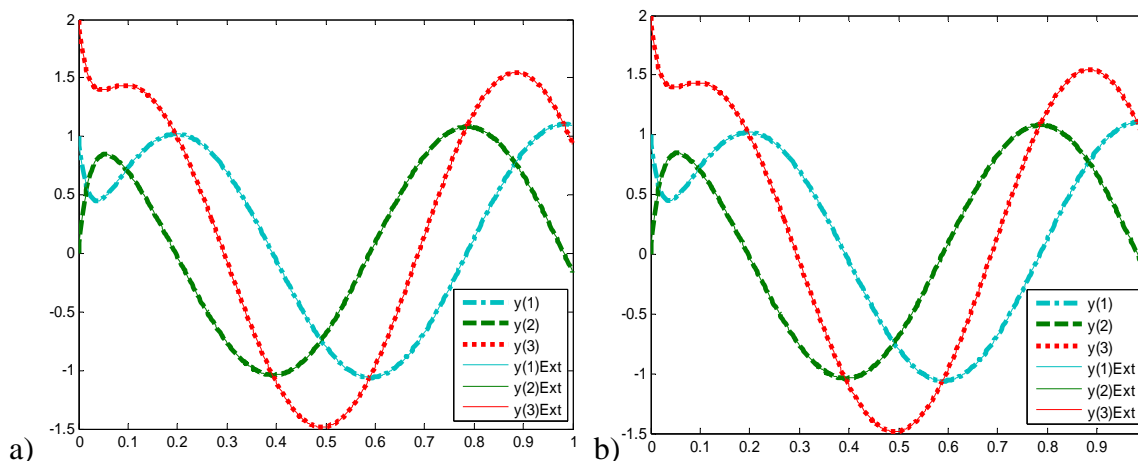| x	Y_i	Radau IIA method [20, 26]	Method (1.29)
	Y_1	$3.70353636647280 \times 10^{-10}$	$3.12638803734444 \times 10^{-13}$
5	Y_2	$3.70363018031838 \times 10^{-10}$	$3.12583292583213 \times 10^{-13}$
	Y_3	$3.70344199751571 \times 10^{-10}$	$3.12638803734444 \times 10^{-13}$
	Y_1	$4.98715513330694 \times 10^{-11}$	$4.27435864480685 \times 10^{-14}$
50	Y_2	$5.00358643407139 \times 10^{-11}$	$4.27435864480685 \times 10^{-14}$
	Y_3	$4.97826224687969 \times 10^{-11}$	$4.21884749357559 \times 10^{-14}$
	Y_1	$2.59459120854899 \times 10^{-13}$	$8.88178419700125 \times 10^{-16}$
150	Y_2	$2.37143638059933 \times 10^{-13}$	$5.99520433297585 \times 10^{-15}$
	Y_3	$1.55916946020795 \times 10^{-14}$	$6.77236045021345 \times 10^{-15}$
	Y_1	$4.30766533554561 \times 10^{-13}$	$7.10542735760101 \times 10^{-15}$
250	Y_2	$4.39759340054025 \times 10^{-13}$	$2.88657986402541 \times 10^{-15}$
	Y3	$8.70636895911048 \times 10^{-13}$	$4.44089209850063 \times 10^{-15}$
	Y_1	$1.29318777908338 \times 10^{-12}$	$5.99520433297585 \times 10^{-15}$
500	Y_2	$9.13435993510348 \times 10^{-14}$	$8.93729534823251 \times 10^{-15}$
	Y3	$1.20203846876166 \times 10^{-12}$	$3.55271367880050 \times 10^{-15}$

Figure 4. *Graphical plots of example 3 using Radau IIA method [20,26] and SDRKC method: a) Solution curve of example 3 using Radau IIA method [20,26], with nfe =500, b) Solution curve of example 3 using method (1.29), with nfe =500*

We solve this problem in the range $0 \leq x \leq 1$. The computed solutions are shown in Table 5, while the graphical plots are displayed in Figure 5. Though, only the first four components $\{y_1, y_2, y_3, y_4\}$ are shown in the Table of values.

Example 5: HIRES

The High irradiance responses problem consists of a stiff system of 8 nonlinear ordinary differential equation which originates from plant physiology and describes how light is involved in morphogenesis. It explains the high irradiance responses of photo morphogenesis on the basis of phytochrome by means of a chemical reaction involving eight reactants in the form of initial value problem see [28,29].

Table 5. Absolute errors in the numerical integration of example 5

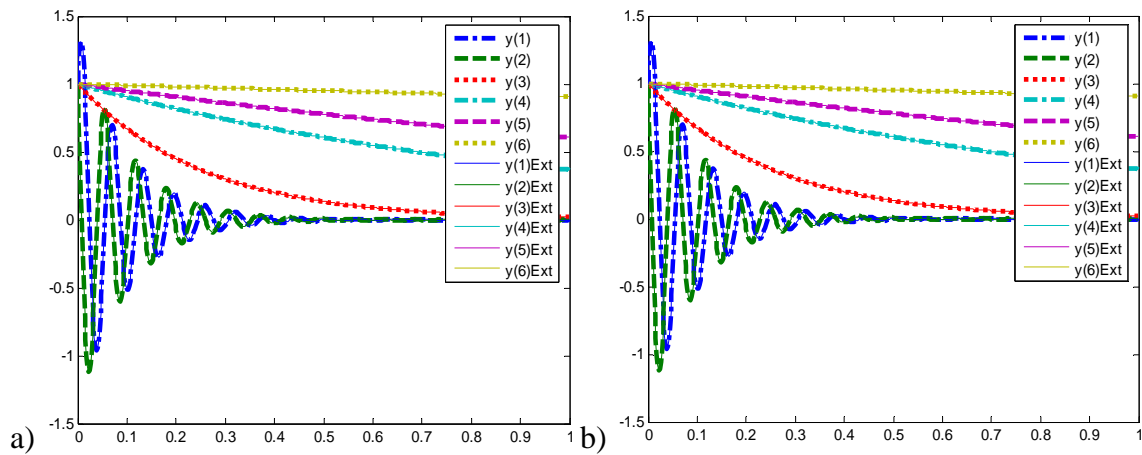| x	Y_i	Radau IIA method [20, 26]	Method (1.29)
5	Y_1	$3.83760043742853 \times 10^{-8}$	$5.84532422465145 \times 10^{-11}$
	Y_2	$2.90659031079721 \times 10^{-8}$	$5.86538248525947 \times 10^{-11}$
	Y_3	$2.22044604925031 \times 10^{-16}$	0
	Y_4	0	$1.11022302462516 \times 10^{-16}$
50	Y_1	$2.32388491339108 \times 10^{-7}$	$1.51528067870998 \times 10^{-10}$
	Y_2	$5.72332533588238 \times 10^{-8}$	$3.82774117957396 \times 10^{-10}$
	Y_3	$1.55431223447522 \times 10^{-15}$	$3.33066907387547 \times 10^{-16}$
	Y_4	$1.11022302462516 \times 10^{-16}$	$4.44089209850063 \times 10^{-16}$
250	Y_1	$1.91869619220464 \times 10^{-8}$	$1.42258618079927 \times 10^{-11}$
	Y_2	$1.09619847643572 \times 10^{-8}$	$3.52476554887904 \times 10^{-11}$
	Y_3	$1.38777878078145 \times 10^{-15}$	$1.66533453693773 \times 10^{-16}$
	Y_4	$7.77156117237610 \times 10^{-16}$	$2.22044604925031 \times 10^{-16}$
500	Y_1	$2.28890865933633 \times 10^{-10}$	$2.65111307337423 \times 10^{-13}$
	Y_2	$1.86748006761241 \times 10^{-10}$	$4.33490943878637 \times 10^{-13}$
	Y3	$3.57353036051222 \times 10^{-16}$	$6.93889390390723 \times 10^{-17}$
	Y_4	$4.44089209850063 \times 10^{-16}$	

Figure 5. *Graphical plots of example 4 using Radau IIA method [20,26] and SDRKC method: a) Solution curve of example 4 using Radau IIA method [20,26], with nfe =500, b) Solution curve of example 4 using method (1.29), with nfe =500*

$$y_1' = -1.71y_1 + 0.43y_2 + 8.32y_3 + 0.0007, \qquad y_1(0) = 1,$$
$$y_2' = 1.71y_1 - 8.75y_2, \qquad y_2(0) = 0,$$
$$y_3' = -10.03y_3 + 0.43y_4 + 0.035y_5, \qquad y_3(0) = 0,$$
$$y_4' = 8.32y_2 + 1.71y_3 - 1.12y_4, \qquad y_4(0) = 0,$$
$$y_5' = -1.745y_5 + 0.43y_6 + 0.43y_7, \qquad y_5(0) = 0,$$
$$y_6' = -280y_6 y_8 + 0.69y_4 + 1.71y_5 - 0.43y_6 + 0.69y_7, \qquad y_6(0) = 0,$$
$$y_7' = 280y_6 y_8 - 1.81y_7, \qquad y_7(0) = 0,$$
$$y_8' = -y_7', \qquad y_8(0) = 0.0057.$$

We solve the problem in the interval [0, 1] and compare the graphical plots with the ODE code of Matlab in Figure 6.
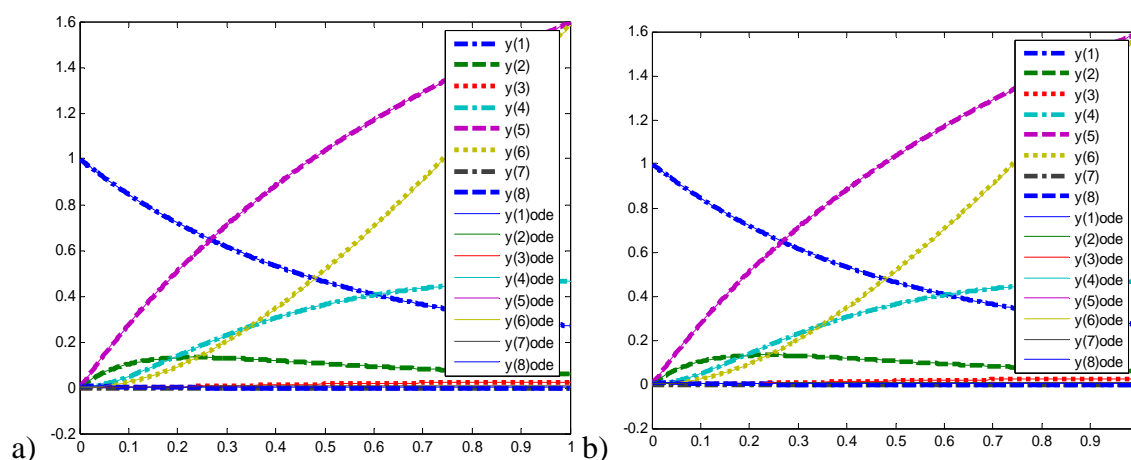

Figure 6. *Graphical plots of example 5 using SDRKCM and the ODE code of Matlab: a) Solution curve of example 5 using method (1.25), with nfe = 500, b) Solution curve of example 5 using method (1.29), with nfe = 500*

Concluding remarks

In conclusion, the methods so derived in this paper so far have been to introduce a class of implicit second-derivative Runge-Kutta collocation methods suitable for the approximate numerical integration of systems, particularly stiff systems of ordinary differential equations. The derived methods proved an efficient way to find numerical solution to systems of initial value problems when the second derivative terms are cheap to evaluate. We present two new methods of orders four and six that are intended for accurate integration of stiff systems of equations. We have also compared the numerical solution obtained with the conventional Radau Runge-Kutta methods which are widely recognized to be among the most efficient methods for stiff systems [3, 20, 26]. The solution curves are also compared with the exact solutions graphically in Figures.

Acknowledgement

The authors are also grateful to the anonymous referee for his/her useful comments and suggestions

References

1. Butcher J. C., *Coefficients for study of Runge-Kutta integration processes*, J. Austral. Math. Society, 1963, 3(2), p. 185 -201.

2. Enright W. H., *Second-derivative multistep methods for stiff ordinary differential equations*, SIAM J. Numer. Anal., 1974, 11(2), p. 321-341.

3. Hairer E., Nørsett S. P., Wanner G., *Solving Ordinary Differential Equations II: Stiff and Differential Algebraic Problems*, Second Revised Edition, Berlin Germany, Springer, 1996.

4. Vigo-Aguiar J., Ramos H., *A family of A-stable Runge-Kutta collocation methods of higher order for initial value problems*, IMA J. Numer Anal., 2007, 27(4), p.798-817.

5. Cash J. R., *High order methods for the numerical integration of ordinary differential equations*, Numer. Math., 1978, 30, p. 385-409.

6. Mitsui T., *Runge-Kutta type integration formulas including the evaluation of the second derivative*, Part I, Publ. RIMS, Kyoto University, Japan, 1982, 18, p. 325-364.

7. Mitsui T., Yakubu D. G., *Two-Step Family of Look-Ahead linear multistep methods for ODEs*, The science and Engineering Review Doshisha University, Japan, 2011, 52(3), p. 181-188.

8. Urabe M., *An implicit one-step method of high-order accuracy for the numerical integration of ordinary differential equations*, J. Numer. Math., 1970, 15, p. 151-164.

9. Yakubu D. G., Kwami A. M., *Implicit two-derivative Runge-Kutta collocation methods for systems of initial value problems*, J. Nigerian Math. Society, 2015, 34, p. 128-142.

10. Dahlquist G., *A special stability problem for linear multistep methods,* BIT Numer. Math., 1963, 3, p. 27-43.

11. Butcher J. C., *Order and stability of Runge-Kutta methods*, Nanjing University Conf., 2014.

12. Onumanyi P., Awoyemi D. O., Jator S. N., Sirisena U. W., *New linear multistep methods with continuous coefficients for first order initial value problems in ODEs*, J. Nigerian Math. Society, 1994, 13, p. 37-51.

13. Lie I., Nørsett S. P., *Superconvergence for multistep collocation*, Math. Comp., 1989, 52, p. 65-80.

14. Kastlunger K. H., Wanner G., *Runge-Kutta processes with multiple nodes*, J. Computing, 1972, 9(9), p. 9-24.

15. Chan R. P. K., Tsai A. Y. J., *On explicit two- derivative Runge-Kutta methods*, Numer. Algor., 2010, 53, p. 171-194.

16. Villadsen J., Michelsen M. L., *Solution of Differential Equation Models by Polynomial Approximations*, Prentice-Hall, INC, Eaglewood Cliffs, New Jersey, 1978, p. 112.

17. Lambert J. D., *Computational Methods in Ordinary Differential Equations,* John Willy, and Sons, New York, 1973.

18. Mitsui T., *A modified version of Urabe's implicit single-step method*, J. Comp. Appl. Math., 1987, 20, p. 325-332.

19. Burrage K., Butcher J. C., *Stability criteria for implicit Runge-Kutta methods*, SIAM J. Numer. Anal., 1979, 16, p. 46-57.

20. Butcher J. C., *Numerical Methods for Ordinary Differential Equations*, 2nd ed. John Wiley & Sons Ltd, 2008, p. 225.

21. Chollom J. P., Jackiewicz Z., *Construction of two step Runge-Kutta (TSRK) methods with large regions of absolute stability*, J. Comp. and Appl. Maths., 2003, 157, p. 125-137.

22. Butcher J. C., Hojjati G., *Second derivative methods with Runge-Kutta stability*, Numer. Algor., 2005, 40, p. 415-429.

23. Ali K. E., Hojjati G., *Third derivative multistep methods for stiff systems*, Intern. J. of Nonlinear Science, 2012, 14(4), p. 443-450.

24. Akinfenwa O. A., Akinnukawe B., Mudasiru S. B., *A family of continuous third derivative block methods for solving stiff systems of first order ordinary differential equations*, J. Nigerian Math. Society, 2015, 34, p. 160-168.

25. Lambert J. D., *Numerical Methods for Ordinary Differential System*, John Willy, & Sons, New York, 1991.

26. Butcher J. C., *Integration processes based on Radau quadrature formulas*, Math. Compt., 1964, 18, p. 233-244.

27. Fatunla S. O., *Numerical integrators for stiff and highly oscillatory differential equations*, J. Math. Compt., 1980, 34(150), p. 373-390.

28. Gonzalez-Pinto S., Gonzalez-Concepcion C., Montijino,J. I., *Iterative schemes for Gauss methods*, J. Comput. Math. Applic.*,* 1994, 27(7), p. 67–81.

29. Schäfer E., Freiburi Br., *A new approach to explain the "High Irradiance Responses" of photomorphogenesis on the basis of phytochrome*, J. Math. Bio., 1975, 2, p. 41-56.