Malaysian Journal of Mathematical Sciences 12(2): 267–283 (2018)

MALAYSIAN JOURNAL OF MATHEMATICAL SCIENCES

Journal homepage: http://einspem.upm.edu.my/journal

An Explicit Time-Stepping Method based on Error Minimization for Solving Stiff System of Ordinary Differential Equations

Rahmanzadeh, M. *[1] and Barfeie, M.[2]

[1] *Department of Chemical Engineering, Sirjan Uuniversity of Technology, Iran*

[2] *Department of Mathematics and Computer Science, Sirjan University of Technology, Iran*

E-mail: rahmanzadeh.mostafa@gmail.com

* *Corresponding author*

Received: 20 May 2017

Accepted: 22 April 2018

ABSTRACT

In this paper, an explicit one step method is presented for numerical solution of stiff systems of ordinary differential equations (ODEs). In this method, the solution of the ODE is considered as a polynomial. The numerical approximation is obtained by minimizing an error function that is defined based on residual error. The stability region of the proposed method is obtained. In contrast to Runge-Kutta (RK) method that use Taylor polynomial, the method has larger stability region and a larger stable step size can be selected to obtain the numerical solutions. Numerical experiments show that the method is more accurate than explicit and implicit methods such as implicit Runge-Kutta Methods (IRK) of order eighth.

Keywords: Ordinary Differential Equations, Stability Region, Runge-Kutta Method, Stiff System, Explicit Method.

1. Introduction

Ordinary differential equations (ODE) appear in science and engineering such as geometry, chemical reaction kinetics, mechanics, population dynamics, electronic circuits, molecular dynamics, and many other areas of application Shampine and Corless (2000), Tasić and Mattheij (2004), Zeeshan and Majeed (2016). The time dependent partial differential equations (PDEs) also can be converted into a system of ODEs by spatial discretization. Numerical methods for solving ODEs are classified into boundary value problems (BVPs), and initial value problems (IVPs). In an initial value problem all of the conditions are specified at the initial point. The conditions of a boundary value problem are given in both initial and final points Ascher et al. (1995). We concentrate here on the case of IVP. However, the proposed method can be used for BVP.

Numerical method forms an important part of solving ordinary differential equations, most especially in cases where there is no closed form solution Ascher and Petzold (1995), Zheng et al. (2011), Zhu and Petzold (1997). There are numerous methods that produce numerical approximations to the solution of ODEs such as Taylor series method, Euler's method, improved Euler's method, Runge-Kutta methods, multi-step methods and the extrapolation method Brugnano and Magherini (2009), Liu (1999), Tasić and Mattheij (2005), Wang (2017).

The local truncation error of these methods depends on higher order derivatives of the solution Barrio (2005), Barrio et al. (2005), Voss and Muir (1999), Zhang (2002). In some differential equations, these derivatives can be absolutely large in value which require that the step size should be taken extremely small in order to achieve suitable accuracy. These types of equations are called stiff differential equation. Many researches have been deal with the development of accurate and stabile methods for solving ODEs Hairer and Wanner (1999), Ibáñez et al. (2011), Ibrahim et al. (2007).

In previous work we introduce RCW method for solving ODEs problem Rahmanzadeh et al. (2013). In this paper, we obtain the stability region of a RCW method and advance it to obtain the numerical solution of stiff systems of ordinary differential equations. In RCW method, the solution of ODE is considered as a polynomial of degree n. Then an error function based on residual error is defined. Finally the coefficients of the polynomial are obtained in such a way that the error becomes minimum. The coefficients of the proposed method are not fixed for all steps and in each step we need to minimize the error and obtain the new coefficients. In contrast to Runge-Kutta methods that use the Taylor polynomial of degree n to solve ODEs, the RCW method has larger stability region. Also numerical experiments show that the method is more accurate than explicit and implicit method such as IRK of order 8. The method is applied for linear and nonlinear ODEs.

2. Theory

A system of first order initial-value problems has the form:

$$\begin{aligned} \frac{dy_1}{dt} &= f_1(t, y_1, y_2, ..., y_m), \\ \frac{dy_2}{dt} &= f_2(t, y_1, y_2, ..., y_m), \\ &\vdots \\ \frac{dy_m}{dt} &= f_m(t, y_1, y_2, ..., y_m), \end{aligned} \tag{1}$$

for $t_0 \leq t \leq t_n$, with the following initial conditions:

$$y_i(t_0) = y_{i,0}; \quad i = 1, \cdots, m. \tag{2}$$

The object aim is to find m functions $y_1, y_2, ..., y_m$ that satisfy the differential equations together with all the initial conditions. There are a number of well-known numerical methods for approximating solutions of (1), such as the Runge-Kutta methods. In stiff ODEs the local truncation error of these methods is large and an extremely small time step h is needed to achieve suitable accuracy. Our goal is to introduce a method that minimizes the error and obtain the result with a larger value of h. To this goal, in the next section we extend the RCW method to solve (1).

Supposedly the initial solution at $t = t_0$ is given and we need to obtain the solution in the next time step $t = t_0 + h$. In the first place, the numerical solution is considered as a polynomial of degree n as follows:

$$y_i(t) \simeq \bar{y}_i(t) = \sum_{j=0}^{n} a_{i,j}(t - t_0)^j, \quad i = 1, ..., m. \tag{3}$$

We need to obtain the coefficient $a_{i,j}, i = 1, ..., m, j = 0, ..., n$. The $a_{i,0}$ and $a_{i,1}$ can be obtained using the initial conditions as follows:

$$\bar{y}_i(t_0) = y_{i,0}, \tag{4}$$
$$\bar{y}_i^{'}(t_0) = f_i(t_0, y_{1,0}, y_{2,0}, ..., y_{n,0}). \tag{5}$$

To find the other coefficients replace y with $\bar{y}$ in (1) and define the residual functions

$$R_i(t, a_{i,2}, a_{i,3}, ..., a_{i,n}) = \frac{d\bar{y}_i}{dt} - f_i(t, \bar{y}_1, \bar{y}_2, ..., \bar{y}_m), \tag{6}$$

then, the error function is introduced as follows:

$$e(a_{1,2}, ..., a_{1,n}, ..., a_{m,2}, ..., a_{m,n}) = \int_{t_0}^{t_0+h} \sum_{i=1}^{m} \left[R_i(t, a_{i,2}, a_{i,3}, ..., a_{i,n})\right]^2 dt. \tag{7}$$

In this paper, the Nelder-Mead simplex algorithm is used to find the coefficients and obtain the approximation solution at $t = t_0 + h$. The Nelder-Mead algorithm or simplex search algorithm, originally published in 1965 (Nelder and Mead, 1965), is one of the best known algorithms for multidimensional optimization. The method does not require any derivative information and is quite simple and very easy to use. In Matlab software the fminsearch command uses the Nelder-Mead simplex algorithm. We use the fminsearch command in order to minimize the functions e and to obtain the coefficients $a_{i,j}$. This procedure is used to obtain the solution for next time steps.

3. Stability analysis

In this section we obtain the stability region of the proposed method. Absolute stability is based on a test equation

$$y' = \lambda y, \quad y(x_0) = y_0, \tag{8}$$

with the exact solution

$$y(t) = y_0 e^{\lambda t}. \tag{9}$$

The behavior of exact solution for $Real(\lambda) < 0$ is that $|\lim_{t\to\infty} y(t)| = 0$. We want that the numerical solution has the same characteristics.

Definition 3.1. *The region of absolute stability of the ODE method is the region in the complex plane R, such that if $h\lambda \in R$ then the numerical solution $y(t) \to 0$ as $t \to \infty$ for all initial values y_0.*

The region of absolute stability is a property of the method and is useful for estimating the timestep size required to obtain qualitatively correct solutions. We obtain the stability region of the proposed method for $n = 1, 2, 3, 4$ and compare the results with the standard Runge-Kutta methods of orders 1 through 4.

To this goal, we apply the RCW method with $n = 1, 2, 3, 4$ to the model problem (8), and obtain the resulting difference equations. For $n = 3$ we have

$$\bar{y}(t) = a_0 + a_1(t - t_n) + a_2(t - t_n)^2 + a_3(t - t_n)^3. \tag{10}$$

The a_0 and a_1 can be obtained using the conditions:

$$\bar{y}(t_n) = y_n \rightarrow a_0 = y_n \tag{11}$$
$$\bar{y}'(t_n) = \lambda y_n \rightarrow a_1 = \lambda y_n. \tag{12}$$

To find a_2 and a_3 we need to minimize the error as comes next

$$e(a_2, a_3) = \int_{t_n}^{t_n+h} [R(t, a_2, a_3)]^2 dt; \tag{13}$$

where

$$R(t, a_2, a_3) = \bar{y}' - \lambda \bar{y}. \tag{14}$$

The minimum of the equation (13) is obtained by solving the following equations

$$\frac{\partial e(a_2, a_3)}{\partial a_1} = 0, \tag{15}$$
$$\frac{\partial e(a_2, a_3)}{\partial a_2} = 0. \tag{16}$$

We have

$$a_2 = -\frac{15\left(2h^3\lambda^5 - 17h^2\lambda^4 + 56h\lambda^3 - 63\lambda^2\right)}{2\left(5h^4\lambda^4 - 60h^3\lambda^3 + 318h^2\lambda^2 - 840h\lambda + 945\right)} y_n, \tag{17}$$
$$a_3 = \frac{7\left(3h^2\lambda^5 - 20h\lambda^4 + 45\lambda^3\right)}{2\left(5h^4\lambda^4 - 60h^3\lambda^3 + 318h^2\lambda^2 - 840h\lambda + 945\right)} y_n. \tag{18}$$

Put $\lambda h = z$ and $t = t_{n+1}$, we have

$$y_{n+1} = p_3(z) y_n, \tag{19}$$

where

$$\begin{aligned} p_3(z) =& 1 + z + \frac{7\left(3z^5 - 20z^4 + 45z^3\right)}{2\left(5z^4 - 60z^3 + 318z^2 - 840z + 945\right)}, \\ & - \frac{15\left(2z^5 - 17z^4 + 56z^3 - 63z^2\right)}{2\left(5z^4 - 60z^3 + 318z^2 - 840z + 945\right)}. \end{aligned}$$

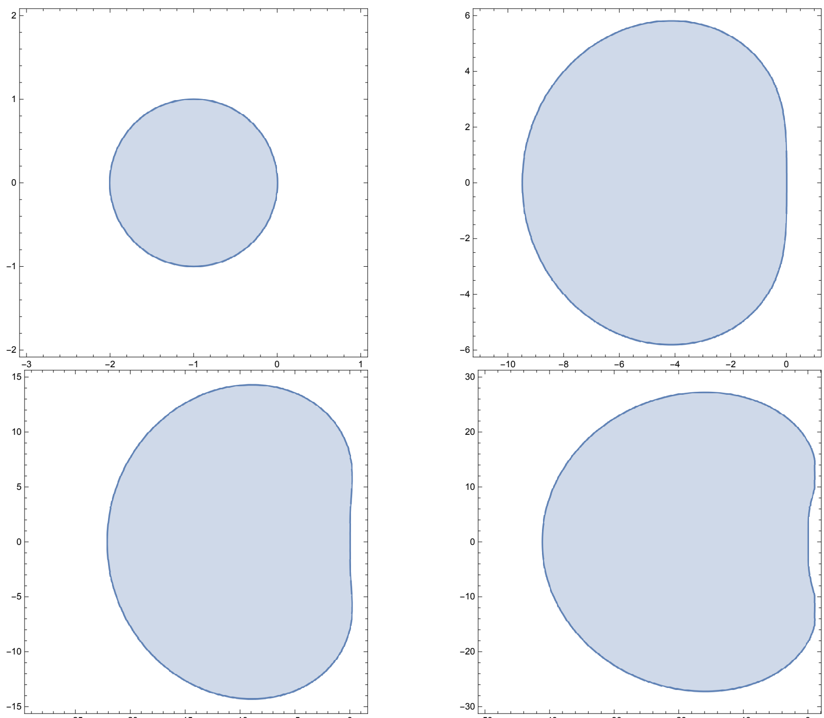

Figure 1: Stability region of RCW method for $n = 1$ (top, left), $n = 2$ (top, right),$n = 3$ (bottom left) and $n = 4$ (bottom, right).

Thus, the region of absolute stability for the RCW method with $n = 3$ is defined by the region in the complex plane such that $|p_3(z)| \le 1$. Figures 1 and 2 show the stability regions of RCW method for $n = 1, 2, 3, 4$ and the RK of orders 1 through 4, respectively. As Figures 3-4 show the RCW has larger stability region and gives a more stable results than that of the RK for ODEs.

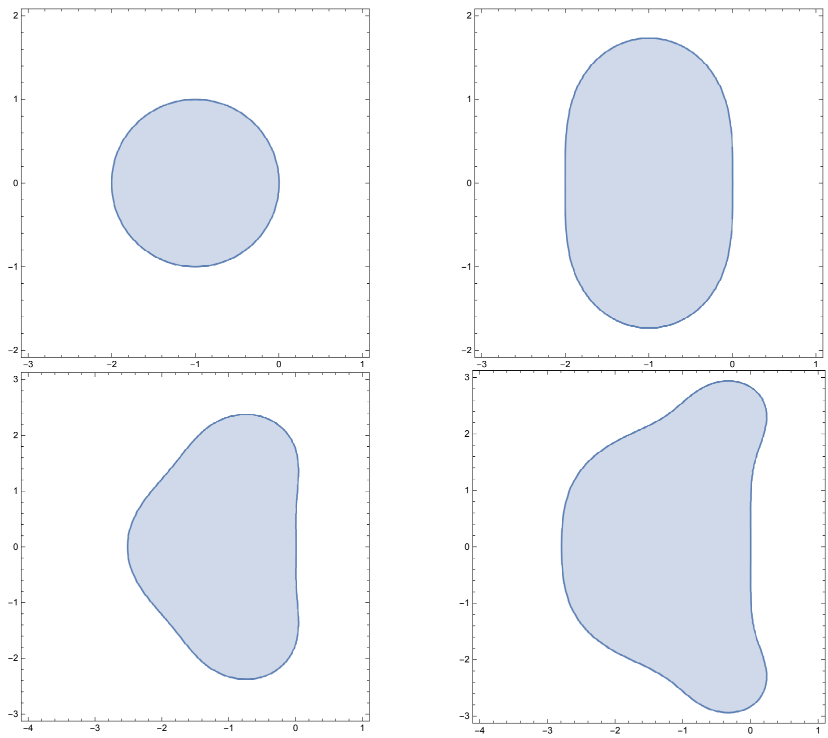

Figure 2: Stability region of RK method of order 1 (top, left), order 2 (top, right), order 3 (bottom left) and order 4 (bottom, right).

4. Numerical examples

In this section, the proposed method is applied to obtain the numerical solution of some linear and nonlinear test problems. In RCW method the numerical solution is considered as a polynomial of degree n that is the best polynomial approximation. Denote y_{RCW} and y_{RK} the approximation solution that is obtained by RK4 and RCW method for $n = 4$, we have

$$|y - y_{RCW}| \leq |y - y_{RK4}| \leq Ch^4. \tag{20}$$

Thus, it is expected that the rate of convergence of RCW method is more better than RK methods.

Example 4.1. *Consider the following systems of first-order differential equations Thohura and Rahman (2013).*

$$y_1^{'} = 9y_1 + 24y_2 - 5\cos(t) - \frac{1}{3}\sin(t), \quad y_1(0) = \frac{4}{3}, \tag{21}$$

$$y_2^{'} = -24y_1 - 51y_2 - 9\cos(t) - \frac{1}{3}\sin(t), \quad y_2(0) = \frac{2}{3}. \tag{22}$$

We use the RK, IRK and RCW methods to approximate the solutions, and compare the results to the exact solutions. The exact solution of this example is

$$y_1(t) = 2e^{-3t} - e^{-39t} + \frac{1}{3}\cos(t), \tag{23}$$

$$y_2(t) = -e^{-3t} + 2e^{-39t} + \frac{1}{3}\cos(t) \tag{24}$$

Figure 3 shows the exact solution of this example that has rapid variation in some part of the domain.

In the first place, we approximate solution by a polynomial with degree $n = 6$, as follows:

$$\bar{y}_i(t) = \sum_{j=0}^{i=6} a_{i,j}(t - t_0)^j, \quad i = 1, 2. \tag{25}$$

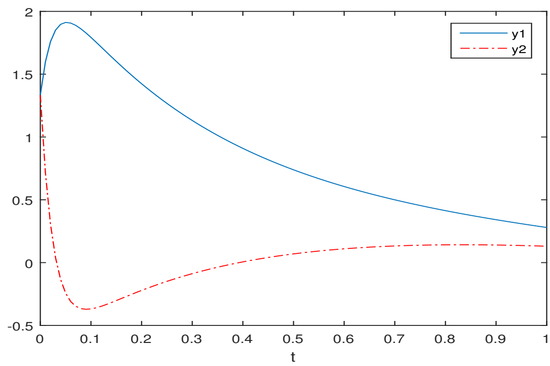

Figure 3: The exact solutions y_1 and y_2 in Example 1.

The coefficients $a_{1,0}$ and $a_{2,0}$ are determined by using the initial conditions in what follows:

$$\bar{y}_1(t_0) = a_{1,0} = y_{1,0} = \frac{4}{3} \quad , \bar{y}_2(t_0) = a_{2,0} = y_{2,0} = \frac{2}{3} \tag{26}$$

$$\bar{y}_1^{'}(t_0) = a_{1,1} = 2e^{-3t_0} - e^{-39t_0} + \frac{1}{3}\cos(t_0), \tag{27}$$

$$\bar{y}_2^{'}(t_0) = a_{2,1} = -e^{-3t_0} + 2e^{-39t_0} + \frac{1}{3}\cos(t_0). \tag{28}$$

In order to obtain other coefficients, we substitute the approximate values $\bar{y}_i, \;\; i = 1, 2$ to equations (21) and (22) and define the residual errors as follows:

$$R_1(t, a_{1,2}, ..., a_{1,6}) = \bar{y}_1^{'} - \left(9\bar{y}_1 + 24\bar{y}_2 - 5\cos(t) - \frac{1}{3}\sin(t)\right), \tag{29}$$

$$R_2(t, a_{2,2}, ..., a_{2,6}) = \bar{y}_2^{'} - \left(-24\bar{y}_1 - 51\bar{y}_2 - 9\cos(t) - \frac{1}{3}\sin(t)\right). \tag{30}$$

The residual errors are used to define an error function as follows:

$$e(a_{1,2}, ..., a_{1,6}, a_{2,2}, ..., a_{2,6}) = \int_0^{h=0.05} \left(R_1^2 + R_2^2\right) dt, \tag{31}$$

By minimizing equation (31) the coefficient $a_{1,2}, ..., a_{1,6}, a_{2,2}, ..., a_{2,6}$ are obtained. In order to obtain the solution at the next step we should replace t_0 by $t_0 + h$ and repeat the procedure. Table (1) show the coefficient for four successive steps.

Table 1: The coefficient $a_{i,j}$ for four successive steps in Example 1.

	a_0	a_1	a_2	a_3	a_4	a_5	a_6
$y_1(N=1)$	1.333333	33	-748.018	9460.765	-79146.8	399907	-905341
$y_2(N=1)$	0.666667	-75	1506.119	-18743.4	154160.7	-761231	1679820
$y_1(N=2)$	1.792971	-3.68861	-8.87429	187.5031	-1638.82	8399.098	-19159.1
$y_2(N=2)$	-1.03196	0.676999	27.25628	-366.559	2899.694	-13307.2	26721.09
$y_1(N=3)$	1.42382	-3.34287	4.462877	-0.84469	-33.8231	214.048	-560.121
$y_2(N=3)$	-0.87464	1.680586	-1.68556	-5.34721	64.17235	-338.161	781.1712
$y_1(N=4)$	1.131515	-2.53741	3.465085	-1.79122	-37.6392	380.4909	-1312.14
$y_2(N=4)$	-0.72497	1.317462	-1.63775	0.523073	23.50436	-217.263	707.0921

Figure 4 shows the residual functions R_1 and R_2 versus time for $h = 0.05$. The residual errors do not increase with time and this show the stability and reliability of this method.

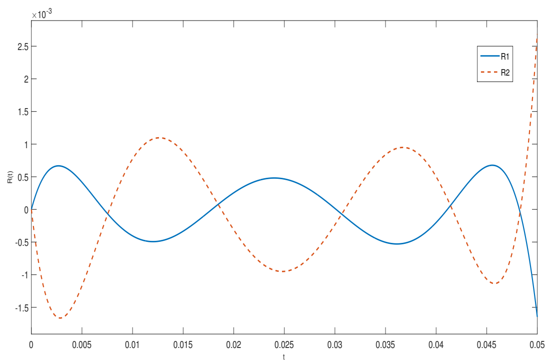

Figure 4: The residual functions R_1 and R_2 versus time in Example 1.

Table 2 shows the numerical solutions of $y_1(x)$ and $y_2(x)$ obtained from the RCW, RK and four stage implicit Runge-kutta methods methods. Also the analytical solution and the difference between the exact value and the approximation value are given. Table 2 shows the RCW method is more accurate than IRK and RK methods. Also, in the case of using RCW method, it is possible to obtain the numerical solution with larger values of h. For example, neither of the IRK and RK methods can obtain the numerical solution with $h = 0.3$.

Table 2: The numerical solutions of $y_1(x)$ and $y_2(x)$ in Example 1.

t=.05	$y_1(analytical) = 1.912058635$		$y_2(analytical) = -0.909076587$	
$\bar{y}_1(t)$	$\bar{y}_2(t)$	error_1	error_2	method
1.912058241	-0.909076888	3.93E-07	3.01E-07	RCW ($h = 0.05$)
1.912056086	-0.909071488	2.55E-06	-5.10E-06	IRK ($s = 4$)($h = 0.05$)
1.736416379	-0.557790252	1.76E-01	-0.3512863	RK
t=.1	$y_1(analytical) = 1.793062585$		$y_2(analytical) = -1.032002453$	
1.792990795	-1.031971618	7.18E-05	-3.08E-05	RCW ($h = 0.1$)
1.792806927	-1.031491136	2.55658E-04	-5.113E-04	IRK ($h = 0.1$)
-2.645181254	7.844542146	4.44E+00	-8.8765446	Rk
t=.15	$y_1(analytical) = 1.601966763$		$y_2(analytical) = -0.961458713$	
1.601164464	-0.961068144	8.023E-04	-3.906E-04	RCW ($h = 0.15$)
1.599648625	-0.956822437	2.318138E-03	-4.6363E-03	IRK ($h = 0.15$)
t=.2	$y_1(analytical) = 1.423902396$		$y_2(analytical) = -0.874681025$	
1.420396706	-0.872963837	3.50569E-03	-1.7172E-03	RCW ($h = 0.2$)
5.17E+78	-1.03E+79	-5.17E+78	1.03E+79	IRK ($h = 0.2$)
t=.25	$y_1(analytical) = 1.267645618$		$y_2(analytical) = -0.795220771$	
1.258154617	-0.790604594	9.491001E-03	-4.6162E-03	RCW ($h = 0.25$)
t=.3	$y_1(analytical) = 1.131576522$		$y_2(analytical) = -0.724998568$	
1.112233204	-0.715665871	1.9343318E-02	-9.3327E-03	RCW ($h = 0.3$)

In order to show the stability of this method for larger value of time step, this example is solved for $h = 0.2$ on $[0, 2]$. The RK and IRK methods cannot obtain the solution for $h = 0.2$ on $[0, 2]$. Table 3 show the numerical solution for different values of t and $h = 0.2$ on $[0, 2]$. As table shows the error is reduced with time.

Table 3: The numerical solutions for different values of t and $h = .2$ in Example 1.

t	$\bar{y}_1(t)$	$\bar{y}_2(t)$	error_1	error_2
0.2	1.420396706	-0.872963837	3.50569E-03	-1.7172E-03
0.4	0.907470234	-0.607245059	1.938353E-03	-9.691E-04
0.6	0.604645846	-0.439878859	1.063802E-03	-5.319E-04
0.8	0.413087649	-0.322661609	5.83827E-04	-2.919E-04
1	0.279354494	-0.229727631	3.20411E-04	-1.602E-04
1.2	0.175257518	-0.148021718	1.75845E-04	-8.79E-05
1.4	8.6550362E-02	-7.1603038E-02	9.65E-05	-4.83E-05
1.6	6.673356E-03	1.529909E-03	5.30E-05	-2.65E-05
1.8	-6.6729937E-02	7.1231984E-02	2.91E-05	-1.45E-05
2	-0.13377406	0.136244836	1.60E-05	-7.98E-06

Example 4.2. *Consider the following systems of first order differential equations*

$$y_1' = y_1 + 100y_2 - \sin(100t), \quad y_1(0) = 0, \tag{32}$$

$$y_2' = -100y_1 - 5y_2 + 5\cos(100t), \quad y_2(0) = 1. \tag{33}$$

The exact solution of this example is $y_1(t) = \sin(100t), y_2(t) = \cos(100t)$.

Figure 5 shows the exact solution of this example. The function y_1 and y_2 in this example have rapid oscillations.

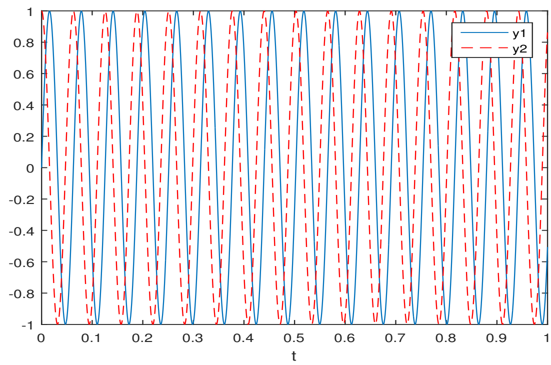

Figure 5: The exact solutions y_1 and y_2 in Example 2.

We use the RCW and IRK methods to solve this example. In addition, we approximate the solution using a polynomial of degree $n = 6$, as follows:

$$\bar{y}_i(t) = \sum_{j=0}^{j=6} a_{i,j}(t - t_0)^j, \quad i = 1, 2. \tag{34}$$

Like Example 1 the coefficients $a_{1,0}$ and $a_{2,0}$ are determined by using the initial conditions in what follows:

$$\bar{y}_1(t_0) = a_{1,0} = y_{1,0} = 0 \quad , \bar{y}_2(t_0) = a_{2,0} = y_{2,0} = 1, \tag{35}$$

$$\bar{y}_1^{'}(t_0) = a_{1,1} = y_{1,0} + 100y_{2,0} - \sin(100t_0), \tag{36}$$

$$\bar{y}_2^{'}(t_0) = a_{2,1} = -100y_{1,0} - 5y_{2,0} + 5\cos(100t_0). \tag{37}$$

Substitute y_i in (32) and (33) with $\bar{y}_i$ in equations (34) and define the residual functions as follows:

$$R_1(t, a_{1,2}, ..., a_{1,6}) = \bar{y}_1^{'} - (\bar{y}_1 + 100\bar{y}_2 - \sin(100t)), \tag{38}$$

$$R_2(t, a_{2,2}, ..., a_{2,6}) = \bar{y}_2^{'} - (-100\bar{y}_1 - 5\bar{y}_2 + 5\cos(100t)). \tag{39}$$

Then, the error function is:

$$e(a_{1,2}, ..., a_{1,6}, a_{2,2}, ..., a_{2,6}) = \int_0^{h=0.1} \left(R_1^2 + R_2^2\right) dt. \tag{40}$$

As previous example by minimizing the error in (40) the numerical solutions are obtained. In this example the error_1 and error_2 are defined as:

$$\text{error}_i = \bar{y}_i - y_i, \quad i = 1, 2. \tag{41}$$

The graph of error_1 and error_2 are plotted in Figures (6) and (7) respectively. The results show that the RCW method is more accurate.

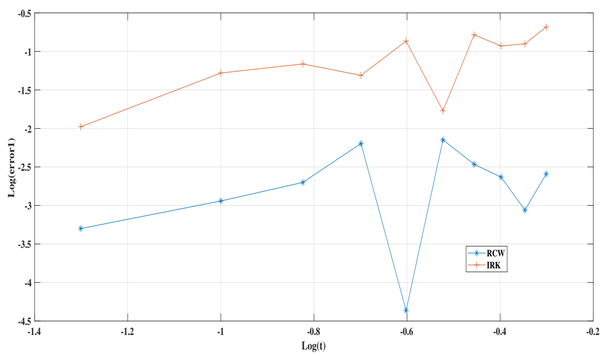

Figure 6: The graph of error_1 for RCW and IRK method in Example 2

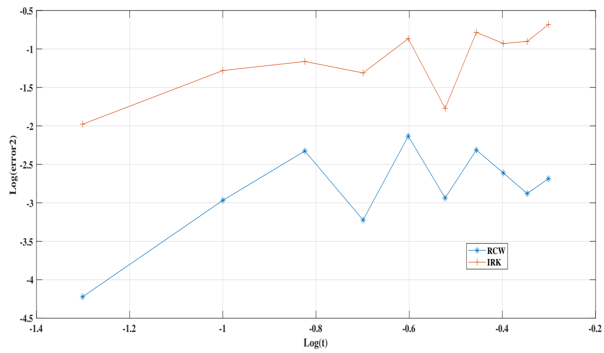

Figure 7: The graph of error_2 for RCW and IRK method in Example 2

Example 4.3. *In order to show the stability of method in this example we consider a nonlinear ODE as follows.*

$$y_1^{'} = 3\frac{y_1 y_2}{t^3} + y_1^2 - e^{6t}, \quad y_1(1) = e^3, \tag{42}$$

$$y_2^{'} = 3y_2^{\frac{2}{3}} + \ln(y_1) - 3t, \quad y_2(1) = 1. \tag{43}$$

The exact solution of this example is $y_1(t) = e^{3t},\ y_2(t) = t^3$.

In this test, we use the RCW and IRK methods with $h = 0.05$. As we have done in Examples 2 and 3 the approximate solution is considered as a polynomial of degree $n = 6$. Table 4 shows the error for RCW method and IRK method. Results show that the error of RCW is less than the error of IRK.

Table 4: The numerical solutions for different values of t and $h = 0.05$ in Example 3.

t	error_1 (RCW)	error_1 (IRK)	error_2 (RCW)	error_2 (IRK)
1.05	-3.908E-14	7.42E-09	-3.92E-13	7.69E-12
1.1	-1.3641E-10	1.27E-07	-4.45E-12	9.55E-11
1.15	-2.8038E-09	2.81E-06	2.21E-13	1.54E-09
1.20	-9.7509E-08	9.74E-05	-3.00E-11	3.90E-08
1.25	-5.9412E-06	5.76531E-03	-1.20E-09	1.69E-06
1.30	-6.7133E-04	0.60453166	2.61E-07	1.30825E-04

Example 4.4. *Consider the following systems of first-order differential equations*

$$y_1^{'} = -1002y_1 + 1000y_2^2 \quad , y_1(0) = 1, \tag{44}$$

$$y_2^{'} = y_1 - y_2(1 + y_2) \quad , y_2(0) = 1. \tag{45}$$

The exact solutions of this example are $y_1(t) = 2e^{-2t}$ and $y_2(t) = -e^{-t}$. We use the RCW method with $n = 4$ and RK4 method to approximate the solutions. Table (5) show the numerical results. As Table (5) show the RCW is more accurate than the new method that is presented in Ascher et al. (1995), Chen et al. (2015).

Table 5: The numerical solutions for different values of t and $h = 0.05$ in Example 3.

h	t	error (RCW)	error (Chen et al. (2015))	error (Guzel and Bayram (2005))
0.05	0.05	-3.28E-09	2.15E-8	8.2E-8
0.05	0.05	-1.63E-9	2.18E-8	2.58E-9
0.20	0.20	-3.35E-06	1.15E-5	7.995E-5
0.20	0.20	-1.86E-06	5.25E-6	2.58E-6

5. Conclusion

An explicit one step method was presented for solving stiff system of ordinary differential equations. The stability region of the proposed method was obtained. In contrast to the methods that use polynomials to approximate numerical solution such as Runge-Kutta methods, the proposed method has larger stability region and accordingly a larger stable step size can be selected to obtain the numerical solutions. The method is applied for some test problems. Numerical experiments show that the method is more accurate than explicit and implicit methods such as implicit Runge-Kutta methods.

References

Ascher, U. M., Mattheij, R. M., and Russell, R. D. (1995). *Numerical solution of boundary value problems for ordinary differential equations*. SIAM.

Ascher, U. M. and Petzold, L. R. (1995). The numerical solution of delay-differential-algebraic equations of retarded and neutral type. *SIAM Journal on Numerical Analysis*, **32**:1635–1657.

Barrio, R. (2005). Performance of the taylor series method for odes/daes. *Applied Mathematics and Computation*, **163**:525–545.

Barrio, R., Blesa, F., and Lara, M. (2005). Vsvo formulation of the taylor method for the numerical solution of odes. *Computers & mathematics with Applications*, **50**:93–111.

Brugnano, L. and Magherini, C. (2009). Recent advances in linear analysis of convergence for splittings for solving ode problems. *Applied Numerical Mathematics*, **59**:542–557.

Chen, W., Xiao, H., Li, H., and Chen, L. (2015). Explicit appropriate basis function method for numerical solution of stiff systems. *Annals of Nuclear Energy*, **75**:353–357.

Guzel, N. and Bayram, M. (2005). On the numerical solution of stiff systems. *Applied mathematics and computation*, **170**:230–236.

Hairer, E. and Wanner, G. (1999). Stiff differential equations solved by radau methods. *Journal of Computational and Applied Mathematics*, **111**:93–111.

Ibáñez, J. J., Hernández, V., Ruiz, P. A., and Arias, E. (2011). A piecewise-linearized algorithm based on the krylov subspace for solving stiff odes. *Journal of Computational and Applied Mathematics*, **235**:1798–1804.

Ibrahim, Z. B., Othman, K. I., and Suleiman, M. (2007). Implicit r-point block backward differentiation formula for solving first-order stiff odes. *Applied Mathematics and Computation*, **186**:558–565.

Liu, Y. (1999). Numerical solution of implicit neutral functional differential equations. *SIAM journal on numerical analysis*, **36**:516–528.

Rahmanzadeh, M., Cai, L., and White, R. E. (2013). A new method for solving initial value problems. *Computers & Chemical Engineering*, **58**:33–39.

Shampine, L. F. and Corless, R. M. (2000). Initial value problems for odes in problem solving environments. *Journal of Computational and Applied Mathematics*, **125**:31–40.

Tasić, B. and Mattheij, R. M. M. (2004). An explicit method for solving flows of ode. *Applied mathematics and computation*, **156**:633–652.

Tasić, B. and Mattheij, R. M. M. (2005). Explicitly solving vectorial odes. *Applied mathematics and computation*, **164**:913–933.

Thohura, S. and Rahman, A. (2013). Numerical approach for solving stiff differential equations: A comparative study. *Global Journal of Human-Social Science Research*, **13**:1–13.

Voss, D. A. and Muir, P. H. (1999). Mono-implicit runge-kutta schemes for the parallel solution of initial value odes. *Journal of computational and applied mathematics*, **102**:235–252.

Wang, W. (2017). Uniform ultimate boundedness of numerical solutions to nonlinear neutral delay differential equations. *Journal of Computational and Applied Mathematics*, **309**:132–144.

Zeeshan, A. and Majeed, A. (2016). Heat transfer analysis of jeffery fluid flow over a stretching sheet with suction/injection and magnetic dipole effect. *Alexandria Engineering Journal*, **55**:2171–2181.

Zhang, C. (2002). Nonlinear stability of natural runge-kutta methods for neutral delay differential equations. *Journal of Computational Mathematics*, **20**:583–590.

Zheng, Q., Zhao, P., and Huang, F. (2011). A family of fourth-order steffensen-type methods with the applications on solving nonlinear odes. *Applied Mathematics and Computation*, **217**:8196–8203.

Zhu, W. and Petzold, L. R. (1997). Asymptotic stability of linear delay differential-algebraic equations and numerical methods. *Applied Numerical Mathematics*, **24**:247–264.