Mathematical Theory and Modeling www.iiste.org
ISSN 2224-5804 (Paper) ISSN 2225-0522 (Online)
Vol.2, No.4, 2012

A Family of Implicit Higher Order Methods for the Numerical Integration of Second Order Differential Equations

Owolabi Kolade Matthew

Department of Mathematics, University of Western Cape, 7535, Bellville, South Africa

*E-mail of the corresponding author: kowolabi@uwc.ac.za; kmowolabi2@gmail.com

Abstract

A family of higher order implicit methods with *k* steps is constructed, which exactly integrate the initial value problems of second order ordinary differential equations directly without reformulation to first order systems. Implicit methods with step numbers $k \in \{2,3,...,6\}$ are considered. For these methods, a study of local truncation error is made with their basic properties. Error and step length control based on Richardson extrapolation technique is carried out. Illustrative examples are solved with the aid of MATLAB package. Findings from the analysis of the basic properties of the methods show that they are consistent, symmetric and zero-stable. The results obtained from numerical examples show that these methods are much more efficient and accurate on comparison. These methods are preferable to some existing methods owing to the fact that they are efficient and simple in terms of derivation and computation

Keywords: Error constant, implicit methods, Order of accuracy, Zero-Stability, Symmetry

1. Introduction

In the last decade, there has been much research activity in the area of numerical solution of higher order linear and nonlinear initial value problems of ordinary differential equations of the form

$$f(t, y, y, ..., y^{(m)}) = 0, y^{(m-1)}(t_0) = \eta_{m-1}$$

$$m = 1,2,.., \{t, y\} \in \Re^n$$

(1)

which are of great interest to Scientists and Engineers. The result of this activity are methods which can be applied to many problems in celestial and quantum mechanics, nuclear and theoretical physics, astrophysics, quantum chemistry, molecular dynamics and transverse motion to mention a few. In literature, most models encountered are often reduced to first order systems of the form

$$y' = f(t, y), \ y(t_0) = y_0, \ t \in [a, b]$$

(2)

before numerical solution is sought [see for instance, Abhulimen and Otunta (2006), Ademiluyi and Kayode (2001), Awoyemi (2005), Chan et al. (2004)].

In this study, our interest is to develop a class of k-steps linear multistep methods for integration of general second order problems without reformulation to systems of first order. We shall be concerned primarily with differential equations of the type

$$y'' = f(t, y, y'), y^{(m)}(t_0) = \eta_m, \; t \in [a, b], \; f(t, y, y') \in \Re^n, \; m = 0,1$$

(3)

Theorem 1 *If f(t,y), f:R x R → R is defined and continuous on all t∈[a, b] and* $-\infty < y < \infty$ *and a constant L exist such that*

$$|f(t, y) - f(t, y^*)| < L|y - y^*|$$

(4)

for every pair (t, y) and (t, y) in the quoted region then, for any y_0∈R the stated initial value problem admits a unique solution which is continuous and differentiable on* [*a*, *b*].

Efforts are made to develop a class of implicit schemes of higher step-numbers with reduced functions evaluation for direct integration of problem (3) for k=2,3,... ,6.

The remainder of the paper is organized in the following way. Under materials and methods, construction of the schemes for approximating the solutions of (3) is presented with the analysis of their basic properties for proper implementation. Some sample problems coded in MATLAB are equally considered. Finally, some concluding comments are made to justify the obtainable results and suitability of the proposed schemes on comparisons.

2. Materials and methods

2.1 *Construction of the schemes*: The proposed numerical method of consideration for direct integration of general second order differential equations of type (3) is of the form

$$y_{n+k} = \alpha_0 y_n + \alpha_1 y_{n+1} + \ldots + \alpha_{k-1} y_{n+k-1} + h^2(\beta_0 f_n + \beta_1 f_{n+1} + \ldots + \beta_k f_{n+k}) \quad ,$$

(5)

taken from the classical K-step method with the algorithm

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h^2 \sum_{j=0}^{k} \beta_j f_{n+j}, \; n = 0,1,\ldots$$

(6)

where y_{n+j} is an approximation to $y(x_{n+j})$ and fn+j =f(x_{n+j}, y_{n+j}, y'_{n+j}). The coefficients α_j and β_j are constants which do not depend on n subject to the conditions

$$\alpha_k = 1, \; |\alpha_0| + |\beta_0| \neq 0$$

are determined to ensure that the methods are symmetric, consistent and zero stable. Also, method (4) is implicit since $\beta_k \neq 0$.

The values of these coefficients are determined from the local truncation error (*lte*)

$$T_{n+k} = y_{n+k} - [\alpha_0 y_n + \alpha_1 y_{n+1} + \ldots + \alpha_{k-1} y_{n+k-1} + h^2(\beta_0 f_n + \beta_1 f_{n+1} + \ldots + \beta_k f_{n+k})]$$

(7)

generated by one-step application of (5) for numerical solution of (3). Clearly, accuracy of these schemes depend on the real constants α_j and β_j. In attempt to obtain the numerical values of these constants, the following steps were adopted;

Taylor series expansion of y_{n+k}, $y_{n+1}, y_{n+2}, \ldots, y_{n+k-1}$ and $f_{n+1}, f_{n+2}, \ldots, f_{n+k}$ about the point (t_n, y_n) yields

$$T_{n+k} = y_n + (kh)y_n^{(1)} + \frac{(kh)^2}{2!}y_n^{(2)} + \ldots + \frac{(kh)^p}{p!}y_n^{(p)} + 0(h^{(p+1)})$$

$$-\sum_{j=0}^{k-1}\alpha_j\{y_n + (jh)y_n^{(1)} + \frac{(jh)^2}{2!}y_n^{(2)} + \ldots + \frac{(jh)^p}{p!}y_n^{(p)} + 0\frac{(jh)^{p+1}}{(p+1)!}y_n^{(p+1)}\}$$

$$-h^2\sum_{j=0}^{k}\beta_j\left\{y_n^{(2)} + (jh)y_n^{(3)} + \frac{(jh)^2}{2!}y_n^{(4)} + \ldots + \frac{(jh)^{p-2}}{(p-2)!}y_n^{(p)} + 0\frac{(jh)^{p-1}}{(p-1)!}y_n^{(p+1)}\right\}$$

(8)

Terms in equal powers of h are collected to have

$$T_{n+k} = \left(1 - \sum_{j=0}^{k-1}\alpha_j\right)y_n + \left(k - \sum_{j=0}^{k-1} j\alpha_j\right)hy_n^{(1)} + \left(\frac{k^2}{2!} - \sum_{j=0}^{k-1}\frac{(j)^2}{2!}\alpha_j - \sum_{j=0}^{k}\beta_j\right)h^2 y_n^{(2)} +$$

$$\left(\frac{k^3}{3!} - \sum_{j=0}^{k-1}\frac{(j)^3}{3!}\alpha_j - \sum_{j=0}^{k} j\beta_j\right)h^3 y_n^{(3)} + \ldots +$$

$$\left(\frac{k^p}{p!} - \sum_{j=0}^{k-1}\frac{(j)^p}{p!}\alpha_j - \sum_{j=0}^{k}\frac{j^{p-2}}{(p-2)}\beta_j\right)h^p y_n^{(p)} + 0\left(h^{p+1}\right)$$

(9)

Accuracy of order **p** is imposed on T_{n+k} to obtain $\mathbf{C_i=0,\ 0 \le i \le p}$. Setting k=2(3)6, j=0(1)6 in equation (9), the obtainable algebraic system of equations are solved with **MATLAB** in the form *AX=B* for various step-numbers to obtain coefficients of the methods parameters displayed in Table 0.

Using the information in Table 0 for k=2(3)6 in (5), we have the following implicit schemes

$$y_{n+2} = 2y_{n+1} - y_n + \frac{h^2}{12}(f_{n+2} + 10f_{n+1} + f_n)$$

(10)

P=4, $C_{p+2} \approx -4.1667\text{x}10^{-3}$

which coincides with Numerov's method of Lambert (see for more details in [7])

$$y_{n+3} = 3y_{n+2} - 3y_{n+1} + y_n + \frac{h^2}{12}(f_{n+3} + 9f_{n+2} - 9f_{n+1} - f_n)$$

(11)

P=5, $C_{p+2} \approx -4.1667\text{x}10^{-3}$

$$y_{n+4} = 4y_{n+3} - 6y_{n+2} + 4y_{n+1} - y_n + \frac{h^2}{12}(f_{n+4} + 8f_{n+3} - 18f_{n+2} + 8f_{n+1} - f_n)$$

(12)

P=6, $C_{p+2} \approx -4.1667\text{x}10^{-3}$

$$y_{n+5} = 5y_{n+4} - 10y_{n+3} + 10y_{n+2} - 5y_{n+1} + y_n + \frac{h^2}{12}(f_{n+5} + 7f_{n+4} - 26f_{n+3} + 26f_{n+2} - 7f_{n+1} - f_n)$$

(13)

P=7, $C_{p+2} \approx -4.1667\text{x}10^{-3}$

$$y_{n+6} = 6y_{n+5} - 15y_{n+4} + 20y_{n+3} - 15y_{n+2} + 6y_{n+1} - y_n + \frac{h^2}{12}(f_{n+6} + 6f_{n+5} - 33f_{n+4} + 52f_{n+3} - 33f_{n+2} + 6f_{n+1} + f_n)$$

(14)

P=7, $C_{p+2} \approx -4.1667\text{x}10^{-3}$

2.2 Analysis of the basic properties of methods (10),...,(14).

To justify the accuracy and applicability of our proposed methods, we need to examine their basic properties which include order of accuracy, error constant, symmetry, consistency and zero stability.

Order of accuracy and error constant:

Definition1. *Linear multistep methods (10)-(14) are said to be of order p, if p is the largest positive integer for which $C_0 = C_1 = \ldots = C_p = C_{p+1} = 0$ but $C_{p+2} \neq 0$.* Hence, our methods are of orders p =4(5)8 with principal truncation error $C_{p+2} \approx -4.1667\text{x}10^{-3}$.

Symmetry: According to Lambert (1976), a class of linear multistep methods (10)-(14) is symmetric if

$$\alpha_j = \alpha_{k-j}$$

$$\beta_j = \beta_{k-j} \qquad \text{, j=0(1) k/2, for even k}$$

(15)

$$\alpha_j = -\alpha_{k-j}$$

$$\beta_j = -\beta_{k-j} \qquad \text{, j=0(1) k, for odd k}$$

(16)

Consistency

Deinition2: *A linear multistep method is consistent if;*

a). It has order p≥1

b). $\sum_{j=0}^{k} \alpha_j = 0$

c). $\rho(r) = \rho'(r) = 0$

d). $\rho''(r) = 2!\delta(r)$

where $\rho(r)$ and $\delta(r)$ are the first and second characteristic polynomials of our methods. Obviously, conditions above are satisfied using the information as contained in Table1 for k=2(3)6.

Zero stability

Definition3: *A linear multistep method is said to be zero-stable if no root ρ(r) has modulus greater than one (that is, if all roots of ρ(r) lie in or on the unit circle).* A numerical solution to class of system (3) is stable if the difference between the numerical and theoretical solutions can be made as small as possible. Hence, methods (10)-(14) are found to be zero-stable since none of their roots has modulus greater than one.

Convergence

Definition 4: *The method defined by (5) is said to be convergent if, for all initial value problems satisfying the hypotheses of the theorem 1, the fixed station limit*

$$h \xrightarrow{\lim} 0 \qquad y_{n_{\max}} = y(t)$$
$$t = a + n_{\max} h$$

(17)

*holds for all t*Є [*a*, *b*] *and for all solutions of the equation (5) satisfying starting conditions*

$y_j = \phi_j(h),\ 0 < j < k-1$,$\lim_{h\to 0}$ $\phi_j(h)$ $=y_0$

(18).

Theorem 2 *The necessary and sufficient conditions for the method (5) or ((10)-(14)) to be convergent is that be both consistent and zero-stable.*

3. Numerical Experiments: The discrete methods described above are implicit in nature, meaning that they require some starting values before they can be implemented. Starting values for y_{n+j}, y'_{n+j}, $2\le j\le 6$ are predicted using Taylor series up to the order of each scheme. For a numerical solution we introduce a partition of [*a*, *b*]: $t_0=a$, $tn=t_0+nh$, $(n=1, 2, \ldots, n_{max})$such that $t_{nmax} =b$ which means that n_{max} and h are linked, $h=(b-a)/n_{max}$.

Accuracy of our methods are demonstrated with five sample initial value problems ranging from general and special, linear and nonlinear and inhomogeneous second order differential equations.

Example 1:

$$y'' - (y')^2 = 0,\ \ y(0) = 1,\ y'(0) = 0.5,\ h = 0.003125$$

Analytical solution: $y(t) = 1 + \frac{1}{2}\ln\left(\frac{2+t}{2-t}\right)$

Example 2:

$$y'' + y = 0, \quad y(0) = 0, \; y'(0) = 1, \; h = 0.025$$

Analytical solution: *y(t)=sin(t)*

Example 3:

$$y'' + \frac{6}{t}y' + \frac{4}{t^2}y = 0, \quad y(1) = 1, \; y'(1) = 1, \; h = 0.003125$$

The analytical solution is: $y(t) = \frac{5}{3t} - \frac{2}{3t^4}$

Example 4:

$$y'' - 100y + 99\sin(t) = 0, \quad y(0) = 1, \; y'(0) = 11$$

Whose analytical solution is: $y(t) = \cos(10t) + \sin(10t) + \sin(t)$

Example 5:

$$y'' + y = 0.001\cos(t), \quad y(0) = 1, \; y'(0) = 0$$

With analytical solution: $y(t) = \cos(t) + 0.0005t\sin(t).$

4. Results and Discussion

Tables 1-5 present the numerical solutions in terms of the global maximum errors obtained for each of the problems considered respectively. The errors of the new methods (**10**)-(**14**) denoted as methods [**A**]-[**E**] are compared with those of block method of Badmus and Yahaya (2009) represented as [**BMY**], exponentially-fitted RK method of Simos (1998) taken to be [**SIM**]and exponentially fitted RK methods of Vanden Berghe et al. (1999) denoted as [**VAN**].

Discussion

In tables 1 and 2, we compare the maximum errors obtained for the proposed schemes in equations (10)-(14) denoted as methods [**A**]-[**E**] respectively for the problems considered, results are given at some selected steps. In columns 3-7 we give the absolute errors.

In Tables 3 and 5, we compare the block method of Badmus and Yahaya ([**BMY**]) and the exponentially-fitted Runge-Kutta methods of Vanden Berghe et al. ([**VAN**]) with the new method [**C**] for problems 3 and 5 respectively. The results in both cases show that our new method is much more efficient on comparison.

Finally, we also compare the exponentially-fitted method of Simos ([**SIM**]) with our new method [**D**] for problem 4, the end-point global error is presented in columns 4-5 of Table 4.

5. **Conclusion**

In this paper, a new approach for constructing a family of linear multistep methods with higher algebraic orders is developed. Using this new approach, we can construct any *k*-step method which directly integrates functions of the form (3) without reformulation to first order systems. Based on the new approach, the methods are symmetric, consistent, zero-stable and convergence.

All computations were carried out with a MATLAB programming language. It is evident from the results presented in Tables 1-5 that the new methods are considerably much more accurate than the other numerical methods that we have considered.

References

Abhulimen, C.E. & Otunta, F. O. (2006). "A Sixth-order Multiderivative Multisteps Methods for Systems of differential Equations", *International Journal of numerical Mathematics*, 1, 248-268.

Ademiluyi, R. A. & Kayode S. J. (2001). "Maximum Order Second-derivative hybrid multistep methods for Integration of Initial Value problems in Ordinary Differential Equations", *Journal of Nigerian Association of Mathematical Physics*, 5, 251-262.

Aruchunan, E. & Sulaiman, J. (2010). "Numerical Solution of Second-order Linear Fredholm Integro-differential Equation using Generalized Minimal Residual method", *American Journal of Applied Science*, 7, 780-783.

Awoyemi, D. O. (2001). "A New Sith-order Algorithm for General Second-order Differential Equations", *International Journal of Computational Mathematics*, 77, 117-124.

Awoyemi, D. O. (2005). "Algorithmic Collocation Approach for Direct Solution of Fourth-order Initial Value problems of ODEs", *International Journal of Computational Mathematics*, 82 271-284.

Awoyemi, D. O. & Kayode, S. J. (2005). "An Implicit Collocation Method for Direct Solution of Second-order ODEs", *Journal of Nigerian Association of Mathematical Physics*, 24, 70-78.

Badmus, A. M. & Yahaya, Y. A. (2009). "An Accurate Uniform Order 6 Blocks Method for Direct Solution of General Second-order ODEs", *Pacific Journal of Science* , 10, 248-254.

Bun, R. A. Vasil'Yer, Y. D. (1992). "ANumerical method for Solving Differential Equations of any Orders", *Computational Maths Physics*, 32, 317-330.

Chan, R. P. K. & Leon, P. (2004). "Order Conditions and Symmetry for Two-step Hybrid Methods", *International Journal of Computational Mathematics* , 81, 1519-1536.

Kayode, S. J. (2010). "A zero-stable Optimal method for Direct Solution of Second-order differential Equation ", *Journal of mathematics & Statistics*, 6, 367-371.

Lambert, J. D. & Watson, A. (1976). "Symmetric Multistep method for periodic Initial Value Problems", *Journal of Inst. Mathematics & Applied*, 18, 189-202

Lambert, J. D. (1991). "Numerical Methods for Ordinary Differential Systems of Initial Value Problems", John Willey & Sons, New York.

Owolabi, K. M. (2011). "An Order Eight Zero-stable Method for direct Integration of Second-order Ordinary Differential Equations ", *Mathematics Applied in Science & Technology*, 3(1), 23-33.

Owolabi, K. M. (2011). "4^{th}-step Implicit formula for Solution of Initial-value problems of Second-order ordinary Differential equations ", *Academic Journal of Mathematics & Computer Science Research*, 4, 270-272.

Simos, T. E. (1998). "An Exponentially-fitted Runge-Kutta Method for the Numerical Integration of Initial-value Problems with periodic or Oscillating Solutions", *Computational Physics Communications*, 115, 1-8.

Vanden Berghe, G., De Meyer, H., Van Daele, M., Van Hecke, T. 1999. Exponentially-fitted explicit

Runge-Kutta methods, Computer Phys Commun. 123:7-15.

Table 0.

K	α_0	α_1	α_2	α_3	α_4	α_5	α_k	β_0	β_1	β_2	β_3	β_4	β_5	β_6	p	C_{p+2}
2	1	-2					1	$\frac{1}{12}$	$\frac{10}{12}$	$\frac{1}{12}$					4	$-\frac{1}{240}$
3	-1	3	-3				-	$-\frac{1}{12}$	$-\frac{9}{12}$	$\frac{9}{12}$	$\frac{1}{12}$				5	-
4	1	-4	6	-4			-	$\frac{1}{12}$	$\frac{8}{12}$	$-\frac{18}{12}$	$\frac{8}{12}$	$\frac{1}{12}$			6	-
5	-1	5	-10	10	-5		-	$-\frac{1}{12}$	$-\frac{7}{12}$	$\frac{26}{12}$	$-\frac{26}{12}$	$\frac{7}{12}$	$\frac{1}{12}$		7	-
6	1	-6	15	-20	15	-6	-	$\frac{1}{12}$	$\frac{6}{12}$	$-\frac{33}{12}$	$\frac{52}{12}$	$-\frac{33}{12}$	$\frac{6}{12}$	$\frac{1}{12}$	8	-

Table 1: Comparison of errors arising from the new methods for example 1

t	Exact	[A]	[B]	[C]	[D]	[E]
0.100	*1.050041676*	*3.9101E-06*	*1.5497E-08*	*2.3801E-09*	*7.3310E-10*	*2.6314E-12*
0.125	*1.100335360*	*7.8321E-06*	*5.8113E-08*	*4.3400E-09*	*3.5810E-09*	*1.1900E-11*
0.150	*1.151140451*	*1.3721E-05*	*1.3682E-07*	*1.9070E-08*	*8.6691E-09*	*1.9211E-11*
0.175	*1.202732563*	*2.1982E-05*	*1.3721E-07*	*3.6951E-07*	*4.6490E-08*	*2.0601E-10*
0.200	*1.255412817*	*3.3021E-05*	*2.2006E-06*	*6.0871E-07*	*7.5481E-08*	*5.5816E-09*

Table 2: Comparison of errors arising from the new methods for example 2

t	Exact	[A]	[B]	[C]	[D]	[E]
0.100	*0.074929707*	*1.5616E-06*	*6.2414E-07*	*8.2001E-10*	*8.9611E-10*	*8.8862E-12*
0.150	*0.149438143*	*5.4621E-06*	*3.1184E-06*	*6.4122E-09*	*7.6048E-09*	*5.3631E-12*
0.200	*0.198669344*	*1.3098E-05*	*8.7225E-06*	*2.6520E-08*	*2.2781E-08*	*1.2370E-10*
0.250	*0.247403994*	*2.5699E-05*	*1.8667E-05*	*8.0170E-08*	*9.9998E-08*	*9.6901E-10*
0.300	*0.295520246*	*4.4484E-05*	*3.4171E-05*	*1.9819e-07*	*2.4736E-07*	*3.5761E-08*

Table 3: Comparison of errors of the new scheme [**C**] with method [**BMY**] for example 3

t	Exact	[C] Computed	[C]	[BMY]
0.025000	*1.022049164*	*1.022049012*	*1.52E-07*	*2.21E-04*
0.015625	*1.014447543*	*1.014447461*	*8.18E-08*	*1.56E-04*
0.012500	*1.011741018*	*1.011741018*	*3.63E-08*	*1.35E-04*
0.006250	*1.006057503*	*1.006057499*	*4.09E-09*	*7.50E-04*
0.003125	*1.003076526*	*1.003076525*	*1.40E-09*	*3.84E-05*

Table 4: Comparison of errors of scheme [**D**]with method [SIM] for example 4

t	Exact	[D] Computed	[D]	[SIM]
1.00000	*-0.541621655*	*-0.424800002*	*1.1E-01*	*1.4E-01*
0.50000	*-0.195836551*	*-0.175498054*	*2.3E-02*	*3.5E-02*
0.02500	*0.044732488*	*0.044721485*	*1.1E-05*	*1.1E-03*
0.12500	*1.388981715*	*1.388981253*	*4.6E-07*	*8.4E-05*
0.06250	*1.458519710*	*1.458551881*	*8.9E-08*	*5.5E-06*
0.03125	*1.290251377*	*1.290251373*	*3.4E-09*	*3.5E-07*

Table 5: Comparison of the end point errors of scheme [**C**] with method [**VAN**] for example 5

T	Exact	[C] Computed	[C]	[VAN]
1.00000	*0.540723041*	*0.539941907*	*7.81E-04*	*1.20E-03*
0.50000	*0.877702418*	*0.877693052*	*9.37E-06*	*7.54E-05*
0.02500	*0.968943347*	*0.968942743*	*6.05E-07*	*4.74E-06*
0.12500	*0.992205459*	*0.992205416*	*4.32E-08*	*2.96E-07*
0.06250	*0.998049463*	*0.998049460*	*2.88E-09*	*1.86E-08*