Jator and Lee *SpringerPlus* 2014, **3**:447
http://www.springerplus.com/content/3/1/447

SpringerPlus
a SpringerOpen Journal

RESEARCH Open Access

Implementing a seventh-order linear multistep method in a predictor-corrector mode or block mode: which is more efficient for the general second order initial value problem

Samuel N Jator[1*] and Leong Lee[2]

*Correspondence: Jators@apsu.edu
[1] Department of Mathematics and Statistics, Austin Peay State University, Clarksville TN 37044, USA
Full list of author information is available at the end of the article

Abstract

A Seventh-Order Linear Multistep Method (SOLMM) is developed and implemented in both predictor-corrector mode and block mode. The two approaches are compared by measuring their total number of function evaluations and CPU times. The stability property of the method is examined. This SOLMM is also compared with existing methods in the literature using standard numerical examples.

Keywords: General second order; Initial value problems; Block form

AMS Subject Classification: 65L05; 65L06

Introduction

Linear multistep methods (LMMs) of the form

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

have been extensively applied to solve the special second order initial value problem (IVP)

$$y'' = f(t, y),\ y(t_0) = y_0,\ y'(t_0) = y'_0, \quad t \epsilon [t_0, t_N] \tag{2}$$

on the discrete set of points $t_n = t_0 + nh, n = 0, ..., N, \quad h = \frac{t_N - t_0}{N}$, (see Lambert and Watson (1976), Ramos and Vigo-Aguiar (2005), Ixaru and Berghe (2004). Despite the successful application of (1) to solving problems of the form (2), fewer methods of the form (1) have been proposed for solving the general second order IVP

$$y'' = f(t, y, y'),\ y(t_0) = y_0,\ y'(t_0) = y'_0. \tag{3}$$

Some of the methods available for directly solving (3) are due to Awoyemi (2001) and Ramos and Vigo-Aguiar (2006). These methods are generally implemented in a step-by-step fashion in a predictor-corrector mode.

In this paper, we construct the continuous form of (1) which has ability to generate several methods which are combined and implemented in block form to solve (3) directly (see Jator and Li (2009) and Jator (2012, 2010, 2007).

Springer

The paper is organized as follows. In Section 'SOLMM', we derive a continuous approximation which is used to obtain the discrete methods that are combined to form the block method. The analysis and computational aspects of the SOLMM is given in Section 'Implementation of the SOLMM'. Numerical examples are given in Section 'Numerical examples' to show the accuracy and efficiency of the method. Finally, the conclusion of the paper is discussed in Section 'Conclusion'.

SOLMM

Continuous form

On interval $t_n \le t \le t_{n+6}$, the exact solution to (3) is approximated by the continuous form of the SOLMM

$$u(t) = \sum_{j=0}^{1} \alpha_j(t) y_{n+j} + h^2 \sum_{j=0}^{6} \beta_j(t) f_{n+j}, \tag{4}$$

whose first derivative is given by

$$u'(t) = \frac{d}{dt}\left\{\sum_{j=0}^{1} \alpha_j(t) y_{n+j} + h^2 \sum_{j=0}^{6} \beta_j(t) f_{n+j}\right\}, \tag{5}$$

where $\alpha_0(t), \alpha_1(t)$, and $\beta_j(t), j = 0, 1, 2$ are continuous coefficients that are uniquely determined. We assume that y_{n+j} is the numerical approximation to the analytical solution $y(t_{n+j})$, y'_{n+j} is an approximation to $y'(t_{n+j})$, and $f_{n+j} = f(t_{n+j}, y_{n+j}, y'_{n+j})$, $j = 0, 1, \ldots, 6$ is supplied by the differential equation. The coefficients of the method (4) are specified by the following theorem.

Theorem 1. *In order to obtain the coefficients of the continuous method (4), a nine by nine system is solved with the aid of Mathematica by demanding that the following conditions are satisfied*

$$u\left(t_{n+j}\right) = y_{n+j}, j = 0, 1,$$

$$u''\left(t_{n+j}\right) = f_{n+j},\ j = 0, 1, \ldots, 6.$$

After some algebraic manipulations, the equivalent form (6) produces the coefficients of (4) whose first derivative is given by (7),

$$u(t) = \sum_{j=0}^{8} \frac{\det(W_j)}{\det(W)} P_j(t), \tag{6}$$

$$u'(t) = \frac{d}{dt}\left(\sum_{j=0}^{8} \frac{\det(W_j)}{\det(W)} P_j(t)\right), \tag{7}$$

where we define the matrix W as

$$W = \begin{pmatrix} P_0(t_n) & \cdots & P_8(t_n) \\ P_0(t_{n+1}) & \cdots & P_8(t_{n+1}) \\ P''_0(t_n) & \cdots & P''_8(t_n) \\ P''_0(t_{n+1}) & \cdots & P''_8(t_{n+1}) \\ \vdots & & \vdots \\ P''_0(t_{n+6}) & \cdots & P''_8(t_{n+6}) \end{pmatrix},$$

and W_j is obtained by replacing the j^{th} column of W by V; $P_j(t) = t^j, j = 0, \ldots, 8$ are basis functions, and V is a vector given by $V = (y_n, y_{n+1}, f_n, f_{n+1}, \ldots, f_{n+6})^T$. We note that T is the transpose.

Proof. See Jator (2012). □

Discrete by-products

The following methods which are used to construct the block form are obtained by evaluating (4) and (5) at $t = \{t_{n+2}, t_{n+3}, t_{n+4}, t_{n+5}, t_{n+6}\}$ and $t = \{t_n, t_{n+1}, t_{n+2}, t_{n+3}, t_{n+4}, t_{n+5}, t_{n+6}\}$ respectively.

$$\begin{cases} y_{n+2} - 2y_{n+1} + y_n = \frac{h^2}{60480}(4315f_n + 53994f_{n+1} - 2307f_{n+2} + 7948f_{n+3} - 4827f_{n+4} \\ \qquad +1578f_{n+5} - 221f_{n+6}) \\ y_{n+3} - 3y_{n+1} + 2y_n = \frac{h^2}{20160}(2803f_n + 37950f_{n+1} + 14913f_{n+2} + 7108f_{n+3} - 3147f_{n+4} \\ \qquad +990f_{n+5} - 137f_{n+6}) \\ y_{n+4} - 4y_{n+1} + 3y_n = \frac{h^2}{10080}(2089f_n + 28878f_{n+1} + 16383f_{n+2} + 13828f_{n+3} - 1257f_{n+4} \\ \qquad +654f_{n+5} - 95f_{n+6}) \\ y_{n+5} - 5y_{n+1} + 4y_n = \frac{h^2}{6048}(1669f_n + 23250f_{n+1} + 15207f_{n+2} + 15004f_{n+3} + 4371f_{n+4} \\ \qquad +1074f_{n+5} - 95f_{n+6}) \\ y_{n+6} - 6y_{n+1} + 5y_n = \frac{h^2}{4032}(1375f_n + 19554f_{n+1} + 13401f_{n+2} + 15004f_{n+3} + 6177f_{n+4} \\ \qquad +4770f_{n+5} + 199f_{n+6}) \end{cases} \tag{8}$$

The derivatives are given by

$$\begin{cases} hy'_n = -y_n + y_{n+1} + \frac{h^2}{120960}(-28549f_n - 57750f_{n+1} + 51453f_{n+2} - 42484f_{n+3} \\ \qquad +23109f_{n+4} - 7254f_{n+5} + 995f_{n+6}) \\ hy'_{n+1} = -y_n + y_{n+1} + \frac{h^2}{120960}(9625f_n + 72474f_{n+1} - 41469f_{n+2} + 32524f_{n+3} \\ \qquad -17313f_{n+4} + 5370f_{n+5} - 731f_{n+6}) \\ hy'_{n+2} = -y_n + y_{n+1} + \frac{h^2}{40320}(2633f_n + 40910f_{n+1} + 17503f_{n+2} + 4f_{n+3} \\ \qquad -905f_{n+4} + 398f_{n+5} - 63f_{n+6}) \\ hy'_{n+3} = -y_n + y_{n+1} + \frac{h^2}{120960}(8441f_n + 117210f_{n+1} + 114147f_{n+2} + 75020f_{n+3} \\ \qquad -16257f_{n+4} + 4410f_{n+5} - 571f_{n+6}) \\ hy'_{n+4} = -y_n + y_{n+1} + \frac{h^2}{120960}(8059f_n + 120426f_{n+1} + 100605f_{n+2} + 150028f_{n+3} \\ \qquad +45381f_{n+4} - 1110f_{n+5} - 29f_{n+6}) \\ hy_{n+5} = -y_n + y_{n+1} + \frac{h^2}{40320}(2867f_n + 38750f_{n+1} + 38401f_{n+2} + 39172f_{n+3} \\ \qquad +46453f_{n+4} + 16382f_5 - 585f_{n+6}) \\ hy'_{n+6} = -y_n + y_{n+1} + \frac{h^2}{120960}(6875f_n + 128874f_{n+1} + 74781f_{n+2} + 192524f_{n+3} \\ \qquad +46437f_{n+4} + 179370f_{n+5} + 36419f_{n+6}) \end{cases} \tag{9}$$

Block form

The methods (8) and (9) are combined and expressed in the form

$$A_1\mathbf{Y}_{\mu+1} = A_0\mathbf{Y}_\mu + h^2B_0\mathbf{F}_\mu + h^2B_1\mathbf{F}_{\mu+1}, \ \ \mu = 0, 1, \ldots \ , \tag{10}$$

where

$$\begin{aligned}
\mathbf{Y}_{\mu+1} &= (\, y_{n+1}, \ldots , y_{n+6}, hy'_{n+1}, \ldots , hy'_{n+6})^{T},\\
\mathbf{F}_{\mu+1} &= (f_{n+1}, \ldots , f_{n+6}, hf'_{n+1}, \ldots , hf'_{n+6})^{T},\\
\mathbf{Y}_{\mu} &= (\, y_{n-5}, y_{n-4}, \ldots , y_{n}, \ldots , hy'_{n-5}, hy'_{n-4}, \ldots , hy'_{n})^{T},\\
\mathbf{F}_{\mu} &= (f_{n-5}, f_{n-4}, \ldots , f_{n}, hf'_{n-5}, hf'_{n-4}, \ldots , hf'_{n})^{T},
\end{aligned}$$

and A_0, A_1, B_0, and B_1 are matrices of dimension 12 whose entries denoted by $\alpha_j = \alpha_{i,j},\ \ \beta_j = \beta_{i,j}, i = 1, \ldots , 12$ are given by the coefficients of (8) and (9).

Order and local truncation error

Define the local truncation error of (10) as

$$\text{Ł}[\, z(t); h] = \mathbf{Z}_{\mu+1} - A_1^{-1}[\, A_0\mathbf{Z}_{\mu} + h^2 B_0 \overline{\mathbf{F}}_{\mu} + h^2 B_1 \overline{\mathbf{F}}_{\mu+1}] \tag{11}$$

where

$$\begin{aligned}
\mathbf{Z}_{\mu+1} &= ((y(t_{n+1}), \ldots , y(t_{n+6}), hy'(t_{n+1}), \ldots , hy'(t_{n+6}))^{T},\\
\overline{\mathbf{F}}_{\mu+1} &= (f(t_{n+1}, y(t_{n+1}), y'(t_{n+1})), \ldots , f(t_{n+6}, y(t_{n+6}), y'(t_{n+6})),\\
&hf'(t_{n+1}, y(t_{n+1}), y'(t_{n+1})), \ldots , hf'(t_{n+6}, y(t_{n+6}), y'(t_{n+6}))^{T},\\
\mathbf{Z}_{\mu} &= (y(t_{n-5}), y(t_{n-4}), \ldots , y(t_{n+6}), hy'(t_{n-5}), hy'(t_{n-4}), \ldots , hy'(t_{n+6}))^{T},\\
\overline{\mathbf{F}}_{\mu} &= (f(t_{n-5}, y(t_{n-5}), y'(t_{n-5})), f(t_{n-4}, y(t_{n-4}), y'(t_{n-4})), \ldots , f(t_{n}, y(t_{n}), y'(t_{n})),\\
&hf'(t_{n-5}, y(t_{n-5}), y'(t_{n-5})), hf'(t_{n-4}, y(t_{n-4}), y'(t_{n-4})), \ldots , hf'(t_{n}, y(t_{n}), y'(t_{n}))^{T},
\end{aligned}$$

and $\text{Ł}[z(t); h] = \ (\text{Ł}_1[z(t); h] , \ldots , \text{Ł}_6[z(t); h] , \text{Ł}_7[\, hz'(t); h] , \ldots , \text{Ł}_{12}[\, hz'(t); h]\,)^{T}$ is a linear difference operator.

Assuming that $z(t)$ is sufficiently differentiable, we can expand the terms in (4) as a Taylor series about the point t to obtain the expression for the local truncation error

$$L[z(x); h] = C_0 z(x) + C_1 hz'(x) + \ldots + C_q h^q z^{(q)}(x) + \ldots \tag{12}$$

where the constant coefficients $C_q = (C_{1,q}, C_{2,q}, \ldots , C_{12,q})^{T}$, $q = 0, 1, \ldots$ are given as follows:

$$\begin{aligned}
C_0 &= \textstyle\sum_{j=0}^{6} \alpha_j\\
C_1 &= \textstyle\sum_{j=1}^{6} j\alpha_j\\
&\ \vdots\\
C_q &= \tfrac{1}{q!}\left[\textstyle\sum_{j=1}^{6} j^q \alpha_j - q(q-1)\sum_{j=1}^{6} j^{q-2}\beta_j\right]
\end{aligned}$$

Definition 1. *Let $p_j, p'_j, j = 1, \ldots , 6$ be positive integers, then, the block method (10) has algebraic order $p = min\{p_1, \ldots , p_6, p'_1, \ldots , p'_6\}$, $p > 1$, provided there exists a corresponding constant C_{p+2} such that the Local Truncation Error E_μ satisfies*

$$\|E_\mu\| = C_{p+2} h^{p+2} + O(h^{p+3})$$

where $\| \cdot \|$ is the maximum norm.

Definition 2. *The block method (10) is said to be consistent if it has order at least one.*

The block method (10) has order and error constant given by the vector $p = 6$ and $C_{p+2} = \| \left(\frac{19}{6048}, \frac{349}{60480}, \frac{127}{15120}, \frac{349}{30240}, \frac{349}{30240}, -\frac{6031}{907200}, \frac{8563}{1814400}, \frac{6163}{1814400}, \frac{6163}{1814400}, \frac{1649}{907200}, \frac{8563}{1814400}, -\frac{6031}{907200} \right)^T \|$.

Linear stability of the SOLMM

The linear-stability of the SOLMM is discussed by applying the method to the test equation $y'' = \lambda y$, where λ is expected to run through the (negative) eigenvalues of the Jacobian matrix $\frac{\partial f}{\partial y}$ (see Sommeijer (1993)). Letting $q = \lambda h^2$, it is easily shown that the application of (10) to the test equation yields

$$Y_{\mu+1} = M(q) Y_\mu \, , M(q) = (A_1 - qB_1)^{-1}(A_0 + qB_0) \tag{13}$$

where the matrix $M(q)$ is the amplification matrix which determines the stability of the method.

Definition 3. *The interval* $[-q_0, 0]$ *is the stability interval, if in this interval* $\rho(q) \leq 1$, *where* $\rho(q)$ *is the spectral radius of* $M(q)$ *and* q_0 *is the stability boundary (see Sommeijer (1993)).*

Remark 1. *We found that* $\rho(q) \leq 1$ *if* $q \in [-4.552, 0]$, *hence, for the SOLMM,* $q_0 = 4.552$.

Implementation of the SOLMM

The SOLMM was implemented in both block mode and predictor-corrector mode using a written code in PERL programming language and executed on a laptop computer with AMD Quad-Core A10-4600M Processor, 8GB of RAM and Windows 8.1 OS. The total program running time was acceptable, as shown in Tables 1, 2 and 3. The computational time complexity and space complexity of the algorithms for both modes of SOLMM used for the examples in this paper are polynomial. Details of the block mode implementation is given in Jator (2012) and the predictor-corrector implementation is discussed next.

Predictor-corrector mode algorithm

The initial block was used to start predictor-corrector algorithm, after which the predictor (14) and corrector (15) were used in a step-by-step fashion to provide the numerical solution from the second block to the end of the interval.

Table 1 Results, with $t \in [0, 1]$, for Example 1

	PC-mode			Block-mode		
N	NFEs	Max error	CPU time	NFEs	Max error	CPU time
6	7	3.14×10^{-3}	2.88×10^{-2}	7	3.14×10^{-3}	3.42×10^{-2}
12	17	4.01×10^{-5}	3.84×10^{-2}	13	1.40×10^{-5}	6.32×10^{-2}
24	43	9.68×10^{-7}	6.04×10^{-2}	25	5.07×10^{-8}	1.26×10^{-1}
48	91	2.47×10^{-5}	5.27×10^{-2}	49	1.92×10^{-10}	1.31×10^{-1}
96	187	8.82×10^{3}	9.09×10^{-2}	97	5.31×10^{-12}	2.64×10^{-1}

Table 2 Results, with $t \epsilon$ [1, 8], for Example 2

	PC-mode			Block-mode		
N	NFEs	Max error	CPU time	NFEs	Max error	CPU time
6	7	2.40×10^{-3}	2.11×10^{-2}	7	2.240×10^{-3}	1.47×10^{-2}
12	17	9.23×10^{-4}	1.18×10^{-2}	13	2.42×10^{-4}	4.13×10^{-2}
24	43	1.51×10^{-2}	2.13×10^{-2}	25	1.23×10^{-5}	3.37×10^{-2}
48	91	8.78×10^{0}	4.05×10^{-2}	49	2.33×10^{-7}	9.35×10^{-2}
96	187	2.66×10^{8}	7.70×10^{-2}	97	1.79×10^{-9}	1.48×10^{-1}

Predictors. The following predictors are derived via Theorem 2.1 by deleting the last row and column of matrix W.

$$\begin{cases} y_{n+6} = -\frac{4090}{221} y_n + \frac{7296}{221} y_{n+1} - \frac{2985}{221} y_{n+2} + \frac{h^2}{663} \left(865 f_n + 11210 f_{n+1} + 1862 f_{n+2} + 3644 f_{n+3} \right. \\ \qquad \left. +301 f_{n+4} + 1018 f_{n+5}\right) \\ y'_{n+6} = \frac{1}{h} \left(-\frac{36861}{442} y_n + \frac{36640}{221} y_{n+1} - \frac{36419}{442} y_{n+2} + \frac{h^2}{278460} \left(1652785 f_n + 20780090 f_{n+1} \right. \right. \\ \qquad \left. \left. -703042 f_{n+2} + 3458396 f_{n+3} - 1724291 f_{n+4} + 1011562 f_{n+5}\right)\right) \end{cases} \tag{14}$$

Correctors. The last members of (8) and (9) are used as correctors.

$$\begin{cases} y_{n+6} = 6 y_{n+1} - 5 y_n + \frac{h^2}{4032} \left(1375 f_n + 19554 f_{n+1} + 13401 f_{n+2} + 15004 f_{n+3} + 6177 f_{n+4} \right. \\ \qquad \left. +4770 f_{n+5} + 199 f_{n+6}\right) \\ y'_{n+6} = \frac{1}{h} \left(- y_n + y_{n+1} + \frac{h^2}{120960} \left(6875 f_n + 128874 f_{n+1} + 74781 f_{n+2} + 192524 f_{n+3} \right. \right. \\ \qquad \left. \left. +46437 f_{n+4} + 179370 f_{n+5} + 36419 f_{n+6}\right)\right) \end{cases} \tag{15}$$

Numerical examples

Example 1. *We consider the IVP given by*

$$y'' - 4y' + 8y = t^3, \ y(0) = 2, \ y'(0) = 4, \ \ [0, 1],$$

$$Exact: y(t) = e^{2t}(2\cos(2t) - \frac{3}{64}\sin(2t)) + \frac{3}{32}t + \frac{3}{16}t^2 + \frac{1}{8}t^3$$

Table 3 Results, with $t \epsilon \left[\sqrt{\frac{\pi}{2}}, 10\right]$, for Example 3

	PC-mode			Block-mode		
N	NFEs	Max error	CPU time	NFEs	Max error	CPU time
180	710	2.34×10^{33}	6.84×10^{-1}	362	1.95×10^{-2}	6.82×10^{1}
360	1430	5.81×10^{59}	7.84×10^{-1}	722	2.13×10^{-4}	5.36×10^{1}
720	2870	5.41×10^{123}	1.15×10^{0}	1442	8.30×10^{-7}	8.46×10^{1}
1440	5750	8.62×10^{257}	1.93×10^{0}	2882	3.40×10^{-9}	1.24×10^{2}
2880	11510	1.68×10^{304}	3.12×10^{0}	5762	1.38×10^{-11}	2.46×10^{2}

Example 2. *We consider the given Bessel's IVP solved on* $[1, 8]$ *(see Vigo-Aguiar and Ramos (2006)).*

$$t^2y'' + ty' + (t^2 - 0.25)y = 0,\ \ y(1) = \sqrt{\frac{2}{\pi}}\sin 1 \simeq 0.6713967071418031$$

$$y'(1) = (2\cos 1 - \sin 1)/\sqrt{2\pi} \simeq 0.0954005144474746$$

$$Exact : y(t) = J_{1/2}(t) = \sqrt{\frac{2}{\pi t}}\sin t$$

The theoretical solution at $t = 8$ is $y(8) = \sqrt{\frac{2}{8\pi}}\sin(8) \simeq 0.279092789108058969$.

Example 3. *We consider the nonlinear Fehlberg problem which was also solved in Sommeijer (1993).*

$$\begin{array}{l} y_1'' = -4t^2y_1 - \frac{2}{\sqrt{y_1^2+y_2^2}}y_2,\ \ y_2'' = \frac{2}{\sqrt{y_1^2+y_2^2}}y_1 - 4t^2y_2 \\ y_1\left(\sqrt{\frac{\pi}{2}}\right) = 0, y_1'\left(\sqrt{\frac{\pi}{2}}\right) = -2\sqrt{\frac{\pi}{2}}, y_2\left(\sqrt{\frac{\pi}{2}}\right) = 1,\ \ y_2'\left(\sqrt{\frac{\pi}{2}}\right) = 0, \\ y_1(t) = \cos(t^2),\ \ y_2(t) = \sin(t^2). \end{array}$$

Comparison of block mode and predictor-corrector mode

The SOLMM is implemented in both predictor-corrector and block modes. The two approaches are compared by measuring their total number of function evaluations (NFEs) and CPU times in seconds. The block mode implementation is shown to be superior to the predictor-corrector mode implementation in terms of accuracy and the number of function evaluations. However, the predictor-corrector mode implementation uses less time than the block implementation. Details of the numerical examples are displayed in Tables 1, 2 and 3.

Comparison of block method with other methods

The theoretical solution at $t = 8$ is $y(8) = \sqrt{\frac{2}{8\pi}}\sin(8) \simeq 0.279092789108058969$. The errors in the solution were obtained at $t = 8$ using the SOLMM of order 7 and compared the the errors in (Vigo-Aguiar and Ramos 2006) which is based on the variable- step Falker method of order eight (VAR (8)) implemented in the predictor-corrector mode. The results given in Table 4 show that the SOLMM is more accurate than the method in (Vigo-Aguiar and Ramos 2006).

The maximum norm of the global error for the y-component is given in the form 10^{-CD}, where CD denotes the the number correct decimal digits at the endpoint (see (Sommeijer 1993)). This problem has also been solved in (Sommeijer 1993) using the eighth-order, eight-stage RKN (H8) method constructed by Hairer (1977). We have chosen to compare this method of order 8 with our method of order 7, because the orders of the methods are very close. The results obtained using the $H8$ are reproduced in Table 5 and compared

Table 4 Absolute errors for Example 2

VAR (8)		SOLMM	
Steps	**Errors**	**Steps**	**Errors**
67	7.1122×10^{-7}	60	2.49×10^{-8}
82	9.2632×10^{-8}	80	3.16×10^{-9}
97	8.7834×10^{-9}	100	6.04×10^{-10}
112	1.2108×10^{-10}	112	2.57×10^{-10}

Table 5 The correct decimal digit at the endpoint for Example 3

H8		SOLMM	
NFEs	CD	NFEs	CD
400	0.3	362	1.7
800	2.6	722	3.7
1600	5.2	1442	6.1
3200	7.6	2882	8.5
6400	10.0	5762	10.9

with the results given by our method. It is seen from Table 5 that our method performs better than those in (Sommeijer 1993) in terms of accuracy (smaller errors) and efficiency (smaller NFEs).

Conclusion

A SOLMM is proposed and implemented in both predictor-corrector and block modes. It is shown that the block mode algorithm is superior to the predictor-corrector mode algorithm in terms of accuracy and the number of function evaluations. However, the predictor-corrector mode implementation uses less time that the block implementation. the Details of the comparison of the numerical examples are displayed in Tables 1, 2, 3, 4 and 5. Our future research will be focus on developing a variable step version of the SOLMM in both modes.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

SJ proposed and derived the method, while LL developed a code for implementing the method. SJ and LL drafted the manuscript. All authors read and approved the final manuscript.

Author details

[1]Department of Mathematics and Statistics, Austin Peay State University, Clarksville TN 37044, USA. [2]Department of Computer Science and Information Technology, Austin Peay State University, Clarksville TN 37044, USA.

Received: 14 July 2014 Accepted: 12 August 2014
Published: 20 August 2014

References

Awoyemi DO (2001) A new sixth-order algorithm for general second order ordinary differential equation. Int J Comput Math 77:117–124

Hairer E (1977) Méthodes de Nyström pour l'équation différentielle $y'' = f(x, y)$. Numer Math 25:283–300

Ixaru L, Berghe GV (2004) Exponential fitting. Kluwer, Dordrecht, Netherlands

Jator SN (2012) A continuous two-step method of order 8 with a Block Extension for $y'' = f(x, y, y')$. Appl Math Comput 219:781–791

Jator SN (2010) Solving second order initial value problems by a hybrid multistep method without predictors. Appl Math Comput 217:4036–4046

Jator SN, Li J (2009) A self-starting linear multistep method for a direct solution of the general second order initial value problem. Intern J Comput Math 86:827–836

Jator SN (2007) A sixth order linear multistep method for the direct Solution of $y'' = f(x, y, y')$. Intern J Pure Appl Math 40:457–472

Lambert JD, Watson A (1976) Symmetric multistep method for periodic initial value problem. J Instit Math Appl 18:189–202

Ramos H, Vigo-Aguiar J (2005) Variable stepsize Stôrmer-Cowell methods. Math Comp Mod 42:837–846

Sommeijer BP (1993) Explicit, high-order Runge-Kutta-Nyström methods for parallel computers. Appl Numer Math 13:221–240

Vigo-Aguiar J, Ramos H (2006) Variable stepsize implementation of multistep methods for $y'' = f(x, y, y')$. J Comput Appl Math 192:114–131

doi:10.1186/2193-1801-3-447
Cite this article as: Jator and Lee: **Implementing a seventh-order linear multistep method in a predictor-corrector mode or block mode: which is more efficient for the general second order initial value problem.** *SpringerPlus* 2014 **3**:447.