Journal of Quality Measurement and Analysis
Jurnal Pengukuran Kualiti dan Analisis

JQMA 10(1) 2014, 87-97

PREDICTOR-CORRECTOR SCHEME IN MODIFIED BLOCK METHOD FOR SOLVING DELAY DIFFERENTIAL EQUATIONS WITH CONSTANT LAG

(Skim Peramal-Pembetul dalam Kaedah Blok Diubah Suai untuk Menyelesaikan Persamaan Pembezaan Tunda dengan Penangguhan Malar)

NURUL HUDA ABDUL AZIZ, ZANARIAH ABDUL MAJID & FUDZIAH ISMAIL

ABSTRACT

In this paper, the numerical solution of delay differential equations using a predictor-corrector scheme in modified block method is presented. In this developed algorithm, each coefficient in the predictor and corrector formula are recalculated when the step size changing. The Runge-Kutta Fehlberg step size strategy has been applied in the algorithm in order to achieve better results in terms of accuracy and total steps. Numerical results are given to illustrate the performance of this modified block method for solving delay differential equations with constant lag.

Keywords: delay differential equations; modified block method; variable step size

ABSTRAK

Dalam makalah ini, penyelesaian berangka bagi persamaan pembezaan tunda menggunakan skim peramal-pembetul dalam kaedah blok diubah suai dipersembahkan. Dalam al-Khwarizmi yang dibangunkan ini, setiap pekali dalam rumus peramal dan pembetul dikira semula apabila saiz langkah berubah. Strategi saiz langkah Runge-Kutta Fehlberg telah disuaikan dalam al-Khwarizmi bagi memperoleh keputusan yang lebih baik dari segi kejituan dan jumlah langkah. Keputusan berangka diberikan untuk menggambarkan prestasi bagi kaedah blok diubah suai dalam menyelesaikan persamaan pembezaan tunda dengan penangguhan tetap.

Kata kunci: persamaan pembezaan tunda; kaedah blok diubah suai; saiz langkah berubah

1. Introduction

Delay differential equations (DDEs) have numerous applications in science and engineering, where the existence of time-delays may arise in dynamic processes. In mathematics, the numerical methods for solving delay differential equations initially come from the adaptation of ordinary differential equations (ODEs). This research will consider the general form of a system of first order DDEs given as follows:

$$\begin{aligned} y_i'(x) &= f\left(x, y_i(x), y_i(x-\tau)\right), \qquad & x \in [a,b], \\ y_i'(x) &= \varphi_i(x), \qquad & x \in [\bar{a}, a], \end{aligned} \tag{1}$$

where $\bar{a} = \min(x-\tau)$ be defined and continuous on $[a,b]$, $\phi_i(x)$ is the given initial function, τ is a lag or time-delay and *i* is the number of equations in a system. The expression $y_i(x-\tau)$ is called the solution of the delay argument and $(x-\tau)$ is called the delay argument. There are three types of conditions that the delay can be represent such as a constant if it is dependent on a positive integer; time dependent if it is dependent on time *x* and state dependent if it is dependent on both *x* and the solution $y(x)$.

Numerical solution of block method has been proposed by several researchers such as Milne (1967), Rosser (1967), and Rao and Mouney (1997). In earlier work, the block method consists of computing the solutions at two points have been studied by Mehrkanoon *et al.* (2010b), Majid and Suleiman (2011) and San *et al.* (2011b) in solving ODE. The investigation of the block method has been extended to the three and four point block method for solving ODE, see Mehrkanoon *et al.* (2010a) and Anuar *et al.* (2011) respectively. These block methods were based on predictor-corrector scheme of multistep method and the advantage were in terms of obtaining several numerical solutions simultaneously at each integration steps.

The attention for the numerical solution of delay differential equations using one-step method has attracted many researchers such as Enright and Hu (1995), Karoui and Vaillancourt (1995), and Ismail and Suleiman (2001). While in the past few years, the studied of DDE using multistep method has gained attention by adapting with the block method. For instance, San *et al.* (2011a) has investigated a coupled block method consists of two and three point in a single code in order to compute the approximations simultaneously. In the work of Ishak *et al.* (2010), the authors had solved the two-point block method that has been introduced by Majid and Suleiman (2011) using variable step size and implemented the six points of Lagrange interpolation to evaluate the delay solution.

In this paper, we solve the problem of DDE as in (1) using the proposed modified block method. The Runge-Kutta Fehlberg strategy is implemented in determining the variable step size so that the accuracy of the error estimation is preserved. The delay argument of constant lags are evaluated with five points using Newton divided difference interpolation. Numerical results are presented and compared with the two-point block method in Ishak *et al.* (2010).

2. Formulation of the Method

The block method can be defined as the interval $[a,b]$ is divided into a series of blocks that contain a sequence of grid points which is given by $a = x_0, ..., x_n, x_{n+1}, x_{n+2}, ..., x_N = b$. In two-point block method, the sequence of grid will distribute into two points in each block. This method is performed when the solution of y_{n+1} and y_{n+2} at grid points x_{n+1} and x_{n+2} are computed simultaneously using the previous back values x_{n-2}, x_{n-1} and x_n. In Figure 1, the current $(k+1)$th block, has the step size 2*h* while the previous (k)th block has the step size 2*rh*. In this method, the uses of *r* is for variable step size implementation in the block method.

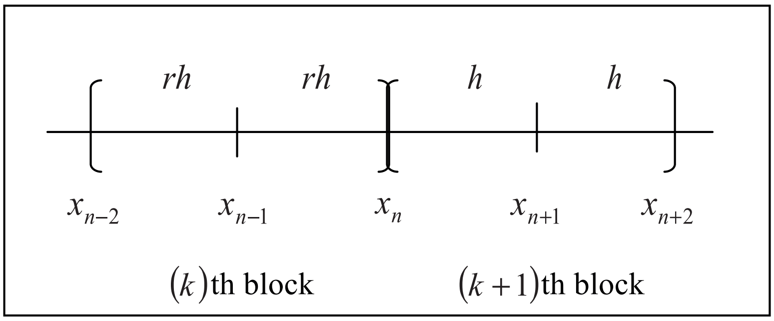

Figure 1: Two-point block method

The derivation of the two-point block method is as follows:

$$y'(x) = f(x,y), \qquad y(a) = y_0, \qquad a \le x \le b. \tag{2}$$

The modified two-point block method will consider the closest points in the interval i.e. $[x_n, x_{n+1}]$ and $[x_{n+1}, x_{n+2}]$ in the integration to obtain the solutions of y_{n+1} and y_{n+2} respectively. While the block method in Ishak *et al.* (2010) consider the points between $[x_n, x_{n+1}]$ and $[x_n, x_{n+2}]$. The formula of the first and second point y_{n+1} and y_{n+2} is then can be obtained by integrating (2) over the respective interval as follows:

$$\int_{x_n}^{x_{n+1}} y'(x)\mathrm{d}x = \int_{x_n}^{x_{n+1}} f(x,y)\mathrm{d}x \qquad \text{and} \qquad \int_{x_{n+1}}^{x_{n+2}} y'(x)\mathrm{d}x = \int_{x_{n+1}}^{x_{n+2}} f(x,y)\mathrm{d}x,$$

or

$$y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} f(x,y)\mathrm{d}x \qquad \text{and} \qquad y(x_{n+2}) - y(x_{n+1}) = \int_{x_{n+1}}^{x_{n+2}} f(x,y)\mathrm{d}x. \tag{3}$$

Let $P_q(x)$ be defined as Lagrange interpolation polynomial as

$$\begin{aligned} P_q(x) &= L_{q,0}(x) f(x_{n+1}) + L_{q,1}(x) f(x_n) + \ldots + L_{q,q}(x) f(x_{n+1-q}) \\ &= \sum_{j=0}^{q} L_{q,j}(x) f(x_{n+1-j}), \end{aligned} \tag{4}$$

where

$$L_{q,j}(x) = \prod_{\substack{i=0 \\ i\neq j}}^{q} \frac{(x - x_{n+1-i})}{(x_{n+1-j} - x_{n+1-i})}, \qquad \text{for } j = 0,1,\ldots,q.$$

In the polynomial $P_q(x)$, the notation q is denoted as a degree of polynomial while $(q+1)$ is denoted as the order of the method. The function $f(x,y)$ in (3) is then be replaced by $P_q(x)$. The interpolation points involved for corrector formula are $\{(x_{n-2}, f_{n-2}),\ldots,(x_{n+2}, f_{n+2})\}$. By taking $s = \frac{x - x_{n+2}}{h}$ and replacing $dx = x\ ds$, the limit of integration for the first and second point corrector formula are changed to $[-2,-1]$ and $[-1,0]$ respectively. Evaluating the integrals using MAPLE, then we obtained the corrector formulae in terms of *r* as follows:

First point:

$$\begin{aligned}
y(x_{n+1}) = y(x_n) + &\frac{h}{240(r+1)(r+2)(2r+1)r^2} \\
&\times\Big[\left(-21r^3 - 3r^2 - 50r^4 - 40r^5\right) f_{n+2} \\
&+\left(672r^3 + 144r^2 + 940r^4 + 320r^5\right) f_{n+1} \\
&+\left(1029r^3 + 564r^2 + 139r + 14 + 790r^4 + 200r^5\right) f_n \\
&+\left(-176r - 28 - 240r^2\right) f_{n-1} + \left(37r + 14 + 15r^2\right) f_{n-2}\Big].
\end{aligned} \tag{5}$$

Second point:

$$\begin{aligned}
y(x_{n+2}) = y(x_{n+1}) + &\frac{h}{240(r+1)(r+2)(2r+1)r^2} \\
&\times\Big[\left(549r^3 + 147r^2 + 610r^4 + 200r^5\right) f_{n+2} \\
&+\left(1632r^3 + 624r^2 + 1300r^4 + 320r^5\right) f_{n+1} \\
&+\left(-501r^3 - 516r^2 - 251r - 46 - 230r^4 - 40r^5\right) f_n \\
&+\left(304r + 92 + 240r^2\right) f_{n-1} + \left(-53r - 46 - 15r^2\right) f_{n-2}.
\end{aligned} \tag{6}$$

The same procedures are applied to obtain the predictor formula of first and second point by involving the interpolation points $\left\{\left(x_{n-3}, f_{n-3}\right), \ldots, \left(x_n, f_n\right)\right\}$.

3. Predictor-Corrector Scheme

Generally, the two-point block method has been formulated from the combination of predictor-corrector pair which can be defined as

$$\text{Predictor:} \qquad \sum_{q=0}^{k-1} \alpha_q y_{n+q} = h \sum_{q=0}^{k-1} \beta_{-q} f_{n-q},$$

$$\text{Corrector:} \qquad \sum_{q=0}^{k} \alpha_q y_{n+q} = h \sum_{q=0}^{k} \beta_{2-q} f_{n+2-q}, \tag{7}$$

with $q = 0$ until k and $(k-1)$ are the number of steps for predictor and corrector respectively. In practice, the predictor which is an explicit method is used to predict the first approximation of y. While the corrector, an implicit method is used to improve the approximation that has been obtained by explicit method. The PE(CE)s scheme has been constructed by using the P and C to indicate the application of predictor and corrector respectively, E to indicate evaluation of the function f and s denotes the number of iteration that is needed in a correcting to convergence. With this notation, the PE(CE)s scheme may be defined as

$$
\begin{aligned}
P: &\quad \begin{cases} {}^{[i]}y_p(x_{n+1}) = y(x_n) + h\sum_{q=0}^{3}\beta_{-q} z(x_{n-q}), \\ {}^{[i]}y_p(x_{n+2}) = {}^{[i]}y_p(x_{n+1}) + h\sum_{q=0}^{3}\beta_{-q} z(x_{n-q}), \end{cases} \\
E: &\quad \begin{cases} {}^{[i]}z_p(x_{n+1}) = f(x_{n+1}, {}^{[i]}y_p(x_{n+1}), {}^{[i]}y_p(x_{n+1}-\tau)), \\ {}^{[i]}z_p(x_{n+2}) = f(x_{n+2}, {}^{[i]}y_p(x_{n+2}), {}^{[i]}y_p(x_{n+2}-\tau)), \end{cases} \\
C: &\quad \begin{cases} {}^{[j]}y_c(x_{n+1}) = y(x_n) + h\sum_{q=0}^{4}\beta_{2-q} z(x_{n+2-q}), \\ {}^{[j]}y_c(x_{n+2}) = {}^{[j]}y_c(x_{n+1}) + h\sum_{q=0}^{4}\beta_{2-q} z(x_{n+2-q}), \end{cases} \\
E: &\quad \begin{cases} {}^{[j]}z_c(x_{n+1}) = f(x_{n+1}, {}^{[j]}y_c(x_{n+1}), {}^{[j]}y_c(x_{n+1}-\tau)), \\ {}^{[j]}z_c(x_{n+2}) = f(x_{n+2}, {}^{[j]}y_c(x_{n+2}), {}^{[j]}y_c(x_{n+2}-\tau)), \end{cases} \\
&\vdots \\
C: &\quad \begin{cases} {}^{[s]}y_c(x_{n+1}) = y(x_n) + h\sum_{q=0}^{4}\beta_{2-q} z(x_{n+2-q}), \\ {}^{[s]}y_c(x_{n+2}) = {}^{[s]}y_c(x_{n+1}) + h\sum_{q=0}^{4}\beta_{2-q} z(x_{n+2-q}), \end{cases} \\
E: &\quad \begin{cases} {}^{[s]}z_c(x_{n+1}) = f(x_{n+1}, {}^{[s]}y_c(x_{n+1}), {}^{[s]}y_c(x_{n+1}-\tau)), \\ {}^{[s]}z_c(x_{n+2}) = f(x_{n+2}, {}^{[s]}y_c(x_{n+2}), {}^{[s]}y_c(x_{n+2}-\tau)), \end{cases}
\end{aligned}
\tag{8}
$$

for $i = 0$ and $j = 1,2,\ldots,s$.

4. Step Size Selection Strategy

In order to achieve the desired accuracy in the whole integration, we have implemented the variable step size strategy of Runge-Kutta Fehlberg in Faires and Burden (1998). The variable step size strategy provides an effective step size selection by optimising the step size taken to achieve an accurate error estimation and hence getting the minimum number of total steps.

In general, the step size selection strategy is typically associated with the initial step size where it must be determined first before we start the integration. Then, follow by finding the values of starting point at x_{n-2}, x_{n-1} and x_n using the one-step method. The approximations of two values in the first block can be obtained by substituting the step size ratio $r=1$ and $q=1$ into the predictor and corrector formula. By defining the local truncation error,

$LTE = \left| y_{n+2}^{(k)} - y_{n+2}^{(k-1)} \right|$ with $y_{n+2}^{(k)}$ is the corrector formula at second point which has the order of the integration method k and $y_{n+2}^{(k-1)}$ is the corrector formula of one order less.

The algorithm then continues to check if it satisfies the successful step condition. If yes, then the new step size, *hnew* will be determined using the step size selection of Runge-Kutta Fehlberg or otherwise, the step size will be reduced to half from the previous step size, *hold*. At each step of integration, the condition of the step entering the last point of interval will be checked using the condition $\left(\left(x_{n+2} + 2.0 \times hnew \right) > b \right)$.

The implementation of the variable step size strategy can be summarised in the algorithm as follows:

Part 1: Computing the initial step size, *hmin*.

Step 1. $$h\min = \frac{\left(\dfrac{TOL}{K[0][1]} \right)^{\frac{1}{EQN+1}}}{4.0^{EQN}}.$$

Step 2. $^{*}h\min = h\min \times \left(1.0 \times 10^{-2} \right).$

Part 2: Computing the new step size, *hnew* in the adaptation of variable step size strategy of Runge-Kutta Fehlberg in 2-point modified block method.

Step 1. **if** $\left(LTE \le TOL \right)$ *Successful steps*

Step 2. $hacc = 0.5 \times \left(\dfrac{TOL}{LTE} \right)^{\frac{1}{4}}.$

Step 3. **if** $\left(hacc \le 0.1 \right)$ $hnew = 0.1 \times hold.$

Step 4. **else if** $\left(hacc \ge 4.0 \right)$ $hnew = 4.0 \times hold.$

Step 5. **else** $hnew = hacc \times hold.$

Step 6. Computing the step size ratio, $r = \dfrac{hold}{h}$ and $q = rold \times \dfrac{hold}{h}.$

Step 7. **else** $\left(LTE > TOL \right)$ *Failure steps*

Step 8. $hnew = 0.5 \times hold.$

Step 9. Computing the step size ratio, $r = \dfrac{hold}{h}$ and $q = rold \times \dfrac{hold}{h}.$

Part 3: Computing the last step size, *hend*.

Step 1. **if** $\left(\left(x_{n+2} + 2.0 \times hnew\right) > b\right)$

Step 2. $$hend = \frac{\left|b - x_{n+2}\right|}{2}.$$

Step 3. **else** repeat Step 6 in Part 2.

K[0][1] is the initial function evaluation and *EQN* is the number of equation in a system.

5. Numerical Results and Discussion

In this section, we have tested three problems of delay differential equations with constant lag in C program.

Problem 1: (Constant lag with $\tau = 1$, Ishak *et al.* (2010))

$$y_1'(x) = y_1(x-1) + y_2(x), \qquad 0 \le x \le 10,$$

$$y_2'(x) = y_1(x) - y_1(x-1), \qquad 0 \le x \le 10,$$

$$y_1(x) = \exp(x), \qquad x \le 0,$$

$$y_2(0) = 1 - \exp(-1).$$

Exact solution:

$$y_1(x) = \exp(x), \qquad x \ge 0,$$

$$y_2(x) = \exp(x) - \exp(x-1), \qquad x \ge 0.$$

Problem 2: (Constant lag with $\tau = \frac{\pi}{2}$, Ishak *et al.* (2010))

$$y_1'(x) = y_2(x), \qquad 2 \le x \le 20,$$

$$y_2'(x) = -\frac{1}{2}y_1(x) - \frac{1}{2} + y_1\left(\frac{1}{2}x - \frac{\pi}{4}\right)^2, \qquad 2 \le x \le 20,$$

$$y_1(x) = \sin(x), \qquad x \le 2,$$

$$y_2(x) = \cos(x), \qquad x \le 2.$$

Exact solution:

$$y_1(x) = \sin(x), \qquad x \ge 2,$$

$$y_2(x) = \cos(x), \qquad x \ge 2.$$

Problem 3: (Constant lag with $\tau = \frac{\pi}{2}$, Ishak *et al.* (2010))

$$y_1'(x) = -y_1\left(x - \frac{\pi}{2}\right), \qquad \frac{\pi}{2} \le x \le 10,$$

$$y_2'(x) = -y_2\left(x - \frac{\pi}{2}\right), \qquad \frac{\pi}{2} \le x \le 10,$$

$$y_1(x) = \sin(x), \qquad x \le \frac{\pi}{2},$$

$$y_2(x) = \cos(x), \qquad x \le \frac{\pi}{2}.$$

Exact solution:

$$y_1(x) = \sin(x), \qquad x \ge \frac{\pi}{2},$$

$$y_2(x) = \cos(x), \qquad x \ge \frac{\pi}{2}.$$

In order to illustrate the efficiency of the proposed method, all the numerical results from the modified block method are compared with the results in Ishak *et al.* (2010) and presented in Table 1 - 3 and Figure 2 - 4. The notations in the tables are defined as follows:

TOL	Tolerance
MTD	The choosing method
HMIN	Minimum step size
HMAX	Maximum step size
TS	Number of successful steps
FS	Number of failure steps
FNC	Number of function evaluation
MAXE	Maximum of mixed error test of the computed solution
AVERR	Average of mixed error test of the computed solution
2PMBM	Implementation of 2-point modified block method in this research.
2PBM	Implementation of 2-point block method in Ishak *et al.* (2010).

Table 1: Numerical results for Problem 1

TOL	MTD	HMIN	HMAX	TS	FS	MAXE	AVERR	FNC
10^{-2}	2PMBM	2.50E-04	6.66E-01	20	0	5.86E-04	1.17E-04	135
	2PBM	-	-	30	0	4.23E-04	5.79E-05	-
10^{-4}	2PMBM	2.50E-05	2.59E-01	37	0	6.27E-06	2.33E-06	251
	2PBM	-	-	49	0	3.85E-06	7.51E-07	-
10^{-6}	2PMBM	2.50E-06	1.03E-01	78	0	1.34E-07	5.66E-08	571
	2PBM	-	-	90	0	1.43E-07	4.45E-08	-
10^{-8}	2PMBM	2.50E-07	4.11E-02	180	0	1.51E-09	7.05E-10	1375
	2PBM	-	-	185	0	1.92E-09	7.46E-10	-
10^{-10}	2PMBM	2.50E-08	1.20E-02	431	0	6.20E-12	2.86E-12	2559
	2PBM	-	-	418	0	1.76E-11	6.46E-12	-

Table 2: Numerical results for Problem 2

TOL	MTD	HMIN	HMAX	TS	FS	MAXE	AVERR	FNC
10^{-2}	2PMBM	2.50E-04	5.86E-01	28	0	2.50E-03	3.90E-04	211
	2PBM	-	-	38	0	1.34E-03	1.82E-04	-
10^{-4}	2PMBM	2.50E-05	2.68E-01	56	0	5.86E-05	9.43E-06	405
	2PBM	-	-	68	0	2.70E-05	3.94E-06	-
10^{-6}	2PMBM	2.50E-06	1.02E-01	125	0	6.17E-07	8.59E-08	897
	2PBM	-	-	138	0	2.88E-07	5.29E-08	-
10^{-8}	2PMBM	2.50E-07	3.95E-02	294	0	6.19E-09	7.04E-10	1835
	2PBM	-	-	307	0	5.31E-09	8.27E-10	-
10^{-10}	2PMBM	2.50E-08	1.55E-02	715	0	9.43E-10	2.38E-10	4261
	2PBM	-	-	723	0	3.59E-11	7.57E-12	-

Table 3: Numerical results for Problem 3

TOL	MTD	HMIN	HMAX	TS	FS	MAXE	AVERR	FNC
10^{-2}	2PMBM	2.50E-04	6.22E-01	19	0	1.58E-03	2.40E-04	99
	2PBM	-	-	29	0	4.94E-04	4.51E-05	-
10^{-4}	2PMBM	2.50E-05	2.47E-01	33	0	4.16E-06	8.78E-07	181
	2PBM	-	-	45	0	6.64E-06	8.31E-07	-
10^{-6}	2PMBM	2.50E-06	9.18E-02	67	0	2.92E-08	8.24E-09	379
	2PBM	-	-	80	0	7.37E-08	1.34E-08	-
10^{-8}	2PMBM	2.50E-07	3.58E-02	147	0	1.21E-09	2.90E-10	857
	2PBM	-	-	161	0	7.98E-10	1.79E-10	-
10^{-10}	2PMBM	2.50E-08	1.42E-02	346	0	7.85E-10	2.15E-10	2047
	2PBM	-	-	358	0	8.25E-12	2.10E-12	-

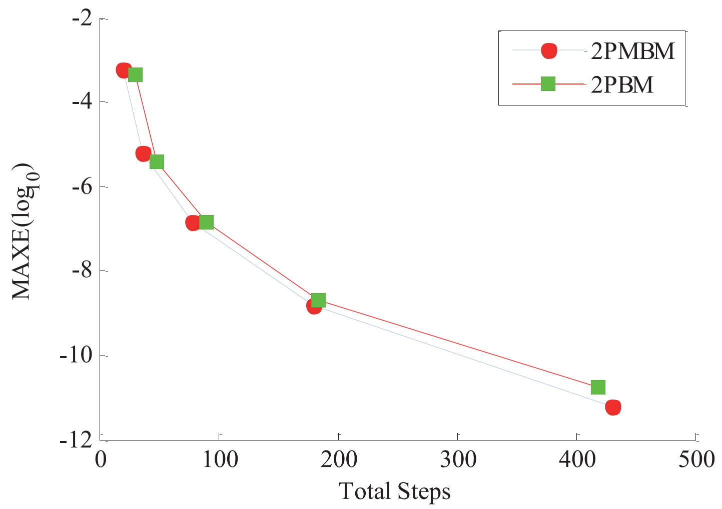

Figure 2: Total steps versus maximum error $(\log_{10})$ for Problem 1

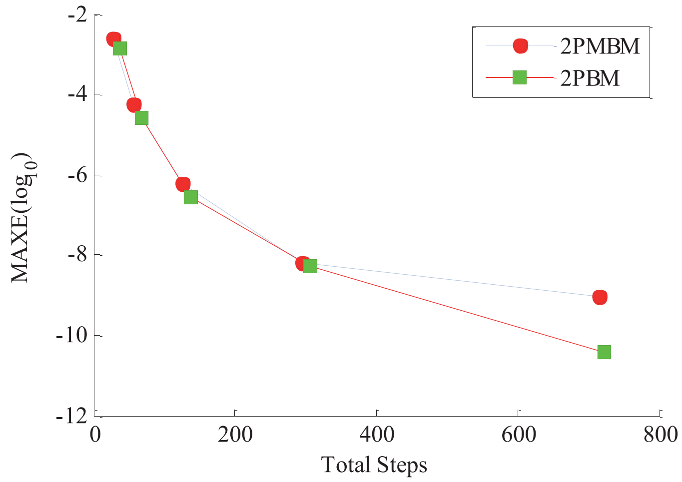

Figure 3: Total steps versus maximum error $(\log_{10})$ for Problem 2

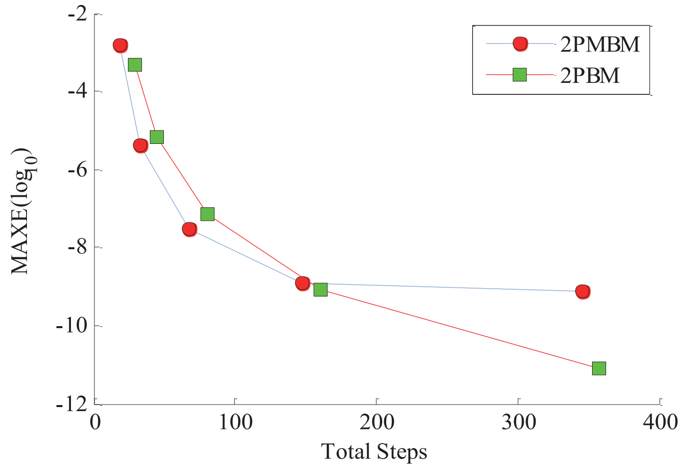

Figure 4: Total steps versus maximum error $(\log_{10})$ for Problem 3

In Table 1, the maximum error for 2PMBM and 2PBM are comparable for all tested tolerances. In terms of total steps, 2PMBM has less number of steps compared to 2PBM. For example, at TOL=10^{-6}, 2PMBM only required 78 steps with maximum error is 1.34E-07, while in 2PBM needs 90 steps with maximum error is 1.43E-07. In Table 2 - 3, both results are comparable in terms of accuracy but when the tolerances at 10^{-10}, the 2PMBM has one order larger for the maximum error compared to 2PBM. The total steps are less for 2PMBM compared to 2PBM.

6. Conclusion

The 2-point modified block method that is based on predictor-corrector scheme has been developed for solving delay differential equations with constant lag. The implementation of the proposed variable step size strategy in adaptation of block method has shown their own efficiency in terms of number of total steps and maximum error over the existing method.

Acknowledgements

The authors would like to acknowledge the financial support received from RUGS Grant and the Ministry of Education Malaysia.

References

Anuar K.H.K., Othman K.I., Ishak F., Ibrahim Z.B. & Majid Z.A. 2011. Development partitioning intervalwise block method for solving ordinary differential equations. *World Academy of Science, Engineering and Technology* **5**(10): 174-176.

Faires D. & Burden R.L. 1998. *Numerical Methods*. 2nd Ed. Pacific Grove: International Thomson Publishing Inc.

Enright W.H. & Hu M. 1995. Interpolating Runge-Kutta methods for vanishing delay differential equations. *Computing Springer-Verlag* **55**: 223-236.

Ishak F., Majid Z.A. & Suleiman M. 2010. Two point block method in variable stepsize technique for solving delay differential equations. *Journal of Materials Science and Engineering* **4** (12): 86 – 90.

Ismail F. & Suleiman M. 2001. Solving delay differential equations using intervalwise partitioning by Runge-Kutta method. *Applied Mathematics and Computation* **121**: 37-53.

Karoui A. & Vaillancourt R. 1995. A numerical method for vanishing-lag delay differential equations. *Applied Numerical Mathematics* **17**: 383-395.

Majid Z.A. & Suleiman M. 2011. Predictor-corrector block iteration method for solving ordinary differential equations. *Sains Malaysiana* **40**(6): 659-664.

Mehrkanoon S., Majid Z.A. & Suleiman M. 2010a. A variable step implicit block multistep method for solving first-order ODEs. *Journal of Computational and Applied Mathematics* **233**: 2387-2394.

Mehrkanoon S., Majid Z.A. & Suleiman M. 2010b. Implementation of 2-point 2-step methods for the solution of first order ODEs. *Applied Mathematical Sciences* **4**(7): 305-316.

Milne W.E. 1967. *Numerical Solution of Differential Equations*. New York: John Wiley.

Rao P.S. & Mouney G. 1997. Data communication in parallel block predictor-corrector methods for solving ODE's. *Parallel Computing* **23**: 1877-1888.

Rosser J.B. 1967. A Runge-Kutta for all seasons. *SIAM Review* **9**(3): 417-452.

San H.C., Majid Z.A. & Othman M. 2011a. Solving delay differential equations using coupled block method. In *Proceedings of the 4th International Conference on Modelling, Simulation and Applied Optimization*, pp. 1-4.

San H.C., Majid Z.A., Suleiman M. & Othman M. 2011b. New computational method for solving ordinary differential equations. *Journal of Quality Measurement and Analysis* **7**(1): 85-96.

Institute for Mathematical Research
Universiti Putra Malaysia
43400 UPM Serdang
Selangor DE, MALAYSIA
*E-mail: am_zana@upm.edu.my**

Mathematics Department, Universiti Putra Malaysia
43400 UPM Serdang
Selangor DE, MALAYSIA
E-mail: am_zana@upm.edu.my, fudziah@upm.edu.my

Institute of Engineering Mathematics
Universiti Malaysia Perlis
02600 Arau
Perlis IK, MALAYSIA
E-mail : nurul_6425@yahoo.com

*Corresponding author