Sains Malaysiana 40(6)(2011): 659–664

Predictor-Corrector Block Iteration Method for Solving Ordinary Differential Equations

(Kaedah Lelaran Blok Peramal-Pembetul Bagi Menyelesaikan Persamaan Terbitan Biasa)

ZANARIAH ABDUL MAJID* & MOHAMED SULEIMAN

ABSTRACT

Predictor-corrector two point block methods are developed for solving first order ordinary differential equations (ODEs) using variable step size. The method will estimate the solutions of initial value problems (IVPs) at two points simultaneously. The existence multistep method involves the computations of the divided differences and integration coefficients when using the variable step size or variable step size and order. The block method developed will be presented as in the form of Adams Bashforth - Moulton type and the coefficients will be stored in the code. The efficiency of the predictor-corrector block method is compared to the standard variable step and order non block multistep method in terms of total number of steps, maximum error, total function calls and execution times.

Keywords: Block method; ordinary differential equations; predictor corrector block

ABSTRAK

Kaedah dua titik blok peramal-pembetul telah dibangunkan bagi penyelesaian persamaan terbitan biasa peringkat pertama menerusi panjang langkah berubah. Kaedah ini akan memberi nilai penghampiran bagi masalah nilai awal pada dua titik secara serentak. Kaedah multilangkah yang sedia ada melibatkan pengiraan beza pembahagi dan pekali kamiran apabila menggunakan saiz langkah berubah atau saiz langkah berubah dan berperingkat. Kaedah blok yang dibangunkan adalah dalam bentuk Adams Bashforth - Moulton dan pekali akan disimpan di dalam kod. Keberkesanan kaedah blok peramal-pembetul akan di bandingkan dengan kaedah multilangkah bukan blok bagi panjang langkah dan peringkat berubah dari segi jumlah langkah, ralat maksimum, jumlah kiraan fungsi dan masa pelaksanaan.

Kata kunci: Blok peramal pembetul; kaedah blok; persamaan terbitan biasa

INTRODUCTION

In this paper, we consider the form of IVPs for systems of first order ODEs as follows

$$y' = f(x, y), \quad y(a) = y_0 \quad a \le x \le b. \tag{1}$$

Shampine and Gordon (1975), Suleiman (1979), Lambert (1993) and Omar (1999) described the algorithm of variable order and step size for the multistep method. The algorithm involved tedious computations of the divided differences and integration coefficients. Majid and Suleiman (2006) have shown that the cost of computing the divided differences and integration coefficients in the multistep method was expensive and the computational cost increases when the method was implemented in variable step size and order.

A block method will compute simultaneously the solution values at several distinct points on the x-axis in the block. Block method for numerical solution had been proposed by several researchers such as Rosser (1976), Worland (1976), Chu and Hamilton (1987), Omar (1999), Majid and Suleiman (2006) and Majid et al. (2003, 2006). Majid et al. (2003, 2006) have introduced the two and three block one step methods based on Newton backward divided difference formulae for solving first order ODEs. The aim of this paper is to introduce the predictor corrector two point block method presented as in the simple form of Adams Moulton method for solving (1) using variable step size.

FORMULATION OF THE TWO POINT BLOCK METHOD

In Figure 1, the two values of y_{n-1} and y_{n+2} were approximated simultaneously in a block by using the same back values from the earlier block.

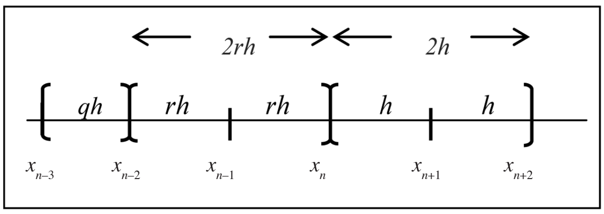

FIGURE 1. Two point block method

The computed block has the step size $2h$ and the previous block has the step sizes $2rh$ and qh. The corrector formulae will involve the set of points $\{x_{n-2}, x_{n-1}, x_n, x_{n+1}, x_{n+2}\}$, while the predictor formulae will involve the set of points $\{x_{n-3}, x_{n-2}, x_{n-1}, x_n\}$. Therefore, the corrector formulae will involve the step sizes of $2rh$ and $2h$ while the predictor formulae will only consider the step sizes qh and $2rh$. The corrector formulae of the two point block method were derived using Lagrange interpolation polynomial of order 5. The two values of y_{n+1} and y_{n+2} can be obtained by integrating (1) over the interval $[x_n, x_{n+1}]$ and $[x_n, x_{n+2}]$, respectively using MAPLE and the following corrector formulae can be obtained as

$$\begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix} + h \begin{bmatrix} \dfrac{18+75r+80r^2}{60(r+1)(2r+1)} & \dfrac{3+15r+20r^2}{240(r+1)(r+2)} \\ \dfrac{4(6+15r+10r^2)}{15(2r+1)(r+1)} & \dfrac{9+15r+5r^2)}{15(r+2)(r+1)} \end{bmatrix} \begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix}$$

$$+h \begin{bmatrix} -\dfrac{(7+30r)}{60(r+1)(r+2)} & \dfrac{(7+45r+100r^2)}{240r^2} \\ \dfrac{4}{15r^2(r+2)(r+1)} & \dfrac{(5r^2-1)}{15r^2} \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix}$$

$$+h \begin{bmatrix} 0 & \dfrac{7+15r}{240(r+1)(2r+1)r^2} \\ 0 & -\dfrac{1}{15r^2(2r+1)(r+1)} \end{bmatrix} \begin{bmatrix} f_{n-3} \\ f_{n-2} \end{bmatrix}. \tag{2}$$

The predictor formulae were derived similar as the corrector formulae and the interpolation points involved are $(x_{n-3}, f_{n-3}), \ldots, (x_n, f_n)$.

VARIABLE STEP SIZE STRATEGY

Shampine and Gordon (1975) step size strategy will be implemented in the methods described above, where the next step size will be restricted to half, double or the same as the current step size. The successful step size will remain constant for at least two blocks before we considered the next step size to be doubled. In the code developed, when the next successful step size is doubled, the ratio r is 0.5 and if the next successful step size remain constant, r is 1. In case of step size failure, r is 2.

Substitute $r = 1$, 2 and 0.5 in (2) will produce the following first and second points of the corrector formulae:

r = 1,

$$\begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix} + h \begin{bmatrix} \dfrac{173}{360} & -\dfrac{19}{720} \\ \dfrac{62}{45} & \dfrac{29}{90} \end{bmatrix} \begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix} + h \begin{bmatrix} \dfrac{11}{720} & \dfrac{57}{90} \\ \dfrac{2}{45} & \dfrac{4}{15} \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix} + h \begin{bmatrix} 0 & -\dfrac{37}{360} \\ 0 & -\dfrac{1}{90} \end{bmatrix} \begin{bmatrix} f_{n-3} \\ f_{n-2} \end{bmatrix} \tag{3}$$

r = 2,

$$\begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix} + h \begin{bmatrix} \dfrac{122}{225} & -\dfrac{113}{2880} \\ \dfrac{304}{225} & \dfrac{59}{180} \end{bmatrix} \begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix} + h \begin{bmatrix} \dfrac{67}{2880} & \dfrac{497}{960} \\ \dfrac{1}{180} & \dfrac{57}{180} \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix} + h \begin{bmatrix} 0 & \dfrac{37}{14400} \\ 0 & -\dfrac{1}{900} \end{bmatrix} \begin{bmatrix} f_{n-3} \\ f_{n-2} \end{bmatrix} \tag{4}$$

r = 0.5,

$$\begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix} + h \begin{bmatrix} \dfrac{151}{360} & -\dfrac{31}{1800} \\ \dfrac{64}{45} & \dfrac{71}{225} \end{bmatrix} \begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix} + h \begin{bmatrix} \dfrac{88}{225} & \dfrac{109}{900} \\ \dfrac{64}{225} & \dfrac{3}{45} \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix} + h \begin{bmatrix} 0 & -\dfrac{29}{360} \\ 0 & -\dfrac{4}{45} \end{bmatrix} \begin{bmatrix} f_{n-3} \\ f_{n-2} \end{bmatrix} \tag{5}$$

The above formulae are in the form of a constant step size multistep method. These formulae will be stored in the code and therefore we don't have to compute the coefficients as the step size changing.

IMPLEMENTATION OF THE METHOD

The first step in the code starts by finding the initial points in the starting block for the method. Each step in the first and second blocks was set to equal distant. Therefore the value r and q in Figure 1 will set equal to one. Initially we used the sequential Euler method to find the three additional points i.e x_{n-2}, x_{n-1} and x_n. The two block method can be applied after the points y_{n+1} and y_{n+2} for the next block has been obtained. Each point in the predictor and the corrector formulae can perform the computations simultaneously within the block as they are independent of each other. The values of y_{n+1} and y_{n+2} will be approximated using the predictor-corrector schemes. If s corrections are needed, then the sequence of computations at any mesh point is $(PE)\ (CE)^1 \ldots (CE)^s$ where P and C indicate the application of the predictor and corrector formulae respectively and E indicate the evaluation of the function f. Below we describe the iterated technique that has been implemented in the code:

Step 1: The predictor equations

$$P: \quad y^p_{n+1} = \sum_{i=0}^{3} \alpha_{n+i} f_{n-i}$$

$$y^p_{n+2} = \sum_{i=0}^{3} \beta_{n+i} f_{n-i}$$

$$E: \quad f^p_{n+1} = (x_{n+1}, y^p_{n+1})$$

$$f^p_{n+2} = (x_{n+2}, y^p_{n+2})$$

Step 2: The corrector equations

$$C:\quad y^c_{n+1} = \sum_{i=0}^{5} \chi_{n+i} f_{n+2-i}$$

$$y^c_{n+1} = \sum_{i=0}^{5} \delta_{n+i} f_{n+2-i}$$

$$E:\quad f^c{}_{n+1} = (x_{n+1}, y^c{}_{n+1})$$

$$f^c{}_{n+2} = (x_{n+2}, y^c{}_{n+2})$$

Step 3: Convergent test: if yes go to *Step 4* else *Step 2*
Step 4: Compute local truncation error, next step size

In the code, we iterate the corrector to convergence. The convergence test employed was $\left|y^{(s+1)}_{n+2} - y^{(s)}_{n+2}\right| < 0.1 \times$ tolerance and s is the number of iterations using $(PE)\,(CE)^1 \ldots (CE)^s$ mode. After the successful convergence test, local errors estimate (*Est*) at the point x_{n+2} will be performed to control the error for the block. We obtained the *Est* by comparing the absolute difference of the corrector formula derived of order k and a similar corrector formula of order $k-1$. The error control for the developed method is at the second point in the block because in general it had given us better results.

The errors calculated in the code are defined as (Omar, 1999)

$$\left(E_i\right)_t = \left|\frac{(y_i)_t - \left(y(x_i)\right)_t}{A + B\left(y(x_i)\right)_t}\right| \tag{8}$$

where $(y)_t$ is the t-th component of the approximate y. A=1, B=0 correspond to the absolute error test. A=1, B=1 correspond to the mixed test and finally A=0, B=1 correspond to the relative error test.

The maximum error is defined as follows:

$$\text{MAXE} = \max_{1\le i\le SSTEP}\left(\max_{1\le i\le N}\left(E_i\right)_t\right) \tag{9}$$

where N is the number of equations in the system and *SSTEP* is the number of successful steps. At each step of integration, a test for checking the end of the interval is made. If b denotes the end of the interval then

$$\text{if } x + 2h \ge b \text{ then } h_{last} = \frac{(b-x)}{2} \tag{10}$$

otherwise h remains as calculated. The interpolation polynomial will be used to find the four back points with h_{last} equally distant and then the two block method will be applied. The technique helped to reach the end point of the interval.

STABILITY REGION

The stability of the two point block method on a linear first order problem is applied to the test equation

$$y' = f = \lambda y. \tag{11}$$

The method is zero stable at r = 1, 2, 0.5 where all the principal roots lie in or on the unit circle. The stability region is investigated when the step size is constant, doubled and halved for the method. The test equation (11) is substituted into the corrector formulae of the block method. The stability polynomials of the block method at r = 1, 2, 0.5 are as follows,
For r = 1 we have,

$$Q_1\left(\bar{h}\right) = t^4\left(1 - \frac{289}{360}\bar{h} + \frac{413}{2160}\bar{h}^2\right) + t^3\left(-1 - \frac{191}{180}\bar{h} - \frac{559}{720}\bar{h}^2\right) + t^2\left(-\frac{19}{120}\bar{h} - \frac{7}{240}\bar{h}^2\right) - \frac{1}{2160}\bar{h}^2 t = 0.$$

For r = 2 we have,

$$Q_1\left(\bar{h}\right) = t^4\left(1 - \frac{87}{100}\bar{h} + \frac{623}{2700}\bar{h}^2\right) + t^3\left(-1 - \frac{5291}{4800}\bar{h} - \frac{289}{540}\bar{h}^2\right) + t^2\left(-\frac{431}{14400}\bar{h} - \frac{533}{86400}\bar{h}^2\right) - \frac{1}{86400}\bar{h}^2 t = 0.$$

Finally, for r = 0.5 we have,

$$Q_1\left(\bar{h}\right) = t^4\left(1 - \frac{147}{200}\bar{h} + \frac{847}{5400}\bar{h}^2\right) + t^3\left(-1 - \frac{407}{600}\bar{h} - \frac{1493}{1080}\bar{h}^2\right) + t^2\left(-\frac{172}{1225}\bar{h} - \frac{179}{130}\bar{h}^2\right) - \frac{8}{675}\bar{h}^2 t = 0.$$

where $\bar{h} = h\lambda$ and the stability region is shown in Figure 2.

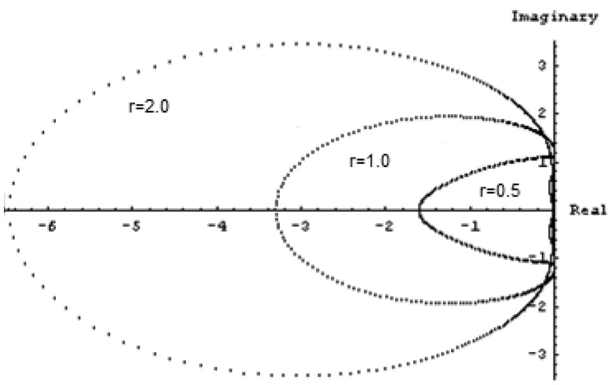

FIGURE 2. Stability Region for two block corrector method

The stability region is inside the boundary of the dotted points. The stability region is larger when the step size is half (r = 2) compared to the step size being double (r = 0.5) or constant (r = 1) . This is expected because the region should get larger with smaller step sizes. The smallest stability region is when the step size being double (r = 0.5) for the method.

NUMERICAL RESULTS

In order to study the efficiency of the proposed block method, we present some numerical experiments for the following problems:

Problem 1: $y_1' = -y_1$,

$y_1(0) = 1, \qquad x \in [0,20]$

Exact Solution: $y_1(x) = e^{-x}$

Problem 2: $y_1' = 0.1(y_1 - \sin x) + \cos x$,

$y_1(0) = 0, \qquad x \in [0,20]$

Exact Solution: $y_1(x) = \sin x$

Problem 3: $y_1' = y_3, y_2' = y_4, y_3' = -\frac{y_1}{r^3}, y_4' = -\frac{y_2}{r^3}, r = \sqrt{y_1^2 + y_2^2}$,

$y_1(0)=1, y_2(0)=0, y_3(0)=0, y_4(0)=1, x\in [0,20]$

Exact Solution: $y_1(x) = \cos x$, $y_2(x) = \sin x$, $y_3(x) = -\sin x$, $y_4(x) = \cos x$.

Problem 4: $y_1' = y_2$,

$y_2' = y_1 - 4xe^x$,

$y_1(0) = 0, y_2(0) = 1, \qquad [0,100]$

Exact Solution: $y_1(x) = x(1-x)e^x$,

$y_2(x) = (1-x-x^2)e^x$,

The following notations are used in the tables:

TOL Tolerance
MTD Method employed
TS Total steps taken
FS Total failure step
MAXE Magnitude of the maximum error of the computed solution
FCN Total function calls
TIME The execution time taken in microseconds to complete the integration in a given range
RSTEP The ratio steps, $\frac{TS_{1PVSO}}{TS_{2BPC}}$
RTIME The ratio execution times, $\frac{TIME_{1PVSO}}{TIME_{2BPC}}$
2BPC Implementation of the two point predictor corrector block method using variable step size
1PVSO Implementation of the one point method of variable step size and order using the integration coefficients

The code was written in C language and executed on DYNIX/ptx operating system. Table 1-4 show the numerical results for the four given problems when solved using the two point predictor corrector block method (2BPC) and conventional non block multistep method (1PVSO) in Omar (1999).

In term of maximum error, 2BPC is better compared to 1PVSO in all tested problems. The total number of steps for 2BPC method has shown to be less than the 1PVSO method. The 2BPC saves considerable amount of computational time and is much faster than 1PVSO although the total function calls is twice than the total function taken by the 1PVSO. This has shown the advantage of the 2PBC method in the form of standard multistep method because the cost per step is cheaper. In Table 5, the ratios are greater than one shows that the 2BPC reduced the total steps taken and execution times compared to 1PVSO. These results are expected since the block method would approximate the solutions at two points simultaneously.

CONCLUSION

In this paper, we have shown the efficiency of the developed predictor-corrector two point block method presented as in the simple form of Adams Bashforth - Moulton method using variable step size is suitable for solving ODEs. The method has shown the superiority in terms of total steps, maximum error and execution times over the one point multistep method.

TABLE 1. Numerical results for solving Problem 1

TOL	MTD	TS	FS	MAXE	FCN	TIME
10^{-2}	2BPC	22	0	5.0529(-4)	171	132
	1PVSO	32	0	2.5139(-2)	97	334
10^{-4}	2BPC	32	0	3.1515(-6)	303	212
	1PVSO	35	0	1.9434(-3)	127	438
10^{-6}	2BPC	68	0	2.8360(-8)	505	357
	1PVSO	84	0	1.2904(-7)	253	659
10^{-8}	2BPC	146	0	1.5336(-10)	981	719
	1PVSO	168	0	1.7610(-9)	505	1166
10^{-10}	2BPC	340	0	1.4077(-12)	2161	1630
	1PVSO	390	0	1.2743(-11)	1171	2778

TABLE 2. Numerical results for solving Problem 2

TOL	MTD	TS	FS	MAXE	FCN	TIME
10^{-2}	2BPC	29	0	6.9519(-4)	201	339
	1PVSO	38	0	1.9992(-2)	115	392
10^{-4}	2BPC	39	0	1.2411(-4)	321	416
	1PVSO	82	0	9.4330(-4)	247	846
10^{-6}	2BPC	158	0	4.0421(-8)	953	1376
	1PVSO	182	0	3.5795(-5)	547	1664
10^{-8}	2BPC	341	2	8.1888(-9)	2091	3064
	1PVSO	435	0	2.9001(-7)	1306	3815
10^{-10}	2BPC	827	1	6.4933(-11)	4963	7388
	1PVSO	1044	0	9.7441(-10)	3133	8949

TABLE 3. Numerical results for solving Problem 3

TOL	MTD	TS	FS	MAXE	FCN	TIME
10^{-2}	2BPC	30	0	9.3294(-2)	309	944
	1PVSO	44	0	9.9994(-1)	133	1336
10^{-4}	2BPC	61	0	1.4804(-3)	513	1322
	1PVSO	83	0	5.1513(-3)	250	1658
10^{-6}	2BPC	137	0	1.9884(-5)	1121	2904
	1PVSO	181	0	1.3466(-3)	544	3408
10^{-8}	2BPC	322	0	2.0891(-7)	2001	5742
	1PVSO	435	0	1.2342(-5)	1306	8081
10^{-10}	2BPC	781	0	2.2126(-9)	4767	13906
	1PVSO	1044	0	1.4101(-7)	3133	19568

TABLE 4. Numerical results for solving Problem 4

TOL	MTD	TS	FS	MAXE	FCN	TIME
10^{-2}	2BPC	111	0	2.4548(-3)	911	1245
	1PVSO	149	0	8.6967(-3)	448	2147
10^{-4}	2BPC	266	0	3.2284(-5)	2131	2762
	1PVSO	394	0	6.5069(-5)	1183	4316
10^{-6}	2BPC	653	0	6.3745(-7)	5177	6500
	1PVSO	868	0	1.0687(-6)	2605	9444
10^{-8}	2BPC	1621	0	7.7037(-9)	12713	16065
	1PVSO	3530	0	7.1469(-9)	10591	37358
10^{-10}	2BPC	4040	0	5.3125(-11)	24903	34556
	1PVSO	8544	0	7.8037(-11)	25633	90211

TABLE 5. The ratios steps and execution times for solving Problem 1 to 4

	PROB 1		PROB 2		PROB 3		PROB 4	
TOL	RSTEP	RTIME	RSTEP	RTIME	RSTEP	RTIME	RSTEP	RTIME
10^{-2}	1.45	2.53	1.31	1.16	1.47	1.42	1.34	1.72
10^{-4}	1.20	2.07	2.10	2.03	1.36	1.25	1.48	1.56
10^{-6}	1.24	1.85	1.15	1.21	1.32	1.17	1.33	1.45
10^{-8}	1.15	1.62	1.28	1.25	1.35	1.41	2.18	2.33
10^{-10}	1.15	1.70	1.26	1.21	1.34	1.41	2.11	2.61

REFERENCES

Chu, M.T. & Hamilton, H. 1987. Parallel Solution of ODE's by Multiblock Methods, *SIAM J. Sci Stat. Comput.* 8:342-354.

Lambert, J.D. 1993. *Numerical Methods For Ordinary Differential Systems. The Initial Value Problem*. New York: John Wiley & Sons, Inc.

Majid, Z.A. & Suleiman, M. 2006. 1-Point Implicit Code of Adams Moulton Type For Solving First Order Ordinary Differential Equations, *Chiang Mei Journal of Sciences* 33(2): 153-159.

Majid, Z.A, Suleiman, M. & Omar, Z. 2006. 3-Point Implicit Block Method for Solving Ordinary Differential Equations. *Bulletin of the Malaysian Mathematical Sciences Society* 29(1/2): 1-9.

Majid, Z.A., Suleiman, M., Ismail, F. & Othman, M. 2003. Two Point Implicit Block Method In Half Gauss Seidel For Solving Ordinary Differential Equations. *Jurnal Matematika* 19(2): 91-100.

Omar, Z., 1999. *Developing Parallel Block Methods For Solving Higher Order ODEs Directly*, Ph.D. Thesis, University Putra Malaysia, Malaysia (unpublished).

Rosser, J.B. 1976. Runge Kutta For All Season, *SIAM Rev*. 9: 417-452.

Shampine, L.F. & Gordon, M.K. 1975. *Computer Solution of Ordinary Differential Equations: The Initial Value Problem*. San Francisco: W. H. Freeman and Company

Suleiman, M.B. 1979. *Generalised Multistep Adams and Backward Differentiation Methods for the Solution of Stiff and Non-Stiff Ordinary Differential Equations*. Ph.D. Thesis. University of Manchester, UK (unpublished).

Worland, P.B. 1976. Parallel Methods for the Numerical Solutions of Ordinary Differential Equations. *IEEE Transactions on Computers* 25: 1045-1048.

Mathematics Department
Faculty of Science
Universiti Putra Malaysia
43400 Serdang, Selangor D.E.
Malaysia

*Corresponding author; email: zanariah@math.upm.edu.my

Received: 5 January 2010
Accepted: 29 November 2010