Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2013, Article ID 917589, 10 pages
http://dx.doi.org/10.1155/2013/917589

Research Article

New Algorithm of Two-Point Block Method for Solving Boundary Value Problem with Dirichlet and Neumann Boundary Conditions

Pei See Phang,[1] Zanariah Abdul Majid,[1,2] Fudziah Ismail,[1,2] Khairil Iskandar Othman,[2] and Mohamed Suleiman[2]

[1] *Department of Mathematics, Universiti Putra Malaysia, Selangor, 43400 Serdang, Malaysia*
[2] *Institute for Mathematical Research, Universiti Putra Malaysia, Selangor, 43400 Serdang, Malaysia*

Correspondence should be addressed to Zanariah Abdul Majid; zanariah@science.upm.edu.my

Received 14 January 2013; Accepted 9 March 2013

Academic Editor: X. Frank Xu

Two-point block method with variable step-size strategy is presented to obtain the solutions for boundary value problems directly. Dirichlet type and Neumann type of boundary conditions are studied in this paper. Multiple shooting techniques adapted with the three-step iterative method are employed for generating the guessing value. Six boundary value problems are solved using the proposed method, and the numerical results are compared to the existing methods. The results suggest a significant improvement in the efficiency of the proposed methods in terms of the number of steps, execution time, and accuracy.

1. Introduction

Numerical solutions for boundary value problem have great importance in scientific computation, as they were widely used to model the real-world problems. There are several methods that can be used to solve the two-point boundary value problems such as the Adomian decomposition method modified by Duan and Rach [1] and Ebaid [2] to solve boundary value problem. Based on the Adomian decomposition method, a new analytical and numerical treatment is introduced to investigate linear and nonlinear two-point boundary value problems. Lang and Xu [3] studied a new quintic B-spline collocation method for linear and nonlinear second-order boundary value problems. Islam and Shirin [4] used Bernoulli polynomials to find the numerical solutions of the second-order linear and nonlinear boundary value problems. Besides that, Liu [5] had solved the boundary value problem with Neumann type using polynomial spline approach. The numerical solutions of second-order boundary value problems by collocation method with the Haar wavelets were presented by Siraj-ul-Islam et al. [6]. A new kind of finite difference scheme presented for special second-order nonlinear two-point boundary value problem has been proposed by Erdogan and Ozis [7], while Prentice [8] considered the error control in a finite difference solution of a two-point boundary value problem.

In this paper, we are concerned for solving two-point second-order boundary value problems (BVPs) with two types of boundary condition, that is, Dirichlet and Neumann type. Two-point second-order boundary value problem is as follows:

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

Dirichlet boundary condition is as follows:

$$y(a) = \alpha, \qquad y(b) = \beta. \tag{2}$$

Neumann boundary condition is as follows:

$$y'(a) = \alpha, \qquad y'(b) = \beta. \tag{3}$$

The BVPs as (1) will be solved using two-point block method with variable step size. The block methods are commonly used to solve the ODEs such as in Majid et al. [9]. The authors have used the two-point block one-step method of

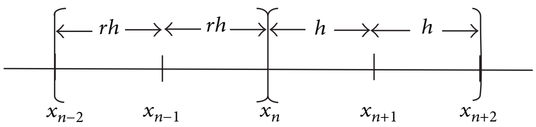

FIGURE 1: Two-point block method.

Runge-Kutta type to solve the general second-order ODEs with variable step-size strategy.

In Majid et al. [10], the authors have solved the BVPs of Dirichlet type only by two-point block method with constant step size. Hence, in this paper, we aim to extend the work in Majid et al. [10] for solving BVPs of Dirichlet and Neumann type using variable step size. This block method has the advantage to solve the second-order differential equation directly and obtain two approximate solutions simultaneously in block. We adapted the multiple shooting techniques to obtain the missing initial value, and the three-step iterative method proposed by Yun [11] was employed to generate the missing guessing value. The numerical computations have been performed using the C language.

2. Formulation of the Two-Point Block Method

In Figure 1, we have divided the interval $[a, b]$ into a series of blocks with each block containing two points. Both approximate solutions are simultaneously found using the same back values. The approximate solution of y_{n+1} and y_{n+2} at the point, x_{n+1} and x_{n+2}, respectively, with step size h will be computed simultaneously using three back values at the points, x_n, x_{n-1} and x_{n-2} with step size rh. The value, y_{n+1} and y_{n+2} will be obtained by integrate (1) once and twice over the intervals $[x_n, x_{n+1}]$ and $[x_n, x_{n+2}]$, respectively, as follows

First point is

$$\begin{aligned} y'(x_{n+1}) - y'(x_n) &= \int_{x_n}^{x_{n+1}} f(x, y, y')\, dx, \\ y(x_{n+1}) - y(x_n) - hy'(x_n) & \\ &= \int_{x_n}^{x_{n+1}} (x_{n+1} - x) f(x, y, y')\, dx. \end{aligned} \tag{4}$$

Second point is

$$\begin{aligned} y'(x_{n+2}) - y'(x_n) &= \int_{x_n}^{x_{n+2}} f(x, y, y')\, dx, \\ y(x_{n+2}) - y(x_n) - 2hy'(x_n) & \\ &= \int_{x_n}^{x_{n+2}} (x_{n+2} - x) f(x, y, y')\, dx. \end{aligned} \tag{5}$$

The method is derived using Lagrange interpolation polynomial and the five interpolating points; that is, (x_{n-2}, f_{n-2}), (x_{n-1}, f_{n-1}), (x_n, f_n), (x_{n+1}, f_{n+1}), and (x_{n+2}, f_{n+2}) are involved in the corrector formulae of the two-point block method. The function $f(x, y, y')$ in (4) and (5) will be replaced by Lagrange interpolating polynomial. Let $s = (x - x_{n+2})/h$, replacing $dx = hds$ and changing the limit of integration from -2 to -1 for the first point and changing the limit of integration to -2 and 0 for the second point.

Evaluate these integrals using MAPLE, the corrector formulae of the two-point block method will be obtained as follows.

First point is

$$\begin{aligned} y'_{n+1} = y'_n &+ \frac{h}{240(1+r)(2+r)(1+2r)r^2} \\ &\times \Big[\left(14 + 37r + 15r^2\right) f_{n-2} \\ &\quad + \left(-28 - 176r - 240r^2\right) f_{n-1} \\ &\quad + \big(14 + 139r + 564r^2 + 1029r^3 \\ &\qquad + 790r^4 + 200r^5\big) f_n \\ &\quad + \left(144r^2 + 672r^3 + 940r^4 + 320r^5\right) f_{n+1} \\ &\quad + \left(-3r^2 - 21r^3 - 50r^4 - 40r^5\right) f_{n+2}\Big], \\ y_{n+1} = y_n + hy'_n &+ \frac{h^2}{240(2+r)(1+r)(1+2r)r^2} \\ &\times \Big[\left(6 + 19r + 8r^2\right) f_{n-2} \\ &\quad + \left(-12 - 88r - 128r^2\right) f_{n-1} \\ &\quad + \left(6 + 69r + 329r^2 + 644r^3 + 538r^4 + 140r^5\right) f_n \\ &\quad + \left(32r^2 + 184r^3 + 324r^4 + 120r^5\right) f_{n+1} \\ &\quad + \left(-r^2 - 8r^3 - 22r^4 - 20r^5\right) f_{n+2}\Big]. \end{aligned} \tag{6}$$

Second point is

$$\begin{aligned} y'_{n+2} = y'_n &+ \frac{h}{15(2+r)(1+r)(1+2r)r^2} \\ &\times \Big[(-2 - r) f_{n-2} + (4 - 8r) f_{n-1} \\ &\quad + \left(-2 - 7r + 3r^2 + 33r^3 + 35r^4 + 10r^5\right) f_n \\ &\quad + \left(48r^2 + 144r^3 + 140r^4 + 40r^5\right) f_{n+1} \\ &\quad + \left(9r^2 + 33r^3 + 35r^4 + 10r^5\right) f_{n+2}\Big], \\ y_{n+2} = y_n + 2hy'_n &+ \frac{h^2}{15r(2+r)(1+r)(1+2r)} \\ &\times \Big[(2 + r) f_{n-2} + (-8 - 16r) f_{n-1} \\ &\quad + \left(6 + 41r + 91r^2 + 76r^3 + 20r^4\right) f_n \end{aligned}$$

$$+\left(32r+112r^2+128r^3+40r^4\right)f_{n+1}$$
$$+\left(2r+7r^2+6r^3\right)f_{n+2}\Big]. \tag{7}$$

The method is the combination of predictor which is one order less than the corrector. The two-point block method with variable step-size strategy will be implemented for solving the boundary value problems via multiple shooting techniques.

3. Analysis of the Method

In this section, stability analysis, stability region, order of the method, and error constant of the two-point block method are discussed.

3.1. Stability Analysis

Definition 1. The method is zero stable provided the roots R_j of the first characteristic polynomial $\rho(R)$ specified as $\rho(R)=\det[\sum_{i=0}^{k}A^{(i)}R^{k-i}]=0$ satisfy $|R_j|\le 1$.

Substitute $r=1$ into (7), and rewrite in matrix from:

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}y_{n+2}\\y'_{n+2}\\y_{n+1}\\y'_{n+1}\end{bmatrix}
=\begin{bmatrix}1&0&0&0\\0&1&0&0\\1&0&0&0\\0&1&0&0\end{bmatrix}\begin{bmatrix}y_n\\y'_n\\y_{n-1}\\y'_{n-1}\end{bmatrix}+h\begin{bmatrix}0&2&0&0\\0&0&0&0\\0&1&0&0\\0&0&0&0\end{bmatrix}\begin{bmatrix}y_n\\y'_n\\y_{n-1}\\y'_{n-1}\end{bmatrix}$$
$$+h\begin{bmatrix}0&0&0&0\\\frac{29}{90}&\frac{124}{90}&\frac{24}{90}&\frac{4}{90}\\0&0&0&0\\\frac{-19}{720}&\frac{346}{720}&\frac{456}{720}&\frac{-74}{720}\end{bmatrix}\begin{bmatrix}f_{n+2}\\f_{n+1}\\f_n\\f_{n-1}\end{bmatrix}
+h\begin{bmatrix}0&0&0&0\\\frac{-1}{90}&0&0&0\\0&0&0&0\\\frac{11}{720}&0&0&0\end{bmatrix}\begin{bmatrix}f_{n-2}\\f_{n-3}\\f_{n-4}\\f_{n-5}\end{bmatrix}$$
$$+h^2\begin{bmatrix}\frac{5}{90}&\frac{104}{90}&\frac{78}{90}&\frac{-8}{90}\\0&0&0&0\\\frac{-17}{1440}&\frac{220}{1440}&\frac{582}{1440}&\frac{-76}{1440}\\0&0&0&0\end{bmatrix}\begin{bmatrix}f_{n-2}\\f_{n-1}\\f_n\\f_{n-1}\end{bmatrix}
+h^2\begin{bmatrix}\frac{1}{90}&0&0&0\\0&0&0&0\\\frac{11}{1440}&0&0&0\\0&0&0&0\end{bmatrix}\begin{bmatrix}f_{n-2}\\f_{n-3}\\f_{n-4}\\f_{n-5}\end{bmatrix}. \tag{8}$$

The first characteristic polynomial of the two-point block method is given as follows:

$$\rho(R)=\det\left[RA^0-A^1\right]=0,$$
$$\text{where } A^0=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix},\qquad A^1=\begin{bmatrix}1&0&0&0\\0&1&0&0\\1&0&0&0\\0&1&0&0\end{bmatrix},$$
$$\rho(R)=\det\begin{bmatrix}R-1&0&0&0\\0&R-1&0&0\\-1&0&R&0\\0&-1&0&R\end{bmatrix},$$
$$0=(R-1)^2R^2,\ \ R=0,0,1,1,\ \ |R|\le 1. \tag{9}$$

From Definition 1 and (9) the two-point one block method is zero stable.

3.2. Stability Region. The stability polynomial of the two-point block method applied to the test equation is as follows:

$$y''=f=\theta y'+\lambda y. \tag{10}$$

The stability polynomials of two-point block method are as follows.

For $r=0.5$:

$$\left(1-\frac{147}{200}H_1+\frac{847}{5400}H_1^2-\frac{83}{450}H_2+\frac{127}{1350}H_1H_2+\frac{211}{13500}H_2^2\right)t^8$$
$$+\left(-2+\frac{19}{100}H_1-\frac{4307}{2700}H_1^2\right)t^7$$
$$\times\left(-\frac{839}{225}H_2-\frac{1633}{1350}H_1H_2-\frac{749}{1125}H_2^2\right)t^7$$
$$+\left(1-\frac{1}{24}H_1+\frac{28463}{27000}H_1^2+\frac{149}{450}H_2\right)t^6$$
$$+\left(-\frac{12719}{6750}H_1H_2+\frac{5611}{13500}H_2^2\right)t^6$$
$$+\left(\frac{44}{75}H_1+\frac{99}{250}H_1^2-\frac{34}{225}H_2-\frac{719}{2250}H_1H_2+\frac{19}{150}H_2^2\right)t^5$$
$$+\left(-\frac{8}{675}H_1^2+\frac{8}{675}H_1H_2-\frac{8}{3375}H_2^2\right)t^4=0, \tag{11}$$

where $H_1=h^2\lambda$ and $H_2=h^2\theta$.

For $r = 1.0$:

$$\left(1-\frac{289}{360}H_1+\frac{413}{2160}H_1^2-\frac{5}{24}H_2+\frac{53}{432}H_1H_2+\frac{239}{10800}H_2^2\right)t^8$$
$$+\left(-2-\frac{31}{120}H_1-\frac{209}{216}H_1^2\right)t^7$$
$$+\left(-\frac{1307}{360}H_2-\frac{161}{270}H_1H_2-\frac{473}{900}H_2^2\right)t^7$$
$$+\left(1+\frac{37}{40}H_1+\frac{25}{36}H_1^2-\frac{41}{360}H_2\right.$$
$$\left.-\frac{283}{360}H_1H_2+\frac{937}{5400}H_2^2\right)t^6+\frac{937}{5400}H_2^2t^6$$
$$+\left(\frac{49}{360}H_1+\frac{89}{1080}H_1^2-\frac{17}{360}H_2-\frac{2}{27}H_1H_2+\frac{17}{900}H_2^2\right)t^5$$
$$-\frac{1}{2160}H_1^2t^4+\left(\frac{1}{2160}H_1H_2-\frac{1}{10800}H_2^2\right)t^4=0, \tag{12}$$

where $H_1 = h^2\lambda$ and $H_2 = h^2\theta$.

For $r = 2.0$:

$$\left(1-\frac{87}{100}H_1+\frac{623}{2700}H_1^2-\frac{103}{450}H_2+\frac{847}{5400}H_1H_2+\frac{137}{4500}H_2^2\right)t^8$$
$$+\left(-2-\frac{223}{960}H_1-\frac{517}{675}H_1^2\right)t^7$$
$$+\left(-\frac{51217}{14400}H_2-\frac{51}{160}H_1H_2-\frac{102029}{216000}H_2^2\right)t^7$$
$$+\left(1+\frac{2579}{2400}H_1+\frac{15001}{28800}H_1^2-\frac{1463}{7200}H_2\right)t^6$$
$$+\left(-\frac{40001}{86400}H_1H_2+\frac{48139}{432000}H_2^2\right)t^6$$
$$+\left(\frac{133}{4800}H_1+\frac{619}{43200}H_1^2-\frac{161}{14400}H_2-\frac{91}{7200}H_1H_2\right)t^5$$
$$+\frac{161}{54000}H_2^2t^5$$
$$+\left(-\frac{1}{86400}H_1^2+\frac{1}{86400}H_1H_2-\frac{1}{432000}H_2^2\right)t^4=0, \tag{13}$$

where $H_1 = h^2\lambda$ and $H_2 = h^2\theta$.

Figure 2 shows the regions of absolute stability for the two-point block method when the step-size ratio $r = 0.5, 1.0$, and 2.0. The stability region is the bounded shaded region and the region is larger as the step-size ratio increases. This is expected since smaller step-size will give larger step size ratio.

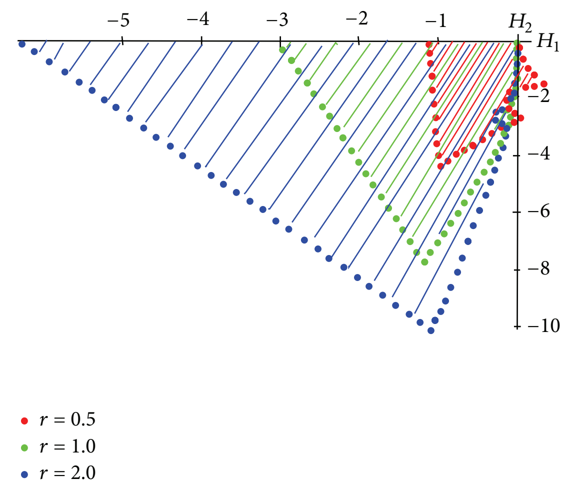

Figure 2: Stability region of the two-point block method.

3.3. Order and Error Constant

Definition 2. The block method is

$$\sum_{j=0}^{k}\alpha_j y_{n+j}-h\sum_{j=0}^{k}\beta_j y'_{n+j}-h^2\sum_{j=0}^{k}\gamma_j y''_{n+j} = C_{p+1}y^{p+1}+O\left(h^{p+2}\right), \tag{14}$$

where

$$C_p=\frac{1}{p!}\sum_{j=0}^{k}j^p\alpha_j-\frac{1}{(p-1)!}\sum_{j=0}^{k}j^{p-1}\beta_j-\frac{1}{(p-2)!}\sum_{j=0}^{k}j^{p-2}\gamma_j,$$
$$C_0=C_1=C_2=\cdots=C_{p+1}=0, \qquad C_{p+2}\neq 0 \tag{15}$$

p is the order of the LMM method, and $O(h^{p+2})$ is the local truncation error.

Substituting $r = 1$ into (7) and apply the formulae, we obtaining

$$C_0=C_1=\cdots=C_6=0,$$
$$C_7=\left[\frac{11}{1440}\quad\frac{37}{10080}\quad\frac{-1}{90}\quad\frac{1}{315}\right]^T. \tag{16}$$

From Definition 2, the order of two-point block method is five with error constant

$$C_7=\left[\frac{11}{1440}\quad\frac{37}{10080}\quad\frac{-1}{90}\quad\frac{1}{315}\right]^T. \tag{17}$$

4. Implementation of the Method

4.1. Multiple Shooting Technique. The initial conditions being imposed at the same point in the independent variable x, but the boundary conditions are imposed at different values of the independent variable. The idea of shooting technique is to form the initial condition from the boundary condition with

the guessing value. Multiple shooting techniques are indeed a combination of several shooting techniques by dividing the given interval $a \le x \le b$ into jth subinterval.

4.1.1. Dirichlet-Type Boundary Condition. The missing initial condition is $y'(a)$. Equation (1) can be written as

$$\frac{d^2 y_j}{dx^2} = f_j\left(x, y_j, y_j'\right), \tag{18}$$

with conditions

$$\begin{aligned}
&1{:}y_1(a) = \alpha, \quad y_1'(a) = s_v \\
&2{:}y_2(x_1) = y(x_1, s_{v-1}), \quad y_2'(x_1) = y'(x_1, s_{v-1}) \\
&\quad\vdots \\
&j{:}y_j\left(x_{j-1}\right) = y\left(x_{j-1}, s_{v-1}\right), \quad y_j'\left(x_{j-1}\right) = y'\left(x_{j-1}, s_{v-1}\right).
\end{aligned} \tag{19}$$

Therefore, we obtain the jth stopping conditions as follows:

$$\begin{gathered}
\left|y(x_1, s_v) - y(x_1, s_{v-1})\right| \le \varepsilon \\
\left|y(x_2, s_v) - y(x_2, s_{v-1})\right| \le \varepsilon \\
\vdots \\
\left|y\left(x_j, s_v\right) - \beta\right| \le \varepsilon.
\end{gathered} \tag{20}$$

The iteration is repeated until we reached the stopping conditions, and the value of s_v will be generated by the three-step iterative method as follows:

$$\begin{aligned}
T_v &= s_v - \frac{y(b, s_v) - \beta}{z(b, s_v)}, \\
U_v &= -\frac{y(b, T_v) - \beta}{z(b, s_v)}, \\
s_{v+1} &= T_v - \frac{y(b, s_v) - \beta}{z(b, s_v)} - \frac{y(b, T_v + U_v) - \beta}{z(b, s_v)},
\end{aligned} \tag{21}$$

where $z(b, s_v)$ is the solution of

$$\begin{gathered}
z'' = \frac{\partial f\left(x, y, y'\right)}{\partial y} z + \frac{\partial f\left(x, y, y'\right)}{\partial y'} z', \\
z(a) = 0, \ z'(a) = 1.
\end{gathered} \tag{22}$$

4.1.2. Neumann-Type Boundary Condition. The missing initial condition is $y(a)$. Equation (1) can be written as

$$\frac{d^2 y_j}{dx^2} = f_j\left(x, y_j, y_j'\right), \tag{23}$$

with conditions given as in (19), but condition 1 will be replaced by

$$y_1(a) = s_v, \qquad y_1'(a) = \alpha. \tag{24}$$

The stopping conditions are

$$\begin{gathered}
\left|y'(x_1, s_v) - y'(x_1, s_{v-1})\right| \le \varepsilon \\
\left|y'(x_2, s_v) - y'(x_2, s_{v-1})\right| \le \varepsilon \\
\vdots \\
\left|y'\left(x_j, s_v\right) - \beta\right| \le \varepsilon.
\end{gathered} \tag{25}$$

The three-step iterative method is

$$\begin{aligned}
T_v &= s_v - \frac{y'(b, s_v) - \beta}{z'(b, s_v)}, \\
U_v &= -\frac{y'(b, T_v) - \beta}{z'(b, s_v)}, \\
s_{v+1} &= T_v - \frac{y'(b, s_v) - \beta}{z'(b, s_v)} - \frac{y'(b, T_v + U_v) - \beta}{z'(b, s_v)},
\end{aligned} \tag{26}$$

where the value of $z'(b, s_v)$ can be obtained from solving the z'' as follows:

$$\begin{gathered}
z'' = \frac{\partial f\left(x, y, y'\right)}{\partial y} z + \frac{\partial f\left(x, y, y'\right)}{\partial y'} z', \\
z(a) = 1, \ z'(a) = 0.
\end{gathered} \tag{27}$$

4.2. Variable Step-Size Strategy. The choices of the next step size will be restricted to half, doubled or the same as the current step size. The adjustment is based on the local truncation error (LTE). If the local truncation error is less or equal to the tolerance (TOL), the choice for the next step will be double or remain the same. In the code developed, when the next step size is doubled, the ratio r is 0.5, while the step size remains constant, r is 1. If this condition fails, the current step size will be halved from the previous step size; that is, r is 2, and the approximate solutions in the block will be recalculate.

Case 1. LTE $\le$ TOL (successful step).

Substituting $r = 1$ in (6) and (7) will produce the following corrector formulae:

$$\begin{aligned}
y_{n+1}' &= y_n' + \frac{h}{720} \\
&\quad \times \left(11 f_{n-2} - 74 f_{n-1} + 456 f_n + 346 f_{n+1} - 19 f_{n+2}\right), \\
y_{n+1} &= y_n + h y_n' + \frac{h^2}{1440} \\
&\quad \times \left(11 f_{n-2} - 76 f_{n-1} + 582 f_n + 220 f_{n+1} - 17 f_{n+2}\right),
\end{aligned}$$

$$y'_{n+2} = y'_n + \frac{h}{90} \times \left(-f_{n-2} + 4f_{n-1} + 24f_n + 124f_{n+1} + 29f_{n+2}\right),$$
$$y_{n+2} = y_n + 2hy'_n + \frac{h^2}{90} \times \left(f_{n-2} - 8f_{n-1} + 78f_n + 104f_{n+1} + 5f_{n+2}\right). \quad (28)$$

Substituting $r = 0.5$ in (6) and (7) will produce the following corrector formulae:

$$y'_{n+1} = y'_n + \frac{h}{1800} \times \left(145f_{n-2} - 704f_{n-1} + 1635f_n + 755f_{n+1} - 31f_{n+2}\right),$$
$$y_{n+1} = y_n + hy'_n + \frac{h^2}{1800} \times \left(70f_{n-2} - 352f_{n-1} + 975f_n + 220f_{n+1} - 13f_{n+2}\right),$$
$$y'_{n+2} = y'_n + \frac{h}{225} \times \left(-20f_{n-2} + 64f_{n-1} + 15f_n + 320f_{n+1} + 71f_{n+2}\right),$$
$$y_{n+2} = y_n + 2hy'_n + \frac{2h^2}{225} \times \left(5f_{n-2} - 32f_{n-1} + 120f_n + 125f_{n+1} + 7f_{n+2}\right). \quad (29)$$

Case 2. LTE > TOL (failure step).

Substituting $r = 2$ in (6) and (7) will produce the following corrector formulae:

$$y'_{n+1} = y'_n + \frac{h}{14400} \times \left(37f_{n-2} - 335f_{n-1} + 7456f_n + 7808f_{n+1} - 565f_{n+2}\right),$$
$$y_{n+1} = y_n + hy'_n + \frac{h^2}{14400} \times \left(19f_{n-2} - 175f_{n-1} + 4965f_n + 2656f_{n+1} - 265f_{n+2}\right),$$
$$y'_{n+2} = y'_n + \frac{h}{900} \times \left(295f_{n-2} + 1216f_{n-1} + 285f_n + 5f_{n+1} - f_{n+2}\right),$$
$$y_{n+2} = y_n + 2hy'_n + \frac{h^2}{450} \times \left(f_{n-2} - 10f_{n-1} + 345f_n + 544f_{n+1} + 20f_{n+2}\right). \quad (30)$$

4.3. Algorithm 2P1BVS

Step 1. Set TOL and s_0.

Step 2. For $n = 1, 2, 3$.

set $x_n = a + nh$, and evaluate the approximate values y_n and z_n with direct Adams-Bashforth method.
Compute functions f_n and z''_n.

Step 3. While $x_n < b$, do Step 4.

Step 4. For $i = 1, 2$.

set $x_{n+i} = x_n + ih$.
Evaluate the approximate values y_{n+i} and z_{n+i} with two-point block method.
Compute functions f_{n+i} and z''_{n+i}.

Step 5. If fulfill stopping condition, go to Step 8.

Step 6. Generate the new guessing values s_v by three-step iterative method. Set $v = v + 1$.

Step 7. If $v \geq 10$, set $b = (a + b)/2$, and go to Step 2.

Step 8. Complete.

This algorithm was developed in C language.

5. Problem Tested

In this section, we have tested the algorithm 2P1BVS to six numerical examples to illustrate its accuracy and efficiency.

Problem 1. We have

$$y'' = y' - \sin y - y^2 + x^4 - 2x^3 + 3 + \sin\left(x^2 - x\right), \quad 0 \leq x \leq 1. \quad (31)$$

Dirichlet boundary condition is as follows: $y(0) = 0$, $y(1) = 0$.

Exact solution is as follows: $y = x^2 - x$.

Problem 2. We have

$$y'' = -y^2 + \sin^2(\pi x) - \pi^2 \sin(\pi x), \quad 0 \leq x \leq 1. \quad (32)$$

Dirichlet boundary condition is as follows: $y(0) = 0$, $y(1) = 0$.

Exact solution is as follows: $y = \sin(\pi x)$.

Problem 3. We have

$$y'' = -\left(2 - 4x^2\right) y, \quad 0 \leq x \leq 1. \quad (33)$$

Neumann boundary condition is as follows: $y'(0) = 0$, $y'(1) = -(-2/\exp(1))$.

Exact solution is as follows: $y = \exp(-x^2)$.

Problem 4. We have

$$y'' = 2y^3, \quad 0 \le x \le 1. \tag{34}$$

Neumann boundary condition is as follows: $y'(0) = -1$, $y'(1) = -1/4$.

Exact solution is as follows: $y = 1/(1 + x)$.

Problem 5. We have

$$y'' = -y - 1, \quad 0 \le x \le b. \tag{35}$$

Neumann boundary condition: $y'(0) = (1 - \cos(1))/\sin(1)$, $y'(b) = \cos(b) + ((1 - \cos(1))/\sin(1))\sin(b) - 1$.

Exact solution is as follows: $y = \cos(x) + ((1 - \cos(1))/\sin(1))\sin(x) - 1$.

Problem 6 (Troesch's Problem). We have

$$y'' = \lambda \sinh(\lambda y), \quad 0 \le x \le b. \tag{36}$$

Dirichlet boundary condition is as follows: $y(0) = 0$, $y(1) = 1$.

Troesch's problem comes from the investigation of the confinement of a plasma column under radiation pressure. The closed form solution to this problem has been given in terms of Jacobian elliptic function by Lin et al. [12].

6. Numerical Result and Discussion

The following notations are used in the tables:

TOL: Tolerance

TS: Total number of steps

MAXE: Maximum error

Time (sec): Execution time in second

TFC: Total function calls

2P1BVS: Two-point block method with variable step size via multiple shooting technique adapted with three-step iterative method

MLAM: Multilevel augmentation method proposed by Chen [13]

COLHW: Collocation method with Haar wavelets proposed by Siraj-ul-Islam [6]

FDM: Finite different method proposed by Erdogan and Ozis [7].

In Problems 1 and 2, we solved the boundary value problem with Dirichlet-type boundary conditions by 2P1BVS and compare our result with MLAM. In Table 1, the 2P1BVS has superiority in terms of accuracy and execution time when compared to MLAM. For example, the maximum error for the 2P1BVS and MLAM with 15 steps is $1.94e-16$ and $2.42e-4$ respectively. This result is expected since 2P1BVS solves the second-order BVPs directly and obtains two approximate solutions simultaneously. In Table 2, we observed that the 2P1BVS can obtain the maximum error $1.07e-10$ with 57 steps, but MLAM can obtain the maximum error $5.00e-8$ with 255 steps.

In Problems 3 and 4, 2P1BVS solved the boundary value problem with Neumann-type boundary conditions when was compared to COLHW. The maximum errors for 2P1BVS in Tables 3 and 4 are comparable to COLHW for the larger tolerance. For example, in Table 3, the maximum error for 2P1BVS with 12 steps and COLHW with 16 steps is $3.64e-4$ and $2.91e-4$, respectively. As the tolerance is getting smaller, the maximum error and total number of steps for 2P1BVS

Table 1: Comparison of the numerical result for solving Problem 1.

	2P1BVS			MLAM		
TOL	TS	MAXE	Time (sec)	TS	MAXE	Time (sec)
$1.00e-2$	12	$5.34e-5$	0.000118	15	$7.43e-4$	1.719
$1.00e-4$	15	$1.94e-16$	0.000130	31	$2.42e-4$	3.750
$1.00e-6$	19	$1.30e-16$	0.000162	63	$6.06e-5$	8.141
$1.00e-8$	22	$1.80e-16$	0.000169	127	$1.51e-5$	16.875
$1.00e-10$	25	$6.71e-17$	0.000203	255	$3.78e-6$	37.328
				511	$9.45e-7$	85.016
				1023	$2.36e-7$	190.12
				2047	$6.76e-8$	435.23

Table 2: Comparison of the numerical result for solving Problem 2.

	2P1BVS			MLAM		
TOL	TS	MAXE	Time (sec)	TS	MAXE	Time (sec)
$1.00e-2$	11	$7.66e-5$	0.000141	15	$2.46e-4$	0.109
$1.00e-4$	16	$2.47e-6$	0.000162	31	$3.94e-5$	0.235
$1.00e-6$	26	$3.55e-8$	0.000227	63	$4.94e-6$	0.328
$1.00e-8$	32	$1.16e-8$	0.000275	127	$6.17e-7$	0.687
$1.00e-10$	57	$1.07e-10$	0.000444	255	$5.00e-8$	1.953

Table 3: Comparison of the numerical result for solving Problem 3.

	2P1BVS			COLHW	
TOL	TS	MAXE	Time (sec)	TS	MAXE
$1.00e-2$	12	$3.64e-4$	0.000092	16	$2.91e-4$
$1.00e-4$	16	$3.83e-5$	0.000111	32	$7.48e-5$
$1.00e-6$	23	$5.36e-7$	0.000149	64	$1.90e-5$
$1.00e-8$	32	$6.31e-8$	0.000193	128	$4.77e-6$
$1.00e-10$	54	$1.09e-9$	0.000306	256	$1.20e-6$

Table 4: Comparison of the numerical result for solving Problem 4.

	2P1BVS			COLHW	
TOL	TS	MAXE	Time (sec)	TS	MAXE
$1.00e-2$	12	$1.72e-5$	0.000090	16	$5.95e-4$
$1.00e-4$	16	$1.33e-5$	0.000109	32	$1.55e-4$
$1.00e-6$	23	$2.55e-7$	0.000141	64	$3.98e-5$
$1.00e-8$	36	$6.20e-9$	0.000218	128	$1.01e-5$
$1.00e-10$	59	$1.72e-10$	0.000315	256	$2.53e-6$

Table 5: Numerical result for solving Problem 5 with different endpoint, b.

TOL	$b = 5$				$b = 10$			
	TS	TFC	MAXE	Time (sec)	TS	TFC	MAXE	Time (sec)
$1.00e-2$	14	81	$2.40e-3$	0.000116	17	111	$4.35e-2$	0.000139
$1.00e-4$	22	129	$3.49e-5$	0.000177	28	177	$3.64e-4$	0.000210
$1.00e-6$	31	179	$1.54e-6$	0.000245	46	273	$3.10e-6$	0.000322
$1.00e-8$	55	323	$8.71e-8$	0.000376	92	553	$2.99e-7$	0.000577
$1.00e-10$	87	509	$3.38e-9$	0.000551	175	1045	$5.57e-9$	0.001083

Table 6: Comparison of the numerical result for solving Troesch's Problem with $\lambda = 0.5$.

x	Solution			Error	
	Exact	FDM	2P1BVS	FDM	2P1BVS
0.1	0.0959443493	0.0959443492	0.0959443493	$5.35e-11$	0.00
0.2	0.1921287477	0.1921287476	0.1921287477	$1.04\ e-10$	0.00
0.3	0.2887944009	0.2887944007	0.2887944009	$1.47\ e-10$	0.00
0.4	0.3861848464	0.3861848462	0.3861848464	$1.80\ e-10$	$9.99e-16$
0.5	0.4845471647	0.4845471645	0.4845471647	$1.99\ e-10$	$3.60e-14$
0.6	0.5841332484	0.5841332482	0.5841332484	$2.02\ e-10$	$5.61e-13$
0.7	0.6852011483	0.6852011481	0.6852011483	$1.86\ e-10$	$5.79e-12$
0.8	0.7880165227	0.7880165225	0.7880165227	$1.49\ e-10$	$4.41e-11$
0.9	0.8928542161	0.8928542161	0.8928542158	$9.12e-11$	$2.66e-10$

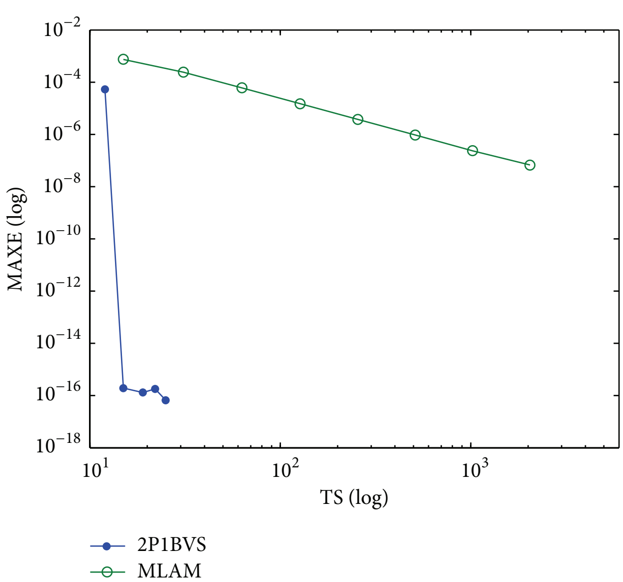

Figure 3: Comparison of maximum error and total step for Problem 1.

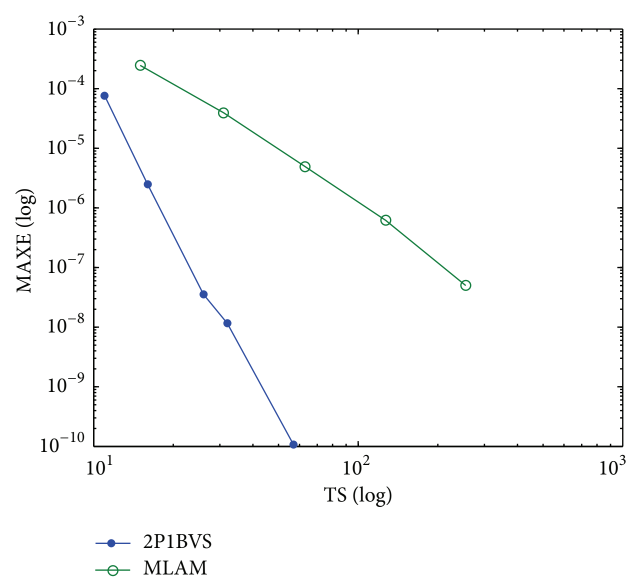

Figure 4: Comparison of maximum error and total step for problem 2.

are setting better than COLHW. In Table 4, the maximum error for 2P1BVS with 108 steps and COLHW with 512 steps is $3.47e-12$ and $6.34e-7$, respectively.

Table 5 displays the numerical results for Problem 5 with larger interval at $b = 5$ and $b = 10$ for solving Problem 5. The 2P1BVS managed to obtain the accuracy within the given tolerances. We also observed that the 2P1BVS manages to converge rapidly for the larger interval, for example, at TOL = 10^{-10}, the execution time is 0.000551 and 0.001083 seconds when $b = 5$ and $b = 10$, respectively.

We consider the cases when $\lambda = 0.5$ and $\lambda = 1.0$ to solve the Troesch problem at TOL = 10^{-5}. The numerical results in Tables 6 and 7 show that the 2P1BVS manage to solve the problem as accurate as or better than FDM. Figures 3, 4, 5, and 6 display the comparison of the maximum errors versus the total steps for the numerical results in Tables 1–7.

Table 7: Comparison of the numerical result for solving Troesch's Problem with $\lambda = 1.0$.

x	Solution			Error	
	Exact	FDM	2P1BVS	FDM	2P1BVS
0.1	0.0846612565	0.0846612556	0.0846612565	$9.30e-10$	0.00
0.2	0.1701713582	0.1701713565	0.1701713582	$1.70e-9$	0.00
0.3	0.2573939080	0.2573939059	0.2573939080	$2.18e-9$	$1.90e-13$
0.4	0.3472228551	0.3472228528	0.3472228551	$2.23e-9$	$1.48e-11$
0.5	0.4405998351	0.4405998333	0.4405998347	$1.80e-9$	$4.46e-10$
0.6	0.5385343980	0.5385343971	0.5385343906	$9.10e-10$	$7.37e-09$
0.7	0.6421286091	0.6421286094	0.6421285278	$3.02e-10$	$8.13e-08$
0.8	0.7526080939	0.7526080954	0.7526080937	$1.47e-9$	$2.43e-10$
0.9	0.8713625196	0.8713625215	0.8713625152	$1.87e-9$	$4.42e-09$

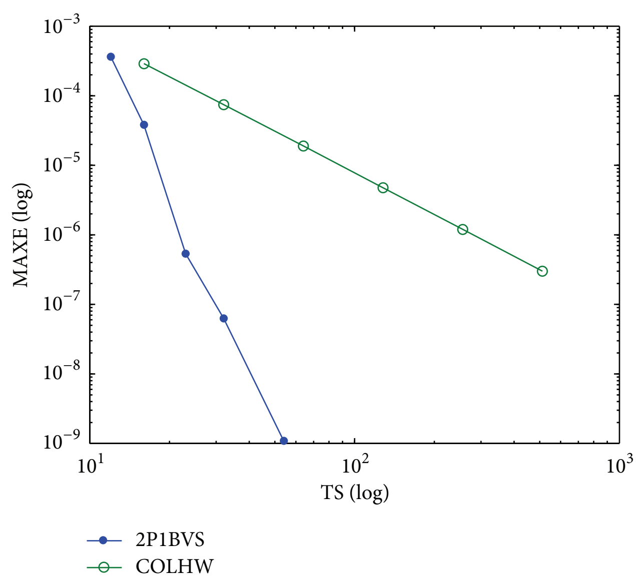

Figure 5: Comparison of maximum error and total step for problem 3.

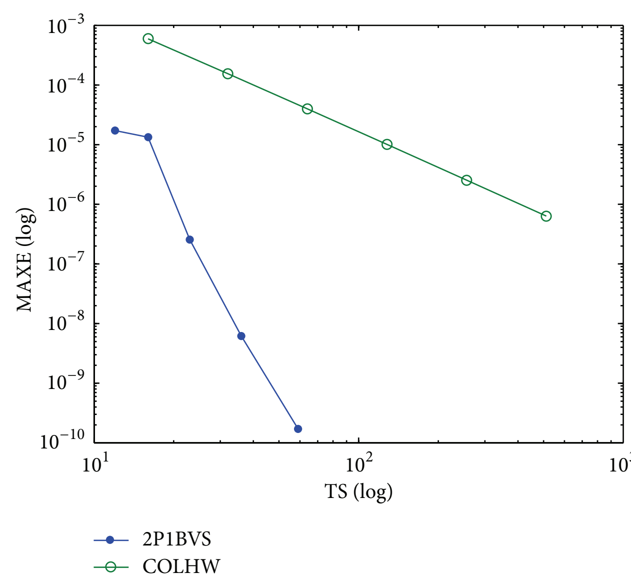

Figure 6: Comparison of maximum error and total step for problem 4.

7. Conclusion

In this paper, we have shown that the proposed two-point block method using variable step size (2P1BVS) is suitable for solving directly two-point second-order boundary value problems in Dirichlet- and Neumann-type-boundary conditions.

Acknowledgments

The author gratefully acknowledged the financial support of Fundamental Research Grant Scheme (FRGS) and MyPhD. scholarship from the Ministry of Higher Education Malaysia.

References

[1] J.-S. Duan and R. Rach, "A new modification of the Adomian decomposition method for solving boundary value problems for higher order nonlinear differential equations," *Applied Mathematics and Computation*, vol. 218, no. 8, pp. 4090–4118, 2011.

[2] A. Ebaid, "A new analytical and numerical treatment for singular two-point boundary value problems via the Adomian decomposition method," *Journal of Computational and Applied Mathematics*, vol. 235, no. 8, pp. 1914–1924, 2011.

[3] F.-G. Lang and X.-P. Xu, "Quintic B-spline collocation method for second order mixed boundary value problem," *Computer Physics Communications*, vol. 183, no. 4, pp. 913–921, 2012.

[4] Md. S. Islam and A. Shirin, "Numerical solutions of a class of second order boundary value problems on using Bernoulli polynomials," *Applied Mathematics*, vol. 2, no. 9, pp. 1059–1067, 2011.

[5] L.-B. Liu, H.-W. Liu, and Y. Chen, "Polynomial spline approach for solving second-order boundary-value problems with Neumann conditions," *Applied Mathematics and Computation*, vol. 217, no. 16, pp. 6872–6882, 2011.

[6] Siraj-ul-Islam, I. Aziz, and B. Šarler, "The numerical solution of second-order boundary-value problems by collocation method with the Haar wavelets," *Mathematical and Computer Modelling*, vol. 52, no. 9-10, pp. 1577–1590, 2010.

[7] U. Erdogan and T. Ozis, "A smart nonstandard finite difference scheme for second order nonlinear boundary value problems,"

Journal of Computational Physics, vol. 230, no. 17, pp. 6464–6474, 2011.

[8] J. S. C. Prentice, "Relative error control in finite-difference solutions of two-point boundary-value problems," *Applied Mathematical Sciences*, vol. 6, no. 17-20, pp. 901–911, 2012.

[9] Z. A. Majid, N. Z. Mokhtar, and M. Suleiman, "Direct two-point block one-step method for solving general second-order ordinary differential equations," *Mathematical Problems in Engineering*, vol. 2012, Article ID 184253, 16 pages, 2012.

[10] Z. A. Majid, P. S. Phang, and M. Suleiman, "Application of block method for solving nonlinear two point boundary value problem," *Advanced Science Letters*, vol. 13, no. 1, pp. 54–757, 2012.

[11] J. H. Yun, "A note on three-step iterative method for nonlinear equations," *Applied Mathematics and Computation*, vol. 202, no. 1, pp. 401–405, 2008.

[12] Y. Lin, J. A. Enszer, and M. A. Stadtherr, "Enclosing all solutions of two-point boundary value problems for ODEs," *Computers and Chemical Engineering*, vol. 32, pp. 1714–1725, 2008.

[13] J. Chen, "Fast multilevel augmentation methods for nonlinear boundary value problems," *Computers & Mathematics with Applications*, vol. 61, no. 3, pp. 612–619, 2011.