Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 2056834, 12 pages
https://doi.org/10.1155/2018/2056834

Research Article

Diagonal Block Method for Solving Two-Point Boundary Value Problems with Robin Boundary Conditions

Nadirah Mohd Nasir,[1,2] Zanariah Abdul Majid ,[1,3] Fudziah Ismail ,[1,3] and Norfifah Bachok [1,3]

[1] *Institute for Mathematical Research, Universiti Putra Malaysia, 43400 Serdang, Selangor Darul Ehsan, Malaysia*
[2] *Faculty of Industrial Sciences & Technology, Universiti Malaysia Pahang, 26300 Gambang, Pahang, Malaysia*
[3] *Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400 Serdang, Selangor Darul Ehsan, Malaysia*

Correspondence should be addressed to Zanariah Abdul Majid; zana_majid99@yahoo.com

Received 3 April 2018; Accepted 2 September 2018; Published 2 October 2018

Academic Editor: Nunzio Salerno

This numerical study presents the diagonal block method of order four for solving the second-order boundary value problems (BVPs) with Robin boundary conditions at two-point concurrently using constant step size. The solution is obtained directly without reducing to a system of first-order differential equations using a combination of predictor-corrector mode via shooting technique. The shooting method was adapted with the Newton divided difference interpolation approach as the strategy of seeking for the new initial estimate. Five numerical examples are included to examine and illustrate the practical usefulness of the proposed method. Numerical tested problem is also highlighted on the diffusion of heat generated application that imposed the Robin boundary conditions. The present findings revealed that the proposed method gives an efficient performance in terms of accuracy, total function calls, and execution time as compared with the existing method.

1. Introduction

This study is focusing on the numerical approach for solving second-order boundary value problems (BVPs) associated with Robin boundary conditions. Generally, this type of BVPs is given as follows:

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

with

$$\begin{aligned} c_1 y'(a) + c_2 y(a) &= \alpha, \\ c_3 y'(b) + c_4 y(b) &= \beta \end{aligned} \tag{2}$$

where $a, b, c_1, c_2, c_3, c_4, \alpha$, and β are all constants and c_1, c_2, c_3, and c_4 all nonzero. In the case of Robin type, both functional value and derivative of the solutions are given in (2). If only functional value is given, then condition in (2) is known as Dirichlet type; otherwise BVPs will be subject to Neumann condition when only derivative values exist. Robin boundary conditions arise in several branches of applications such as in electromagnetic problem and heat transfer problem where these Robin type conditions are called impedance boundary conditions and the convective boundary conditions are, respectively, as explained in [1]. The Bernoulli polynomial together with Galerkin approximation in solving linear and nonlinear Robin boundary condition problems had been studied by Islam and Shirin [2]. The eminent scholars including Duan et al. [3] and Rach et al. [4] had derived the Adomian decomposition methods for solving BVPs imposing this condition where the approach involved analytic stage and numeric simulations. Meanwhile, Bhatta and Sastri [5] and Lang and Xu [6] had transformed respective BVPs into discretization form and solved them using symmetric global (continuous) spline and Quintic B-spline collocation method, respectively.

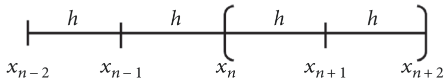


FIGURE 1: Two-point block method.

To optimize the computational cost, the development of proposed algorithm must at least satisfy the facility to generate solutions at several points simultaneously as suggested in Fatunla [7]. This implementation has been shown in the literature discussed by Phang et al. [8], Majid et al. [9], and Omar and Adeyeye [10]. They obtained the approximate solution of (1) at two points concurrently. Therefore, the advantages of outcome from their discussion have motivated us in this study.

Diagonal block method for solving differential equations was widely studied in the previous literatures. These include the discussion on solving first-order differential equations in [11, 12]. Solving second-order ordinary differential equations using diagonal block method has been discussed in [13]. The formulation in [12, 13] is based on backward differentiation approach. In [11], the author has derived the diagonal block method using Lagrange interpolation polynomial for solving first-order ordinary differential equations. In this research, we have extended the derivation in [11] to obtain the formulation of direct integration for solving second-order differential equations. Then the new formulae obtained have been implemented to solve the boundary value problems. The investigation will be focusing on the Newton divided difference interpolation as an iterative technique in seeking as well as updating the missing initial guesses while performing the shooting technique.

The organization of this paper is as follows. Section 2 presents the derivation of the two-point diagonal block method. Section 3 elaborates on the analysis of the method including the order, consistency, and stability. Section 4 presents the idea of the procedure of shooting method together with Newton divided difference interpolation as an iterative part. For validation and a clear overview, five tested problems will be discussed in Section 5. Finally, Section 6 concludes the finding from this study.

2. Derivation of the Diagonal Block Method

The interval of $x \in [a, b]$ is divided into a series of blocks so that each block contains two subintervals with equal distance step size, h, as depicted in Figure 1. Both numerical solutions of y_{n+1} and y_{n+2} in these subintervals are simultaneously generated using the appropriate number of back values. The approximate solution of y_{n+1} at the point x_{n+1} will be computed using three back values at the points, x_n, x_{n-1}, and x_{n-2}. While the approximation of y_{n+2} at the point x_{n+2} required an additional one back value, x_{n+1}. The derivation formulae of y_{n+1} and y_{n+2} will be obtained by integrate equation (1) once and twice over the intervals $[x_n, x_{n+1}]$ and $[x_n, x_{n+2}]$, respectively, as follows.

First Point

$$\int_{x_n}^{x_{n+1}} y''(x)\,dx = \int_{x_n}^{x_{n+1}} f(x, y, y')\,dx \tag{3}$$

$$y'(x_{n+1}) = y'(x_n) + \int_{x_n}^{x_{n+1}} f(x, y, y')\,dx \tag{4}$$

$$\int_{x_n}^{x_{n+1}} \int_{x_n}^{x} y''(x)\,dx dx = \int_{x_n}^{x_{n+1}} \int_{x_n}^{x} f(x, y, y')\,dx dx \tag{5}$$

$$\begin{aligned} &y(x_{n+1}) \\ &\quad = y(x_n) + hy'(x_n) \\ &\qquad + \int_{x_n}^{x_{n+1}} (x_{n+1} - x)\, f(x, y, y')\,dx, \end{aligned} \tag{6}$$

Second Point

$$\int_{x_n}^{x_{n+2}} y''(x)\,dx = \int_{x_n}^{x_{n+2}} f(x, y, y')\,dx \tag{7}$$

$$y'(x_{n+2}) = y'(x_n) + \int_{x_n}^{x_{n+2}} f(x, y, y')\,dx \tag{8}$$

$$\int_{x_n}^{x_{n+2}} \int_{x_n}^{x} y''(x)\,dx dx = \int_{x_n}^{x_{n+2}} \int_{x_n}^{x} f(x, y, y')\,dx dx \tag{9}$$

$$\begin{aligned} &y(x_{n+2}) \\ &\quad = y(x_n) + 2hy'(x_n) \\ &\qquad + \int_{x_n}^{x_{n+2}} (x_{n+2} - x)\, f(x, y, y')\,dx. \end{aligned} \tag{10}$$

The function $f(x, y, y')$ in these equations will be approximated using Lagrange interpolating polynomial, $P(x)$. By following the standard mathematical process, replace the $f(x, y, y')$ in (4) and (6) with Lagrange interpolation polynomial that interpolates the set of points

$$\{(x_{n-2}, f_{n-2}), (x_{n-1}, f_{n-1}), (x_n, f_n), (x_{n+1}, f_{n+1})\}. \tag{11}$$

Now, introducing the variable substitution $x = x_{n+1} + sh$ and $dx = hds$ into interpolating polynomial, hence, evaluate these integrals using MAPLE with the limit of the integration from −1 to 0. This will generate the corrector formula for the first point. Similar procedure applied to the $f(x, y, y')$ in (8) and (10) with Lagrange interpolation polynomial that interpolates the set of points

$$\begin{aligned} &\{(x_{n-2}, f_{n-2}), (x_{n-1}, f_{n-1}), (x_n, f_n), (x_{n+1}, f_{n+1}), \\ &\quad (x_{n+2}, f_{n+2})\}. \end{aligned} \tag{12}$$

Next, introduce the variable substitutions $x = x_{n+2} + sh$ and $dx = hds$ into interpolating polynomial. Again, evaluate these integrals using MAPLE with the limit of the integration from −2 to 0. This will generate the corrector formula for the second point. This two-point one block method is the combination of $PE(CE)^r$ mode where P is the predictor formula, C is the corrector, and E is the evaluation of the function, f. The predictor formulae were derived similarly as the corrector formulae but with the order being one less. In this study, we called the proposed predictor-corrector formula as 2PDD4 method. The 2PDD4 formulae were given as follows.

Predictor

$$\begin{aligned} y'_{n+1} &= y'_n + \frac{h}{12}\left[5f_{n-2} - 16f_{n-1} + 23f_n\right] \\ y_{n+1} &= y_n + hy'_n + \frac{h^2}{24}\left[3f_{n-2} - 10f_{n-1} + 19f_n\right] \end{aligned} \tag{13}$$

$$\begin{aligned} y'_{n+2} &= y'_n + \frac{h}{3}\left[-f_{n-2} + 4f_{n-1} - 5f_n + 8f_{n+1}\right] \\ y_{n+2} &= y_n + 2hy'_n \\ &\quad + \frac{h^2}{45}\left[-2f_{n-2} + 6f_{n-1} + 24f_n + 62f_{n+1}\right]. \end{aligned} \tag{14}$$

Corrector

$$\begin{aligned} y'_{n+1} &= y'_n + \frac{h}{48}\left[2f_{n-2} - 10f_{n-1} + 38f_n + 18f_{n+1}\right] \\ y_{n+1} &= y_n + hy'_n + \frac{h^2}{720}\left[14f_{n-2} - 72f_{n-1} + 342f_n \right. \\ &\quad \left. + 76f_{n+1}\right] \end{aligned} \tag{15}$$

$$\begin{aligned} y'_{n+2} &= y'_n + \frac{h}{270}\left[-3f_{n-2} + 12f_{n-1} + 72f_n + 372f_{n+1} \right. \\ &\quad \left. + 87f_{n+2}\right] \\ y_{n+2} &= y_n + 2hy'_n + \frac{h^2}{270}\left[3f_{n-2} - 24f_{n-1} + 234f_n \right. \\ &\quad \left. + 312f_{n+1} + 15f_{n+2}\right]. \end{aligned} \tag{16}$$

One-step method will be used at the beginning of the proposed block method in order to get the starting initial points since multistep method needs more than one previous points before generating the remaining values over the interval.

3. Analysis of the Method

In this section, the order, consistency, stability, and convergence of the proposed two-point block method are discussed.

3.1. Order of the Method. The main proposed method of (13)–(16) is classified as a member of the Linear Multistep Method (LMM) which generally can be represented as

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h \sum_{j=0}^{k} \beta_j y'_{n+j} + h^2 \sum_{j=0}^{k} \gamma_j y''_{n+j}. \tag{17}$$

The local truncation error (LTE) associated with LMM in (17) is defined in the form of linear difference operator as

$$\begin{aligned} L\left[y(x), h\right] = \sum_{j=0}^{k} &\left[\alpha_j y(x + jh) - h\beta_j y'(x + jh) \right. \\ &\left. - h^2 \gamma_j y''(x + jh)\right]. \end{aligned} \tag{18}$$

Assume that $y(x)$ is sufficiently differentiable, so that expanding the terms in (18) using Taylor's series about the point x will give the following expression:

$$\begin{aligned} y(x + jh) &= y(x) + jhy'(x) + \frac{j^2}{2!}h^2 y''(x) + \dots \\ &\quad + \frac{(jh)^p}{p!} y^{(p)}(x) + O\left(h^{p+2}\right) \end{aligned} \tag{19}$$

$$\begin{aligned} hy'(x + jh) &= hy'(x) + jh^2 y''(x) + \frac{j^2}{2!}h^3 y^{(3)}(x) \\ &\quad + \dots + \frac{j^{p-1}}{(p-1)!} h^p y^{(p)}(x) \\ &\quad + O\left(h^{p+2}\right) \end{aligned} \tag{20}$$

$$\begin{aligned} h^2 y''(x + jh) &= h^2 y''(x) + jh^3 y'''(x) \\ &\quad + \frac{j^2}{2!}h^4 y^{(4)}(x) + \dots \\ &\quad + \frac{j^{p-2}}{(p-2)!} h^p y^{(p)}(x) \\ &\quad + O\left(h^{p+2}\right). \end{aligned} \tag{21}$$

Substituting (19)–(21) into (18) results in

$$\begin{aligned} L\left[y(x), h\right] = \sum_{j=0}^{k} &\left[\alpha_j \left(y(x) + jhy'(x) + \frac{j^2}{2!}h^2 y''(x) \right.\right. \\ &\left. + \dots + \frac{(jh)^p}{p!} y^{(p)}(x)\right) - \beta_j \left(hy'(x) + jh^2 y''(x) \right. \\ &\left. + \frac{j^2}{2!}h^3 y^{(3)}(x) + \dots + \frac{j^{p-1}}{(p-1)!} h^p y^{(p)}(x)\right) \\ &- \gamma_j \left(h^2 y''(x) + jh^3 y'''(x) + \frac{j^2}{2!}h^4 y^{(4)}(x) + \dots \right. \\ &\left.\left. + \frac{j^{p-2}}{(p-2)!} h^p y^{(p)}(x)\right)\right] + O\left(h^{p+2}\right). \end{aligned} \tag{22}$$

Simplifying this gives

$$\begin{aligned} L\left[y(x), h\right] = \sum_{j=0}^{k} &\left[\alpha_j y(x) + \left(j\alpha_j - \beta_j\right) hy'(x) \right. \\ &+ \left(\frac{j^2}{2!}\alpha_j - j\beta_j - \gamma_j\right) h^2 y''(x) \end{aligned}$$

$$\begin{aligned}&+\left(\frac{j^3}{3!}\alpha_j-\frac{j^2}{2!}\beta_j-j\gamma_j\right)h^3y^{(3)}(x)+\dots\\&+\left(\frac{j^p}{p!}\alpha_j-\frac{j^{p-1}}{(p-1)!}\beta_j-\frac{j^{p-2}}{(p-2)!}\gamma_j\right)h^py^{(p)}(x)\Bigg]\\&+O\left(h^{p+2}\right)\end{aligned}\tag{23}$$

and now we obtained the expression

$$\begin{aligned}L\left[y(x),h\right]=C_0y(x)+C_1hy'(x)+\dots\\+C_ph^py^{(p)}(x)+C_{p+1}h^{p+1}y^{(p+1)}(x)\\+O\left(h^{p+2}\right)\end{aligned}\tag{24}$$

where

$$\begin{aligned}C_0&=\sum_{j=0}^{k}\alpha_j=\alpha_0+\alpha_1+\dots+\alpha_k\\C_1&=\sum_{j=0}^{k}\left(j\alpha_j-\beta_j\right)=\left(\alpha_1+2\alpha_2+\dots+k\alpha_k\right)\\&\quad-\left(\beta_0+\beta_1+\dots+\beta_k\right)\\&\;\vdots\\C_p&=\sum_{j=0}^{k}\left(\frac{j^p}{p!}\alpha_j-\frac{j^{p-1}}{(p-1)!}\beta_j-\frac{j^{p-2}}{(p-2)!}\gamma_j\right).\end{aligned}\tag{25}$$

Definition 1. According to Fatunla [7] and Lambert [14], the method in (17) is said to be of order p with error constant C_{p+2} if

$$\begin{aligned}C_0=C_1=C_2=\dots=C_{p+1}=0,\\C_{p+2}\neq0.\end{aligned}\tag{26}$$

This concept was used to calculate the order and error constant of the proposed formula as stated in (13)–(16). Now, rewrite the corrector formula in matrix difference form as

$$\begin{aligned}&\begin{bmatrix}0&0&0&0\\0&0&-1&1\\0&0&0&0\\0&0&-1&0\end{bmatrix}\begin{bmatrix}y_{n-2}\\y_{n-1}\\y_n\\y_{n+1}\end{bmatrix}+\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\1&0&0&0\end{bmatrix}\begin{bmatrix}y_{n+2}\\y_{n+3}\\y_{n+4}\\y_{n+5}\end{bmatrix}\\&=h\begin{bmatrix}0&0&1&-1\\0&0&1&0\\0&0&1&0\\0&0&2&0\end{bmatrix}\begin{bmatrix}y'_{n-2}\\y'_{n-1}\\y'_n\\y'_{n+1}\end{bmatrix}+h\begin{bmatrix}0&0&0&0\\0&0&0&0\\-1&0&0&0\\0&0&0&0\end{bmatrix}\begin{bmatrix}y'_{n+2}\\y'_{n+3}\\y'_{n+4}\\y'_{n+5}\end{bmatrix}\\&+h^2\begin{bmatrix}\frac{2}{48}&-\frac{10}{48}&\frac{38}{48}&\frac{18}{48}\\\frac{14}{720}&-\frac{72}{720}&\frac{342}{720}&\frac{76}{720}\\-\frac{3}{270}&\frac{12}{270}&\frac{72}{270}&\frac{372}{270}\\\frac{3}{270}&-\frac{24}{270}&\frac{234}{270}&\frac{312}{270}\end{bmatrix}\begin{bmatrix}f_{n-2}\\f_{n-1}\\f_n\\f_{n+1}\end{bmatrix}\\&+h^2\begin{bmatrix}0&0&0&0\\0&0&0&0\\\frac{87}{270}&0&0&0\\\frac{15}{270}&0&0&0\end{bmatrix}\begin{bmatrix}f_{n+2}\\f_{n+3}\\f_{n+4}\\f_{n+5}\end{bmatrix}\end{aligned}\tag{27}$$

with $\alpha_0=\alpha_1=\alpha_5=\alpha_6=\alpha_7=\beta_0=\beta_1=\beta_5=\beta_6=\beta_7=\gamma_5=\gamma_6=\gamma_7=[0,0,0,0]^T$.

By choosing $k=7$ the calculation gives

$$C_0=\sum_{j=0}^{7}\alpha_j=\alpha_0+\alpha_1+\alpha_2+\alpha_3+\alpha_4+\alpha_5+\alpha_6+\alpha_7$$

$$=\frac{1}{0!}\left(\begin{bmatrix}0\\-1\\0\\-1\end{bmatrix}+\begin{bmatrix}0\\1\\0\\0\end{bmatrix}+\begin{bmatrix}0\\0\\0\\1\end{bmatrix}\right)=\begin{bmatrix}0\\0\\0\\0\end{bmatrix}.$$

$$\begin{aligned}C_1&=\sum_{j=0}^{7}\left(\frac{j^1}{1!}\alpha_j-\frac{j^0}{0!}\beta_j\right)=\frac{1}{1!}\left(2\begin{bmatrix}0\\-1\\0\\-1\end{bmatrix}+3\begin{bmatrix}0\\1\\0\\0\end{bmatrix}\right.\\&\left.+4\begin{bmatrix}0\\0\\0\\1\end{bmatrix}\right)-\frac{1}{0!}\left(\begin{bmatrix}1\\1\\1\\2\end{bmatrix}+\begin{bmatrix}-1\\0\\0\\0\end{bmatrix}+\begin{bmatrix}0\\0\\-1\\0\end{bmatrix}\right)\\&=\begin{bmatrix}0\\0\\0\\0\end{bmatrix}.\end{aligned}$$

$$C_2 = \sum_{j=0}^{7}\left(\frac{j^2}{2!}\alpha_j - \frac{j^1}{1!}\beta_j - \frac{j^0}{0!}\gamma_j\right) = \frac{1}{2!}\left(2^2\begin{bmatrix}0\\-1\\0\\-1\end{bmatrix} + 3^2\begin{bmatrix}0\\1\\0\\0\end{bmatrix} + 4^2\begin{bmatrix}0\\0\\0\\1\end{bmatrix}\right) - \frac{1}{1!}\left(2\begin{bmatrix}1\\1\\1\\2\end{bmatrix} + 3\begin{bmatrix}-1\\0\\0\\0\end{bmatrix} + 4\begin{bmatrix}0\\0\\-1\\0\end{bmatrix}\right) - \frac{1}{0!}\left(\begin{bmatrix}\frac{2}{48}\\\frac{14}{720}\\-\frac{3}{270}\\\frac{3}{270}\end{bmatrix} + \begin{bmatrix}-\frac{10}{48}\\-\frac{72}{720}\\\frac{12}{270}\\-\frac{24}{270}\end{bmatrix} + \begin{bmatrix}\frac{38}{48}\\\frac{342}{720}\\\frac{72}{270}\\\frac{234}{270}\end{bmatrix} + \begin{bmatrix}\frac{18}{48}\\\frac{76}{720}\\\frac{372}{270}\\\frac{312}{270}\end{bmatrix} + \begin{bmatrix}0\\0\\\frac{87}{270}\\\frac{15}{270}\end{bmatrix}\right) = [0,0,0,0]^T.$$

$$\vdots$$

$$C_5 = \sum_{j=0}^{7}\left(\frac{j^5}{5!}\alpha_j - \frac{j^4}{4!}\beta_j - \frac{j^3}{3!}\gamma_j\right) = [0,0,0,0]^T.$$

$$C_6 = \sum_{j=0}^{7}\left(\frac{j^6}{6!}\alpha_j - \frac{j^5}{5!}\beta_j - \frac{j^4}{4!}\gamma_j\right) = \left[-\frac{19}{720}, -\frac{17}{1440}, 0, 0\right]^T. \tag{28}$$

Thus, the 2PDD4 satisfied order four with the error constant,

$$C_{p+2} = C_6 = \left[-\frac{19}{720}, -\frac{17}{1440}, 0, 0\right]^T. \tag{29}$$

3.2. Consistency of the Method

Definition 2. The linear multistep method is said to be consistent if it has order $p \geq 1$ according to [14].

Since the order of the proposed method is $p = 4 \geq 1$, the method is consistent.

3.3. Stability of the Method

Definition 3. The linear multistep method is zero stable provided that the root ξ_j, $j = 0(1)k$ of the first characteristics polynomial $\rho(\xi)$ specified as $\rho(\xi) = \det|\sum_{j=0}^{k} A^{(j)}\xi^{(k-j)}| = 0$ satisfies $|\xi_j| \leq 1$ and for those roots with $|\xi_j| = 1$, the multiplicity must not exceed two as in Lambert [14] and See et al. [15].

Rewrite (15) to (16) in matrix form as follows:

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}y'_{n+1}\\y_{n+1}\\y'_{n+2}\\y_{n+2}\end{bmatrix} = \begin{bmatrix}0&0&1&0\\0&0&0&1\\0&0&1&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}y'_{n-1}\\y_{n-1}\\y'_n\\y_n\end{bmatrix} + h\begin{bmatrix}0&0&0&0\\0&0&1&0\\0&0&0&0\\0&0&2&0\end{bmatrix}\begin{bmatrix}y'_{n-1}\\y_{n-1}\\y'_n\\y_n\end{bmatrix} + h\begin{bmatrix}0&\frac{2}{48}&-\frac{10}{48}&\frac{38}{48}\\0&0&0&0\\0&-\frac{3}{270}&\frac{12}{270}&\frac{72}{270}\\0&0&0&0\end{bmatrix}\begin{bmatrix}f_{n-3}\\f_{n-2}\\f_{n-1}\\f_n\end{bmatrix} + h\begin{bmatrix}\frac{18}{48}&0&0&0\\0&0&0&0\\\frac{372}{270}&\frac{87}{270}&0&0\\0&0&0&0\end{bmatrix}\begin{bmatrix}f_{n+1}\\f_{n+2}\\f_{n+3}\\f_{n+4}\end{bmatrix} + h^2\begin{bmatrix}0&0&0&0\\0&\frac{14}{720}&-\frac{72}{720}&\frac{342}{720}\\0&0&0&0\\0&\frac{3}{270}&-\frac{24}{270}&\frac{234}{270}\end{bmatrix}\begin{bmatrix}f_{n-3}\\f_{n-2}\\f_{n-1}\\f_n\end{bmatrix} + h^2\begin{bmatrix}0&0&0&0\\\frac{76}{720}&0&0&0\\0&0&0&0\\\frac{312}{270}&\frac{15}{270}&0&0\end{bmatrix}\begin{bmatrix}f_{n+1}\\f_{n+2}\\f_{n+3}\\f_{n+4}\end{bmatrix}. \tag{30}$$

From (30), the first characteristic polynomial $p(\xi) = \det|\xi A^0 - A^1| = 0$, where

$$A^0 = \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix},$$

$$A^1 = \begin{bmatrix}0&0&1&0\\0&0&0&1\\0&0&1&0\\0&0&0&1\end{bmatrix}$$

$$p(\xi) = \det \begin{bmatrix} \xi & 0 & -1 & 0 \\ 0 & \xi & 0 & -1 \\ 0 & 0 & \xi - 1 & 0 \\ 0 & 0 & 0 & \xi - 1 \end{bmatrix}$$
$$0 = \xi^2 (\xi - 1)^2, \quad \xi = 0, 0, 1, 1. \tag{31}$$

According to Definition 3 and (31), the diagonal two-point one block method is zero stable.

The test equation used in order to obtain the stability polynomial of the two-point block follows the idea from [11] as follows:

$$y'' = f = \theta y' + \lambda y. \tag{32}$$

The stability polynomial obtained is as follows:

$$\begin{aligned} & t^8 \left(1 + \frac{1777}{32400} H1H2 - \frac{29}{180} H2 - \frac{251}{360} H1 \right. \\ & \quad \left. + \frac{19}{3240} H2^2 + \frac{29}{240} H1^2 \right) + t^7 \left(-2 - \frac{2173}{2025} H1H2 \right. \\ & \quad \left. - \frac{331}{90} H2 - \frac{131}{360} H1 - \frac{2257}{3240} H2^2 - \frac{1273}{1080} H1^2 \right) \\ & \quad + t^6 \left(1 - \frac{29}{180} H2 + \frac{59}{72} H1 - \frac{1}{360} H2^2 \right. \\ & \quad \left. - \frac{5863}{5400} H1H2 + \frac{163}{180} H1^2 \right) + t^5 \left(\frac{29}{120} H1 \right. \\ & \quad \left. + \frac{17}{3240} H2^2 - \frac{148}{2025} H1H2 + \frac{11}{72} H1^2 \right) \\ & \quad + t^4 \left(-\frac{23}{32400} H1H2 - \frac{1}{1620} H2^2 - \frac{1}{2160} H1^2 \right) \end{aligned} \tag{33}$$

where $H1 = h\theta$ and $H2 = h^2\lambda$.

The boundary of the absolute stability region in $H1 - H2$ plane is determined by substituting t in the stability polynomial with $1, -1$ and $e^{i\theta}$ for $0 \leq \theta \leq 2\pi$. Figure 2 illustrates the regions of the absolute stability for the block method.

The stability region shows that the proposed numerical method will be able to produce reasonable results with the given values of the time steps.

3.4. Convergence of the Method

Definition 4. The linear multistep method is convergent if and only if it is consistent and zero stable [14].

Since the consistency and zero stability of the method have been achieved, in conclusion, the proposed two-point block method is convergent.

4. Implementation of the Method

In this study, shooting technique will be adapted throughout the solution process where the calculation starts with deciding a set of initial guesses. The behaviour of shooting method is 'hit or miss' the target. Due to that, we attempt to obtain the result as close as possible to the required solution with less number of guessing values. Therefore, choosing the right initial guesses and implementing the best method for refining the guessing value are two important elements in this shooting procedure. Initial values of y and y' need to be guessed for the case of Robin type since none of them are given in (2). Starting with the initial guess, s_1 for $y(a)$, then the initial guess of $y'(a)$ is given explicitly from the first boundary condition as follows:

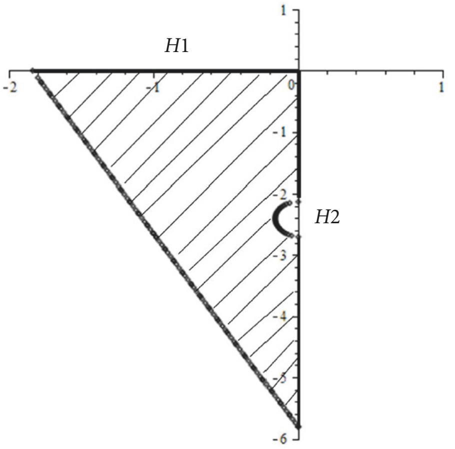


FIGURE 2: Stability region of the two-point diagonal block method.

$$\begin{aligned} & y'' = f\left(x, y, y'\right), \quad \text{with } a \leq x \leq b \\ & y_1(a) = s_1, \\ & y_1'(a) = V_1 - C_1 y_1(a) \end{aligned} \tag{34}$$

where $V_1 = \alpha/c_1$ and $C_1 = c_2/c_1$. Compute the approximate solutions using the formula in (13) to (16). We obtained the first stopping condition as

$$\left| h\left(y_1(b), y_1'(b)\right) - \beta \right| \leq TOL, \tag{35}$$

where $h(y(b), y'(b)) = y'(b) + C_2 y(b)$, with $C_2 = c_4/c_3$. If this is sufficiently close to the condition in (35), then the BVPs are solved. Otherwise, update the new set of guessing values and the process continues as described in the following algorithm.

Algorithm of 2PDD4

Step 1. Set TOL and $y_1(a) = s_1$, $y_1'(a) = V_1 - C_1 y_1(a)$.

Step 2. Set $x_n = x_0 + nh$ and calculate the approximate values, y_{n+1} and y_{n+2}, using the formula in (13) to (16) with $PE(CE)^r$ where $r = 1, 2, \ldots$ and r depend on the test of convergence at each iteration.

Step 3. If $x_n < b$, then repeat Step 2. If $x_n = b$, then go to Step 4.

Step 4. If the following stopping condition is fulfilled: $|h(y_j(b), y_j'(b)) - \beta| \leq TOL$, then go to Step 6.

Else, continue to Step 5.

Step 5. Generate the new guessing values, $y_j(a) = s_j$ and $y_j'(a) = V_1 - C_1 y_j(a)$ for $j = 2, 3, \ldots N$ based on the previous guess using Newton divided difference interpolation formula. Repeat Step 2.

Step 6. Complete.

The first two estimates were decided to be $s_1 = 0$ and $s_2 = 1$ based on the consideration in Roberts [16]. The calculation for the corrector part involved the test for convergence. The formulae for the calculation of error and convergence test are defined as follows.

Error

$$\left| \frac{(y_i)_t - (y(x_i))_t}{A + B(y_i)_t} \right|. \tag{36}$$

Convergence Test

$$\left| \frac{(y_{n+1,r})_t - (y_{n+1,r-1})_t}{A + B(y_{n+1,r})_t} \right| < 0.1 \times TOL \tag{37}$$

where r is the number of iterations and $(y)_t$ is the t^{th} component of the approximation. $A = 1, B = 0$ result in absolute error test, $A = 1, B = 1$ result in mixed error test, and $A = 0, B = 1$ result in relative error test. All the calculations were done using the C programing code.

5. Numerical Results

In this section, we have applied the algorithm of 2PDD4 to five numerical tested problems to illustrate its accuracy and efficiency. Problem 1 used mixed error test and Problem 2 applied the relative error test whereas Problems 3, 4, and 5 used absolute error test throughout the calculation for obtaining the required result. The following notations are used in the following result.

MAXE: Maximum absolute error as stated with $*$

AVE: Average absolute error

h: Step size

TOL: Tolerance

TS: Total steps at last iteration

FCN: Total function call

ITN: Total iteration of guess

TIME: Time computation in seconds

2PDD4: Direct two-point diagonal block method of order four proposed in this study

2PDAM4: Direct two-step Adams Moulton method of order four as in Phang et al. [17]

DAM4: Direct Adams Moulton method of order four as in Majid et al. [18]

BP5: Bernoulli polynomials of degree five proposed by Islam and Shirin [2].

Problem 1. One has linear second-order differential equation

$$y'' = y - 2\cos(x), \quad \frac{\pi}{2} \leq x \leq \pi \tag{38}$$

with $y'(\pi/2) + 3y(\pi/2) = -1$ and $y'(\pi) + 4y(\pi) = -4$.

Exact solution: $y(x) = \cos(x)$.

Source: Islam and Shirin [2].

Problem 2. One has nonlinear second-order differential equation

$$y'' = \frac{1}{2}e^{-x}\left(y^2 + (y')^2\right), \quad 0 \leq x \leq 1 \tag{39}$$

with $-y'(0) + y(0) = 0$ and $y'(1) + y(1) = 2e$.

Exact solution: $y(x) = e^x$.

Source: Duan et al. [3].

Problem 3. One has nonlinear second-order differential equation

$$y'' = -\frac{1}{8}\left(e^{-2y} + 4(y')^2\right), \quad 0 \leq x \leq 1 \tag{40}$$

with $-2y'(0) + y(0) = -1$ and $2y'(1) + y(1) = 2/3 + \log(3/2)$.

Exact solution: $y(x) = \log((2+x)/2)$.

Source: Duan et al. [3].

Problem 4. One has nonlinear second-order differential equation

$$y'' = -\exp(-2y), \quad 0 \leq x \leq 1 \tag{41}$$

with $y'(0) - y(0) = 1$ and $y'(1) + y(1) = 0.5 + \ln(2)$.

Exact solution: $y(x) = \ln(1+x)$. Source: Bhatta and Sastri [5].

Problem 5. One has nonlinear second-order differential equation

$$y''(x) = \pi^2 e^{y(x)}, \quad 0 \leq x \leq 1 \tag{42}$$

with $2y'(0) + y(0) = -2\pi$ and $-y'(1) + 2y(1) = -\pi$.

Exact solution: $y(x) = -2\ln(\cos((\pi/2)x - \pi/4)) - \ln(2)$.

Source: Lang and Xu [6].

The general equation of Problem 5 given as follows:

$$y''(x) = be^{ay(x)}, \quad \text{with } y(0) = y(1) = 0, \tag{43}$$

arises in applications involving the diffusion of heat generated by positive temperature-dependent sources. According to discussion in Agarwal and O'Regan [19], p.295, if $a = 1$, the diffusion of heat arises in the two different analyses as Joule losses either in electrically conducting solids or in frictional heating. For the first analysis, b represents the square of

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

x	Error at h = 0.1			Error at h = 0.01			
	2PDD4	2PDAM4	DAM4	2PDD4	2PDAM4	DAM4	BP5
1.5708	4.64e-07	2.60e-07	1.49e-06	2.05e-10	2.05e-10	2.05e-10	0.00e+00
1.7279	6.76e-07	5.81e-07	2.37e-07	1.70e-10	1.71e-10	1.71e-10	2.74e-06
1.8850	8.47e-07∗	8.32e-07∗	7.10e-07	1.56e-10	1.63e-10	1.63e-10	8.52e-06
2.0420	6.32e-07	7.43e-07	1.11e-06	1.49e-10	1.66e-10	1.69e-10	6.63e-06
2.1991	5.36e-07	6.73e-07	1.39e-06	1.48e-10	1.78e-10	1.91e-10	4.21e-06
2.3562	3.68e-07	6.25e-07	1.56e-06	1.52e-10	2.00e-10	2.29e-10	1.10e-05∗
2.5133	3.43e-07	5.78e-07	1.62e-06∗	1.60e-10	2.31e-10	2.86e-10	5.35e-06
2.6704	1.94e-07	5.28e-07	1.59e-06	1.73e-10	2.72e-10	3.65e-10	6.10e-06
2.8274	2.06e-07	4.72e-07	1.45e-06	1.91e-10	3.25e-10	4.68e-10	9.42e-06
2.9845	6.27e-08	3.96e-07	1.18e-06	2.16e-10	3.92e-10	6.03e-10	2.45e-06
3.1416	9.16e-08	3.05e-07	7.81e-07	2.47e-10∗	4.76e-10∗	7.76e-10∗	0.00e+00
AVE	4.42e-07	5.99e-07	1.31e-06	1.75e-10	2.46e-10	3.16e-10	6.27e-06
TS	6	6	10	51	51	100	-
FCN	96	144	96	114	212	114	-
ITN	3	3	3	1	1	1	-
TIME	0.0030	0.0150	0.0310	0.0150	0.0470	0.0470	-

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

x	Error at h = 0.1			Error at h = 0.01		
	2PDD4	2PDAM4	DAM4	2PDD4	2PDAM4	DAM4
0.00	4.90e-07∗	3.05e-05	1.84e-07	0.00e+00	0.00e+00	0.00e+00
0.10	2.61e-07	3.04e-05	4.17e-07	5.32e-12	2.79e-11	8.54e-12
0.20	2.63e-08	3.06e-05	6.61e-07	5.87e-12	1.35e-10	2.23e-11
0.30	2.58e-08	3.13e-05	6.76e-07	6.69e-12	3.24e-10	4.57e-11
0.40	9.67e-09	3.21e-05∗	6.78e-07∗	7.76e-12	5.88e-10	7.80e-11
0.50	1.30e-08	3.19e-05	6.68e-07	9.08e-12	9.22e-10	1.19e-10
0.60	7.25e-11	3.15e-05	6.46e-07	1.06e-11	1.32e-09	1.68e-10
0.70	6.15e-09	2.99e-05	6.14e-07	1.24e-11	1.79e-09	2.24e-10
0.80	4.33e-09	2.81e-05	5.73e-07	1.44e-11	2.32e-09	2.88e-10
0.90	3.91e-09	2.47e-05	5.22e-07	1.66e-11	2.91e-09	3.59e-10
1.00	4.63e-09	2.11e-05	4.62e-07	1.90e-11∗	3.56e-09∗	4.38e-10∗
AVE	8.44e-08	3.22e-05	6.10e-07	1.01e-11	1.22e-09	1.55e-10
TS	6	6	10	51	51	100
FCN	120	212	88	228	424	228
ITN	4	4	3	2	2	2
TIME	0.015	0.031	0.015	0.031	0.047	0.046

the constant current and $e^{y(x)}$ the temperature-dependent resistance. Meanwhile, for the second part, b represents the square of the constant shear stress and $e^{y(x)}$ the temperature-dependent fluidity. In addition, if $ab = 0$ and $ab > 0$, then problem in (43) has a unique solution.

Data in Tables 1–4 summarized the result obtained for all tested problems at $h = 0.1$ and $h = 0.01$ with $TOL = 10^{-6}$. In addition, the values ending with ∗ in Tables 1–4 denote the maximum error for those particular results. In order to investigate the competency of this proposed method, 2PDD4 has been compared with 2PDAM4 and DAM4. These two existing methods have the same order as 2PDD4 and will solve the BVP problems directly. The same shooting procedure has been implemented in the algorithm as explained in Section 4.

In Table 1, the maximum error of 2PDD4 is comparable with 2PDAM4 and DAM4 but better than BP5. The total function calls and execution times for 2PDD4 are less compared to 2PDAM4 and DAM4. As can be seen in Table 2, the total function calls and number of initial guessing for 2PDD4 are greater than DAM4 at $h = 0.1$ but comparable in terms of computation time. Besides that, 2PDD4 achieved an excellent accuracy result compared to 2PDAM4. In detail, 2PDD4 has

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

x	Error at $h = 0.1$			Error at $h = 0.01$		
	2PDD4	2PDAM4	DAM4	2PDD4	2PDAM4	DAM4
0.00	0.00e+00	0.00e+00	1.28e-06	0.00e+00	0.00e+00	0.00e+00
0.10	1.68e-08	1.68e-08	1.33e-06	3.72e-13	1.29e-10	3.30e-13
0.20	3.03e-08	3.03e-08	1.38e-06	4.26e-13	7.01e-10	7.56e-14
0.30	4.47e-08*	2.36e-08	1.43e-06	5.04e-13	1.64e-09	3.62e-13
0.40	3.12e-08	2.16e-08	1.45e-06	5.99e-13	2.86e-09	9.31e-13
0.50	3.55e-08	2.82e-08	1.46e-06*	7.05e-13	4.32e-09	1.59e-12
0.60	2.34e-08	3.73e-08	1.45e-06	8.18e-13	5.96e-09	2.32e-12
0.70	2.23e-08	5.12e-08	1.43e-06	9.36e-13	7.74e-09	3.09e-12
0.80	1.12e-08	6.63e-08	1.40e-06	1.06e-12	9.64e-09	3.90e-12
0.90	7.25e-09	8.41e-08	1.36e-06	1.18e-12	1.16e-08	4.73e-12
1.00	2.96e-09	1.03e-07*	1.33e-06	1.31e-12*	1.37e-08*	5.57e-12*
AVE	2.26e-08	4.62e-08	1.53e-06	7.46e-13	5.20e-09	2.05e-12
TS	6	6	10	51	51	100
FCN	32	80	32	114	604	114
ITN	1	1	3	1	1	1
TIME	0.0040	0.0470	0.0470	0.0400	0.0620	0.0780

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

x	Error at $h = 0.1$			Error at $h = 0.01$		
	2PDD4	2PDAM4	DAM4	2PDD4	2PDAM4	DAM4
0.00	0.00e+00	2.07e-06	1.18e-05	0.00e+00	0.00e+00	0.00e+00
0.10	5.30e-07	1.77e-06	1.25e-05	1.27e-11	2.41e-11	9.41e-11
0.20	9.48e-07	1.62e-06	1.36e-05	1.20e-11	8.26e-11	3.66e-10
0.30	1.66e-06*	2.03e-06	1.46e-05	1.01e-11	1.71e-10	7.57e-10
0.40	1.33e-06	2.36e-06	1.47e-05*	7.54e-12	2.79e-10	1.23e-09
0.50	1.53e-06	2.39e-06	1.44e-05	4.68e-12	4.00e-10	1.75e-09
0.60	1.34e-06	2.39e-06*	1.37e-05	1.61e-12	5.33e-10	2.33e-09
0.70	1.38e-06	2.29e-06	1.29e-05	1.62e-12	6.74e-10	2.93e-09
0.80	1.25e-06	2.18e-06	1.20e-05	4.97e-12	8.23e-10	3.58e-09
0.90	1.23e-06	2.03e-06	1.10e-05	8.42e-12	9.80e-10	4.25e-09
1.00	1.14e-06	1.88e-06	1.01e-05	1.20e-11*	1.14e-09*	4.96e-09*
AVE	1.23e-06	2.30e-06	1.41e-05	7.55e-12	4.59e-10	2.00e-09
TS	6	6	10	51	51	100
FCN	32	48	32	114	212	114
ITN	1	4	4	1	1	1
TIME	0.0030	0.0150	0.0150	0.0050	0.0310	0.0460

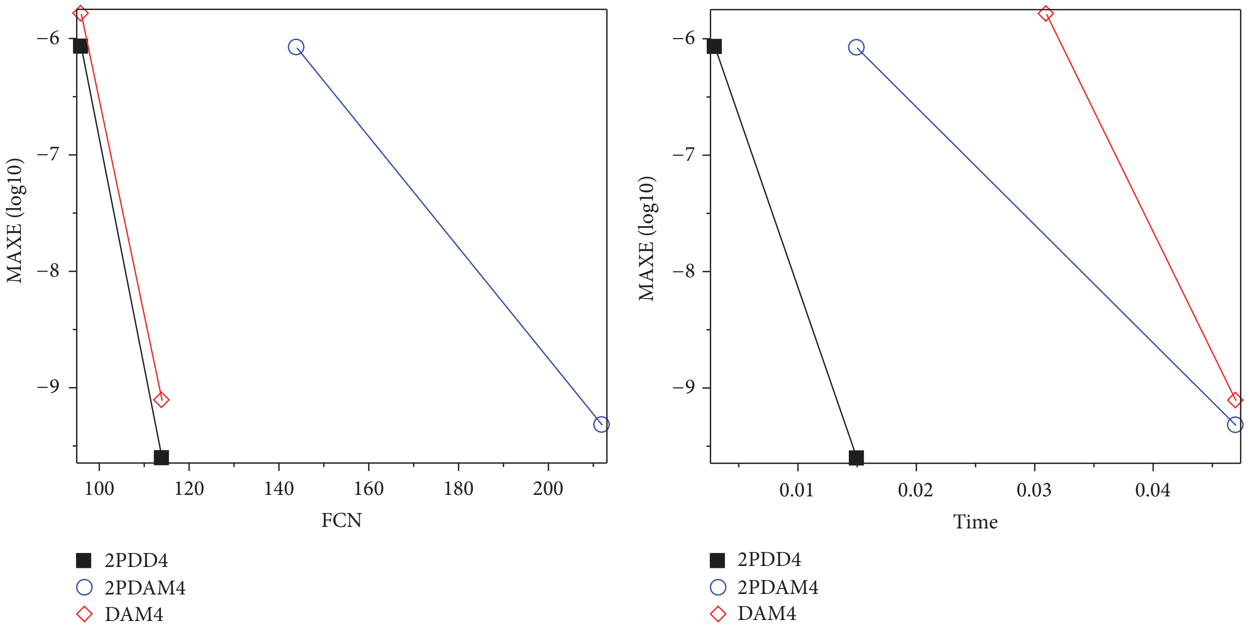
less three- or two-decimal accuracy than 2PDAM4 for solving Problem 2 with $h = 0.1$ and $h = 0.01$, respectively. Method 2PDD4 needs less total function call and obtained smallest maximum error compared to other methods as displayed in Table 3. Table 4 has shown that 2PDD4 required only one initial guessing as compared to 2PDAM4 and DAM4 for solving Problem 4 at $h = 0.1$. Due to this efficient iterative performance, 2PDD4 gives a superiority result in terms of execution times and total function calls.

Overall, a similar number of total steps are observed from all the data generated by 2PDD4 and 2PDAM4 as presented in Tables 1–4. These results justify that both methods have potential to reduce the total step approximately by half when computing the results at two-point simultaneously.

Finally, Table 5 describes the comparison of maximum error obtained in Lang and Xu [6] using the Quintic b-spline collocation method of order four with 2PDD4 method. The first three results demonstrate that the maximum error for both methods is comparable. However, at $h = 0.025$ and $h = 0.0125$, the results showed that 2PDD4 are able to achieve smaller maximum error than the method in [6]. Figures 3, 4, 5, and 6 display clear information about the comparison

Table 5: Comparison of MAXE for solving Problem 5.

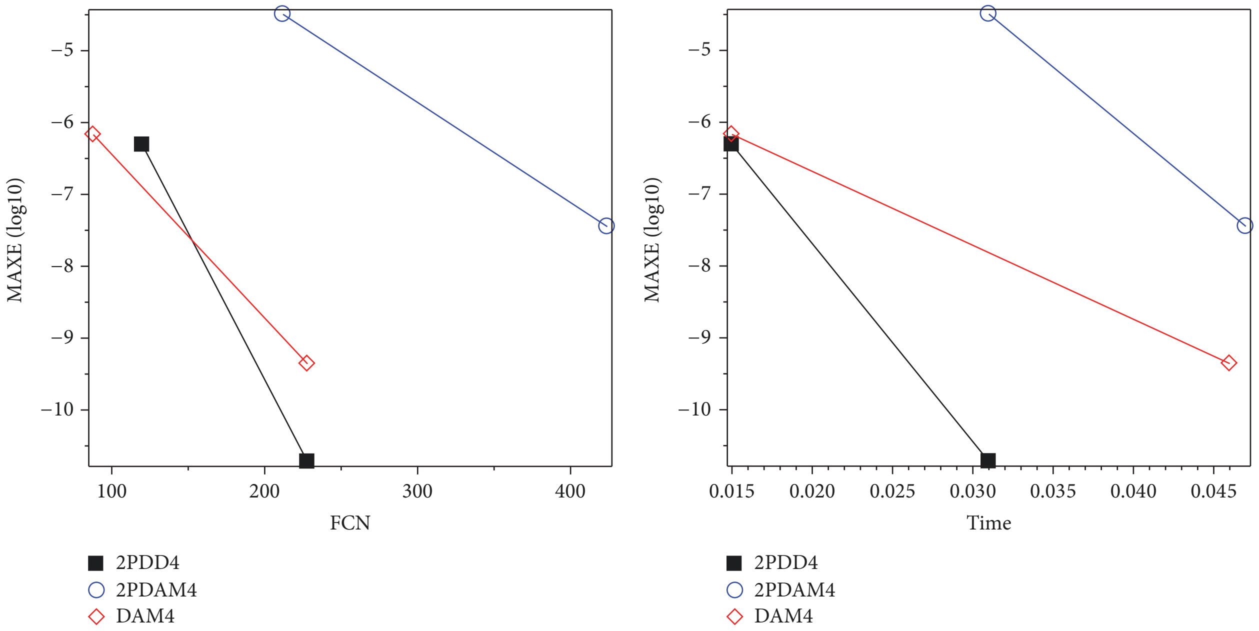
n	$h = 1/n$	MAXE Lang and Xu[6]	MAXE of 2PDD4
5	0.2	5.085000E-03	1.929712E-02
10	0.1	3.405000E-04	5.903279E-04
20	0.05	2.154000E-05	2.478867E-05
40	0.025	1.348000E-06	1.042691E-07
80	0.0125	8.425000E-08	6.295665E-09



(a) Comparison of FCN and MAXE

(b) Comparison of TIME and MAXE

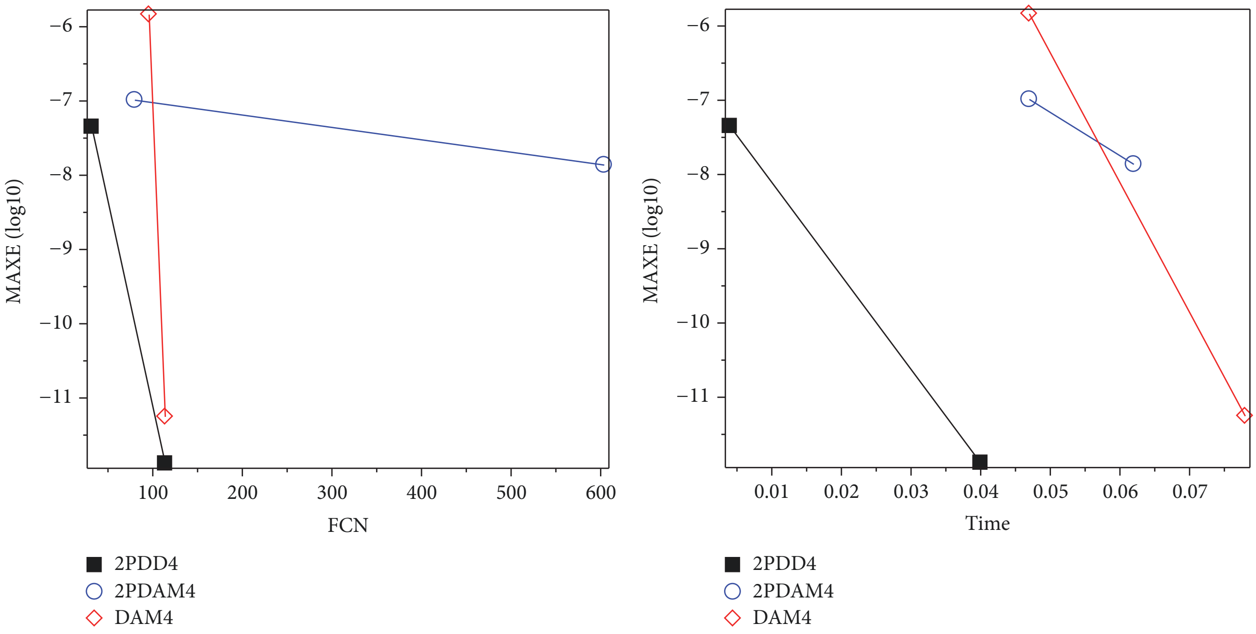
Figure 3: Graph of numerical result for solving Problem 1.



(a) Comparison of FCN and MAXE

(b) Comparison of TIME and MAXE

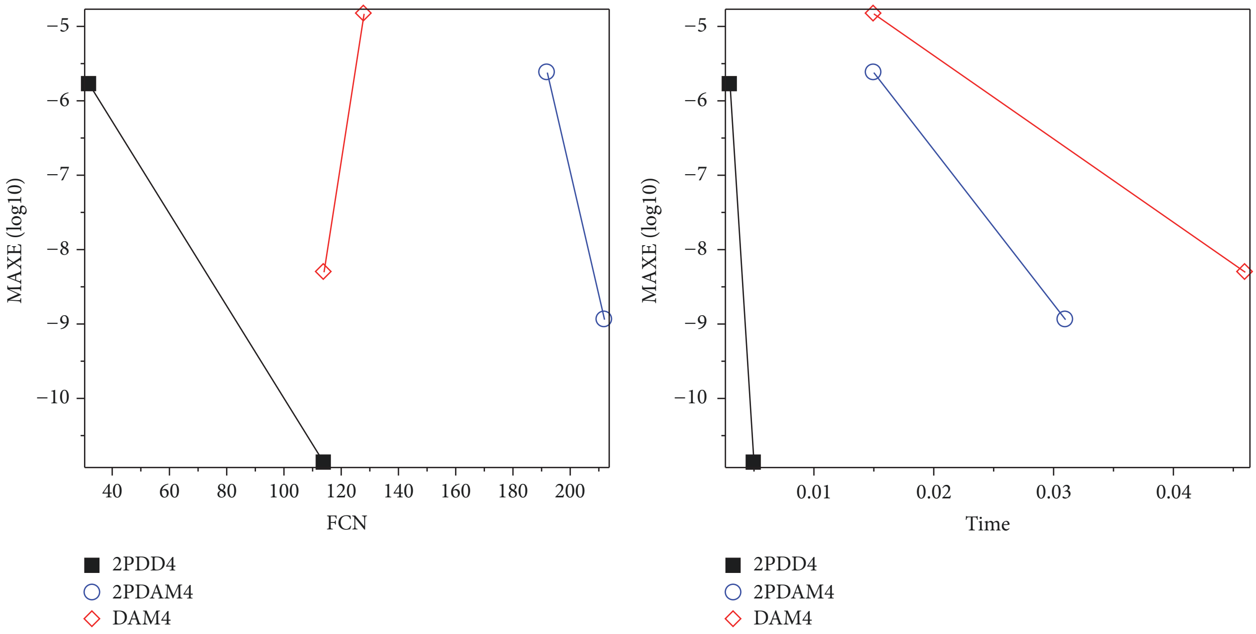
Figure 4: Graph of numerical result for solving Problem 2.



(a) Comparison of FCN and MAXE

(b) Comparison of TIME and MAXE

Figure 5: Graph of numerical result for solving Problem 3.



(a) Comparison of FCN and MAXE

(b) Comparison of TIME and MAXE

Figure 6: Graph of numerical result for solving Problem 4.

of the maximum errors versus the total function call as well as time for the numerical results summarized in Tables 1–4. Comparison result for Problem 5 is as depicted in Figure 7.

6. Conclusion

This study has presented a diagonal two-point block method formula of order four to solve directly the linear and nonlinear second-order Robin boundary conditions. Overview from the results showed a significant finding that the proposed diagonal block method manages to give faster execution time and less number of total function calls in all tested problems. The proposed method is able to achieve good accuracy when solving the problems. Future work should be devoted to solving third- and fourth-order Robin boundary value problems.

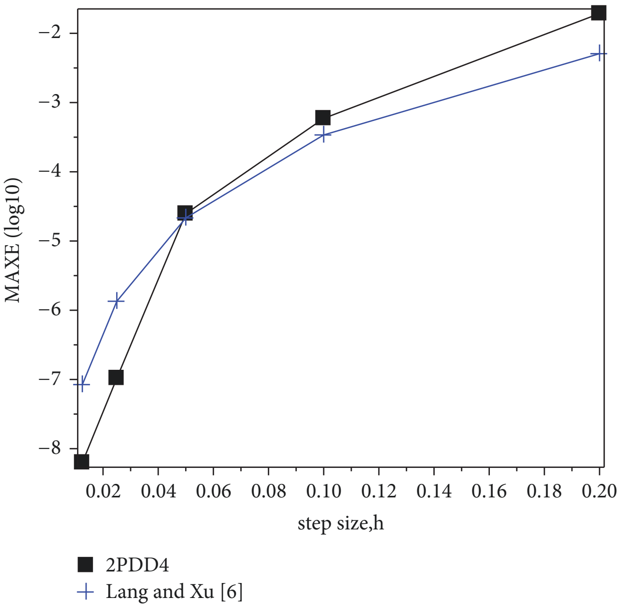


Figure 7: Graph of numerical result for solving Problem 5.

Data Availability

All data generated or analysed from the five tested problems in the article during this study are included in this published article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors gratefully acknowledge the financial support received from Putra Grant (Project Code: GP-IPS/2018/9625100), Universiti Putra Malaysia (UPM), and SLAB Scholarship sponsored by the Ministry of Higher Education (MOHE), Malaysia.

References

[1] T. T. Akano and O. A. Fakinlede, "Numerical computation of Sturm-liouville problem with Robin boundary condition," in *Proceedings of the World Academy of Science, Engineering and Technology, International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering*, vol. 9, pp. 684–688, 2015.

[2] M. 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.

[3] J. Duan, R. Rach, A. Wazwaz, T. Chaolu, and Z. Wang, "A new modified Adomian decomposition method and its multistage form for solving nonlinear boundary value problems with Robin boundary conditions," *Applied Mathematical Modelling*, vol. 37, no. 20–21, pp. 8687–8708, 2013.

[4] R. Rach, J.-S. Duan, and A.-M. Wazwaz, "Solution of higher-order, multipoint, nonlinear boundary value problems with high-order robin-type boundary conditions by the adomian decomposition method," *Applied Mathematics & Information Sciences*, vol. 10, no. 4, pp. 1231–1242, 2016.

[5] S. K. Bhatta and K. S. Sastri, "Symmetric spline procedures for boundary value problems with mixed boundary conditions," *Journal of Computational and Applied Mathematics*, vol. 45, no. 3, pp. 237–250, 1993.

[6] 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.

[7] S. O. Fatunla, "A class of block methods for second order ivps," *International Journal of Computer Mathematics*, vol. 55, no. 1-2, pp. 119–133, 1995.

[8] P. S. Phang, Z. A. Majid, and M. Suleiman, "On the solution of two point boundary value problems with two point direct method," *International Journal of Modern Physics: Conference Series*, vol. 9, pp. 566–573, 2012.

[9] Z. A. Majid, M. Mughti Hasni, and N. Senu, "Solving second order linear Dirichlet and Neumann boundary value problems by block method," *IAENG International Journal of Applied Mathematics*, vol. 43, no. 2, pp. 71–76, 2013.

[10] Z. Omar and O. Adeyeye, "Solving two-point second order boundary value problems using two-step block method with starting and non-starting values," *International Journal of Applied Engineering Research*, vol. 11, no. 4, pp. 2407–2410, 2016.

[11] Z. A. Majid, *Parallel block methods for solving ordinary differential equations [Ph.D. thesis]*, Universiti Putra Malaysia, 2004.

[12] I. S. M. Zawawi, Z. B. Ibrahim, F. Ismail, and Z. A. Majid, "Diagonally Implicit Block Backward Differentiation Formulas for Solving Ordinary Differential Equations," *International Journal of Mathematics and Mathematical Sciences*, vol. 2012, Article ID 767328, 8 pages, 2012.

[13] N. Zainuddin, Z. B. Ibrahim, and K. I. Othman, "Diagonally implicit block backward differentiation formula for solving linear second order ordinary differential equations," in *Proceedings of the 3rd international conference on fundamental and applied sciences (icfas 2014): innovative research in applied sciences for a sustainable future*, pp. 69–75, Kuala Lumpur, Malaysia, 2014.

[14] J. D. Lambert, *Computational Methods in Ordinary Differential Equations*, John Wiley & Sons, New York, NY, USA, 1973.

[15] P. P. See, Z. Abdul Majid, and M. Suleiman, "Three-step block method for solving nonlinear boundary value problems," *Abstract and Applied Analysis*, vol. 2014, Article ID 379829, 8 pages, 2014.

[16] J. Roberts, *Ordinary differential equations*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1979.

[17] P. S. Phang, Z. A. Majid, and M. Suleiman, "Solving nonlinear two point boundary value problem using two step direct method (menyelesaikan masalah nilai sempadan dua titik tak linear menggunakan kaedah langsung dua langkah," *Journal of Quality Measurement and Analysis*, vol. 7, no. 1, pp. 129–140, 2011.

[18] Z. A. Majid, P. P. See, and M. Suleiman, "Solving directly two point non linear boundary value problems using direct Adams Moulton method," *Journal of Mathematics and Statistics*, vol. 7, no. 2, pp. 124–128, 2011.

[19] R. P. Agarwal and D. ORegan, *An introduction to ordinary differential equations*, Springer Science & Business Media, 2008.