

Solving Directly Two Point Non Linear Boundary Value Problems Using Direct Adams Moulton Method

Zanariah Abdul Majid, Phang Pei See and Mohamed Suleiman
Department of Mathematics, Faculty Science, University Putra Malaysia,
43400 UPM Serdang, Selangor Darul Ehsan, Malaysia

Abstract: Problem statement: In this study, a direct method of Adams Moulton type was developed for solving non linear two point Boundary Value Problems (BVPs) directly. Most of the existence researches involving BVPs will reduced the problem to a system of first order Ordinary Differential Equations (ODEs). This approach is very well established but it obviously will enlarge the systems of first order equations. However, the direct method in this research will solved the second order BVPs directly without reducing it to first order ODEs. **Approach:** Lagrange interpolation polynomial was applied in the derivation of the proposed method. The method was implemented using constant step size via shooting technique in order to determine the approximated solutions. The shooting technique will employ the Newton's method for checking the convergent of the guessing values for the next iteration. **Results:** Numerical results confirmed that the direct method gave better accuracy and converged faster compared to the existing method. **Conclusion:** The proposed direct method is suitable for solving two point non linear boundary value problems.

Key words: Boundary Value Problems (BVPs), direct method, shooting method, Ordinary Differential Equations (ODEs), non linear boundary

INTRODUCTION

This study is concerned with the numerical solution of solving directly the second order two-point boundary value problems of the form as follows Eq. 1 and 2:

$$y'' = f(x, y, y') \quad (1)$$
$$a \leq x \leq b$$

with the boundary conditions:

$$y(a) = \alpha, y(b) = \beta. \quad (2)$$

Many problems in applied sciences and engineering are modeled as two-point boundary value problems such as the boundary layer theory in fluid mechanic, heat power transmission theory, space technology and reaction kinetics. A recent application of the boundary value problem can be found in Adegbe and Alao (2007) Jamshidi and Rostami (2008); Hassan (2009). Since the boundary value problem has wide application in scientific research, therefore faster and accurate numerical solutions of boundary value problem are very importance.

There are several methods that can be used to solve the two point boundary value problems numerically. It had been proposed by Attili and Syam (2008); Ha (2001); Jafri *et al.* (2009) and Taiwo and Ogunlaran (2008). Ha (2001) had solved the two-point boundary value problem using fourth order Runge-Kutta method via shooting technique. The second order boundary value problem has been reduced to a system of first order equation. Jafri *et al.* (2009) has consider solving directly two-point boundary value problem for second order ordinary differential equations using multistep method in term of backward difference formulas.

In this study, we propose a direct method of Adams Moulton type via shooting technique to solve the boundary value problem directly. The Newton method will be implemented as the iterative method to estimate the guessing values. The given equations in (1) will be treated in their original second order form and therefore the requirement of the storage is lower.

METERIALS AND METHODS

Formulation of the method: The point, y_{n+1} at x_{n+1} can be obtained by integrating (1) once and twice, i.e., integrate once Eq. 3-6:

Corresponding Author: Zanariah Abdul Majid, Department of Mathematics, Faculty of Science, University Putra Malaysia, 43400 UPM Serdang, Selangor Darul Ehsan, Malaysia

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

Therefore:

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

Integrate twice:

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

Therefore:

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

The function $f(x, y, y')$ in (4) and (6) will be approximated using Lagrange interpolating polynomial and the interpolation points involved are four points, five points and six points ie $\{x_{n-2}, x_{n-1}, x_{n+1}\}, \{x_{n-3}, x_{n-1}, x_n, x_{n+1}\}$ and $\{x_{n-4}, x_{n-3}, x_{n-2}, x_{n-1}, x_n, x_{n+2}\}$ respectively. Taking $s = \frac{x - x_{n+1}}{h}$ and by replacing $dx = h ds$, the value of x_{n+1} can be obtained by integrating (4) and (6) over the interval $[x_n, x_{n+1}]$ using mathematica software and the corrector formulae can be obtained. The method is the combination of predictor of one order less than the corrector.

Direct method of order 4:

Predictor:

$$y'_{n+1} = y'_n + \frac{h}{12}(23f_n - 16f_{n-1} + 5f_{n-2})$$

$$y_{n+1} = y_n + hy'_n + \frac{h^2}{24}(19f_n - 10f_{n-1} + 3f_{n-2})$$

Corrector:

$$y'_{n+1} = y'_n + \frac{h}{24}(9f_{n+1} + 19f_n - 5f_{n-1} + f_{n-2})$$

$$y_{n+1} = y_n + hy'_n + \frac{h^2}{360}(38f_{n+1} + 171f_n - 36f_{n-1} + 7f_{n-2})$$

Direct method of order 5:

Predictor:

$$y'_{n+1} = y'_n + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3})$$

$$y_{n+1} = y_n + hy'_n + \frac{h^2}{360}(323f_n - 264f_{n-1} + 159f_{n-2} - 38f_{n-3})$$

Corrector:

$$y'_{n+1} = y'_n - \frac{h}{720} \begin{pmatrix} -251f_{n+1} - 646f_n + 264f_{n-1} \\ -106f_{n-2} + 19f_{n-3} \end{pmatrix}$$

$$y_{n+1} = y_n + hy'_n - \frac{h^2}{1440} \begin{pmatrix} -135f_{n+1} - 752f_n + 264f_{n-1} \\ -96f_{n-2} + 17f_{n-3} \end{pmatrix}$$

Direct method of order 6:

Predictor:

$$y'_{n+1} = y'_n + \frac{h}{720} \begin{pmatrix} 1901f_n - 2774f_{n-1} + 2616f_{n-2} \\ -1274f_{n-3} + 251f_{n-4} \end{pmatrix}$$

$$y_{n+1} = y_n + hy'_n + \frac{h^2}{1440} \begin{pmatrix} 1427f_n - 1596f_{n-1} + 1446f_{n-2} \\ -692f_{n-3} + 135f_{n-4} \end{pmatrix}$$

Corrector:

$$y'_{n+1} = y'_n + \frac{h}{1440} \begin{pmatrix} 475f_{n+1} + 1427f_n - 798f_{n-1} \\ + 482f_{n-2} - 173f_{n-3} + 27f_{n-4} \end{pmatrix}$$

$$y_{n+1} = y_n + hy'_n + \frac{h^2}{10080} \begin{pmatrix} 863f_{n+1} + 5674f_n - 2542f_{n-1} \\ + 1492f_{n-2} - 52f_{n-3} + 82f_{n-4} \end{pmatrix}$$

The Euler method will be used only once at the beginning of the code to find the additional points for the starting initial points. Then, the predictor and corrector direct method can be applied until the end of the interval. This direct method will be adapted for solving the boundary value problems via shooting techniques. Shooting technique will allow for new guessing and for each new guessing of the y' , the Euler method will be used again to find the starting initial points.

Implementation of the method: The values of y'_{n+1} and y_{n+1} will be implemented using the predictor and

corrector schemes. The sequence of computation involved was PECE where P and C indicate the application of the predictor and corrector formula respectively and E indicate the evaluation of the function f. Shooting techniques is an analogy with the procedure of firing objects as a stationary target. It solves the problem with trial and error. To form an initial value problem out of boundary value problem (1), the initial value of y' need to be guessed. The solution depends on both x and s_0 . Starting with the initial guess, s_0 , the approximated solution of the derivative $y'(a)$ gives Eq. 7-11:

$$y'' = f(x, y, y') \tag{7}$$

$$y(a) = \alpha, \quad y'(a) = s_0, \quad a \leq x \leq b$$

Equation 6 can be written as:

$$\frac{d^2y(x,s)}{ds^2} = f(x, y(x,s), y'(x,s)) \tag{8}$$

$$y(a,s) = \alpha, \quad \frac{dy(a,s)}{dx} = s$$

For the first initial guessing, s_0 , we considered Richard *et al.* (1981) Eq. 9:

$$s_0 = \frac{\beta - \alpha}{b - a} \tag{9}$$

The solution of (8) will coincide with the solution of (1-2) if we could find the value of $s = s_v$ such that:

$$\varphi(s) = y(b, s_v) - \beta = 0 \tag{10}$$

Newton method will be used to get a very rapidly converging iteration. We compute the $\{s_v\}$ defined as:

$$s_{v+1} = s_v - \frac{\varphi(s_v)}{\varphi'(s_v)}$$

Differentiate (8) with respect to s and let $z(x,s) = \frac{dy}{ds}(x,s)$, thus (8) is simplify as follows:

$$z'' = \frac{d}{dy} f(x, y, y')z + \frac{d}{dy'} f(x, y, y')z' \tag{11}$$

$$a \leq x \leq b, z(a) = 0, \quad z'(a) = 1$$

Therefore, the new guess can be calculated base on the previous guess using:

$$s_{n+1} = s_n + \frac{y(b, s_n) - \beta}{z(b, s_n)}, \quad i = 1, 2, \dots$$

Both of Eq. 8 and 11 will be solved simultaneously using the direct method. The process is repeated over and over until the error $|\beta - y(b, s_n)| \leq \text{tolerance}$. The algorithms of the proposed method were developed in C language.

Tested problem: Three tested problems are presented and those problems are referred Ha (2001). The problems will be tested to the direct method of Adams Moulton type (DAM) mentioned earlier. However, Ha (2001) used the fourth-order Runge-Kutta method via shooting technique using generalized Newton's method and reduced the given problems to first order system of equations.

Problem 1: $y'' = \frac{32 + 2x^3 - yy'}{8}, 1 \leq x \leq 3$ with the boundary condition $y(1) = 17, y(3) = \frac{43}{3}$. The problem has exact solution: $y = x^2 + \frac{16}{x}$.

Problem 2: $y'' = 2y^3 - 6y - 2x^2, 1 \leq x \leq 2$ with the boundary condition $y(1) = 2, y(2) = \frac{5}{2}$. The problem has exact solution: $y = x + \frac{1}{x}$.

Problem 3: $y'' = y^3 - yy', 1 \leq x \leq 2$ with the boundary condition $y(1) = \frac{1}{2}, y(2) = \frac{1}{3}$. The problem has exact solution: $y = \frac{1}{x+1}$.

RESULTS

The following notations are used in the tables:

MAXE	Maximum error
AVE	Average error
Error x_{last}	The error for last x at last iteration
ITN	Total iteration of guess
TS	Total step at last iteration
s_v	Guess at last iteration
HA	The numerical results Ha (2001)
DAM4	Direct Adams Moulton method of order 4
DAM5	Direct Adams Moulton method of order 5
DAM6	Direct Adams Moulton method of order 6

DISCUSSION

In Problem 1, we chose $h = 0.01$ and error bound = 10^{-5} . Table 1 showed the value of iteration for calculating s_v in DAM4, DAM5 and DAM6 methods when using approach in (9) and obtained $s_0 = -1.3335$. The final calculating s_v is converged to the values of -14.0000 for DAM4, DAM5 and DAM6. The iteration needed for calculating s_v when $s_0 = -1.3335$ is five. The DAM methods manage to give good numerical results at each step. Using the same h and error bound 10^{-5} we tried the initial guess $s_0 = 0.25, 0.5$ and 1.0 as Ha (2001). The comparison for iterating those initial guess with Ha (2001) is showed in Table 2-4.

The maximum errors and number of iteration for DAM4 using $s_0 = 0.25, 0.5, 1.0$ are much better than HA as shown in Table 2-4. The DAM4, DAM5 and DAM6 methods have converging final $s_v = -14.0000$ for each initial guess in Table 2-4. It is also observed that all the DAM method has the same number of iteration and the error at x_{last} is very small. The DAM methods manage to obtain better accuracy as the order increased.

In Problem 2, we used the same approach as in (9) and obtained the initial guess $s_0 = 0.5$. Table 5 shows a good numerical result when using initial values $s_0 = 0.5$. The final calculating s_v is converged to the values of -0.00000 for all DAM4, DAM5 and DAM6. The iteration needed for calculating s_v when $s_0 = 0.5$ is fourteen. The error estimated at the last x is very accurate i.e., $3.129e-13, 5.118e-12$ and $6.267e-11$ for DAM4, DAM5 and DAM6 respectively. The DAM method manages to give good numerical results at each order. Then we chose $h = 0.05$ and error bound = 10^{-5} as Ha (2001); and tried the initial guess $s_0 = 0.6975$. The maximum error and number of iteration for DAM4 is better compared to HA in Table 6. The DAM4 only need 40 iterations in order to converge while HA took 9982 iterations. The DAM methods have better accuracy as the order increase and the converging final is $s_v = 0.00000$. The DAM5 and DAM6 only need 41 and 42 iterations in order to converge.

Table 7 show the numerical results of DAM4, DAM5 and DAM6 method when initial guess $s_0 = 0.1667$ using approach in (6) when solving Problem 3. This problem was tested using $h = 0.05$ and error bound 10^{-5} . The final calculating of s_v is converged to the values of -0.25000 . The iteration needed for calculating s_v when $s_0 = 0.1667$ is three. The DAM methods manage to give good numerical results at each order.

We tried the initial guess of $s_0 = 3.99, 4.00, 4.10, 4.15$ and 4.175 when solving Problem 3. The comparison for iterating those initial guess with Ha (2001) is showed in Table 8-12.

Table 1: The numerical results when using $s_0 = -1.3335$ for solving problem 1

	DAM4	DAM5	DAM6
MAXE	2.379e-08	8.536e-09	5.196e-09
AVE	1.230e-08	2.786e-09	2.085e-09
Error x_{last}	3.734e-11	3.733e-11	3.733e-11
TS	200	200	200
ITN	5	5	5
s_v	-14.0000	-14.0000	-14.0000

Table 2: Comparison of the iterations with $s_0 = 0.25$ when solving problem 1

	HA	DAM4	DAM5	DAM6
MAXE	1.100e-05	2.373e-08	8.526e-09	5.184e-09
AVE	4.800e-06	1.217e-08	2.678e-09	1.985e-09
Error x_{last}	1.100e-05	2.023e-10	2.022e-10	2.022e-10
TS	200	200	200	200
ITN	20	5	5	5
s_v	-13.9999	-14.0000	-14.0000	-14.0000

Table 3: Comparison of the iterations with $s_0 = 0.5$ when solving problem 1

	HA	DAM4	DAM5	DAM6
MAXE	1.100e-5	2.370e-008	8.523e-009	5.180e-009
AVE	4.800e-6	1.212e-008	2.650e-009	1.961e-009
Error x_{last}	1.100e-5	2.593e-010	2.593e-010	2.593e-010
TS	200	200	200	200
ITN	20	5	5	5
s_v	-13.9999	-14.0000	-14.0000	-14.0000

Table 4: Comparison of the iterations with $s_0 = 1.0$ when solving problem 1

	HA	DAM4	DAM5	DAM6
MAXE	1.400e-5	2.364e-008	8.514e-009	5.169e-009
AVE	4.850e-6	1.200e-008	2.587e-009	1.913e-009
Error x_{last}	1.400e-5	4.206e-010	4.206e-010	4.205e-010
TS	200	200	200	200
ITN	20	5	5	5
s_v	-13.9999	-14.0000	-14.0000	-14.0000

Table 5: The numerical results when using $s_0 = 0.5$ for solving problem 2

	DAM4	DAM5	DAM6
MAXE	6.239e-07	2.303e-07	4.205e-07
AVE	4.564e-07	2.562e-07	5.895e-07
Error x_{last}	3.129e-13	5.118e-12	6.267e-11
TS	20	20	20
ITN	14	14	14
s_v	-0.0000	-0.0000	-0.0000

Table 6: Comparison of the numerical results when $s_0 = 0.6975$ for solving problem 2

	HA	DAM4	DAM5	DAM6
MAXE	3.650e-03	6.239e-07	2.303e-07	4.205e-07
AVE	2.324e-03	4.564e-07	2.562e-07	5.895e-07
Error x_{last}	4.800e-06	8.780e-13	1.368e-11	3.400e-11
TS	20	20	20	20
ITN	9982	40	41	42
s_v	-0.00606	-0.00000	-0.00000	-0.00000

Table 7: The numerical results when using $s_0 = 0.1667$ for solving problem 3

	DAM4	DAM5	DAM6
MAXE	1.795e-08	4.259e-08	4.259e-08
AVE	1.299e-08	4.593e-08	4.593e-08
Error x_{last}	4.769e-09	4.768e-09	4.768e-09
TS	20	20	20
ITN	3	3	3
s_v	-0.25000	-0.25000	-0.25000

Table 8: Comparison of the numerical results when $s_0 = 3.99$ for solving problem 3

	HA	DAM4	DAM5	DAM6
MAXE	1.190e-4	3.243e-007	3.206e-007	3.190e-007
AVE	8.030e-5	1.808e-007	2.501e-007	3.102e-007
Error x_{last}	7.000e-6	3.243e-007	3.206e-007	3.190e-007
TS	20	20	20	20
ITN	Not stated	5	5	5
s_v	-0.24935	-0.25000	-0.25000	-0.25000

Table 9: Comparison of the numerical results when $s_0 = 4.00$ for solving problem 3

	HA	DAM4	DAM5	DAM6
MAXE	1.180e-4	3.382e-007	3.344e-007	3.326e-007
AVE	7.970e-5	1.891e-007	2.590e-007	3.200e-007
Error x_{last}	6.000e-6	3.382e-007	3.344e-007	3.326e-007
TS	20	20	20	20
ITN	Not stated	5	5	5
s_v	-0.24935	-0.25000	-0.25000	-0.25000

Table 10: Comparison of the numerical results when $s_0 = 4.10$ for solving problem 3

	HA	DAM4	DAM5	DAM6
MAXE	1.130e-4	4.863e-007	4.822e-007	4.807e-007
AVE	7.345e-5	2.764e-007	3.545e-007	4.264e-007
Error x_{last}	6.000e-6	4.863e-007	4.822e-007	4.807e-007
TS	20	20	20	20
ITN	Not stated	5	5	5
s_v	-0.24935	-0.25000	-0.25000	-0.25000

Table 11: Comparison of the numerical results when $s_0 = 4.15$ for solving problem 3

	HA	DAM4	DAM5	DAM6
MAXE	1.120e-4	5.582e-007	5.544e-007	5.529e-007
AVE	7.270e-5	3.189e-007	4.011e-007	4.783e-007
Error x_{last}	8.000e-6	5.582e-007	5.544e-007	5.529e-007
TS	20	20	20	20
ITN	Not stated	5	5	5
s_v	-0.24935	-0.25000	-0.25000	-0.25000

Table 12: Comparison of the numerical results when $s_0 = 4.175$ for solving problem 3

	HA	DAM4	DAM5	DAM6
MAXE	1.190e-4	5.906e-007	5.870e-007	5.856e-007
AVE	8.070e-5	3.379e-007	4.222e-007	5.018e-007
Error x_{last}	8.000e-6	5.906e-007	5.870e-007	5.856e-007
TS	20	20	20	20
ITN	Not stated	5	5	5
s_v	-0.24935	-0.25000	-0.25000	-0.25000

In Table 8-12, we could observe that the maximum error for DAM4 is better than HA. The approximating error at x last for DAM4 is much better than in HA. The iteration needed in DAM4, DAM5 and DAM6 for calculating s_v is five. The converging final for all the DAM methods is -0.25000 but HA converge to -0.24935.

CONCLUSION

In this study, we have shown the proposed direct Adams Moulton method via shooting technique using Newton's method is suitable for solving two point second order non linear boundary value problems. This proposed method is simple, efficient and economically.

REFERENCES

- Adegbe, K.S. and F.I. Alao, 2007. On thermal explosion of sensitized reaction with variable heat loss. *Am. J. Applied Sci.*, 4: 53-55. DOI: 10.3844/ajassp.2007.53.55
- Attili, B.S. and M.I. Syam, 2008. Efficient shooting method for solving two point boundary value problems. *Chaos. Solitons Fractals*, 35: 895-903. DOI: 10.1016/J.CHAOS.2006.05.094
- Ha, S.N., 2001. A nonlinear shooting method for two point boundary value problems. *Comput. Math. Appl.*, 42: 1411-1420. DOI: 10.1016/S0898-1221(01)00250-4
- Hassan, A.M., 2009. Soret and dufour effects on natural convection flow past a vertical surface in a porous medium with variable surface temperature. *J. Math. Stat.*, 5: 190-198. DOI: 10.3844/jmssp.2009.190.198
- Jafri, M.D., M. Suleiman, Z.A. Majid and Z.B. Ibrahim, 2009. Solving directly two point boundary value problems using direct multistep method. *Sains Malaysian*, 38: 723-728.
- Jamshidi, N. and M. Rostami, 2008. Gait optimization of biped robot during double support phase by pure dynamic synthesis. *Am. J. Applied Sci.*, 5: 1175-1181. DOI: 10.3844/ajassp.2008.1175.1181
- Richard, L., J. Burden, D. Faires and A.C. Reynolds, 1981. *Numerical Analysis*. 2nd Edn., Prindle, Weber and Schmidt, Boston, ISBN-10: 087150314X, pp: 598.
- Taiwo, O.A. and O.M. Ogunlaran, 2008. Numerical solution of fourth order linear ordinary differential equations by cubic spline collocation method. *J. Math. Stat.*, 4: 264-268. DOI: 10.3844/jmssp.2008.264.268