International Journal of Systems Science and Applied Mathematics
2026, Vol. 11, No. 3, pp. 53–61
https://doi.org/10.11648/j.ijssam.20261103.11

Research Article

Numerical Solution of the Mildly Non-linear Boundary Value Problems (MNBVP) Using Newton-Lieberstein Algorithm

Adebayo Kayode James[1, *], Akinmuyise Matthew Folorunsho[2], Dele-Rotimi Adejoke Olumide[3]

[1]Department of Mathematics, Ekiti State University, Ado-Ekiti, Nigeria
[2]Department of Mathematics, Adeyemi Federal University of Education, Ondo, Nigeria
[3]Department of Mathematics, Bamidele Olumilua University of Education, Science and Technology, Ikere-Ekiti, Nigeria

Abstract

Several numerical and classical approaches have been employed to determine the Boundary Value Problems solutions, but not many of these methods have been adopted to work out Mildly Non-Linear Boundary Value Problems (MNBVP); hence, there is a necessity to think of ways of achieving this feat. The MNBVP are problems that are neither linear nor explicitly nonlinear. This paper discusses the process of intertwining the Newton-Lieberstein method into the upwind differencing technique in solving mildly non-linear boundary value problems. The method requires two independent stages that involve different techniques. First, transform the BVP to a tri-diagonal system. Secondly, solve the system of equations by applying the Newton-Lieberstein Algorithm. The resulting equations from the nonlinear boundary problems considered in this paper will generate a nonlinear but will relate structurally to tridiagonal equations. Here, the authors developed a method described as a hybridized iterative algorithm, for which the number of steps cannot be determined a priori. It is a variant of the Newton's technique. Unlike other classical methods, it applies to nonlinear systems and to systems with more than a thousand equations, after which round off error tends to accumulate. In real life, mathematical problems can now be expressed in the BVPs that sprang largely in the fields of physics, science, and engineering, such as fluid dynamics, electric circuits, the motion of rockets or satellites, and other areas of application. The algorithm proposed will estimate the approximate solutions irrespective of the number of equations involved. Numerical results showed the robustness, efficiency, and flexibility of the intertwined technique.

Keywords

Boundary Value Problem, Mildly Non-Linear Boundary Value Problem, Newton-Lieberstein Method, Tridiagonal Systems, Upwind Differencing

1. Introduction

In recent years, several problems have arisen from different fields like science and engineering that are presented by mathematical models. Most of these models are presented in the form of state differential equations, which could either in first or higher-order ordinary differential equations (ODEs). This research focuses on finding solution to the ODEs second-order

*Correspondence: Adebayo Kayode James (kayode.adebayo@eksu.edu.ng)

Received: 7 August 2026; **Accepted:** 17 August 2026; **Published:** 27 August 2026

boundary value problems (BVPs), usually of the form

$$y'' = f(x, y, y')\ y(a) = y_0,\ y'(a) = y'_0\ a < x < b. \quad (1)$$

where the values of a, b, y_0, and y'_0 are specified constants and importantly, $a \neq b$.

As opined by Omar, Z., (2005) [1] and Dennis G. Z., (2009) [2], it's sacrosanct to find a solution $y(x)$ in (1) that is continuous on the interval $x \in [a, b]$ and that will satisfy the boundary conditions, to which an alternative approach is proposed by Nur Z. M., (2011) [3]. While IVP has a unique solution, BVP may have unique, multiple, or no solutions. In mathematical modeling, Boundary Value Problems (BVPs) are basic and have covered a wide range in practice in the Biological Sciences physical sciences, and Engineering. They involve finding solution to differential equations with conditions given in the boundaries of the domain. This eventually makes them important to accurately describe real-life phenomena such as structural analysis, heat conduction, and fluid dynamics.

The approach adopted in this work is to solve (1) by intertwining the Newton's-Lieberstein method after the BVP has been reduced to a tridiagonal system. Several methods have been employed to compute the approximation solution of four points simultaneously in form of a block. The basic idea of the one-step block method was carefully laid out by Rosser, J. B., (1967) [4], as a block of new approximation values was introduced simultaneously into the laid framework. The approach was also discussed in Worland, P. B., (1976) and Majid, et al., (2003), [5] and [6]. In the work of Majid, Z. A., (2006) [7], the two-point implicit one-step block technique was described by the author for the solution first-order ODEs based on an integral formula using the nearest point in the block.

Several researchers have already solved (1) directly, such as Chakravarti, P. C., (1971), Suleiman, M. B., (1989), and Fatunla, S. O., (1991) [8-10]. In that case, higher-order system of ODEs can be resolved to a system of first-order equations and then solved using first-order ODEs. This approach enlarges the system of first-order ODEs and requires extra computational work.

For a second-order ODE, any of the following can, and will, be used for boundary conditions Trench, W. F., (2013), [11]:

Dirichlet or First Kind Boundary Condition:

$$y(x_0) = y_0,\ y(x_1) = y_1, \quad (2)$$

Regular Boundary Condition:

$$y(x_0) = y_0,\ y'(x_1) = y_1, \quad (3)$$

Neumann or Second Kind Boundary Condition:

$$y'(x_0) = y_0,\ y'(x_1) = y'_1, \quad (4)$$

Periodic Boundary Condition:

$$y'(x_0) = y'_0\ y(x_1) = y_1, \quad (5)$$

So, our discussion will be directed towards a linear differential equation.

2. Mildly Non-Linear Boundary Value Problem

The Mildly Non-linear Boundary Value Problems (MNBVP) can be in various forms depending on the transcendental function associated with it. However, the commonest studied (MNBVP) is of the form:

$$y'' = f(x, y, y') \quad (6)$$

$$y(a) = \alpha,\ y(b) = \beta. \quad (7)$$

where a is not equal to b. if $a = b$ in (7), the system is not a BVP rather an Initial Value Problem. To ensure the uniqueness of (6) and (7), it is assumed that.

$$\frac{\partial f(x,y)}{\partial y} \geq 0, x \in [a, b], \text{ and } -\infty < y < \infty. \quad (8)$$

When (8) is valid, (6) and (7) are referred to as a Mildly Nonlinear problem.

While there are many theories to back the solution of both the linear and nonlinear BVPs numerically though appears rather more difficult than those meant to substantiate the results for initial value problems, Ibrahim S. M., (1992), [12]. Attention will be on theorems to support the linear boundary value problems of a special type, as in (8) and (7).

Theorem 1: Let I be an interval $a < x < b$. Let α and β be constants, and let $P(x), Q(x)$, and $R(x)$ be continuous on $a \leq x \leq b$. Considering the linear BVP

$$y'' + P(x)y' + Q(x)y = R(x), x \in I \quad (9)$$

$$y(a) = \alpha, \text{ and } y(b) = \beta. \quad (10)$$

It is pertinent to note that, for numerical reasons, *a priori*, the solution of the problem (9) and (10) exists, and this is characterized by a unique solution. Owing to this, in addition to the above, let us assumed that, if $a \leq x \leq b$,

$$Q(x) \leq 0 \quad (11)$$

This is a sufficient reason to establish the existence and the uniqueness of the solution.

If it is assumed that y' is an independent variable, then

$$\frac{\partial}{\partial y}[-P(x)y' - Q(x)y + R(x)] = -Q(x) \quad (12)$$

Hence, the condition (8) will imply either $-Q(x) \geq 0$ or $Q(x) \leq 0$.

3. Transforming Boundary Value Problems to Tri-diagonal Systems

The numerical solution of (6) and (7) follows in the manner prescribed with the three-point central approximation

$$\frac{y_{i+1}-y_{i-1}}{2h} = y_i' \text{ (for two points central)} \quad (13)$$

$$\frac{y_{i-1}-2y_i+y_{i+1}}{h^2} = f(x_i, y_i) \text{ (for three-point central)} \quad (14)$$

Replacing (13) and (14) in (9) gives rise to

$$\frac{y_{i-1}-2y_i+y_{i+1}}{h^2} + P(x_i)\frac{y_{i+1}-y_{i-1}}{2h} + Q(x_i)y_i = R(x_i) \quad (15)$$

For $i = 1, 2, 3, \dots, n-1$, the resulting nonlinear (15) can now be solved using the Newton-Lieberstein's method.

To illustrate the processes of changing the differential equation into an argument matrix that will later be solved using the Newton-Lieberstein's method, the following examples will be employed to demonstrate that.

For $n = 2, 3, 4, \dots$ the general form for the system of linear algebraic of n equations of the n unknowns $x_1, x_2, \dots, x_n$ is given as

$$\left.\begin{array}{c} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_1 = f_1(x_1) \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_2 = f_2(x_2) \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_3 = f_3(x_3) \\ \vdots \; \vdots \; \vdots \; \ddots \; \vdots \; \vdots \\ a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n = f_n(x_n) \end{array}\right\} \quad (16)$$

From (16), the coefficient matrix, A, the variable vector, x, and the solution vector, b are given by

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix}, \; x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix},$$

and

$$b = \begin{pmatrix} f_1(x_1) \\ f_2(x_2) \\ f_3(x_3) \\ \vdots \\ f_n(x_n) \end{pmatrix} \quad (17)$$

Then, the (16) can be compactly written as

$$Ax = b \quad (18)$$

The system (15) or its equivalent (18), for $i = 1, 2, 3, \dots, n$, $j = 1, 2, 3, \dots, n-1$, is called a tri-diagonal if all the entries in A are zeros except $a_{ii}, a_{j,j+1}$, and $a_{j+1,j}$. A system of simultaneous equations with nonzero coefficients only on the lead diagonal, on its lower diagonal, and on its upper diagonal is called a tri-diagonal system of equations.

The term tri-diagonal is most appropriate for the coefficient matrix, A, in (18), provided it's of the form that its lead diagonal a_{ii}, the super-diagonal $a_{j,j+1}$, i.e., diagonal above the lead diagonal, and the sub-diagonal $a_{j+1,j}$, i.e., diagonal below the only non-zero entries.

For practical importance, to ensure a unique solution for the tri-diagonal system, it is expedient that the conditions (19), (20), and (21) must all hold.

Theorem 2: Let (18) be a tri-diagonal system with the entries of the lead diagonal being negative while the entries of the sub-diagonal and super-diagonal are positive, such that

$$a_{ii} < 0, i = 1, 2, 3, \dots, n \quad (19)$$

$$a_{i,i+1} > 0, i = 1, 2, 3, \dots, n-1 \quad (20)$$

$$a_{i+1,i} > 0, i = 1, 2, 3, \dots, n-1. \quad (21)$$

Furthermore, to ensure (19) through (21) hold, the leading diagonal entries are made to dominate the matrix such that the absolute value of the leading diagonal entries will be greater than or will be equal to the sum of all the entries of the other row, with firm inequality for at least one row of the coefficient matrix, Paul E., (2020), [13]. Precisely, let

$$-a_{11} \geq a_{12} \quad (22)$$

$$-a_{n,n} \geq a_{n,n-1} \quad (23)$$

$$-a_{ii} \geq a_{i,i-1}, i = 2, 3, \dots., n-1, \quad (24)$$

With the firm inequality condition holding for at least one of (22) through (24), the resulting linear algebraic system is said to have a unique solution.

The afore knowledge of theorem 2 before starting the computation processes cannot be underestimated because, for systems that having no solutions, computations in such systems will lead to no right or meaningful direction. If a system of equation should have more than a solution, applying a particular computational method to solve it could cause a shift from one result to another thereby, confusion.

On considering (16) with the coefficient

$$A = \begin{pmatrix} a_{11}\; a_{12}\; 0 \\ a_{21}\; a_{22}\; a_{23} \\ a_{33}\; a_{34} \\ \cdots\; \cdots\; \cdots \\ \cdots\; \cdots\; \cdots \\ \cdots\; \cdots\; \cdots \\ a_{n-1,n-2}\; a_{n-1,n-1}\; a_{n-1,n} \\ 0\; a_{n,n-1}\; a_{nn} \end{pmatrix} \quad (25)$$

satisfying the conditions in Theorem 2. If $f_i(x_i)$ for all $i =$

$1, 2, ..., n$, are not all constants and the derivative, $f_i'(x_i) \geq 0$, then the system is referred to as a mildly nonlinear problem. According to Radzi, H. M., (2011) [14], all such mildly nonlinear system have solutions and their solutions are said to be unique.

4. Newton-Lieberstein Method Algorithm

The Newton-Lieberstein method for solving a mildly nonlinear system is itemized in the algorithmic process below as:

Step 1: Initialize guess values for the variables $x_1^{(0)}, x_2^{(0)}, x_3^{(0)}, ..., x_n^{(0)}$ and a value for ω within the range $[0, 2]$.

Step 2: For $k = 0, 1, 2, 3, ...,$ iterate with the relations

$$x_1^{(k+1)} = x_1^{(k)} - \omega\left(\frac{a_{11}x_1^{(k)} + a_{12}x_2^{(k)} - f_1(x_1^{(k)})}{a_{11} - f_1'(x_1^{(k)})}\right)$$

$$x_2^{(k+1)} = x_2^{(k)} - \omega\left(\frac{a_{21}x_1^{(k)} + a_{22}x_2^{(k)} + a_{23}x_3^{(k)} - f_2(x_2^{(k)})}{a_{22} - f_2'(x_2^{(k)})}\right)$$

$$x_3^{(k+1)} = x_3^{(k)} - \omega\left(\frac{a_{32}x_2^{(k)} + a_{33}x_3^{(k)} + a_{34}x_4^{(k)} - f_3(x_3^{(k)})}{a_{33} - f_3'(x_3^{(k)})}\right)$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$x_{n-1}^{(k+1)} = x_{n-1}^{(k)} - \omega\left(\frac{a_{n-1,n-2}x_{n-2}^{(k)} + a_{n-1,n-1}x_{n-1}^{(k)} + a_{n-1,n}x_n^{(k)} - f_{n-1}(x_{n-1}^{(k)})}{a_{n-1,n-1} - f_{n-1}'(x_{n-1}^{(k)})}\right)$$

$$x_n^{(k+1)} = x_n^{(k)} - \omega\left(\frac{a_{n,n-1}x_{n-1}^{(k)} + a_{nn}x_n^{(k)} - f_n(x_n^{(k)})}{a_{nn} - f_n'(x_n^{(k)})}\right)$$

Step 3: Determine if $\left|x_i^{(k+1)} - x_i^{(k)}\right| < \epsilon$, for $i = 1, 2, 3, ..., n$ by setting a prescribed convergence tolerance, ϵ. Otherwise, update and repeat step 2 iteratively.

Step 4: If step 3 is satisfied, substitute the values of $x_1^{(k+1)}, x_2^{(k+1)}, x_3^{(k+1)}, ..., x_n^{(k+1)}$ into the original system of equations. This is to ascertain that the values obtained for the variables are an approximate solution to the given problem.

5. Results and Discussions

Problem 1: Solve the Mildly Nonlinear Boundary Value Problem

$$y'' = e^y \tag{26}$$

under the conditions

$$y(0) = 0, y(1) = 0. \tag{27}$$

It is expedient to mention that the Newton-Lieberstein Method will perform well if the BVP is reduced to a tridiagonal system with a unique solution and the solution will exist. To determine the existence of solution of Problem 1 and if the result is unique, the function $f(x, y) = e^y$, such that its derivative $\frac{\partial}{\partial y}f(x, y) = e^y > 0$. Thus, the solution to the problem exists, and it is unique, Adebayo, (2025), [15].

To solve the problem numerically, the interval $[0, 1]$ must be divided into 10 equal parts with the step size, $h = 0.1$ gives rise to $x_0 = 0, x_1 = 0.1, x_2 = 0.2, x_3 = 0.3, x_4 = 0.4, x_5 = 0.5, x_6 = 0.6, x_7 = 0.7, x_8 = 0.8, x_9 = 0.9, x_{10} = 1.0$.

From the condition in (27), let $y(x_i) = y_i$ such that $y_0 = 0$ and $y_1 = 0$. To approximate y_i where $i = 1, 2, ..., n - 1$, i.e., $y_1, y_2, ..., y_9$, at each interior grid point, the differential equation will be approximated by the nonlinear difference equation introduced to (26) that yields

$$\frac{y_{i-1} - 2y_i + y_{i+1}}{h^2} - e^{y_i} = 0 \text{ for } i = 1, 2, ..., 9 \tag{28}$$

Simplifying (28) at $h = 0.1$ further gives

$$\frac{y_{i-1} - 2y_i + y_{i+1}}{(0.1)^2} - e^{y_i} = 0$$

$$y_{i-1} - 2y_i + y_{i+1} - 0.01e^y = 0 \tag{29}$$

For $i = 1, 2, ..., n - 1$ and the zero boundary conditions, yield the mildly nonlinear system as

$$\left.\begin{aligned}
y_0 - 2y_1 + y_2 - 0.01e^{y_1} &= 0\\
y_1 - 2y_2 + y_3 - 0.01e^{y_2} &= 0\\
y_2 - 2y_3 + y_4 - 0.01e^{y_3} &= 0\\
y_3 - 2y_4 + y_5 - 0.01e^{y_4} &= 0\\
y_4 - 2y_5 + y_6 - 0.01e^{y_5} &= 0\\
y_5 - 2y_6 + y_7 - 0.01e^{y_6} &= 0\\
y_6 - 2y_7 + y_8 - 0.01e^{y_7} &= 0\\
y_7 - 2y_8 + y_9 - 0.01e^{y_8} &= 0\\
y_8 - 2y_9 + y_{10} - 0.01e^{y_9} &= 0\\
y_9 - 2y_{10} + y_{11} - 0.01e^{y_{10}} &= 0
\end{aligned}\right\} \tag{30}$$

Putting the conditions (20) in (30) and simplifying, one obtains (30) as

$$\left.\begin{aligned} -2y_1 + y_2 &= 0.01e^{y_1} \\ y_1 - 2y_2 + y_3 &= 0.01e^{y_2} \\ y_2 - 2y_3 + y_4 &= 0.01e^{y_3} \\ y_3 - 2y_4 + y_5 &= 0.01e^{y_4} \\ y_4 - 2y_5 + y_6 &= 0.01e^{y_5} \\ y_5 - 2y_6 + y_7 &= 0.01e^{y_6} \\ y_6 - 2y_7 + y_8 &= 0.01e^{y_7} \\ y_7 - 2y_8 + y_9 &= 0.01e^{y_8} \\ y_8 - 2y_9 + y_{10} &= 0.01e^{y_9} \\ y_9 - 2y_{10} &= 0.01e^{y_{10}} - 1 \end{aligned}\right\} \tag{31}$$

Reducing (31) into an augmented matrix gives:

$$\left.\begin{pmatrix} -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} 0.01e^{y_1} \\ 0.01e^{y_2} \\ 0.01e^{y_3} \\ 0.01e^{y_4} \\ 0.01e^{y_5} \\ 0.01e^{y_6} \\ 0.01e^{y_7} \\ 0.01e^{y_8} \\ 0.01e^{y_9} \\ 0.01e^{y_{10}} - 1 \end{pmatrix}\right\} \tag{32}$$

At this point, the first leg of the intertwining, which is the difference approximation, ends, and the second leg of the intertwining, which is applying Newton-Lieberstein's Method to solve the argument matrix (32), begins, such that the coefficient matrix of (32) is tri-diagonally Nonlinear BVP will now be solved using the Newton-Lieberstein Method thus:

Step 1: Let $w = 1.5$, a value that is chosen from $[0, 2]$, initial guess values at $k = 0$, and

$y_1^0 = 0, y_2^0 = 0, y_3^0 = 0, y_4^0 = 0$, $y_5^0 = 0, y_6^0 = 0, y_7^0 = 0, y_8^0 = 0$, $y_9^0 = 0$, and $y_{10}^0 = 0$.

Step 2: Update the variable values using the Newton-Lieberstein algorithm, thus:

$$y_1^{(1)} = y_1^{(0)} - w\frac{-2y_1^{(0)} + y_2^{(0)} - f_1\left(y_1^{(0)}\right)}{a_{11} - f_1'\left(y_1^{(k)}\right)}$$

$$= 0 - 1.5\frac{-2(0) + 0 - 0.01}{-2 - 0.01}$$

$$= -0.00746268656716$$

$$y_2^{(1)} = y_2^{(0)} - w\frac{y_1^{(1)} - 2y_2^{(0)} + y_3^0 - f_2(y_2^{(0)})}{a_{22} - f_2'(x_2^{(k)})}$$

$$= 0 - 1.5\frac{-0.00746268656716 - 0 + 0 - 0.01}{-2 - 0.01}$$

$$= -0.01303185564713$$

$$y_3^{(1)} = y_3^{(0)} - w\frac{y_2^{(1)} - 2y_3^{(0)} + y_4^0 - f_3(y_3^{(0)})}{a_{33} - f_3'(x_3^{(k)})}$$

$$= 0 - 1.5\frac{-0.01303185564713 - 0 + 0 - 0.01}{-2 - 0.01}$$

$$= -0.01718795197547$$

$$y_4^{(1)} = y_4^{(0)} - w\frac{y_3^{(1)} - 2y_4^{(0)} + y_5^0 - f_4(y_4^{(0)})}{a_{44} - f_4'(x_4^{(k)})}$$

$$= 0 - 1.5\frac{-0.01718795197547 - 0 + 0 - 0.01}{-2 - 0.01}$$

$$= -0.02028951639960$$

$$y_5^{(1)} = y_5^{(0)} - w\frac{y_4^{(1)} - 2y_5^{(0)} + y_6^0 - f_5\left(y_5^{(0)}\right)}{a_{55} - f_5'\left(x_5^{(k)}\right)}$$

$$= 0 - 1.5\frac{-0.02028951639960 - 0 + 0 - 0.01}{-2 - 0.01}$$

$$= -0.02260411671612$$

$$y_6^{(1)} = y_6^{(0)} - w\frac{y_5^{(1)} - 2x_6^{(0)} + y_7^0 - f_6\left(y_6^{(0)}\right)}{a_{66} - f_6'\left(x_6^{(k)}\right)}$$

$$= 0 - 1.5\frac{-0.02260411671612 - 0 + 0 - 0.01}{-2 - 0.01}$$

$$= -0.02433143038516$$

$$y_7^{(1)} = y_7^{(0)} - w\frac{y_6^{(1)} - 2y_7^{(0)} + y_8^0 - f_7(y_7^{(0)})}{a_{77} - f_7'(x_7^{(k)})}$$

$$= 0 - 1.5\frac{-0.02433143038516 - 0 + 0 - 0.01}{-2 - 0.01}$$

$$= -0.02560242094974$$

$$y_8^{(1)} = y_8^{(0)} - w\frac{y_7^{(1)} - 2y_8^{(0)} + y_9^0 - f_8(y_8^{(0)})}{a_{22} - f_2'(x_2^{(k)})}$$

$$= 0 - 1.5\frac{-0.02560242094974 - 0 + 0 - 0.01}{-2 - 0.01}$$

$$= -02654997852770$$

$$y_9^{(1)} = y_9^{(0)} - w\frac{y_8^{(1)} - 2y_9^{(0)} + y_{10}^0 - f_9\left(y_9^{(0)}\right)}{a_{22} - f_2'\left(x_2^{(k)}\right)}$$

$$= 0 - 1.5\frac{-02654997852770 - 0 + 0 - 0.01}{-2 - 0.01}$$

$$= -0.02727610337888$$

$$y_{10}^{(1)} = y_{10}^{(0)} - w\frac{y_9^{(1)} - 2y_{10}^{(0)} - f_{10}\left(y_{10}^{(0)}\right)}{a_{22} - f_2'\left(x_2^{(k)}\right)}$$

$$= 0 - 1.5\frac{-0.02727610337888 - 0 + 0 - 0.01}{-2 - 0.01}$$

$$= -0.02781798759618$$

This process must be performed 10 times, and the values of the variables at the 10th iteration is the expected result. However, this are done using a program, and the summary of the results of the 10 iterations is presented in the below.

$$\text{At } k = 1, \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} -0.00746268656716 \\ -0.01303185564713 \\ -0.01718795197547 \\ -0.02028951639960 \\ -0.02260411671612 \\ -0.02433143038516 \\ -0.02560242094974 \\ -02654997852770 \\ -0.02727610337888 \\ -0.02781798759618 \end{pmatrix}$$

$$\text{At } k = 2, \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} -0.013457038 \\ -0.02381748 \\ -0.031786713 \\ -0.37911386 \\ -0.04261460 \\ -0.46223482 \\ -0.48990599 \\ -0.05111080 \\ -0.52734264 \\ -0.32902138 \end{pmatrix}.$$

$$\text{At } k = 3, \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} -0.018509415 \\ -0.033091557 \\ -0.04456232 \\ -0.053572597 \\ -0.06064049 \\ -0.066177527 \\ -0.070509925 \\ -0.07389579 \\ -0.060807761 \\ -0.03639510 \end{pmatrix}.$$

$$\text{At } k = 4, \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} -0.022904842 \\ -0.04127090 \\ -0.055970032 \\ -0.067713356 \\ -0.077079271 \\ -0.08453695 \\ -0.090465994 \\ -0.083428914 \\ -0.06649490 \\ -0.038893526 \end{pmatrix}.$$

$$\text{At } k = 5, \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} -0.026811829 \\ -0.04861249 \\ -0.066302275 \\ -0.080628098 \\ -0.092207743 \\ -0.10155069 \\ -0.10030804 \\ -0.090256022 \\ -0.070612951 \\ -0.040718074 \end{pmatrix}.$$

$$\text{At } k = 6, \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} -0.030337872 \\ -0.05528620 \\ -0.075759199 \\ -0.09252530 \\ -0.10622856 \\ -0.11086082 \\ -0.10743556 \\ -0.095238343 \\ -0.073635423 \\ -0.042061828 \end{pmatrix}.$$

$$\text{At } k = 7, \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} -0.033555992 \\ -0.061410921 \\ -0.08448492 \\ -0.10355936 \\ -0.11440822 \\ -0.11763509 \\ -0.11264988 \\ -0.098896849 \\ -0.075858547 \\ -0.043049341 \end{pmatrix}.$$

$$\text{At } k=8,\begin{pmatrix} y_1\\ y_2\\ y_3\\ y_4\\ y_5\\ y_6\\ y_7\\ y_8\\ y_9\\ y_{10}\end{pmatrix}=\begin{pmatrix} -0.036518377\\ -0.067073661\\ -0.092587466\\ -0.11019932\\ -0.12033447\\ -0.12256645\\ -0.11645651\\ -0.10156953\\ -0.077479473\\ -0.043765485\end{pmatrix}.$$

$$\text{At } k=9,\begin{pmatrix} y_1\\ y_2\\ y_3\\ y_4\\ y_5\\ y_6\\ y_7\\ y_8\\ y_9\\ y_{10}\end{pmatrix}=\begin{pmatrix} -0.039263874\\ -0.072340497\\ -0.097425847\\ -0.11491687\\ -0.12457613\\ -0.12611043\\ -0.11919505\\ -0.10348809\\ -0.078635943\\ -0.044270635\end{pmatrix}.$$

$$\text{At } k=10,\begin{pmatrix} y_1\\ y_2\\ y_3\\ y_4\\ y_5\\ y_6\\ y_7\\ y_8\\ y_9\\ y_{10}\end{pmatrix}=\begin{pmatrix} -0.041822363\\ -0.075229088\\ -0.100685522\\ -0.118159167\\ -0.127522654\\ -0.128583546\\ -0.121104998\\ -0.104818354\\ -0.079427894\\ -0.044609196\end{pmatrix}.$$

This is now the solution of the reduced tridiagonal system, and the tridiagonal system is the solution of the BVP differential equation. The method becomes very relevant, especially when the system of equations has many variables, as we have in the following problems. These problems cannot be solved analytically; rather, the use of a computer program becomes very relevant.

Problem 2: Consider the mildly nonlinear system

$$-50y_1 + 25y_2 = e^{y_1}$$

$$25y_1 - 50y_2 + 25y_3 = e^{y_2}$$

$$25y_2 - 50y_3 + 25y_4 = e^{y_3}$$

$$25y_3 - 50y_4 = e^{y_4}$$

With zero initial guess and $w = 1.3$, the method converged to a numerical solution after only eight iterations, to four decimal point precision as: $y_1 = -0.0731$ $y_2 = -0.1089$, $y_3 = -0.1089$, and $y_4 = -0.0731$.

Problem 3: Consider the mildly non-linear equation

$$-2y_1 + y_2 = 5e^{y_1}$$

$$y_1 - 2y_2 + y_3 = 4e^{y_2}$$

$$y_{i-1} - 2y_i + y_{i+1} = e^{y_i} \quad i = 3, 4, \dots, 97, 98$$

$$y_{98} - 2y_{99} + y_{100} = 8e^{y_{99}}$$

$$y_{99} - 2y_{100} = 13e^{y_{100}}$$

The system in problem 3 is a typical mildly nonlinear problem with 100 equations, which becomes readily solvable using the method proposed herein.

6. Conclusion

This study focused on solving mildly nonlinear boundary value problems (BVPs) that result in tridiagonal linear algebraic systems using the Newton-Lieberstein method. The results demonstrated that this method provides efficient and accurate solutions with faster convergence.

The key findings include that the Newton-Lieberstein method was able to handle mildly nonlinear systems effectively, the iterative approach improved accuracy with each step, and the method proved computationally efficient compared to traditional numerical techniques. The results obtained from the test problems confirmed that the method provides stable and reliable solutions, making it an effective choice for solving tridiagonal systems in engineering, physics, and applied mathematics. In practice, especially for systems of equations with numerous variables, it is advisable to use $\omega = 1.0, 1.3, 1.5,$ or 1.7 for iterations not less than 10 and then choose value for ω which is yields the most rapidly convergence for the remainder of the iterations.

The findings confirm that the Newton-Lieberstein method is an effective approach for solving tridiagonal linear systems. It enhances accuracy, stability, and efficiency, making it valuable for complex numerical problems. The study contributes to the field of numerical analysis by providing a method that balances computational efficiency with precision, ensuring that BVPs can be solved more effectively.

Abbreviations

BVP	Boundary Value Problem
MNBVP	Mildly Nonlinear Boundary Value Problem
ODE	Ordinary Differential Equation

Author Contributions

Adebayo Kayode James: Conceptualization, Methodology, Resources, Writing – review & editing

Akinmuyise Matthew Folorunsho: Data curation, Validation, Writing – original draft

Dele-Rotimi Adejoke Olumide: Formal Analysis, Investigation, Methodology

Data Availability Statement

The data supporting the outcome of this research work has been reported in this manuscript.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Omar, Z. and Suleiman, M. B., Solving higher order ordinary differential equations using parallel 2-point explicit block method, (2005), Matematika, Jabatan UTM, 21, 15-23.

[2] Dennis G. Zill and Michael R. Cullen, Differential Equations with Boundary-Value Problems, Seventh Edition, (2009), Library of Congress Control Number: 2008924835 ISBN-13: 978-0-495-10836-8. Printed in Canada.

[3] Nur Zahidah Mukhtar, Zanariah Abdul Majid, Fudziah Ismail, Numerical Solution for Solving Second Order Ordinary Differential Equations Using Block Method, (2011), International Conference Mathematical and Computational Biology, International Journal of Modern Physics: Conference Series, World Scientific Publishing Company, Vol. 9 (2012) 560–565, https://doi.org/10.1142/S2010194512005661

[4] Rosser, J. B. and Runge-Kutta, (1967), for all seasons, in SIAM Rev. 9, 417-452.

[5] Worland, P. B., Parallel methods for the numerical solutions of ordinary differential equations, (1976), IEEE Transactions on Computers, 25(10), 1045-1048, https://doi.org/10.1109/TC.1976.1674545

[6] Majid, Z. A., Suleiman, M. B., Ismail, F., and Othman, M., 2-point implicit block one-step method half Gauss-Seidel for solving first order ordinary differential equations, (2003), Matematika, Jabatan UTM, 19, 91-100, https://doi.org/10.54216/IJNS.250420

[7] Majid, Z. A. Suleiman, M. B., and Omar, Z., 3-point implicit block method for solving ordinary differential equations, (2006), Bull. Malays. Math. Sci. Soc, 29, 23-31.

[8] Chakravarti, P. C. and Worland, P. B., A class of self-starting methods for the numerical solution of $y'' = f(x,y)$, (1971), BIT, 11(4), 368-383, https://doi.org/10.1007/BF01939405

[9] Suleiman, M. B., Solving non-stiff higher order ODEs directly by the direct integration method, (1989), Applied Mathematics and Computation, 33, 197-219.

[10] Fatunla, S. O. (1991) Block Method for Second Order Differential Equations. International Journal of Computer Mathematics, 41, 55-63. http://dx.doi.org/10.1080/00207169108804026

[11] Trench, William F., Elementary Differential Equations with Boundary Value Problems, (2013). Textbooks Collection. 8. https://digitalcommons.usf.edu/oa_textbooks/8

[12] Ibrahim S. M., Numerical Solution of Boundary Value Problems, (1992), A Thesis Submitted To The Department Of Mathematics With Computer Science Federal University Of Technology, Minna, In Partial Fulfillment Of The Requirements For The A Ward Of Bachelor Of Technology Degree (B. Tech) In Mathematics With Computer Science.

[13] Paul Eloe, Two-Point Boundary Value Problems for Finite Difference Equations, Uniqueness of Solutions Implies Existence of Solutions, (2020), International Journal of Difference Equations ISSN 0973-6069, Volume 15, Number 2, pp. 377–387 (2020) http://campus.mst.edu/ijde

[14] Radzi, H. M., Majid, Z. A., Ismail, F., and Suleiman, M. B., Four-step implicit block method of Runge-Kutta type for solving first order ordinary differential equations. (2011), Proceedings of Fourth International Conference on Modeling, Simulation and Applied Optimization 2011, 139-143.

[15] Adebayo Kayode James, Alabi Taiye John, Adisa Isaac Olabisi, Ademoroti Albert Olalekan, and Dele-Rotimi Adejoke Olumide, Intertwining the Upwind Differencing and the Newton-Lieberstein's Methods to Solve Mildly Non-linear Boundary Value Problems (MNBVP) (2025), Quest Journals Journal of Research in Applied Mathematics Volume 11 ~ Issue 7 (July 2025) pp: 53-56, www.questjournals.org

Biography

Adebayo Kayode James is an Associate Professor at Ekiti State University, Mathematics Department, Faculty of Physical Sciences. He completed his Ph.D. in Mathematics from Ekiti State University in 2016 and his Master of Mathematics in Control Theory from the same institution in 2010. In addition, he is a member of several Mathematics bodies in and outside Nigeria. He has participated in multiple international research collaboration projects in recent years. He currently serves as a reviewer for quite a lot of journals.

Akinmuyise Matthew Folorunsho is a Senior Lecturer at Adeyemi Federal University of Education, Ondo, Ondo State, in the Department of Mathematics. He completed his Ph.D. in Mathematics from Ekiti State University in 2020, and his Master's of Mathematics in Control Theory from Ekiti State University in the year 2010. He is a member of several Mathematics bodies with research covering Optimization, Control Theory, and applied Mathematics.

Dele-Rotimi Adejoke Olumide is a Senior Lecturer in the Department of Mathematical Sciences at Bamidele Olumilua University of Education, Science and Technology (BOUESTI), Ikere-Ekiti, Nigeria. She completed his Ph.D. in Mathematics from Ekiti State University in 2024, and his Master of Mathematics in Control Theory from the same institution in 2016. Her research interests include Optimization, Artificial Intelligence, Operations Research, Computational Mathematics, and Applied Mathematics. She has served in several research groups in her institution. Presently, she is the Head of the Department of Mathematical Sciences in BOUESTI, Ikere Ekiti, Nigeria. She is a member of several Mathematics bodies in and outside Nigeria.

Research Field

Adebayo Kayode James: Optimization, Control Theory

Akinmuyise Matthew Folorunsho: Optimization; Artificial intelligence, Operations Research, Computational mathematics, and Applied mathematics

Dele-Rotimi Adejoke Olumide: Optimization; Artificial intelligence, Operations Research, Computational mathematics, and Applied mathematics