Numer. Math. Theor. Meth. Appl. Vol. 13, No. 4, pp. 946-972
doi: 10.4208/nmtma.OA-2019-0157 November 2020

One-Step Hybrid Block Method Containing Third Derivatives and Improving Strategies for Solving Bratu's and Troesch's Problems

Mufutau Ajani Rufai[1,*] and Higinio Ramos[2,3]

[1] *Dipartimento di Matematica, Università degli Studi di Bari Aldo Moro, 70125 Bari, Italy*
[2] *Scientific Computing Group, Universidad de Salamanca, Plaza de la Merced 37008 Salamanca, Spain*
[3] *Escuela Politécnica Superior de Zamora, Campus Viriato, 49022 Zamora, Spain*

Received 5 October 2019; Accepted (in revised version) 20 April 2020

Abstract. In this paper, we develop a one-step hybrid block method for solving boundary value problems, which is applied to the classical one-dimensional Bratu's and Troesch's problems. The convergence analysis of the new technique is discussed, and some improving strategies are considered to get better performance of the method. The proposed approach produces discrete approximations at the grid points, obtained after solving an algebraic system of equations. The solution of this system is obtained through a homotopy-type strategy used to provide the starting points needed by Newton's method. Some numerical experiments are presented to show the performance and effectiveness of the proposed approach in comparison with other methods that appeared in the literature.

AMS subject classifications: 65L10, 65L20

Key words: Ordinary differential equations, Bratu's problem, Troesch's problem, boundary value problems, hybrid block method, homotopy-type strategy.

1. Introduction

This paper aims at obtaining numerical solutions for second-order boundary value problems (BVPs) where the differential equation is of the special form

$$y''(x) = f(x, y(x)), \quad x \in [a, b], \tag{1.1}$$

*Corresponding author. *Email addresses:* mufutau.rufai@uniba.it (M.A. Rufai), higra@usal.es (H. Ramos)

subject to Dirichlet boundary conditions given by

$$y(a) = y_a, \quad y(b) = b_b, \tag{1.2}$$

although, instead of the above ones, mixed boundary conditions of the form

$$g_1(y(a), y'(a)) = v_a, \quad g_2(y(b), y'(b)) = v_b \tag{1.3}$$

might be considered.

One of those problems is the well-known Bratu's problem, which is given by

$$y''(x) + \lambda \mathrm{e}^{y(x)} = 0, \quad y(0) = y(1) = 0, \quad x \in [0,1]. \tag{1.4}$$

According to Jacobsen and Schmitt [13], and Boyd [2], the theoretical solution for this problem is given by

$$y(x) = -2\log\left[\frac{\cosh\left(\left(x-\frac{1}{2}\right)\frac{\theta}{2}\right)}{\cosh\left(\frac{\theta}{4}\right)}\right], \tag{1.5}$$

where θ is the solution of the algebraic equation

$$\theta = \sqrt{2\lambda}\cosh\left(\frac{\theta}{4}\right).$$

In addition, we will also consider the Troesch's problem, given by

$$y''(x) = \lambda\sinh(\lambda\, y(x)), \quad y(0) = y(1) = 0, \quad x \in [0,1]. \tag{1.6}$$

The close form of the solution for the problem in (1.6) is presented in Khuri [18] as follows

$$y(x) = \frac{2}{\lambda}\sinh^{-1}\left[\frac{y'(0)}{2}\,\mathrm{sc}\left(\lambda x|1-\frac{1}{4}\left(y'(0)\right)^2\right)\right], \tag{1.7}$$

where $y'(0) = 2(1-\mathrm{m})^{1/2}$ and the constant m satisfies the following transcendental equation

$$\frac{\sinh\left(\frac{\lambda}{2}\right)}{(1-\mathrm{m})^{1/2}} = \mathrm{sc}(\lambda|m), \tag{1.8}$$

where $\mathrm{sc}(\lambda|m)$ stands for one of the elliptic Jacobi functions (see [9]).

Problems (1.5) and (1.6) arise in engineering and science. For example, Bratu's problems of the form (1.4) emerge in the thermal reaction process in flammable non-deformable material, like the strong fuel ignition (see [13, 16]). It additionally shows up in the electro-spinning process for the generation of ultra-fine polymer fibers (see [37]), the Chandrasekhar model of the extension of the universe, chemical reactor theory, and nanotechnology (see [26] and the references therein for more details). Troesch's problem in (1.6) is a two-point nonlinear BVP that emerges in the control of a plasma segment by radiation weight and in the theory of gas porous electrodes. This problem was presented and defined for the first time by Troesch in [35].

It is imperative to note that most of the differential equations arising from the modelling of physical phenomena do not have known analytical solutions, or in the case of having them, it can be challenging to tackle. Thus, the need for the derivation of numerical approaches to get approximate solutions becomes an important task. There are mainly three different types of approximation techniques for solving boundary value problems of ODEs: the shooting-type methods, finite-difference methods, and the class of methods based on approximating the solution by a linear combination of trial functions (of which collocation methods, Galerkin method, variational iteration method, and the Rayleigh-Ritz method are the most typical examples). The shooting method transforms the boundary-value ODE into a system of first-order ODEs, which must be solved by some initial-value solver. The finite-difference approach constructs a finite difference approximation of the exact ODE at selected points on a discrete grid, including the boundary conditions. In this way a system of coupled finite difference equations results, which must be solved to get the approximate solutions at the grid points.

Prominent researchers like Buckmire [3] applied the finite difference method to solve the problem in (1.5), Khuri [19] implemented a variational iteration method to give approximate solutions to (1.5), Hassan and Erturk used differential transformation to solve the problem in (1.5). Various numerical methods have been applied by different scholars to solve Troesch's problem. Among the various techniques, we can mention the simple shooting method in Chang [5], the discontinuous Galerkin finite-element technique in [27], or the stochastic numerical treatment proposed by Temimi [33]. Recently, Jator and Manathunga [15] implemented a block Nyström-type integrator of order seven for solving (1.5), Kafri *et al.* [17] applied embedding Green's functions into fixed-point iteration technique for the solution of Troesch's problem. There are a lot of works for solving (1.5) or (1.6) separately. In the present manuscript, we develop a one-step hybrid block method for solving BVPs, which is applied to the classical one-dimensional Bratu's and Troesch's problems.

This manuscript is organized as follows. In Section 2, we develop a one-step hybrid block Falkner method for solving two-point boundary value problems. Some improvement strategies are given in Section 3 when the differential equation is of the type in (1.1), and implementation details of the proposed method are explained in Section 4. In Section 5, we present some numerical experiments to demonstrate the efficiency and reliability of the proposed strategy. Finally, some conclusions and future research work are given in Section 6.

2. Development of the one-step hybrid block method

In this section, we will develop the block method for solving problems of the general form

$$y''(x) = f(x, y(x), y'(x)), \quad x \in [a, b], \tag{2.1}$$

subject to the boundary conditions in (1.2), and later we will present some strategies in order to improve its performance for solving special problems of the type in (1.1).

We are interested in obtaining approximations of the true solution $y(x)$ of (2.1)-(1.2) at the grid points $a = x_0 < \cdots < x_N = b$ of the integration interval $[a, b]$, taking a constant step-size $h = x_{j+1} - x_j, j = 0, \dots, N-1$. To derive the block method, we consider that the true solution can be approximated on the interval $[x_n, x_{n+1}]$ by a polynomial $p(x)$, that is,

$$y(x) \simeq p(x) = \sum_{n=0}^{8} a_n x^n, \tag{2.2}$$

from which, we get

$$y'(x) \simeq p'(x) = \sum_{n=1}^{8} a_n n x^{n-1}, \tag{2.3}$$

$$y''(x) \simeq p''(x) = \sum_{n=2}^{8} a_n n(n-1) x^{n-2}, \tag{2.4}$$

$$y'''(x) \simeq p'''(x) = \sum_{n=3}^{8} a_n n(n-1)(n-2) x^{n-3}, \tag{2.5}$$

where $a_n \in \mathbb{R}$ are real unknown coefficients that will be determined imposing collocation conditions at selected points (the degree of the polynomial is taken on the basis of those conditions). Consider the intermediate points

$$x_{n+r} = x_n + \frac{1}{4}h, \quad x_{n+s} = x_n + \frac{1}{2}h, \quad x_{n+t} = x_n + \frac{3}{4}h$$

on $[x_n, x_{n+1}]$ and the approximation in (2.2) and its first derivative in (2.3) applied to the point x_n, its second derivative in (2.4) applied to the points $x_n, x_{n+r}, x_{n+s}, x_{n+t}$, x_{n+1}, and its third derivative in (2.5) applied to the points x_n, x_{n+1}. In this way, we get a system of nine equations with nine real unknowns $a_n, n = 0, \dots, 8$, given by

$$\begin{array}{lll} p(x_n) = y_n, & p'(x_n) = y'_n, & p''(x_n) = f_n, \\ p''(x_{n+r}) = f_{n+r}, & p''(x_{n+s}) = f_{n+s}, & p''(x_{n+t}) = f_{n+t}, \\ p''(x_{n+1}) = f_{n+1}, & p'''(x_n) = g_n, & p'''(x_{n+1}) = g_{n+1}, \end{array}$$

where the notations y_{n+j}, f_{n+j} and g_{n+j} stand respectively for approximations of $y(x_{n+j}), y''(x_{n+j})$ and

$$y'''(x_{n+j}) = f'(x_{n+j}, y(x_{n+j}), y'(x_{n+j})).$$

This system of nine equations can be written in matrix form as

$$Ax = y,$$

where

$$A = \begin{pmatrix} 1 & x_n & x_n^2 & x_n^3 & x_n^4 & x_n^5 & x_n^6 & x_n^7 & x_n^8 \\ 0 & 1 & 2x_n & 3x_n^2 & 4x_n^3 & 5x_n^4 & 6x_n^5 & 7x_n^6 & 8x_n^7 \\ 0 & 0 & 2 & 6x_n & 12x_n^2 & 20x_n^3 & 30x_n^4 & 42x_n^5 & 56x_n^6 \\ 0 & 0 & 2 & 6x_{n+r} & 12x_{n+r}^2 & 20x_{n+r}^3 & 30x_{n+r}^4 & 42x_{n+r}^5 & 56x_{n+r}^6 \\ 0 & 0 & 2 & 6x_{n+s} & 12x_{n+s}^2 & 20x_{n+s}^3 & 30x_{n+s}^4 & 42x_{n+s}^5 & 56x_{n+s}^6 \\ 0 & 0 & 2 & 6x_{n+t} & 12x_{n+t}^2 & 20x_{n+t}^3 & 30x_{n+t}^4 & 42x_{n+t}^5 & 56x_{n+t}^6 \\ 0 & 0 & 2 & 6x_{n+1} & 12x_{n+1}^2 & 20x_{n+1}^3 & 30x_{n+1}^4 & 42x_{n+1}^5 & 56x_{n+1}^6 \\ 0 & 0 & 0 & 6 & 24x_n & 60x_n^2 & 120x_n^3 & 210x_n^4 & 336x_n^5 \\ 0 & 0 & 0 & 6 & 24x_{n+1} & 60x_{n+1}^2 & 120x_{n+1}^3 & 210x_{n+1}^4 & 336x_{n+1}^5 \end{pmatrix},$$

$$x = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8)^\top,$$

$$y = (y_n, y_n', f_n, f_{n+r}, f_{n+s}, f_{n+t}, f_{n+1}, g_n, g_{n+1})^\top.$$

Solving the above system of equations, we readily obtain the values of the coefficients $a_n, n = 0, \ldots, 8$. After obtaining the values of these coefficients and changing the variable, $x = x_n + zh$, the polynomial in (2.2) may be written as

$$\begin{aligned} p(x_n + zh) = \alpha_0(z)y_n + h\alpha_1(z)y_n' + h^2(\beta_0(z)f_n + \beta_r(z)f_{n+r} + \beta_s(z)f_{n+s} \\ + \beta_t(z)f_{n+t} + \beta_1(z)f_{n+1}) + h^3(\gamma_0(z)g_n + \gamma_1(z)g_{n+1}), \end{aligned} \tag{2.6}$$

where

$$\begin{aligned} &\alpha_0(z) = 1, \quad \alpha_1(z) = t, \\ &\beta_0(z) = \frac{1}{3780}\left(1890z^2 - 17465z^4 + 46410z^5 - 54096z^6 + 30400z^7 - 6720z^8\right), \\ &\beta_r(z) = \frac{64}{945}\left(105z^4 - 336z^5 + 434z^6 - 260z^7 + 60z^8\right), \\ &\beta_s(z) = \frac{-4}{35}\left(35z^4 - 154z^5 + 238z^6 - 160z^7 + 40z^8\right), \\ &\beta_t(z) = \frac{64}{945}\left(35z^4 - 168z^5 + 294z^6 - 220z^7 + 60z^8\right), \\ &\beta_1(z) = \frac{1}{3780}\left(-3255z^4 + 16086z^5 - 29456z^6 + 23360z^7 - 6720z^8\right), \\ &\gamma_0(z) = \frac{1}{252}\left(42z^3 - 196z^4 + 399z^5 - 420z^6 + 224z^7 - 48z^8\right), \\ &\gamma_1(z) = \frac{1}{252}\left(21z^4 - 105z^5 + 196z^6 - 160z^7 + 48z^8\right). \end{aligned}$$

Now, taking $z = 1$ in the above formula we evaluate $p(x)$ at the point $x_{n+1} = x_n + h$, and thus we obtain the first of the formulas that approximates the solution $y(x_{n+1})$:

$$y_{n+1} = y_n + hy_n' + \frac{h^2}{3780}(768f_{n+r} + 432f_{n+s} + 256f_{n+t} + 419f_n + 15f_{n+1}) + \frac{h^3}{252}g_n. \tag{2.7}$$

Similarly, taking $z = 1$ in the derivative of the above formula we evaluate $p'(x)$ at the point $x_{n+1} = x_n + h$, thus obtaining an approximation for the first derivative of the solution $y(x)$ at the point x_{n+1}:

$$
\begin{aligned}
hy'_{n+1} = hy'_n &+ \frac{h^2}{3780}(1024f_{n+r} + 864f_{n+s} + 1024f_{n+t} + 434f_n + 434f_{n+1}) \\
&+ \frac{h^3}{3780}(15g_n - 15g_{n+1}).
\end{aligned} \tag{2.8}
$$

Until now, we have obtained the two main formulas, one for approximating the solution and another for approximating the first derivative at the final point of the interval $[x_n, x_{n+1}]$. Note that we have eight unknowns, that is, $(y_{n+j}, y'_{n+j}), j = r, s, t, 1$. Therefore, to obtain a one-step hybrid block method, we need to consider additional formulas. For this, we consider the evaluation of $p(x)$ and $p'(x)$ at the points $x_{n+r}, x_{n+s}, x_{n+t}$. In this way, we obtain a total of eight formulas that form the block hybrid method. The rest of the formulas are as follows

$$
\begin{aligned}
y_{n+r} &= y_n + \frac{h}{4}y'_n + \frac{h^2}{15482880}(181440f_{n+r} - 62424f_{n+s} + 33472f_{n+t} \\
&\quad + 343124f_n - 11772f_{n+1}) + \frac{h^3}{15482880}(11715g_n + 1125g_{n+1}), \\
y_{n+s} &= y_n + \frac{h}{2}y'_n + \frac{h^2}{120960}(8576f_{n+r} + 384f_{n+t} + 6312f_n - 152f_{n+1}) \\
&\quad + \frac{h^3}{120960}(225g_n + 15g_{n+1}), \\
y_{n+t} &= y_n + \frac{3h}{4}y'_n + \frac{h^2}{573440}(78912f_{n+r} + 30456f_{n+s} + 6720f_{n+t} \\
&\quad + 46668f_n - 1476f_{n+1}) + \frac{h^3}{573440}(1665g_n + 135g_{n+1}),
\end{aligned} \tag{2.9}
$$

$$
\begin{aligned}
hy'_{n+r} &= hy'_n + \frac{h^2}{967680}(139456f_{n+r} - 35424f_{n+s} + 18496f_{n+t} \\
&\quad + 125846f_n - 6454f_{n+1}) + \frac{h^3}{967680}(5055g_n + 615g_{n+1}), \\
hy'_{n+s} &= hy'_n + \frac{h^2}{60480}(17152f_{n+r} + 6912f_{n+s} - 768f_{n+t} + 6762f_n + 182f_{n+1}) \\
&\quad + \frac{h^3}{60480}(225g_n - 15g_{n+1}), \\
hy'_{n+t} &= hy'_n + \frac{h^2}{35840}(9024f_{n+r} + 9504f_{n+s} + 4544f_{n+t} + 4354f_n - 546f_{n+1}) \\
&\quad + \frac{h^3}{35840}(165g_n + 45g_{n+1}).
\end{aligned} \tag{2.10}
$$

For each of the above formulas, we can obtain the local truncation error in the usual form: passing all the terms to the left, substituting the approximate values for the true ones, and expanding the resulting formula by Taylor series in powers of h. In this way, we obtain the following local truncation errors

$$\begin{aligned}
\mathcal{L}[y(x_{n+1}),h] &= -\frac{h^9 y^{(9)}(x_n)}{203212800} + \mathcal{O}\big(h^{10}\big),\\
\mathcal{L}[y'(x_{n+1}),h] &= \frac{h^{10} y^{(10)}(x_n)}{1625702400} + \mathcal{O}\big(h^{11}\big),\\
\mathcal{L}[y(x_{n+r}),h] &= -\frac{3991 h^9 y^{(9)}(x_n)}{3329438515200} + \mathcal{O}\big(h^{10}\big),\\
\mathcal{L}[y(x_{n+s}),h] &= -\frac{h^9 y^{(9)}(x_n)}{406425600} + \mathcal{O}\big(h^{10}\big),\\
\mathcal{L}[y(x_{n+t}),h] &= -\frac{153 h^9 y^{(9)}(x_n)}{41104179200} + \mathcal{O}\big(h^{10}\big),\\
\mathcal{L}[y'(x_{n+r}),h] &= -\frac{3 h^9 y^{(9)}(x_n)}{293601280} + \mathcal{O}\big(h^{10}\big),\\
\mathcal{L}[y'(x_{n+s}),h] &= \frac{h^{10} y^{(10)}(x_n)}{3251404800} + \mathcal{O}\big(h^{11}\big),\\
\mathcal{L}[y'(x_{n+t}),h] &= -\frac{3 h^9 y^{(9)}(x_n)}{293601280} + \mathcal{O}\big(h^{10}\big),
\end{aligned} \tag{2.11}$$

which indicates that the proposed method has seventh order.

To be applied for solving a boundary-value problem, the above formulas are considered altogether along with the grid points on the integration interval at the same time, thus resulting in a global method that provides an approximate solution over all the grid points simultaneously. Considering the grid points $a = x_0 < \cdots < x_N = b$ with $N \in \mathbb{N}$, N a positive integer, we take the formulas in (2.7)-(2.10) for $n = 0, \ldots, N-1$, which results in a system of $8N$ equations. It is also clear that the number of unknowns is $8N$ (the approximate values of the solution and the first derivative at the grid and intermediate points).

2.1. Convergence analysis

This subsection is devoted to proving the convergence of the proposed one-step method. We start by defining convergence, and then we will show that the proposed method is convergent by compactly writing the main formulas in (2.7) and (2.8), and the additional ones in (2.9) and (2.10) in a matrix-vector form.

Definition 2.1. Let $y(x)$ be the solution of the considered boundary value problem and $\{y_j\}_{j=0}^N$ the approximations provided by the proposed method. The numerical method is said to be a p-th order convergent method if for h sufficiently small, there exists a constant K independent of h such that

$$\max_{0\le j\le N} |y(x_j) - y_j| \le K h^p.$$

Note that in this case, we have that $\max_{0\leq j\leq N}|y(x_j)-y_j|\to 0$ as $h\to 0$.

Let D represent the $8N\times 8N$ matrix defined by

$$D=\begin{bmatrix} D_{1,1} & D_{1,2} & \dots & D_{1,2N}\\ \vdots & \vdots & & \vdots\\ D_{2N,1} & D_{2N,2} & \dots & D_{2N,2N}\end{bmatrix},$$

where the elements $D_{i,j}$ are 4×4 submatrices, except the $D_{i,N}, i=1,\dots,2N$ which have size 4×3, and the $D_{i,2N}, i=1,\dots,2N$, which have size 4×5. Those submatrices are given as follows:

$$D_{N,N}=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\\0&0&0\end{bmatrix},\qquad D_{i,i-1}=\begin{bmatrix}0&0&0&-1\\0&0&0&-1\\0&0&0&-1\\0&0&0&-1\end{bmatrix},\quad i=2,\dots,N,$$

$$D_{2N-1,2N}=h\begin{bmatrix}0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\1&0&0&0&0\end{bmatrix},\qquad D_{i,i+1}=h\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\1&0&0&0\end{bmatrix},\quad i=N+1,\dots,2N-2,$$

$$D_{2N,2N}=h\begin{bmatrix}-1&1&0&0&0\\-1&0&1&0&0\\-1&0&0&1&0\\-1&0&0&0&1\end{bmatrix},\qquad D_{i,i}=h\begin{bmatrix}-1&1&0&0\\-1&0&1&0\\-1&0&0&1\\-1&0&0&0\end{bmatrix},\quad i=N+1,\dots,2N-1,$$

$$D_{N,2N}=h\begin{bmatrix}-\frac{1}{4}&0&0&0&0\\-\frac{2}{4}&0&0&0&0\\-\frac{3}{4}&0&0&0&0\\-1&0&0&0&0\end{bmatrix},\quad D_{i,N+i}=h\begin{bmatrix}-\frac{1}{4}&0&0&0\\-\frac{2}{4}&0&0&0\\-\frac{3}{4}&0&0&0\\-1&0&0&0\end{bmatrix},\quad i=1,\dots,N-1,$$

and $D_{i,i}=I, i=1,\dots,N-1$, where I is the identity matrix.

For the rest of submatrices not included above it is $D_{i,j}=\mathbb{O}$, that is, they are null matrices.

On the other hand, let U be a $8N\times(8N+2)$ matrix defined by

$$U=\begin{bmatrix} U_{1,1} & U_{1,2} & \dots & U_{1,2N}\\ \vdots & \vdots & & \vdots\\ U_{2N,1} & U_{2N,2} & \dots & U_{2N,2N}\end{bmatrix},$$

where the elements $U_{i,j}$ are 4×4 submatrices except the $U_{i,1}, U_{i,N+1}, i=1,\dots,2N$, which have size 4×5. Those submatrices are given as follows:

$$U_{1,1}=\begin{bmatrix}\frac{-343124}{15482880}&\frac{-181440}{15482880}&\frac{62424}{15482880}&\frac{-33472}{15482880}&\frac{11772}{15482880}\\ \frac{-6312}{120960}&\frac{-8576}{120960}&0&\frac{-384}{120960}&\frac{152}{120960}\\ \frac{-46668}{573440}&\frac{-78912}{573440}&\frac{-30456}{573440}&\frac{-6720}{573440}&\frac{1476}{573440}\\ \frac{-419}{3780}&\frac{-768}{3780}&\frac{-432}{3780}&\frac{-256}{3780}&\frac{-15}{3780}\end{bmatrix},$$

$$U_{i,i}=\begin{bmatrix}\frac{-181440}{15482880}&\frac{62424}{15482880}&\frac{-33472}{15482880}&\frac{11772}{15482880}\\ \frac{-8576}{120960}&0&\frac{-384}{120960}&\frac{152}{120960}\\ \frac{-78912}{573440}&\frac{-30456}{573440}&\frac{-6720}{573440}&\frac{1476}{573440}\\ \frac{-768}{3780}&\frac{-432}{3780}&\frac{-256}{3780}&\frac{-15}{3780}\end{bmatrix},\qquad i=2,\dots,N,$$

$$U_{N+1,1}=\begin{bmatrix}\frac{-125846}{967680}&\frac{-139456}{967680}&\frac{35424}{967680}&\frac{-18496}{967680}&\frac{6454}{967680}\\ \frac{-6762}{60480}&\frac{-17152}{60480}&\frac{-6912}{60480}&\frac{768}{60480}&\frac{-182}{60480}\\ \frac{-4354}{35840}&\frac{-9024}{35840}&\frac{-9504}{35840}&\frac{-4544}{35840}&\frac{546}{35840}\\ \frac{-434}{3780}&\frac{-1024}{3780}&\frac{-864}{3780}&\frac{-1024}{3780}&\frac{-434}{3780}\end{bmatrix},$$

$$U_{N+j,j}=\begin{bmatrix}\frac{-139456}{967680}&\frac{35424}{967680}&\frac{-18496}{967680}&\frac{6454}{967680}\\ \frac{-17152}{60480}&\frac{-6912}{60480}&\frac{768}{60480}&\frac{-182}{60480}\\ \frac{-9024}{35840}&\frac{-9504}{35840}&\frac{-4544}{35840}&\frac{546}{35840}\\ \frac{-1024}{3780}&\frac{-864}{3780}&\frac{-1024}{3780}&\frac{-434}{3780}\end{bmatrix},\qquad j=2,\dots,N,$$

$$U_{2,1}=\begin{bmatrix}0\,0\,0\,0&\frac{-343124}{15482880}\\ 0\,0\,0\,0&\frac{-6312}{120960}\\ 0\,0\,0\,0&\frac{-46668}{573440}\\ 0\,0\,0\,0&\frac{-419}{3780}\end{bmatrix},\qquad U_{i,i-1}=\begin{bmatrix}0\,0\,0&\frac{-343124}{15482880}\\ 0\,0\,0&\frac{-6312}{120960}\\ 0\,0\,0&\frac{-46668}{573440}\\ 0\,0\,0&\frac{-419}{3780}\end{bmatrix},\quad i=3,\dots,N,$$

$$U_{N+2,1}=\begin{bmatrix}0\,0\,0\,0&\frac{-125846}{967680}\\ 0\,0\,0\,0&\frac{-6762}{60480}\\ 0\,0\,0\,0&\frac{-4354}{35840}\\ 0\,0\,0\,0&\frac{-434}{3780}\end{bmatrix},\qquad U_{N+j,j-1}=\begin{bmatrix}0\,0\,0&\frac{-125846}{967680}\\ 0\,0\,0&\frac{-6762}{60480}\\ 0\,0\,0&\frac{-4354}{35840}\\ 0\,0\,0&\frac{-434}{3780}\end{bmatrix},\quad j=3,\dots,N,$$

$$U_{1,N+1}=h\begin{bmatrix}\frac{-11715}{15482880}&0\,0\,0&\frac{-1125}{15482880}\\ \frac{-225}{120960}&0\,0\,0&\frac{-15}{120960}\\ \frac{-1665}{573440}&0\,0\,0&\frac{-135}{573440}\\ \frac{-1}{252}&0\,0\,0&0\end{bmatrix},\quad U_{i,N+i}=h\begin{bmatrix}0\,0\,0&\frac{-1125}{15482880}\\ 0\,0\,0&\frac{-15}{120960}\\ 0\,0\,0&\frac{-135}{573440}\\ 0\,0\,0&0\end{bmatrix},\quad i=2\dots,N,$$

$$U_{2,N+1}=h\begin{bmatrix}0\,0\,0\,0&\frac{-11715}{15482880}\\ 0\,0\,0\,0&\frac{-225}{120960}\\ 0\,0\,0\,0&\frac{-1665}{573440}\\ 0\,0\,0\,0&\frac{-1}{252}\end{bmatrix},\qquad U_{i,N+i-1}=h\begin{bmatrix}0\,0\,0&\frac{-11715}{15482880}\\ 0\,0\,0&\frac{-225}{120960}\\ 0\,0\,0&\frac{-1665}{573440}\\ 0\,0\,0&\frac{-1}{252}\end{bmatrix},\quad i=3,\dots,N,$$

$$U_{N+1,N+1}=h\begin{bmatrix}\frac{-5055}{967680}&0\,0\,0&\frac{-615}{967680}\\ \frac{-225}{60480}&0\,0\,0&\frac{15}{60480}\\ \frac{-165}{35840}&0\,0\,0&\frac{-45}{35840}\\ \frac{-15}{3780}&0\,0\,0&\frac{15}{3780}\end{bmatrix},\qquad U_{N+i,N+i}=h\begin{bmatrix}0\,0\,0&\frac{-615}{967680}\\ 0\,0\,0&\frac{15}{60480}\\ 0\,0\,0&\frac{-45}{35840}\\ 0\,0\,0&\frac{15}{3780}\end{bmatrix},\quad i=2,\dots,N,$$

$$U_{N+2,N+1} = h \begin{bmatrix} 0\,0\,0\,0\,\frac{-5055}{967680} \\ 0\,0\,0\,0\,\frac{-225}{60480} \\ 0\,0\,0\,0\,\frac{-165}{35840} \\ 0\,0\,0\,0\,\frac{-15}{3780} \end{bmatrix}, \qquad U_{N+i,N+i-1} = h \begin{bmatrix} 0\,0\,0\,\frac{-5055}{967680} \\ 0\,0\,0\,\frac{-225}{60480} \\ 0\,0\,0\,\frac{-165}{35840} \\ 0\,0\,0\,\frac{-15}{3780} \end{bmatrix}, \quad i = 3, \ldots, N.$$

For the rest of submatrices $U_{i,j}$ not included above it is $U_{i,j} = \mathbb{O}$, that is, they are null matrices.

Note that all those submatrices $D_{i,j}$ and $U_{i,j}$ contain the coefficients of the formulas in (2.9)-(2.7) for $n = 0, 1, \ldots, N-1$, followed by the formulas in (2.10)-(2.8) for $n = 0, 1, \ldots, N-1$, in this order. We also define the following vectors of exact values

$$\begin{aligned} Y &= \big(y(x_{1/4}), y(x_{1/2}), y(x_{3/4}), \ldots, y(x_{N-1+3/4}), y'(x_0), y'(x_{1/4}), \ldots, y'(x_N)\big)^\top, \\ F &= \Big(f(x_0, y(x_0), y'(x_0)), f(x_{1/4}, y(x_{1/4}), y'(x_{1/4})), \ldots, f(x_N, y(x_N), y'(x_N)), \\ &\qquad g(x_0, y(x_0), y'(x_0)), g(x_{1/4}, y(x_{1/4}), y'(x_{1/4})), \ldots, g(x_N, y(x_N), y'(x_N))\Big)^\top. \end{aligned}$$

Note that Y has $(4N-1)+(4N+1) = 8N$ components, while F has $(4N+1)+(4N+1) = 8N+2$ components, because due to the boundary conditions in (1.2), $y(x_0)$ and $y(x_N)$ are known values, $y(x_0) = y_a, y(x_N) = y_b$.

By using the above notations, the exact form of the system that provides the approximate values of the problem at hand is given by

$$D_{8N\times 8N} Y_{8N} + h^2 U_{8N\times(8N+2)} F_{8N+2} + C_{8N} = L(h)_{8N}, \tag{2.12}$$

where we have included the dimensions for clarity. C_{8N} is a vector containing the known values, which is given by

$$C_{8N} = (-y_a, -y_a, -y_a, -y_a, 0, \ldots, 0, y_b, 0, \ldots, 0)^\top,$$

and $L(h)_{8N}$ corresponds to the local truncation errors of the formulas, that is,

$$L(h)_{8N} = (u_1, u_2, u_3, u_4, u_5, \ldots, u_{4N}, v_1, v_2, v_3, v_4, v_5, \ldots, v_{4N})^\top,$$

where

$$\begin{aligned} u_1 &= -\frac{3991 h^9 y^{(9)}(x_0)}{3329438515200} + \mathcal{O}(h^{10}), & u_2 &= -\frac{h^9 y^{(9)}(x_0)}{406425600} + \mathcal{O}(h^{10}), \\ u_3 &= -\frac{153 h^9 y^{(9)}(x_0)}{41104179200} + \mathcal{O}(h^{10}), & u_4 &= -\frac{h^9 y^{(9)}(x_0)}{203212800} + \mathcal{O}(h^{10}), \\ u_5 &= -\frac{3991 h^9 y^{(9)}(x_1)}{3329438515200} + \mathcal{O}(h^{10}), & u_{4N} &= -\frac{h^9 y^{(9)}(x_{N-1})}{203212800} + \mathcal{O}(h^{10}), \\ v_1 &= -\frac{3 h^9 y^{(9)}(x_0)}{293601280} + \mathcal{O}(h^{10}), & v_2 &= \frac{h^{10} y^{(10)}(x_0)}{3251404800} + \mathcal{O}(h^{11}), \\ v_3 &= -\frac{3 h^9 y^{(9)}(x_0)}{293601280} + \mathcal{O}(h^{10}), & v_4 &= \frac{h^{10} y^{(10)}(x_0)}{1625702400} + \mathcal{O}(h^{11}), \end{aligned}$$

$$v_5 = -\frac{3h^9 y^{(9)}(x_1)}{293601280} + \mathcal{O}(h^{10}), \qquad v_{4N} = \frac{h^{10} y^{(10)}(x_{N-1})}{1625702400} + \mathcal{O}(h^{11}).$$

On the other hand, the system to get the approximate values of the problem is represented by

$$D_{8N\times 8N}\bar{Y}_{8N} + h^2 U_{8N\times(8N+2)}\bar{F}_{8N+2} + C_{8N} = 0, \tag{2.13}$$

where $\bar{Y}_{8N}$ approximates the vector Y_{8N}, that is,

$$\begin{aligned}
&\bar{Y}_{8N} = \left(y_{1/4}, y_{1/2}, y_{3/4}, y_1, \ldots, y_{N-1+3/4}, y'_0, y'_{1/4}, \ldots, y'_N\right)^\top, \\
&\bar{F}_{8N+2} = \left(f_0, f_{1/4}, f_{1/2}, f_{3/4}, f_1, \ldots, f_N, g_0, g_{1/4}, g_{1/2}, g_{3/4}, g_1, \ldots, g_N\right)^\top.
\end{aligned}$$

On subtracting (2.13) from (2.12) and simplifying we get

$$D_{8N\times 8N}E_{8N} + h^2 U_{8N\times(8N+2)}\left(F - \bar{F}\right)_{8N+2} = L(h)_{8N}, \tag{2.14}$$

where

$$E_{8N} = Y_{8N} - \bar{Y}_{8N} = \left(e_{1/4}, e_{1/2}, \ldots, e_{N-1+3/4}, e'_0, e'_{1/4}, \ldots, e'_N\right)^\top$$

contains the errors of the solution and its first derivative at the grid points.

By using the Mean-Value Theorem (see [10, 22]), we can write for $i = 0, \frac{1}{4}, \ldots, N$

$$\begin{aligned}
f\left(x_i, y(x_i), y'(x_i)\right) - f\left(x_i, y_i, y'_i\right) &= \left(y(x_i) - y_i\right)\frac{\partial f}{\partial y}(\xi_i) + \left(y'(x_i) - y'_i\right)\frac{\partial f}{\partial y'}(\xi_i), \\
g\left(x_i, y(x_i), y'(x_i)\right) - g\left(x_i, y_i, y'_i\right) &= \left(y(x_i) - y_i\right)\frac{\partial g}{\partial y}(\eta_i) + \left(y'(x_i) - y'_i\right)\frac{\partial g}{\partial y'}(\eta_i),
\end{aligned}$$

where ξ_i and η_i are intermediate points on the line segment joining $(x_i, y(x_i), y'(x_i))$ to (x_i, y_i, y'_i). Thus, we have that

$$F - \bar{F} = \begin{pmatrix}
\frac{\partial f}{\partial y}(\xi_0) & 0 & \ldots & 0 & \frac{\partial f}{\partial y'}(\xi_0) & 0 & \ldots & 0 \\
0 & \frac{\partial f}{\partial y}(\xi_{1/4}) & \ldots & 0 & 0 & \frac{\partial f}{\partial y'}(\xi_{1/4}) & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & \frac{\partial f}{\partial y}(\xi_N) & 0 & 0 & \ldots & \frac{\partial f}{\partial y'}(\xi_N) \\
\frac{\partial g}{\partial y}(\eta_0) & 0 & \ldots & 0 & \frac{\partial g}{\partial y'}(\eta_0) & 0 & \ldots & 0 \\
0 & \frac{\partial g}{\partial y}(\eta_{1/4}) & \ldots & 0 & 0 & \frac{\partial g}{\partial y'}(\eta_{1/4}) & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & \frac{\partial g}{\partial y}(\eta_N) & 0 & 0 & \ldots & \frac{\partial g}{\partial y'}(\eta_N)
\end{pmatrix}
\begin{pmatrix} e_0 \\ e_{1/4} \\ \vdots \\ e_N \\ e'_0 \\ e'_{1/4} \\ \vdots \\ e'_N \end{pmatrix}$$

$$
= \begin{pmatrix}
0 & \dots & 0 & \frac{\partial f}{\partial y'}(\xi_0) & 0 & \dots & 0 & 0 \\
\frac{\partial f}{\partial y}(\xi_{1/4}) & \dots & 0 & 0 & \frac{\partial f}{\partial y'}(\xi_{1/4}) & \dots & 0 & 0 \\
\vdots & \ddots & 0 & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & \dots & \frac{\partial f}{\partial y}(\xi_{N-1+3/4}) & 0 & 0 & \dots & \frac{\partial f}{\partial y'}(\xi_{N-1+3/4}) & 0 \\
0 & \dots & 0 & 0 & 0 & \dots & 0 & \frac{\partial f}{\partial y'}(\xi_N) \\
0 & \dots & 0 & \frac{\partial g}{\partial y'}(\eta_0) & 0 & \dots & 0 & 0 \\
\frac{\partial g}{\partial y}(\eta_{1/4}) & \dots & 0 & 0 & \frac{\partial g}{\partial y'}(\eta_{1/4}) & \dots & 0 & 0 \\
\vdots & \ddots & 0 & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & \dots & \frac{\partial g}{\partial y}(\eta_{N-1+3/4}) & 0 & 0 & \dots & \frac{\partial g}{\partial y'}(\eta_{N-1+3/4}) & 0 \\
0 & \dots & 0 & 0 & 0 & \dots & 0 & \frac{\partial g}{\partial y'}(\eta_N)
\end{pmatrix} E_{8N}
$$

$$
= J_{(8N+2)\times 8N} E_{8N},
$$

where the second identity has been achieved through the fact that we know the exact boundary conditions, that is, $e_0 = y(x_0) - y_0 = 0$ and $e_N = y(x_N) - y_N = 0$.

Finally, using the above result, the equation in (2.14) may be rewritten as follows

$$
\left(D_{8N\times 8N} + h^2 U_{8N\times(8N+2)} J_{(8N+2)\times 8N}\right) E_{8N} = L(h)_{8N}, \tag{2.15}
$$

and setting $M = D + h^2 UJ$ we simply get that

$$
M_{8N\times 8N} E_{8N} = L(h)_{8N}. \tag{2.16}
$$

Let us see that except for a few selected values of $h > 0$, matrix M is invertible. If we use the abbreviate notation $D_N = D_{8N\times 8N}$, given the form of this matrix where the submatrices have many zeros, it is easy to verify that for $N = 1$ the determinant is $|D_1| = -h^5$. Now, by induction, it can be proved that $|D_N| = -Nh^{N+4}$, and thus D_N is invertible as long as it is $h > 0$.

Now the matrix M may be rewritten as

$$
M = D + h^2 UJ = (Id - C)D,
$$

where Id is the identity matrix of order $8N$, and $C = -h^2 UJD^{-1}$. Thus, we have that $|M| = |Id - C||D|$.

As $|\lambda Id - C| = \prod_{i=1}^{8N}(\lambda - \lambda_i)$ is the characteristic polynomial of C in order to have $|Id - C| \neq 0$, if we take $\lambda = 1$, it is sufficient to choose h such that

$$
h^2 \notin \left\{1/\bar{\lambda}_i : \bar{\lambda}_i \text{ is an eigenvalue of } UJD^{-1}\right\}.
$$

For such values of h the equation in (2.16) may be rewritten as

$$
E = \left(M^{-1}\right) L(h). \tag{2.17}
$$

We consider the maximum norm in $\mathbb{R}$, $\|E\| = \max_i |e_i|$, and the corresponding matrix induced norm in $\mathbb{R}^{8N\times 8N}$. After expanding each term of M^{-1} in series around h it can be shown that $\|M^{-1}\| = \mathcal{O}(h^{-1})$. Note that the norm of matrix J must be bounded, which in fact is related with the boundedness of $\frac{\partial f}{\partial y}, \frac{\partial f}{\partial y'}, \frac{\partial g}{\partial y}, \frac{\partial g}{\partial y'}$.

Consequently, from the equation in (2.17) and the form of the vector $L(h)$ in (2.11), assuming that $y(x)$ has in $[a, b]$ bounded derivatives up to the ninth order, we have that

$$\|E\| \leq \|(M^{-1})\|\|L(h)\| = \mathcal{O}(h^{-1})\mathcal{O}(h^8) \leq Kh^7.$$

Therefore, the proposed method is a seventh-order convergent method.

3. Some improving strategies

This section explains the adopted strategies to improve the performance of the developed method for solving problems of the form in (1.1).

Note that the developed method obtained from the formulas in (2.7)-(2.10) may be applied for solving problems where the differential equation is of the general form $y'' = f(x, y(x), y'(x))$. But for the kind of problems considered here the first derivative does not appear in the function f, that is, the differential equation is of the special second order, as in (1.1). In this situation, we can put aside the formulas that contain the first derivatives at the intermediate points, and consider only the formulas in (2.7)-(2.9). With this strategy the resulting algebraic system to be solved has $5N + 2$ equations (obtained from the formulas in (2.7)-(2.9) for $n = 0, \ldots, N - 1$ together with the two boundary conditions) and the same number of unknowns (the approximate values of the solution at the grid and intermediate points, and the approximate values of the first derivative at the grid points). This procedure results in a saving in computational time.

A complementary strategy in order to gain efficiency consists in a reformulation of the formulas in (2.7)-(2.9). To do that we consider those formulas in (2.7)-(2.9) as an algebraic system and solve this system in terms of the variables $f_{n+r}, f_{n+s}, f_{n+t}, f_{n+1}, g_{n+1}$. After some simplifications we get

$$\begin{aligned}
&h^2 f_{n+r} - \left(-\frac{f_n h^2}{3} - \frac{5 g_n h^3}{384} - \frac{1151 h y'_n}{576} - \frac{13 h y'_{n+1}}{192} + \frac{51 y_{n+s}}{2}\right.\\
&\qquad\qquad \left.-\frac{304 y_{n+r}}{9} - \frac{112 y_{n+t}}{27} + \frac{9983 y_n}{864} + \frac{251 y_{n+1}}{288}\right) = 0,\\
&h^2 f_{n+s} - \left(\frac{7 f_n h^2}{18} + \frac{g_n h^3}{72} + \frac{463 h y'_n}{108} + \frac{h y'_{n+1}}{4} - 40 y_{n+s}\right.\\
&\qquad\qquad \left.+\frac{2048 y_{n+t}}{81} + \frac{2951 y_n}{162} - \frac{7 y_{n+1}}{2}\right) = 0,\\
&h^2 f_{n+t} - \left(-\frac{11 f_n h^2}{16} - \frac{3 g_n h^3}{128} - \frac{531 h y'_n}{64} - \frac{99 h y'_{n+1}}{64} + \frac{27 y_{n+s}}{2} + 48 y_{n+r}\right.\\
&\qquad\qquad \left.-48 y_{n+t} - \frac{1299 y_n}{32} + \frac{867 y_{n+1}}{32}\right) = 0,\\
&h^2 f_{n+1} - \left(-\frac{31 f_n h^2}{3} - \frac{g_n h^3}{3} - \frac{1184 h y'_n}{9} + \frac{68 h y'_{n+1}}{3} - 384 y_{n+s}\right.
\end{aligned} \tag{3.1}$$

$$+\frac{8192y_{n+r}}{9}+\frac{8192y_{n+t}}{27}-\frac{18236y_n}{27}-\frac{1388y_{n+1}}{9}\Big)=0,$$

$$h^3g_{n+1}-\Big(-\frac{952f_nh^2}{3}-\frac{31g_nh^3}{3}-\frac{36092hy'_n}{9}+308hy'_{n+1}-10752y_{n+s}$$
$$+\frac{81920y_{n+r}}{3}+\frac{180224y_{n+t}}{27}-\frac{552824y_n}{27}-\frac{8264y_{n+1}}{3}\Big)=0.$$

The resulting formulas are from a theoretical point of view equivalent to those in (2.7)-(2.9), the theoretical basis of this fact can be seen in the article by Ramos and Popescu [30]. Note that the numerical results obtained with the different formulations of the method may not be the same, due to the various calculations involved.

To get the approximate solution in the interval $[x_0, x_N]$ we have to solve the system obtained from the formulas in (3.1) for $n = 0, \dots, N-1$, together with the two boundary conditions. To solve this system with $5N+2$ equations, a Newton-type method is usually appropriate. By using this approach, the nonlinearity of the functions f and g is reflected fewer times in the system. On the contrary, taking the formulas in (2.7)-(2.9) the number of occurrences of the functions f and g is higher, which might complicate the system, and thus its resolution.

Although we will describe the implementation procedure in the following section, just to check the performance of the different approaches, we have considered the problem in (1.4) for $\lambda = 1$ taking $h = \frac{1}{10}$.

In Table 1 we show the performance of the three approaches considered, where we have included the number of intervals, N, the number of equations of each algebraic system, $Eqns$, the computational times, CPU, and the maximum absolute errors on the grid points, MAE. Note that for this problem the number of equations is not $8N+2$ or $5N+2$, as the two equations corresponding to the boundary values are in fact constants, and accordingly, y_0 and y_N are not considered unknowns. We see that the best performance corresponds to the method that uses the simplified equations in (3.1). From now on, we would use this method in the numerical examples, which will be denoted as $1HBM$.

Table 1: Comparison of the different approaches for solving Eq. (1.4) for $\lambda = 1, h = 0.1$.

	Formulas (2.7)-(2.10)	Formulas (2.7)-(2.9)	Formulas (3.1)
N	10	10	10
$Eqns$	80	50	50
CPU	0.171	0.125	0.094
MAE	3.46×10^{-15}	3.46×10^{-15}	1.66×10^{-15}

4. Implementation

To obtain the numerical approximations to the considered problem, we have to solve the system of $5N+2$ equations given by (3.1) for $n = 0, \dots, N-1$, together with the two boundary conditions in (1.2) or (1.3). If we denote the system in (3.1) by $F_n = 0$, considering the boundary conditions of the general form in (1.3), the algebraic system may be formulated as

$$\begin{cases} g_1(y_0, y_0') - v_a = 0, \\ F_0 = 0, \\ F_1 = 0, \\ \quad \cdots \\ F_{N-1} = 0, \\ g_2(y_N, y_N') - v_b = 0, \end{cases} \tag{4.1}$$

which will be named in shortly as $\mathbf{F} = 0$. The $5N + 2$ unknowns will be denoted by

$$\begin{aligned} \mathbf{Y} =& \big(y_0, y_0', y_r, y_s, y_t, y_1, y_1', y_{1+r}, y_{1+s}, y_{1+t}, y_2, y_2', \ldots, \\ & y_{N-1+r}, y_{N-1+s}, y_{N-1+t}, y_N, y_N'\big). \end{aligned}$$

In general, we will use Newton's method for solving the system $\mathbf{F} = 0$, although any other of the many schemes available in the literature could be used.

The Newton iteration is given by

$$\mathbf{Y}^{i+1} = \mathbf{Y}^i - \left(\mathbf{J}^i\right)^{-1} \mathbf{F}^i,$$

where $\mathbf{J}$ denotes the jacobian matrix of $\mathbf{F}$, which is given by

$$\mathbf{J} = \begin{pmatrix} \frac{\partial g_1}{\partial y_0} & \frac{\partial g_1}{\partial y_0'} & & & \cdots & & & \\ & & J_0 & & \cdots & & & \\ & & & J_1 & \cdots & & & \\ & & & & \cdots & & & \\ & & & & \cdots & J_{N-1} & & \\ & & & & \cdots & & \frac{\partial g_2}{\partial y_N} & \frac{\partial g_2}{\partial y_N'} \end{pmatrix}$$

with the $J_n, n = 0, \ldots, N-1$ submatrices of dimensions 5×7 associated to the set of equations in (3.1), given by

$$J_n = \begin{pmatrix} j_{11} & j_{12} & j_{13} & j_{14} & j_{15} & j_{16} & j_{17} \\ j_{21} & j_{22} & j_{23} & j_{24} & j_{25} & j_{26} & j_{27} \\ j_{31} & j_{32} & j_{33} & j_{34} & j_{35} & j_{36} & j_{37} \\ j_{41} & j_{42} & j_{43} & j_{44} & j_{45} & j_{46} & j_{47} \\ j_{51} & j_{52} & j_{53} & j_{54} & j_{55} & j_{56} & j_{57} \end{pmatrix},$$

where

$$j_{11} = \frac{h^2}{3} f_2^n + \frac{5h^3}{384} g_2^n - \frac{9983}{864}, \quad j_{12} = \frac{h^2}{3} f_2^n + \frac{5h^3}{384} g_2^n + \frac{1151h}{576},$$

$$j_{13}=h^2f_2^{n+r}+\frac{304}{9},\quad j_{14}=-\frac{51}{2},\quad j_{15}=\frac{112}{27},\quad j_{16}=-\frac{251}{288},\quad j_{17}=\frac{13h}{192},$$
$$j_{21}=-\frac{7h^2}{18}f_2^n-\frac{h^3}{72}g_2^n-\frac{2951}{162},\quad j_{22}=-\frac{7h^2}{18}f_2^n-\frac{h^3}{72}g_2^n-\frac{463h}{108},$$
$$j_{23}=0,\quad j_{24}=h^2f_2^{n+s}+40,\quad j_{25}=-\frac{2048}{81},\quad j_{26}=\frac{7}{2},\quad j_{27}=-\frac{h}{4},$$
$$j_{31}=\frac{11h^2}{16}f_2^n+\frac{3h^3}{128}g_2^n+\frac{1299}{32},\quad j_{32}=\frac{11h^2}{16}f_2^n+\frac{3h^3}{128}g_2^n+\frac{531h}{64},$$
$$j_{33}=-48,\quad j_{34}=-\frac{27}{2},\quad j_{35}=h^2f_2^{n+t}+48,\quad j_{36}=-\frac{867}{32},\quad j_{37}=\frac{99h}{64},$$
$$j_{41}=\frac{31h^2}{3}f_2^n+\frac{h^3}{3}g_2^n+\frac{18236}{27},\quad j_{42}=\frac{31h^2}{3}f_2^n+\frac{h^3}{3}g_2^n+\frac{1184h}{9},$$
$$j_{43}=-\frac{8192}{9},\quad j_{44}=384,\quad j_{45}=-\frac{8192}{27},$$
$$j_{46}=h^2f_2^{n+1}+\frac{1388}{9},\quad j_{47}=h^2f_3^{n+1}-\frac{68h}{3},$$
$$j_{51}=\frac{952h^2}{3}f_2^n+\frac{31h^3}{3}g_2^n+\frac{552824}{27},\quad j_{52}=\frac{952h^2}{3}f_2^n+\frac{31h^3}{3}g_2^n+\frac{36092h}{9},$$
$$j_{53}=-\frac{81920}{3},\quad j_{54}=10752,\quad j_{55}=-\frac{180224}{27},$$
$$j_{56}=h^3g_2^{n+1}+\frac{8264}{3},\quad j_{57}=h^3g_3^{n+1}-308h,$$

and the notations f_j^{n+i}, g_j^{n+i} refer respectively to the partial derivatives of $f(x,y,y')$ or $g(x,y,y')$ with respect to y for $j=2$ and with respect to y' for $j=3$, evaluated at the corresponding point $x_{n+i}, i=0,r,s,t,1$.

For the use of the iterative Newton's method, initial guesses reasonably close to the true roots are needed. In case of Dirichlet conditions it is $y_0=y_a, y_N=y_b$, and thus the system is reduced to $5N$ equations with $5N$ unknowns. In this case, it might be considered a linear interpolation to get as initial starting points the following

$$\begin{aligned}
y_j^{(0)} &= y_0+\frac{y_N-y_0}{b-a}jh, && j=1,\dots,N-1,\\
y_{j+r}^{(0)} &= y_0+\frac{y_N-y_0}{b-a}(j+r)h, && j=0,\dots,N-1,\\
y_{j+s}^{(0)} &= y_0+\frac{y_N-y_0}{b-a}(j+s)h, && j=0,\dots,N-1,\\
y_{j+t}^{(0)} &= y_0+\frac{y_N-y_0}{b-a}(j+t)h, && j=0,\dots,N-1,\\
{y'}_j^{(0)} &= \frac{y_N-y_0}{b-a}, && j=0,\dots,N.
\end{aligned}$$

Nevertheless, the above strategy may behave very poorly. For the general case in (1.3) we propose a strategy similar to that in [8, 24, 31, 32] where a homotopy-type procedure is used. We consider a family of nonlinear BVPs $P_j, j=0,\dots,m$, such that for $j=0$ the problem P_0 admits only the solution $y(x)=0$, while for $j=m$ we recover the original problem. In this way, we get a family of BVPs given by

$$P_j \equiv \begin{cases} y'' = f(x,y,y') - f(x,0,0) + \dfrac{j}{m} f(x,0,0), \\ g_1(y(a),y'(a)) = \dfrac{j}{m} v_a, \\ g_2(y(b),y'(b)) = \dfrac{j}{m} v_b, \end{cases} \tag{4.2}$$

for $j=0,\dots,m$. Each of these problems P_j for $j=1,\dots,m$ is solved using the method presented in the previous section, taking as starting guesses the values obtained after solving the previous problem P_{j-1}. For $j=m$ the nonlinear algebraic system corresponding to the original BVP is solved taking as starting guesses the values obtained after solving the problem P_{m-1}.

This strategy has the sole purpose of providing suitable starting values for Newton's method. Sometimes it is enough to take $m=1$ (which means that the starting values are taken as zero, although any other constant values according to the data of the problem could be considered).

5. Numerical experiments

In this section, we present the approximate solutions for the Bratu's and Troesch's problems using the proposed modified one-step hybrid block method for various values of λ. Besides, we will include comparisons with different numerical approaches available in the literature, to show the efficiency of the proposed method.

5.1. Problem 1

As a first problem, we will consider the equation in (1.4) whose theoretical solution is given in (1.5).

This problem is solved using the new one-step hybrid block method $1HBM$ with step-size $h=\frac{1}{10}$. The reason for considering $h=\frac{1}{10}$ is to compare the absolute errors (AEs) provided by our method with those obtained by Caglar *et al.* [4] using a B-spline approach. In order to compare the accuracy of our present method to a non-polynomial spline method proposed by Jalilian [14] we use step-sizes $h=\frac{1}{4},\frac{1}{8},\frac{1}{16},\frac{1}{32}$. We note that for the step-sizes, $h=\frac{1}{4},\frac{1}{8},\frac{1}{16},\frac{1}{32}$, the $1HBM$ method has less maximum absolute errors ($MAEs$) compared with the step-sizes $h=\frac{1}{8},\frac{1}{16},\frac{1}{32},\frac{1}{64}$ used in [14]. The numerical outcomes presented in Tables 2-5 taking $m=1$ show that $1HBM$ is more efficient in terms of accuracy than the approaches in [4] and [14].

Comparison of the theoretical versus approximate solutions for various values of λ with $m=1, h=0.1$ and $m=8$, are shown in Figs. 1 and 2 respectively. Concerning the boundedness of matrix J to guaranty the convergence of the proposed scheme, we have that

$$f(x,y,y') = -\lambda e^{y}, \quad g(x,y,y') = -\lambda e^{y} y',$$

Table 2: Comparison of the absolute errors (AEs) on Problem 1 at different grid points for $\lambda = 1, h = 0.1$.

x	AEs in $(1HBM)$	AEs in [14]	AEs in [4]
0.1	8.25728×10^{-16}	5.7700×10^{-10}	2.9800×10^{-6}
0.2	1.7486×10^{-15}	2.4700×10^{-10}	5.4600×10^{-6}
0.3	2.62290×10^{-15}	4.5600×10^{-11}	7.3300×10^{-6}
0.4	3.24740×10^{-15}	9.6400×10^{-11}	8.5000×10^{-6}
0.5	3.46945×10^{-15}	1.4600×10^{-10}	8.8900×10^{-6}
0.6	3.24740×10^{-15}	9.6400×10^{-11}	8.5000×10^{-6}
0.7	2.6229×10^{-15}	4.5600×10^{-11}	7.3300×10^{-6}
0.8	1.74860×10^{-15}	2.4700×10^{-10}	5.4600×10^{-6}
0.9	8.18789×10^{-16}	5.7700×10^{-10}	2.9800×10^{-6}

Table 3: Comparison of the absolute errors (AEs) on Problem 1 at different grid points for $\lambda = 2, h = 0.1$.

x	AEs in $(1HBM)$	AEs in [14]	AEs in [4]
0.1	1.84297×10^{-14}	9.71000×10^{-9}	1.72000×10^{-5}
0.2	5.82312×10^{-14}	1.41000×10^{-10}	3.26000×10^{-6}
0.3	1.12577×10^{-13}	1.98000×10^{-8}	4.49000×10^{-6}
0.4	1.61260×10^{-13}	2.42000×10^{-8}	5.28000×10^{-5}
0.5	1.80911×10^{-13}	2.60000×10^{-8}	5.56000×10^{-5}
0.6	1.61260×10^{-13}	9.6400×10^{-11}	5.28000×10^{-5}
0.7	1.12521×10^{-13}	1.98000×10^{-8}	4.49000×10^{-5}
0.8	5.82312×10^{-14}	1.41000×10^{-8}	3.26000×10^{-5}
0.9	1.84019×10^{-14}	9.71000×10^{-9}	1.72000×10^{-5}

Table 4: Comparison of the absolute errors (AEs) on Problem 1 at different grid points for $\lambda = 3.51, h = 0.1$.

x	AEs in $(1HBM)$	AEs in [14]	AEs in [4]
0.1	2.67258×10^{-12}	6.61000×10^{-6}	3.84000×10^{-2}
0.2	1.38389×10^{-12}	5.83000×10^{-6}	7.48000×10^{-2}
0.3	6.78757×10^{-12}	6.19000×10^{-6}	1.06000×10^{-1}
0.4	3.02771×10^{-11}	6.89000×10^{-6}	1.27000×10^{-1}
0.5	4.74032×10^{-11}	7.31000×10^{-10}	1.35000×10^{-1}
0.6	3.02771×10^{-11}	6.89000×10^{-6}	1.27000×10^{-6}
0.7	6.78746×10^{-12}	6.19000×10^{-6}	1.06×10^{-1}
0.8	1.38389×10^{-12}	1.37000×10^{-6}	7.48000×10^{-2}
0.9	2.67264×10^{-12}	6.61000×10^{-6}	3.84000×10^{-2}

Table 5: Comparison of the maximum absolute errors (MAEs) on Problem 1.

N	Methods	MAEs, $\lambda = 3.51$	MAEs, $\lambda = 2$	MAEs, $\lambda = 1$
4	$(1HBM)$	4.57168×10^{-7}	2.97576×10^{-10}	5.81088×10^{-12}
8	non-polynomial spline [14]	3.51000×10^{-5}	4.53000×10^{-8}	5.64000×10^{-9}
8	$(1HBM)$	2.70407×10^{-10}	1.08646×10^{-12}	2.16771×10^{-14}
16	non-polynomial spline [14]	1.45000×10^{-7}	1.76000×10^{-9}	4.66000×10^{-11}
16	$(1HBM)$	1.15996×10^{-12}	4.16334×10^{-15}	1.11022×10^{-16}
32	non-polynomial spline [14]	1.02000×10^{-9}	2.13000×10^{-11}	8.33000×10^{-13}
32	$(1HBM)$	6.43929×10^{-15}	5.55112×10^{-17}	2.77556×10^{-17}
64	non-polynomial spline [14]	1.48000×10^{-11}	2.87000×10^{-13}	9.21000×10^{-15}

and thus,

$$\frac{\partial f}{\partial y} = -\lambda e^y, \quad \frac{\partial f}{\partial y'} = 0, \quad \frac{\partial g}{\partial y} = -\lambda e^y y', \quad \frac{\partial g}{\partial y'} = -\lambda e^y. \tag{5.1}$$

On the other hand, taking in mind the boundary values, we have

$$y'(x) = \int_0^x -\lambda e^y dx, \quad x \in [0,1],$$

and thus, as the solution is bounded, it will be so the derivative. The partial derivatives in (5.1) are bounded, and so is matrix J.

Fig. 3 shows the efficiency curves for $N = 10, 20, 30$ of the maximum absolute errors in logarithmic scale versus CPU times for the proposed $1HBM$ and the Block Nyström Method (BNM) in [15]. We observed that the proposed $1HBM$ presents the best performance.

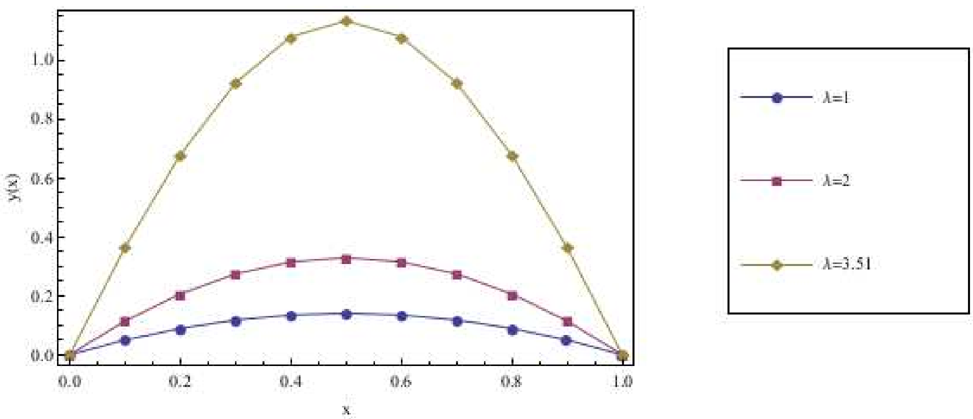

Figure 1: Exact and discrete solutions with the method $1HBM$ for $\lambda = 1, 2, 3.51$, on Problem 1 with $N = 10$.

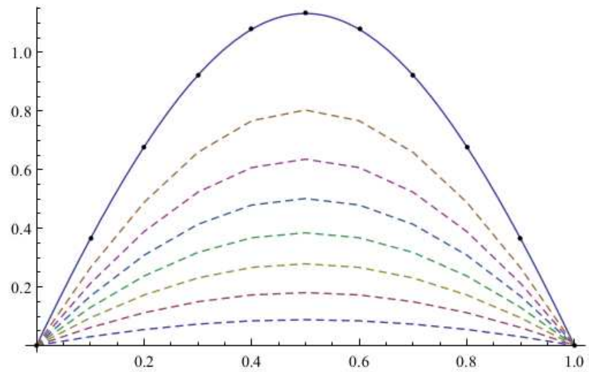

Figure 2: Exact and discrete solutions of Problem 1 (for $N = 10, \lambda = 3.51$) using the homotopy-type approach with $m = 8$.

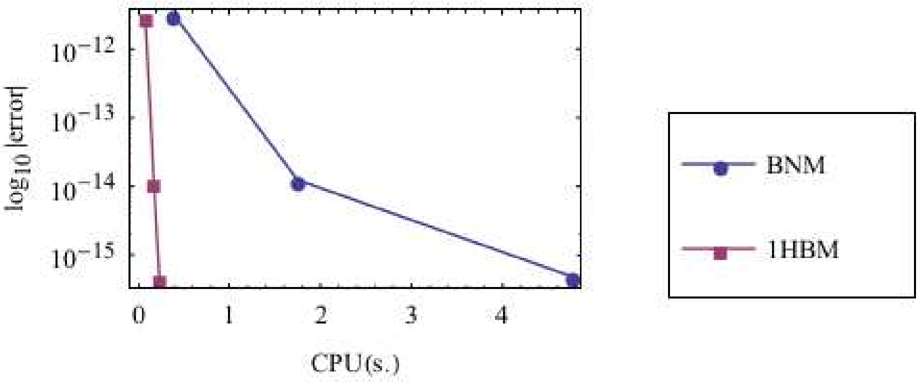

Figure 3: Efficiency plot showing the maximum absolute errors (MAEs) versus CPU times for Problem 1.

5.2. Problem 2

In the second problem, we apply our newly $1HBM$ to give numerical solution to non-linear Bratu's BVP given by

$$y''(x) + \lambda \mathrm{e}^{y(x)} = 0,$$

subject to the following initial conditions

$$y(0) = y(1) = 0, \quad 0 \leq x \leq 1,$$

whose exact solution for $\lambda = -\pi^2$ is

$$y(x) = -\log(1 - \cos(\pi(x + 0.5))).$$

Problem 2 has been solved earlier by Raja *et al.* [28] using the active-set method (ASM), the genetic algorithm (GA), and a hybrid approach denoted by GA-ASM with step-size

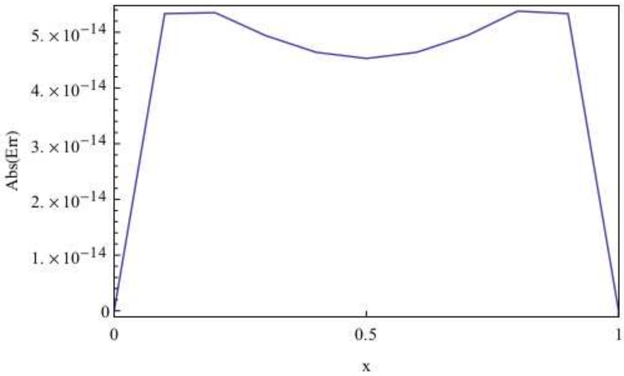

Figure 4: Absolute errors of the $1HBM$ for $\lambda = -\pi^2$ on Problem 2 with $N = 10$.

Table 6: Comparison of the absolute errors (AEs) on Problem 2 at different grid points for $\lambda = -\pi^2$.

x	$(1HBM)$	GA-ASM in [28]	ASM in [28]	GA in [28]
0.1	5.33462×10^{-14}	8.22940×10^{-03}	1.49360×10^{-06}	7.68000×10^{-07}
0.2	5.35127×10^{-14}	5.95130×10^{-03}	8.63660×10^{-07}	2.52870×10^{-07}
0.3	4.94049×10^{-14}	4.15280×10^{-03}	9.89070×10^{-07}	3.09370×10^{-08}
0.4	4.64073×10^{-14}	3.0633×10^{-03}	5.19550×10^{-07}	6.82250×10^{-07}
0.5	4.52971×10^{-14}	2.39900×10^{-03}	5.40640×10^{-07}	6.44070×10^{-07}
0.6	4.64073×10^{-14}	1.75990×10^{-03}	5.49350×10^{-07}	1.56430×10^{-07}
0.7	4.94049×10^{-14}	9.45720×10^{-04}	2.45520×10^{-07}	7.04670×10^{-07}
0.8	5.37903×10^{-14}	6.32720×10^{-05}	3.34720×10^{-07}	9.73410×10^{-07}
0.9	5.33462×10^{-14}	7.61860×10^{-04}	2.61370×10^{-07}	1.29300×10^{-06}

$h = \frac{1}{10}$. We have solved the same problem using the newly developed technique $1HBM$ with same step-size. The results presented in Table 6 show that the proposed approach is more efficient and accurate than those given in [28]. Fig. 4 displays the plot of the absolute errors on problem 2 for $m = 1, h = \frac{1}{10}$.

5.3. Problem 3

To test the practical relevance and applicability of the proposed $1HBM$ method and to exhibit its convergence computationally, we consider the Troesch's problem. This problem has been solved numerically earlier by many researchers using different numerical methods for different values of the parameter λ. Among them are Khuri [18], Khuri and Sayfy [20], Mirmoradia *et al.* [25], and Zarebnia and Sajjadian [38]. This problem is given in Eq. (1.6) and its solution in closed form is presented in Eqs. (1.7) and (1.8).

Fig. 5 shows the plot of the absolute errors for $\lambda = 2, m = 1, h = 0.1$. Concerning the boundedness of matrix J to guaranty the convergence of the proposed scheme, we

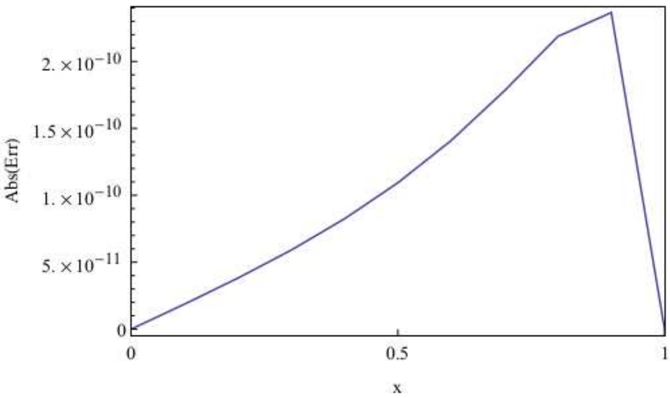

Figure 5: Absolute errors of the $1HBM$ method for Troesch's problem with $\lambda = 2, m = 1, N = 10$.

have that

$$f(x, y, y') = \lambda \sinh(\lambda y), \quad g(x, y, y') = \lambda^2 \cosh(\lambda y) y',$$

and thus,

$$\frac{\partial f}{\partial y} = \lambda^2 \cosh(\lambda y), \quad \frac{\partial f}{\partial y'} = 0, \quad \frac{\partial g}{\partial y} = \lambda^3 \sinh(\lambda y) y', \quad \frac{\partial g}{\partial y'} = \lambda^2 \cosh(\lambda y). \tag{5.2}$$

On the other hand, taking in mind the boundary values, we have

$$y'(x) = \int_0^x \lambda \sinh(\lambda y) dx, \quad x \in [0, 1],$$

and thus, as the solution is bounded, it will be so the derivative and the partial derivatives in (5.2).

Tables 7 and 8 report the comparisons between the theoretical solution and the numerical approximations provided by the proposed $1HBM$ taking $h = 0.1$.

Table 7: Numerical results of Troesch's problem for the case $\lambda = 1$ and $m = 1, h = 0.1$.

x	$(1HBM)$	SGM in [38]	HPM in [25]	LM in [20]
0.10000	0.084661256551	0.084661250	0.084934415	0.0846631
0.20000	0.170171358178	0.170171338	0.170697546	0.1701750
0.30000	0.257393908080	0.257393933	0.258133224	0.2573995
0.40000	0.347222855110	0.347222839	0.348116627	0.3472304
0.50000	0.440599835168	0.440599836	0.441572740	0.4406094
0.60000	0.538534398076	0.538534416	0.539498234	0.5385460
0.70000	0.642128609191	0.642128589	0.642987984	0.6421421
0.80000	0.752608094046	0.752608114	0.753267551	0.7526227
0.90000	0.871362519798	0.871362527	0.871733059	0.8713749

Table 8: Numerical results of Troesch's problem for the case $\lambda = 2$ and $m = 1, h = 0.1$.

x	$(1HBM)$	Exact	Absolute Errors
0.10000	0.05220874953937186	0.05220874955804888	1.86770×10^{-11}
0.20000	0.10651864918058021	0.10651864921867965	3.80994×10^{-11}
0.30000	0.16514082653330660	0.16514082659236876	5.90621×10^{-11}
0.40000	0.23052172534051973	0.23052172542298302	8.24633×10^{-11}
0.50000	0.30550472585623295	0.30550472596559070	1.09358×10^{-10}
0.60000	0.39356327187974344	0.39356327202069340	1.40950×10^{-10}
0.70000	0.49917297576130920	0.49917297593948345	1.78174×10^{-10}
0.80000	0.62846511544672280	0.62846511566546390	2.18741×10^{-10}
0.90000	0.79049400208269970	0.79049400231928270	2.36583×10^{-10}

Our technique yields higher accuracy than the Laplace-based Method (LM) proposed in [20], the Homotopy Perturbation Method (HPM) presented in [25], and the sinc-Galerkin method (SGM) reported in [38].

In Tables 9 and 10 we show and contrast the AEs of the $1HBM$ and Laplace Method (LM) proposed in [20], the Homotopy Perturbation Method (HPM) in [25], and the sinc-Galerkin strategy (SGM) in [38]. We have used the $1HBM$ to give numerical results to Troesch's problem for $\lambda = 0.5$ and $\lambda = 1$ with $N = 10$ on the mesh points. From the AEs displayed in Tables 9 and 10, we noticed that $1HBM$ methodology gives better numerical results and more proficiency in terms of accuracy than those methods used for comparisons.

Figs. 6 and 7 display the comparison of the analytical versus numerical solutions for different values of λ with $m = 1, h = 0.1$ and $m = 5, h = 0.05$ respectively.

Fig. 8 includes the efficiency curves for $N = 10, 20, 30$ of the maximum absolute errors in logarithmic scale versus CPU times for the proposed $1HBM$ and the BNM method in [15]. Again the proposed method performs better.

Table 9: Comparison of the absolute errors (AEs) for $\lambda = 0.5$ on Troesch's problem.

x	AEs with $(1HBM)$	AEs with SGM [38]	AEs with HPM [25]	AEs with LM [20]
0.10000	6.93433×10^{-12}	7.67445×10^{-4}	7.71124×10^{-4}	7.70000×10^{-4}
0.20000	1.22436×10^{-11}	1.49487×10^{-3}	1.50193×10^{-3}	1.50000×10^{-3}
0.30000	1.78599×10^{-11}	2.14100×10^{-3}	2.15084×10^{-3}	2.10000×10^{-3}
0.40000	2.34778×10^{-11}	2.66191×10^{-3}	2.67371×10^{-3}	2.70000×10^{-3}
0.50000	2.91577×10^{-11}	3.00978×10^{-3}	3.02252×10^{-3}	3.00000×10^{-3}
0.60000	3.75988×10^{-12}	3.13128×10^{-3}	3.14381×10^{-3}	3.10000×10^{-3}
0.70000	1.21607×10^{-11}	2.96601×10^{-3}	2.97716×10^{-3}	3.00000×10^{-3}
0.80000	7.85405×10^{-12}	2.44474×10^{-3}	2.45332×10^{-3}	2.40000×10^{-3}
0.90000	8.62710×10^{-12}	1.48723×10^{-3}	1.49210×10^{-3}	1.50000×10^{-3}

Table 10: Comparison of the absolute errors (AEs) for $\lambda = 1$ on Troesch's problem.

x	AEs with $1HBM$	AEs with SGM [38]	AEs with HPM [25]	AEs with LM [20]
0.10000	5.51466×10^{-9}	2.86425×10^{-3}	3.13742×10^{-3}	2.90000×10^{-3}
0.20000	1.07024×10^{-8}	5.64047×10^{-3}	6.61667×10^{-3}	5.90000×10^{-3}
0.30000	1.60636×10^{-8}	8.22657×10^{-3}	8.96586×10^{-3}	8.20000×10^{-3}
0.40000	2.15824×10^{-8}	1.01049×10^{-2}	1.13844×10^{-2}	1.00000×10^{-2}
0.50000	2.73293×10^{-8}	1.22527×10^{-2}	1.32256×10^{-2}	1.20000×10^{-2}
0.60000	2.42145×10^{-8}	1.32604×10^{-2}	1.42242×10^{-2}	1.30000×10^{-2}
0.70000	2.80028×10^{-8}	1.31574×10^{-2}	1.40168×10^{-2}	1.30000×10^{-2}
0.80000	2.56935×10^{-8}	1.14397×10^{-2}	1.20992×10^{-2}	1.10000×10^{-2}
0.90000	1.98350×10^{-8}	7.39251×10^{-3}	7.76304×10^{-3}	7.40000×10^{-5}

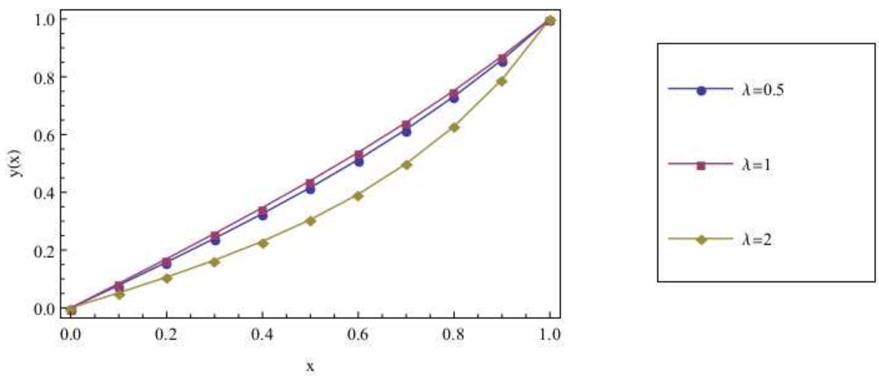

Figure 6: Exact and discrete solutions with the method $1HBM$ for $\lambda = 0.5, 1, 2$, on Problem 3 with $N = 10$.

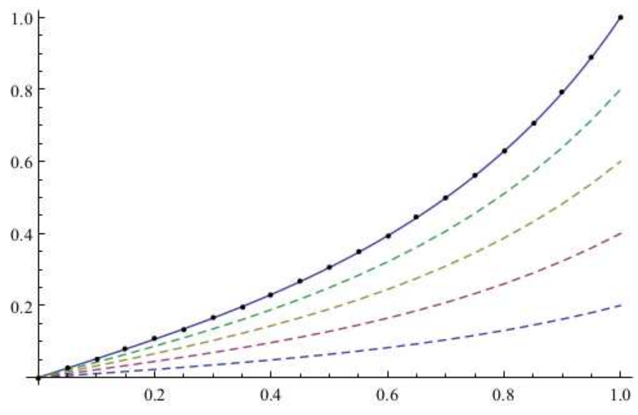

Figure 7: Exact and discrete solutions of Problem 3 (for $N = 20,\ \lambda = 2$) using the homotopy-type approach with $m = 5$.

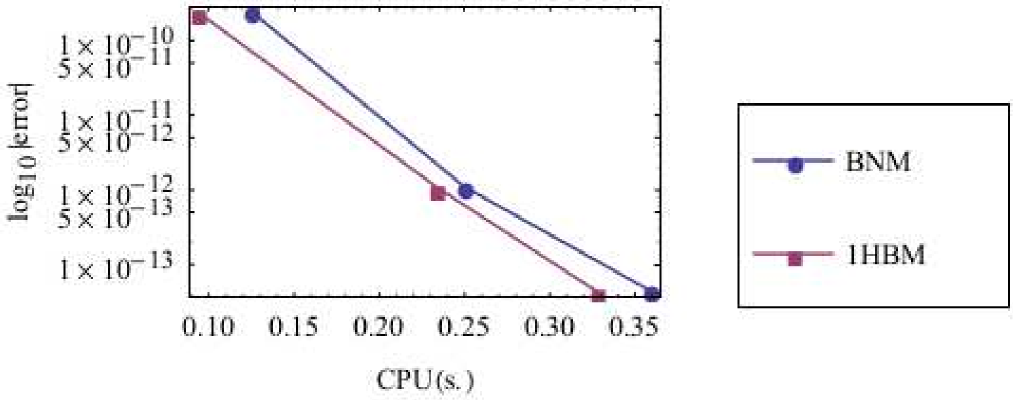

Figure 8: Efficiency plot showing the maximum absolute errors (MAEs) versus CPU times for Problem 3.

6. Conclusions

A new one-step hybrid block method considering some improving strategies ($1HBM$) has been developed and effectively used for numerically solving the Bratu's and Troesch's problems. Some test problems have been solved, illustrating the performance of the proposed method and demonstrating its reliability. The presented numerical outcomes in Tables 2-10 and Figs. 1-8 show that our approach is much superior to other existing numerical strategies, in terms of accuracy and efficiency. For future research work, we note that the two dimensional Bratu's and Troesch's problems for large values of the sensitivity parameter λ might be solved considering the type of strategies presented in this work.

References

[1] I. H. Abdel-Halim Hassan, V. S. Erturk, *Applying differential transformation method to the one-dimensional planar Bratu problem*, Int. J. Contemp. Math. Sci. 2 (2007) 1493-1504.

[2] J. P. Boyd, *Chebyshev polynomial expansions for simultaneous approximation of two branches of a function with application to the one-dimensional Bratu equation*, Appl. Math. Comput. 142 (2003) 189-200.

[3] R. Buckmire, *Application of a Mickens finite-difference scheme to the cylindrical Bratu-Gelfand problem*, Numer. Methods Partial Differential Equations 20 (2004) 327-337.

[4] H. Caglar, N. Caglar, M. Ozer, A. Valaristos, A. N. Anagnostopoulos, *B-spline method for solving Bratu's problem*, Int. J. Comput. Math. 87 (8) (2010) 1885-1891.

[5] Shih-Hsiang Chang, *Numerical solution of Troesch's problem by simple shooting method*, Appl. Math. Comput. 216(11) (2010) 3303-3306.

[6] S. H. Chen, J. Hu, L. Chen, C. P. Wang, *Existence results for n-point boundary value problem of second order ordinary differential equations*, J. Comput. Appl. Math. 180 (2005) 425-432.

[7] X. Y. CHENG, C. K. ZHONG, *Existence of positive solutions for a second order ordinary differential system*, J. Math. Anal. Appl. 312 (2005) 14-23.

[8] S. CUOMO, A. MARASCO, *A numerical approach to nonlinear two-point boundary value problems for ODEs*, Comput. Math. Appl. 55 (2008) 2476-2489.

[9] A. C. DIXON, *The Elementary Properties of the Elliptic Functions, with Examples*, Macmillan, 1894.

[10] H. DYM, *Linear Algebra in Action*, AMS, Providence, Rhode Island, 2007.

[11] D. GREENSPAN, V. CASULLI, *Numerical Analysis for Applied Mathematics, Science and Engineering*, Addison-Wewley, 1988.

[12] ROBERT A. VAN GORDER, *The variational iteration method is a special case of the homotopy analysis method*, Applied Mathematics Letters 45 (2015) 81-85.

[13] J. JACOBSEN AND K. SCHMITT, *The Liouville-Bratu-Gelfand problem for radial operators*, Journal of Differential Equations 184 (2002) 283-298.

[14] R. JALILIAN, *Non-polynomial spline method for solving Bratu's problem*, Comput. Phys. Commun. 181 (2010) 1868-1872.

[15] S. N. JATOR, V. MANATHUNGA, *Block Nyström type integrator for Bratu's equation*, Journal of Computational and Applied Mathematics 327 (2017) 341-349.

[16] H. Q. KAFRI, S. A. KHURI, *Bratus problem: A novel approach using fixed-point iterations and Greens functions*, Computer Physics Communications 198 (2016) 97-104.

[17] H. Q. KAFRI, S. A. KHURI, A. SAYFY, *A new approach based on embedding Green's functions into fixed-point iterations for highly accurate solution to Troesch's problem*, International Journal for Computational Methods in Engineering Science and Mechanics 17 (2) (2016) 93-105.

[18] S. A. KHURI, *A numerical algorithm for solving Troesch's problem*, International Journal of Computer Mathematics 80 (2003) 493-498.

[19] S. A. KHURI, *A new approach to Bratu's problem*, Appl. Math. Comput. 147 (2004) 131-136.

[20] S. A. KHURI, A. SAYFY, *Troesch's problem: A B-spline collocation approach*, Journal of Mathematical and Computer Modelling, 54 (9-10) (2011) 1907-1918.

[21] S. A. KHURI AND A. SAYFY, *A novel fixed point scheme: Proper setting of variational iteration method for BVPs*, Applied Mathematics Letters 48 (2015) 75-84.

[22] J. D. LAMBERT, *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem*, John Wiley, New York, 1991.

[23] FENG-GONG LANG, XIAO-PING XU, *Quintic B-spline collocation method for second order mixed boundary value problem*, Computer Physics Communications 183 (2012) 913-921.

[24] A. MARASCO, A. ROMANO, *Scientific Computing with Mathematica: Mathematical Problems for Ordinary Differential Equations, in: Modeling and Simulation in Science, Engineering and Technology*, Birkhuser Boston, Inc., Boston, MA, 2001.

[25] S. H. MIRMORADIA, I. HOSSEINPOURA, S. GHANBARPOURB, A. BARARIA, *Application of an approximate analytical method to nonlinear Troesch's problem*, Applied Mathematical Sciences 3 (32) (2009) 1579-1585.

[26] A. MOHSEN, *A simple solution of the Bratu problem*, Comput. Math. Appl. 67 (2014) 26-33.

[27] M. A. Z. RAJA, *Stochastic numerical treatment for solving Troesch's problem*, Information Sciences 279 (2014) 860-873.

[28] M. A. ZAHOOR RAJA, R. SAMAR, E. S. ALAIDAROUS, E. SHIVANIAN, *Bio-inspired computing platform for reliable solution of Bratu-type equations arising in the modeling of electrically conducting solids*, Applied Mathematical Modelling 40 (2016) 596-5977.

[29] H. RAMOS, Z. KALOGIRATOU, T. MONOVASILIS, T. E. SIMOS, *An optimized two-step hybrid block method for solving general second order initial value problems*, Numer. Alg. 72 (2016) 1089-1102.

[30] H. RAMOS, P. POPESCU, *How many k-step linear block methods exist and which of them is the most efficient and simplest one?*, Appl. Math. Comput. 316 (2018) 296-309.

[31] H. RAMOS, M. A. RUFAI, *A third-derivative two-step block Falkner-type method for solving general second-order boundary-value systems*, Mathematics and Computers in Simulation, 165 (2019) 139-155.

[32] H. RAMOS, M. A. RUFAI, *Numerical solution of boundary value problems by using an optimized two-step block method*, Numer. Alg. 84 (2020) 229-251.

[33] TEMIMI, *A discontinuous Galerkin finite element method for solving the Troeschs problem*, Appl. Math. Comput. 219 (2012) 521-529.

[34] H. B. THOMPSON, C. TISDELL, *Boundary value problems for systems of difference equations associated with systems of second-order ordinary differential equations*, Appl. Math. Lett. 15 (6) (2002) 761-766.

[35] B. A. TROESCH, *A simple approach to a sensitive two-point boundary value problem*, Journal of Computational Physics 21(3) (1976) 279-290.

[36] R. A. USMANI, *A method of high-order accuracy for the numerical integration of boundary value problems*, BIT 13 (1973) 458-469.

[37] Y. WAN, Q. GUO, N. PAN, *Thermo-electro-hydrodynamic model for electro-spinning process*, Int. J. Nonlinear Sci. Numer. Simul. 5 (2004) 5-8.

[38] M. ZAREBNIA, M. SAJJADIAN, *The sinc-Galerkin method for solving Troesch's problem*, Journal of Mathematical and Computer Modelling 56(9-10) (2012) 218-228.