Available online at www.sciencedirect.com
ScienceDirect
Mathematics and Computers in Simulation 165 (2019) 139–155
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Original articles

A third-derivative two-step block Falkner-type method for solving general second-order boundary-value systems

Higinio Ramos[a,b,*], M.A. Rufai[c]

[a] *Scientific Computing Group, Universidad de Salamanca, Plaza de la Merced, 37008 Salamanca, Spain*
[b] *Escuela Politécnica Superior de Zamora, Campus Viriato, 49022 Zamora, Spain*
[c] *The Federal University of Technology, Akure P.M.B. 704, Department of Mathematical Sciences, Akure, Ondo State, Nigeria*

Received 14 May 2017; received in revised form 29 January 2019; accepted 8 March 2019
Available online 18 March 2019

Abstract

In this article, a third derivative continuous 2-step block Falkner-type method for the general solution of second order boundary value problems of ordinary differential equations (ODEs) with different types of boundary conditions is developed. The approaches of collocation and interpolation are adopted to derive the new Falkner-type method, which is then implemented in a block mode to get approximations at all the grid points simultaneously. This method is said to be a global method since it simultaneously produces a solution over the entire interval, although it may also be categorized as a boundary value method (see Brugnano and Trigiante (1998)). The order and the convergence analysis of the proposed method are studied. The new Falkner-type scheme is applied to solve linear and non-linear systems of second-order boundary value problems of ODEs considering different types of boundary conditions. Numerical results obtained through the implementation of the scheme are very much close to the theoretical solution and found favourably compared with various existing methods in the literature.

Keywords: Two-step Falkner-type method; Second order differential equations; Modified block method; Linear and nonlinear BVPs

1. Introduction

This paper aims at obtaining directly numerical solutions for second-order boundary value problems (BVPs) with a second-order differential equation of the general form

$$y'' = f(x, y, y'), \qquad x \in [a, b]\,. \tag{1}$$

Concerning the boundary conditions we consider different possibilities, of the following types:

(a) Dirichlet boundary conditions, where the solution $y(x)$ is specified at the ends of the integration interval:

$$y(a) = y_a, \qquad y(b) = y_b\,. \tag{2}$$

* Corresponding author at: Scientific Computing Group, Universidad de Salamanca, Plaza de la Merced, 37008 Salamanca, Spain.
E-mail addresses: higra@usal.es (H. Ramos), mrufai@aims.ac.tz (M.A. Rufai).

https://doi.org/10.1016/j.matcom.2019.03.003
0378-4754/

(b) Neumann boundary conditions, where the derivative of the solution $y'(x)$ is specified at the ends of the integration interval:

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

(c) Robin boundary conditions, that consider combinations of $y(x)$ and $y'(x)$ at the ends of the integration interval:

$$g_1(y(a), y'(a)) = v_a, \qquad g_2(y(b), y'(b)) = v_b\,. \tag{4}$$

We assume that the function f in (1) satisfies the conditions to assure that the existence and uniqueness theorems are verified (for this, we refer the reader to [9,13,17]).

Numerous boundary-value problems for ordinary differential equations of the form in (1) arise in engineering and science, mathematical models are often derived to assist in the understanding and solving those problems. It is imperative to note that most of the differential equations arising from the modelling of physical phenomena do not always have known analytical solutions. Thus, the need for the derivation of numerical approaches to get approximate solutions becomes necessary. There are mainly three different types of approximation methods for solving boundary value problems of ODEs: the shooting method, 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, and 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 a suitable 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 simultaneously, thus obtaining the approximate solution at the grid points. Prominent researchers like Chen et al. [7], Cheng and Zhong [8], Lomtatidze and Malaguti [20] and Thompson [30] have applied finite difference methods to solve the problem in (1) together with selected boundary conditions. The drawback of these methods is that they require great computational costs to obtain high accuracy.

According to Carla et al. [6], the problem in (1) could be solved by reducing it to first-order boundary value problem with twice dimension. Notable scholars such as Brugnano et al. [4], Amodio et al. [1], Ascher et al. [2] among others, had transformed the equation in (1) to a first-order boundary value problem with doubled dimension in order to be able to get numerical solutions.

In this manuscript, we derive a new 2-step Falkner-type block method that uses third derivatives, to obtain directly the approximate solution of a general second-order BVP with any of the boundary conditions in (2), (3), or (4). Falkner-type methods, block methods, or even block Falkner-type methods have been used efficiently for solving initial-value problems (see [3,12,23,25–28]), but not for solving BVPs. The 2-step Falkner block technique is not so much costly in terms of the number of function evaluations, compared to some existing methods, and the major advantage of this method over transformation approach is that it gives a better numerical performance when solving directly problems of the form of the equation in (1).

The remaining part of this manuscript is outlined as follows. In Section 2 we introduce the procedure to develop the proposed two-step block Falkner method (which will be named $2BF$ for short). Some characteristics of the new method are given in Section 3, and the implementation of the proposed approach is explained in Section 4. Section 5 presents some numerical examples to show the efficiency and reliability of the proposed technique. Finally, some conclusions are reported in Section 6.

2. Development of the method

In this section, a 2-step block Falkner-type method for solving the problem in (1) is derived. Consider the interval $[x_n, x_{n+2}]$ where $x_{n+j} = x_n + jh$, and let us assume that the solution $y(x)$ on this interval is approximated by a polynomial $p(x)$ of the form

$$y(x) \simeq p(x) = \sum_{j=0}^{7} a_j x^j\,, \tag{5}$$

where the a_j are unknown coefficients that will be determined. To do that we impose that the following equations must be satisfied:

$$p(x_{n+1}) = y_{n+1}\,,\; p'(x_{n+1}) = y'_{n+1}\,, \tag{6}$$

$$p''(x_{n+j}) = f_{n+j}\,, \quad p'''(x_{n+j}) = g_{n+j}\,, \; j = 0, 1, 2, \tag{7}$$

where the y_{n+j}, y'_{n+j}, $f_{n+j} = f(x_{n+j}, y_{n+j}, y'_{n+j})$, $g_{n+j} = g(x_{n+j}, y_{n+j}, y'_{n+j})$ are respectively approximations for $y(x_{n+j})$, $y'(x_{n+j})$, $y''(x_{n+j})$ and $y'''(x_{n+j})$ with

$$g(x, y, y') = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}\, y' + \frac{\partial f}{\partial y'}\, f(x, y, y')\,.$$

The undetermined coefficients a_j are obtained by solving the system of equations in (6)–(7), and then the obtained values are substituted into (5). After some manipulations, the continuous representation of the proposed block Falkner method is obtained as

$$y(x) \simeq p(x) = \alpha_0(x) y_{n+1} + \alpha_1(x) h y'_{n+1} + h^2 \left[\sum_{j=0}^{2} \beta_j(x) f_{n+j} \right] + h^3 \left[\sum_{j=0}^{2} \gamma_j(x) g_{n+j} \right], \tag{8}$$

where $\alpha_0(x), \alpha_1(x)$ and $\beta_j(x)$, $\gamma_j(x)$, $j = 0, 1, 2$, are continuous coefficients and h is the chosen step-size.

2.1. Main formulas

The main formulas are obtained by substituting the values of $\alpha_0(x), \alpha_1(x)$ and $\beta_j(x)$, $\gamma_j(x)$, $j = 0, 1, 2$ into Eq. (8) and evaluating $p(x_n + 2h)$ and $p'(x_n + 2h)$ to get approximations for $y(x_n + 2h)$ and $y'(x_n + 2h)$. Thus, we obtain the following 2-step Falkner-type formulas

$$\begin{aligned} y_{n+2} &= y_{n+1} + h {y'}_{n+1} + h^2 \left(\frac{37}{1680} f_n + \frac{11}{30} f_{n+1} + \frac{187}{1680} f_{n+2} \right) \\ &\quad + h^3 \left(\frac{1}{168} g_n + \frac{19}{210} g_{n+1} - \frac{2}{105} g_{n+2} \right), \end{aligned} \tag{9}$$

$$\begin{aligned} {y'}_{n+2} &= y'_{n+1} + h \left(\frac{11}{240} f_n + \frac{8}{15} f_{n+1} + \frac{101}{240} f_{n+2} \right) \\ &\quad + h^2 \left(\frac{1}{80} g_n + \frac{1}{6} g_{n+1} - \frac{13}{240} g_{n+2} \right). \end{aligned} \tag{10}$$

These formulas must be considered along with the grid points in order to get a solution of the BVP. That is, if we consider the grid points $a = x_0 < x_1 < x_2 < \cdots < x_{N-2} < x_{N-1} < x_N = b$ with $N \in \mathbb{N}$, we take the formulas in (9)–(10) for $n = 0, 1, 2, \ldots, N-2$. This makes a total of $2(N-1)$ equations. As the number of unknowns is $2N+2$ and we have two boundary conditions, we still need two additional formulas.

2.2. Additional formulas

In order to get the two additional formulas to form the block Falkner-type method for solving the BVP we proceed by evaluating the continuous scheme (8) and its first derivative at the point $x = x_n$. We obtain

$$\begin{aligned} y_n &= y_{n+1} - h {y'}_{n+1} + h^2 \left(\frac{187}{1680} f_n + \frac{11}{30} f_{n+1} + \frac{37}{1680} f_{n+2} \right) \\ &\quad + h^3 \left(\frac{2}{105} g_n - \frac{19}{210} g_{n+1} - \frac{1}{168} g_{n+2} \right), \end{aligned} \tag{11}$$

$$\begin{aligned} {y'}_{n} &= y'_{n+1} - h \left(\frac{101}{240} f_n + \frac{8}{15} f_{n+1} + \frac{11}{240} f_{n+2} \right) \\ &\quad - h^2 \left(\frac{13}{240} g_n - \frac{1}{6} g_{n+1} - \frac{1}{80} g_{n+2} \right). \end{aligned} \tag{12}$$

These are general formulas, but we need only two more formulas. Thus, we particularize these formulas for a specific value of n. This value of n may be any from 0 to $N-2$. In our approach, we have considered $n = 0$, and thus the resulting two additional formulas are

$$\begin{aligned} y_0 &= y_1 - h{y'}_1 + h^2\left(\frac{187}{1680} f_0 + \frac{11}{30} f_1 + \frac{37}{1680} f_2\right) \\ &\quad + h^3\left(\frac{2}{105} g_0 - \frac{19}{210} g_1 - \frac{1}{168} g_2\right), \end{aligned} \tag{13}$$

$$\begin{aligned} {y'}_0 &= y'_1 - h\left(\frac{101}{240} f_0 + \frac{8}{15} f_1 + \frac{11}{240} f_2\right) \\ &\quad - h^2\left(\frac{13}{240} g_0 - \frac{1}{6} g_1 - \frac{1}{80} g_2\right). \end{aligned} \tag{14}$$

Considering altogether the formulas in (9)–(10) for $n = 0, 1, \ldots, N-2$, the two boundary conditions (one of those in (2), (3) or (4)), and the formulas in (13)–(14) we get the $2BF$ method for solving the BVP. This method consists in a system of $2N+2$ equations with $2N+2$ unknowns: $y_0, y_1, \ldots, y_N, y'_0, y'_1, \ldots, y'_N$. This system may be linear or not according to the type of function f on the right-hand side of the differential equation in (1). In case we have a linear system we use any of the available linear solvers in the literature, and in the case of a nonlinear system, we use Newton's method. In the latter case, some starting values are needed to initialize the iterative solver. In the implementation section, we will describe how these values are provided, depending on the type of boundary conditions.

3. Characteristics of the method

3.1. Order and local truncation errors

Although we have particularized the equations in (11)–(12) for $n = 0$, as we can choose any value of $n = 0, 1, \ldots, N-2$ we consider these formulas in general. Equations in (9)–(12) may be written in the following form

$$P\,Y_n = h\,Q\,Y'_n + h^2\,R\,F_n + h^3\,S\,G_n\,, \tag{15}$$

where P, Q, R, S are matrices of coefficients with dimensions 4×3, and

$$\begin{aligned} Y_n &= (y_n, y_{n+1}, y_{n+2})^T, \\ Y'_n &= \left(y'_n, y'_{n+1}, y'_{n+2}\right)^T, \\ F_n &= (f_n, f_{n+1}, f_{n+2})^T \\ G_n &= (g_n, g_{n+1}, g_{n+2})^T\,. \end{aligned}$$

The corresponding matrices for the formula in (15) are given by

$$P = \begin{pmatrix} 0 & -1 & 1 \\ 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad Q = \begin{pmatrix} 0 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 0 \end{pmatrix},$$

$$R = \begin{pmatrix} \frac{37}{1680} & \frac{11}{30} & \frac{187}{1680} \\ -\frac{11}{240} & -\frac{8}{15} & -\frac{101}{240} \\ \frac{187}{1680} & \frac{11}{30} & \frac{37}{1680} \\ \frac{101}{240} & \frac{8}{15} & \frac{11}{240} \end{pmatrix}, \quad S = \begin{pmatrix} \frac{1}{168} & \frac{19}{210} & -\frac{2}{105} \\ -\frac{1}{80} & -\frac{1}{6} & \frac{13}{240} \\ \frac{2}{105} & -\frac{19}{210} & -\frac{1}{168} \\ \frac{13}{240} & -\frac{1}{6} & -\frac{1}{80} \end{pmatrix}.$$

Following the lines proposed by Ramos et al. [24] for a Falkner-type method, assuming that $y(x)$ is a sufficiently differentiable function, we denote the linear difference operator $\mathbb{L}$ associated to the formulas in (15) as follows:

$$\mathbb{L}[y(x); h] = \sum_{j=0}^{2}\left[\bar{\alpha}_j\, y\,(x_n + jh) - h\bar{\beta}_j\, y'\,(x_n + jh) - h^2\bar{\gamma}_j\, y''\,(x_n + jh) - h^3\bar{\delta}_j\, y'''\,(x_n + jh)\right], \tag{16}$$

where $\bar{\alpha}_j, \bar{\beta}_j, \bar{\gamma}_j$ and $\bar{\delta}_j$ are respectively the vector columns of matrices P, Q, R and S.

Expanding Eq. (16) using Taylor series about x_n we obtain that $\mathbb{L}[y(x); h]$ may be written as

$$\mathbb{L}[y(x); h] = \bar{C}_0 y(x_n) + \bar{C}_1 h y'(x_n) + \bar{C}_2 h^2 y''(x_n) + \cdots + \bar{C}_q h^q y^q(x_n) + \ldots, \tag{17}$$

where

$$\bar{C}_q = \frac{1}{q!}\left[\sum_{j=1}^{2} j^q \bar{\alpha}_j - q\sum_{j=1}^{2} j^{q-1}\bar{\beta}_j - q(q-1)\sum_{j=1}^{2} j^{q-2}\bar{\gamma}_j - q(q-1)(q-2)\sum_{j=1}^{2} j^{q-3}\bar{\delta}_j\right], \tag{18}$$

for $q = 0, 1, 2, 3, \ldots$.

Definition 3.1. The linear difference operator and the associated formulas are said to be of order p if $\bar{C}_0 = \bar{C}_1 = \cdots = \bar{C}_{p+1} = 0$, $\bar{C}_{p+2} \neq 0$. The $\bar{C}_i$ are column vectors of scalars of size 4, and $\bar{C}_{p+2}$ is the vector of error constants (see Rufai [29]).

From (18), we readily obtain that $\bar{C}_0 = \bar{C}_1 = \cdots = \bar{C}_7 = 0$ and

$$\bar{C}_8 = \left(\frac{29}{604800}, \frac{1}{9450}, \frac{29}{604800}, -\frac{1}{9450}\right)^T .$$

We note that all the proposed formulas in Eqs. (9)–(12) are of order $p = 6$.

3.2. Convergence analysis

It is well known that if a numerical method does not converge then it is of little use. This subsection is devoted to prove the convergence of the proposed method. We begin by exposing the definition of convergence, and then we will show that the proposed method is convergent by compactly writing the main formulas in (9)–(10) and the additional ones in (13)–(14) in matrix–vector form (see [16]).

Definition 3.2. 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 pth-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 $\max_{0\le j\le N} |y(x_j) - y_j| \to 0$ as $h \to 0$.

We will consider that the boundary conditions are of the type in (2), which is a very common case. For other cases the proof can be made similarly, making the appropriate changes. We assume that these are exact boundary conditions, and thus we have the known values $y_0 = y(x_0) = y_a$ and $y_N = y(x_N) = y_b$, while the unknowns will be $y_1, y_2, \ldots, y_{N-1}, y'_0, y'_1, \ldots, y'_N$. To proof convergence, we introduce the following notations. Let D represent the $2N \times 2N$ matrix defined by

$$D = \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix},$$

where D_{11} and D_{21} are submatrices of dimension $N \times (N-1)$ given respectively by

$$D_{11} = \begin{pmatrix} -1 & 0 & 0 & 0 & \ldots & 0 & 0 \\ -1 & 1 & 0 & 0 & \ldots & 0 & 0 \\ 0 & -1 & 1 & 0 & \ldots & 0 & 0 \\ 0 & 0 & -1 & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & -1 & 1 & 0 \\ 0 & 0 & 0 & \ldots & 0 & -1 & 1 \\ 0 & 0 & 0 & \ldots & 0 & 0 & -1 \end{pmatrix},$$

$$D_{21} = \begin{pmatrix} 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 \end{pmatrix},$$

and D_{12}, D_{22} are submatrices of dimension $N \times (N+1)$ given respectively by

$$D_{12} = h \begin{pmatrix} 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & -1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & -1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -1 & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & -1 & 0 \end{pmatrix},$$

$$D_{22} = \begin{pmatrix} 1 & -1 & 0 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & -1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -1 & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & -1 & 1 \end{pmatrix}.$$

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

$$U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix},$$

where the U_{ij} are submatrices of dimension $N \times (N+1)$ with similar structure given respectively by

$$U_{11} = h \begin{pmatrix} -\frac{187}{1680} & -\frac{11}{30} & -\frac{37}{1680} & 0 & 0 & \dots & 0 \\ -\frac{37}{1680} & -\frac{11}{30} & -\frac{187}{1680} & 0 & 0 & \dots & 0 \\ 0 & -\frac{37}{1680} & -\frac{11}{30} & -\frac{187}{1680} & 0 & \dots & 0 \\ 0 & 0 & -\frac{37}{1680} & -\frac{11}{30} & -\frac{187}{1680} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & -\frac{37}{1680} & -\frac{11}{30} & -\frac{187}{1680} \end{pmatrix},$$

$$U_{12} = h^2 \begin{pmatrix} -\frac{2}{105} & \frac{19}{210} & \frac{1}{168} & 0 & 0 & \dots & 0 \\ -\frac{1}{168} & -\frac{19}{210} & \frac{2}{105} & 0 & 0 & \dots & 0 \\ 0 & -\frac{1}{168} & -\frac{19}{210} & \frac{2}{105} & 0 & \dots & 0 \\ 0 & 0 & -\frac{1}{168} & -\frac{19}{210} & \frac{2}{105} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & -\frac{1}{168} & -\frac{19}{210} & \frac{2}{105} \end{pmatrix},$$

$$U_{21}=\begin{pmatrix} \frac{101}{240} & \frac{8}{15} & \frac{11}{240} & 0 & 0 & \dots & 0\\ -\frac{11}{240} & -\frac{8}{15} & -\frac{101}{240} & 0 & 0 & \dots & 0\\ 0 & -\frac{11}{240} & -\frac{8}{15} & -\frac{101}{240} & 0 & \dots & 0\\ 0 & 0 & -\frac{11}{240} & -\frac{8}{15} & -\frac{101}{240} & \dots & 0\\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots\\ 0 & 0 & 0 & \dots & -\frac{11}{240} & -\frac{8}{15} & -\frac{101}{240} \end{pmatrix},$$

$$U_{22}=h\begin{pmatrix} \frac{13}{240} & -\frac{1}{6} & -\frac{1}{80} & 0 & 0 & \dots & 0\\ -\frac{1}{80} & -\frac{1}{6} & \frac{13}{240} & 0 & 0 & \dots & 0\\ 0 & -\frac{1}{80} & -\frac{1}{6} & \frac{13}{240} & 0 & \dots & 0\\ 0 & 0 & -\frac{1}{80} & -\frac{1}{6} & \frac{13}{240} & \dots & 0\\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots\\ 0 & 0 & 0 & \dots & -\frac{1}{80} & -\frac{1}{6} & \frac{13}{240} \end{pmatrix}.$$

Now we consider the exact solution of the problem, $y(x)$, and define the $2N$-vector

$$Y=\left(y(x_1),\dots,y(x_{N-1}),y'(x_0),\dots,y'(x_N)\right)^T,$$

and the $(2N+2)$-vector

$$F=\left(f(x_0,y(x_0),y'(x_0)),\dots,f(x_N,y(x_N),y'(x_N)),g(x_0,y(x_0),y'(x_0)),\dots,g(x_N,y(x_N),y'(x_N))\right)^T.$$

With the above notations, the exact form of the system given by the equation in (13), the equations in (9) for $n=0,1,\dots,N-2$, the equation in (14), and the equations in (10) for $n=0,1,\dots,N-2$, in this order, can be expressed as (note that we have included as subscripts the corresponding dimensions to make it clearer)

$$D_{2N\times 2N}Y_{2N}+h\,U_{2N\times(2N+2)}F_{2N+2}+C_{2N}=L(h)_{2N}, \tag{19}$$

where C_{2N} is a vector containing the known values, which in this case is

$$C_{2N}=(y_a,\dots,y_b,0,\dots,0)^T,$$

and $L(h)_{2N}$ corresponds to the local truncation errors of the formulas, that is,

$$L(h)_{2N}=\begin{pmatrix} \frac{29}{604800}y^{(8)}(x_0)h^8+\mathcal{O}(h^9)\\ \frac{29}{604800}y^{(8)}(x_0)h^8+\mathcal{O}(h^9)\\ \frac{29}{604800}y^{(8)}(x_1)h^8+\mathcal{O}(h^9)\\ \vdots\\ \frac{29}{604800}y^{(8)}(x_{N-2})h^8+\mathcal{O}(h^9)\\ \frac{-1}{9450}y^{(8)}(x_0)h^7+\mathcal{O}(h^8)\\ \frac{1}{9450}y^{(8)}(x_0)h^7+\mathcal{O}(h^8)\\ \frac{1}{9450}y^{(8)}(x_1)h^7+\mathcal{O}(h^8)\\ \vdots\\ \frac{1}{9450}y^{(8)}(x_{N-2})h^7+\mathcal{O}(h^8) \end{pmatrix}. \tag{20}$$

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

$$D_{2N\times 2N}\bar{Y}_{2N}+h\,U_{2N\times(2N+2)}\bar{F}_{2N+2}+C_{2N}=0, \tag{21}$$

where $\bar{Y}_{2N}$ approximates the vector Y_{2N},

$$\bar{Y}_{2N}=\left(y_1,\dots,y_{N-1},y'_0,\dots,y'_N\right)^T,$$

and $\bar{F}_{2N+2}$ approximates F_{2N+2},

$$\bar{F}_{2N+2} = (f_0, \dots, f_N, g_0, \dots, g_N)^T \ .$$

On subtracting (21) from (19) and simplifying we get

$$D_{2N\times 2N} E_{2N} + h\, U_{2N\times(2N+2)} \left(F - \bar{F}\right)_{2N+2} = L(h)_{2N}, \tag{22}$$

where $E_{2N} = \bar{Y}_{2N} - Y_{2N} = \left(e_1, \dots, e_{N-1}, e'_0, \dots, e'_N\right)^T$ contains the errors at the grid points.

By using the Mean-Value Theorem, we can write for $i = 0, 1 \dots, N$ that

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

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

$$F - \bar{F} = \begin{pmatrix} \frac{\partial f}{\partial y}(\xi_0) & 0 & \dots & 0 & \frac{\partial f}{\partial y'}(\xi_0) & 0 & \dots & 0 \\ 0 & \frac{\partial f}{\partial y}(\xi_1) & \dots & 0 & 0 & \frac{\partial f}{\partial y'}(\xi_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{\partial f}{\partial y}(\xi_N) & 0 & 0 & \dots & \frac{\partial f}{\partial y'}(\xi_N) \\ \frac{\partial g}{\partial y}(\eta_0) & 0 & \dots & 0 & \frac{\partial g}{\partial y'}(\eta_0) & 0 & \dots & 0 \\ 0 & \frac{\partial g}{\partial y}(\eta_1) & \dots & 0 & 0 & \frac{\partial g}{\partial y'}(\eta_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{\partial g}{\partial y}(\eta_N) & 0 & 0 & \dots & \frac{\partial g}{\partial y'}(\eta_N) \end{pmatrix} \begin{pmatrix} e_0 \\ e_1 \\ \vdots \\ e_N \\ e'_0 \\ e'_1 \\ \vdots \\ e'_N \end{pmatrix}$$

$$= \begin{pmatrix} 0 & \dots & 0 & \frac{\partial f}{\partial y'}(\xi_0) & 0 & \dots & 0 & 0 \\ \frac{\partial f}{\partial y}(\xi_1) & \dots & 0 & 0 & \frac{\partial f}{\partial y'}(\xi_1) & \dots & 0 & 0 \\ \vdots & \ddots & 0 & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & \frac{\partial f}{\partial y}(\xi_{N-1}) & 0 & 0 & \dots & \frac{\partial f}{\partial y'}(\xi_{N-1}) & 0 \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 & \frac{\partial f}{\partial y'}(\xi_N) \\ 0 & \dots & 0 & \frac{\partial g}{\partial y'}(\eta_0) & 0 & \dots & 0 & 0 \\ \frac{\partial g}{\partial y}(\eta_1) & \dots & 0 & 0 & \frac{\partial g}{\partial y'}(\eta_1) & \dots & 0 & 0 \\ \vdots & \ddots & 0 & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & \frac{\partial g}{\partial y}(\eta_{N-1}) & 0 & 0 & \dots & \frac{\partial g}{\partial y'}(\eta_{N-1}) & 0 \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 & \frac{\partial g}{\partial y'}(\eta_N) \end{pmatrix} \begin{pmatrix} e_1 \\ \vdots \\ e_{N-1} \\ e'_0 \\ e'_1 \\ \vdots \\ e'_N \end{pmatrix}$$

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

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 (22) may be rewritten as follows

$$D_{2N\times 2N} E_{2N} + h\, U_{2N\times(2N+2)} J_{(2N+2)\times 2N} E_{2N} = L(h)_{2N}, \tag{23}$$

and setting $M_{2N\times 2N} = D_{2N\times 2N} + h\, U_{2N\times(2N+2)} J_{(2N+2)\times 2N}$ we have that for sufficiently small values of h, M is invertible and thus

$$E_{2N} = \left(M^{-1}\right)_{2N\times 2N} L(h)_{2N}\,. \tag{24}$$

We consider the maximum norm in $\mathbb{R}^{2N}$, $\|E\| = \max_i |e_i|$, and the corresponding matrix induced norm in $\mathbb{R}^{2N\times 2N}$. After expanding each term of $\left(M^{-1}\right)_{2N\times 2N}$ in series around h it can be shown after tedious calculations that

$\| \left(M^{-1}\right)_{2N\times 2N} \| = \mathcal{O}(h^{-1})$. This is essentially related to the fact that the uniform norm of the inverse of D grows like h^{-1}, as one can verify rather simply.

Consequently, from the equation in (24) and the form of the vector $L(h)_{2N}$ in (20), assuming that $y(x)$ has in $[a, b]$ bounded derivatives up to the eighth order, we have that

$$\begin{aligned}\|E_{2N}\| &\leq \| \left(M^{-1}\right)_{2N\times 2N} \| \, \|L(h)_{2N}\| \\ &= \mathcal{O}(h^{-1})\,\mathcal{O}(h^{7}) \\ &\leq K\,h^{6}.\end{aligned}$$

Therefore, the proposed method $2BF$ is a sixth-order convergent method.

4. Implementation

In order to give numerical approximations to the considered problems, we have to solve the system of $2N+2$ equations with $2N+2$ unknowns given by (9)–(10), $j = 0, 1, \ldots, N-2$, (13)–(14) and the two boundary conditions (either (2), or (3), or (4)). If the function f in (1) is linear we use any available linear solver, in our case we have considered the one in the *Mathematica 8.0* system. If f is nonlinear we use Newton's method. If the system to be solved is denoted by $\mathbf{F}(\mathbf{y}) = 0$, the stopping criterion considered has been $\|\mathbf{F}(\mathbf{y}^{i})\| \leq 10^{-16}$, with a maximum number of iterations established in $MaxIter = 100$. When using Newton's method it is important to consider initial guesses reasonably close to the true roots. We distinguish two situations:

- In case of Dirichlet conditions it is $y_0 = y_a$, $y_N = y_b$, and thus the system is reduced to $2N$ equations with $2N$ unknowns. In this case we consider as initial starting points:

$$y_j^{(0)} = y_0 + \frac{y_N - y_0}{b-a}\, jh\,, \quad j = 1, 2, \ldots, N-1; \tag{25}$$

$$y_j'^{(0)} = \frac{y_N - y_0}{b-a}\,, \quad j = 0, 1, 2, \ldots, N\,. \tag{26}$$

- In case of Neumann of Robin conditions, we may consider that they are of the general type in (4). In this situation, we adopt a strategy similar to the one in [10] or [22], where a homotopy-type procedure was used. We consider a family of nonlinear BVPs P_j, $j = 0, 1, 2, \ldots, s$, such that for $j = 0$ the problem P_0 admits only the solution $y(x) = 0$, while when $j = s$ we recover the original problem. In this way, we have a family of BVPs given by

$$P_j \equiv \begin{cases} y'' = f(x, y, y') - f(x, 0, 0) + \dfrac{j}{s}\, f(x, 0, 0)\,, \\ g_1(y(a), y'(a)) = \dfrac{j}{s}\, v_a\,, \\ g_2(y(b), y'(b)) = \dfrac{j}{s}\, v_b, \end{cases} \tag{27}$$

  for $j = 0, 1, \ldots, s$. Each of these problems P_j for $j = 1, 2, \ldots, s$ is solved using the method developed in this paper, taking as starting guesses the values obtained after solving the previous problem P_{j-1}. For $j = s$ the nonlinear system corresponding to the original BVP is solved taking as starting guesses the values obtained after solving the problem P_{s-1}.

This strategy has the sole purpose of providing suitable starting values for the Newton's method. If one has another way of supplying the starting values it can be used, even sometimes it is enough just to take these starting values as zero (which may be accomplished taking $s = 1$). There are no established criteria for choosing the number s of intermediate problems, it depends on the difficulty of the problem. Nevertheless, in our numerical experiments we have taken in general $s = 1$ (that is, just the original problem), and in the more complicated cases with $s = 6$ we got good results.

5. Numerical examples

This section is devoted to determine the accuracy, suitability and applicability of the new proposed $2BF$ method for solving the problem in (1) for different types of boundary conditions. We have tested the performance of the

Table 1
Comparison of the maximum absolute errors (MAE) on Problem 1.

h	MAE with $2BF$	MAE in [31]
$\frac{1}{2}$	1.51722×10^{-7}	0.26400×10^{-4}
$\frac{1}{4}$	2.11789×10^{-9}	0.16600×10^{-5}
$\frac{1}{8}$	3.78544×10^{-11}	0.10700×10^{-6}
$\frac{1}{16}$	6.27165×10^{-13}	0.67200×10^{-8}
$\frac{1}{32}$	1.03251×10^{-14}	0.42200×10^{-9}

proposed method on different BVPs including linear and non-linear, scalar and systems of second-order. In each example, we determine the absolute errors of the approximate solutions and the numerical results obtained are compared with various existing approaches in the literature. The accuracy of the proposed method may be seen in Tables 1–8.

5.1. Problem 1

We consider the following BVP that was solved by Usmani [31],

$$y''(x) = y(x) + x^2 - 2, \;\; y(0) = 0, \;\; y(1) = 1, \quad 0 \leq x \leq 1\,,$$

whose exact solution is given by

$$y(x) = \frac{e^2x^2 - x^2 + 2e^{1-x} - 2e^{x+1}}{1 - e^2}\,.$$

The numerical results with the proposed method are compared with those provided with a method of high-order accuracy, discussed in Usmani [31] which solved the same problem using the same values of h in Table 1. The local truncation errors in the method by Usmani were also of the form $T_{n+1} = \mathbb{O}(h^8)$, but the difference with the proposed method is that it was designed for linear equations and consists of a system of N equations. The differential equation of the problem solved by Usmani was simpler, $y''(x) = f(x)y(x) + g(x)$, without the appearance of the first derivative. It was observed that the maximum absolute errors (MAE) obtained with our method are smaller than those with the method in [31].

5.2. Problem 2

We now consider the nonlinear BVP that was solved by Ha [14]

$$y''(x) = y^3(x) - y(x)y'(x),$$

subject to the boundary conditions

$$y(1) = \frac{1}{2}, \;\; y(2) = \frac{1}{3}, \;\; 1 \leq x \leq 2,$$

whose exact solution is given by

$$y(x) = \frac{1}{x+1}\,.$$

Ha uses a shooting method that requires two initial-value problems which must be solved at each iteration. As an initial-value solver, he considers the classical Runge–Kutta method, which is combined with a generalized Newton's method that uses a relaxation parameter. The difficulty with this method relies on determining a suitable relaxation parameter and a suitable initial guess for the velocity. In fact, for this problem, the author finds some troubles because there exists a critical value between 4.15 and 4.20 for which using an initial velocity greater than this value the method does not converge. Table 2 shows the absolute errors at different points taking $h = 0.05$, that is $N = 20$. We have included the best results in [14] which were obtained for an initial guess for the velocity $v_0 = 4$. We see that the proposed method greatly outperforms those results.

Table 2
Comparison of the maximum absolute errors (MAE) on Problem 2.

x	MAE with $2BF$	MAE in [14] taking $v_0 = 4$
1.0	0	0
1.1	1.93168×10^{-12}	0.0000550
1.2	2.91617×10^{-12}	0.0000910
1.3	3.27344×10^{-12}	0.0001110
1.4	3.23480×10^{-12}	0.0001180
1.5	2.94503×10^{-12}	0.0001160
1.6	2.49561×10^{-12}	0.0001050
1.7	1.94511×10^{-12}	0.0000880
1.8	1.33088×10^{-12}	0.0000650
1.9	6.77347×10^{-13}	0.0000370
2.0	0	0.0000060

Table 3
Comparison of the maximum absolute errors (MAE) on Problem 3.

h	BC_1 with $2BF$	BC_2 with $2BF$	BC_1 in [19]	BC_2 in [19]
$\frac{1}{5}$	1.06656×10^{-8}	1.47864×10^{-8}	2.88500×10^{-6}	7.69400×10^{-6}
$\frac{1}{10}$	1.70827×10^{-10}	3.47774×10^{-10}	1.83500×10^{-7}	4.37200×10^{-7}
$\frac{1}{20}$	2.76146×10^{-12}	5.99476×10^{-12}	1.15100×10^{-8}	2.66600×10^{-8}
$\frac{1}{40}$	4.39648×10^{-14}	9.72555×10^{-14}	7.20200×10^{-10}	1.65600×10^{-9}
$\frac{1}{80}$	6.80012×10^{-16}	4.44089×10^{-15}	4.49800×10^{-11}	1.03100×10^{-10}

5.3. Problem 3

In the third example, we solve the problem appeared in [19] given by

$$y''(x) + (x^2 - 6x - 1)y'(x) + (5x - x^2 + 6)y(x) = e^x - x^2 + 5x + 6, \quad 0 \le x \le 1\,,$$

subject to boundary conditions either of Neumann or Robin types:

$$BC_1 = \{y'(0) = 1,\ \ y'(1) = 2e\}$$

or

$$BC_2 = \{y(0) + y'(0) = 2,\ \ 2y(1) - y'(1) = 2\}\,.$$

In both cases the exact solution is

$$y(x) = xe^x + 1.$$

Problem 3 was also solved by Lang and Xu [19] using a quintic B-spline collocation method that needs to solve a system of $N+5$ equations. In order to provide the numerical solution, we consider different step-sizes and compare the numerical results of the proposed method with the method in [19]. From the data in Table 3, it could be seen clearly that $2BF$ method gives better numerical results.

5.4. Problem 4

Here, we consider a nonlinear BVP with Neumann boundary conditions

$$y''(x) = 2y^3(x),\ \ y'(0) = -1,\ \ y'(1) = -1/4, \qquad 0 \le x \le 1,$$

that was presented in [15] and previously in [18], whose exact solution is given by

$$y(x) = \frac{1}{1+x}\,.$$

Table 4
Comparison of the maximum absolute errors (MAE) on Problem 4.

h	MAE with 2BF	MAE in [15]
$\frac{1}{16}$	1.40300×10^{-8}	5.94920×10^{-4}
$\frac{1}{32}$	2.23986×10^{-10}	1.55450×10^{-4}
$\frac{1}{64}$	3.45612×10^{-12}	3.97570×10^{-5}
$\frac{1}{128}$	5.34017×10^{-14}	1.00550×10^{-5}
$\frac{1}{256}$	9.99201×10^{-16}	2.52830×10^{-6}
$\frac{1}{512}$	7.77156×10^{-16}	6.33925×10^{-7}

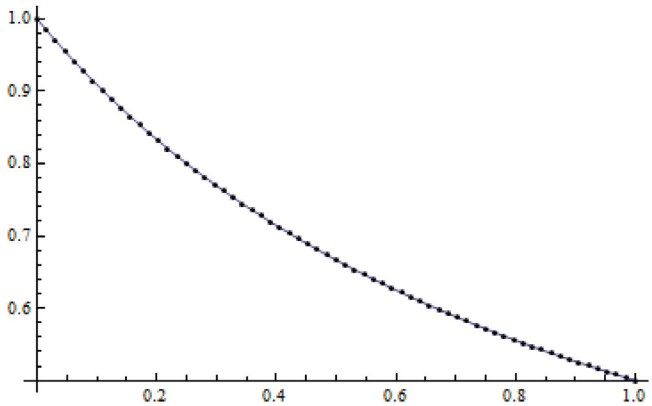

Fig. 1. Comparison of the exact solution of problem 4 with the discrete solution provided by $2BF$ for $N = 64$.

The maximum absolute errors (MAE) for various values of h are displayed in Table 4. This problem was also solved by Siraj-ul-Islam et al. [15] using a collocation method combined with the Haar wavelets. The MAE with this later method for $N = 16$ and $N = 512$ collocation points are 5.94920×10^{-4} and 6.33925×10^{-7} while the MAE with our $2BF$ approach as seen in Table 4 are 1.40300×10^{-8} and 7.77156×10^{-16} for the same number of collocation points. It is obvious from the results in Table 4 that our method is more accurate than the collocation method with the Haar wavelets proposed in [15]. Comparison of the analytical versus approximate solution is displayed in Fig. 1.

For this nonlinear problem, we have used the strategy described in the implementation section concerning the starting values. The sequence of approximate solutions for $s = 6$ is shown in Fig. 2.

5.5. Problem 5

Now we consider a non-linear second order two point boundary value problem

$$y''(x) = -\frac{1}{2} y(x) y'(x),$$

subject to the mixed boundary conditions of Robin type

$$\{2y(0) - y'(0) = -1.44, \;\; y(4) + 0.5y'(4) = -6\} \qquad 0 \le x \le 4\,,$$

whose exact solution is

$$y(x) = \frac{4}{x-5}\,.$$

We have solved this problem with the proposed method and the strategy described in the implementation section for $s = 1$, that is, the starting guesses have been taken equal to zero for the approximate values of the solution and the derivative. Table 5 shows the maximum absolute errors (MAE) with our method for different step-sizes $\left(h = \frac{4}{10}, \frac{4}{20}, \frac{4}{40}, \frac{4}{80}, \frac{4}{100}\right)$. This problem was also solved by Lang and Xu [19] using a quintic B-spline collocation method. From the data in Table 5 we can see that the new approach outperforms the method in [19].

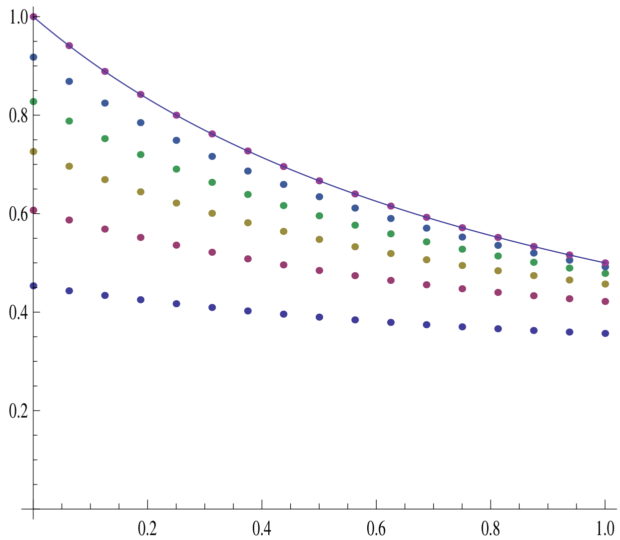

Fig. 2. Sequence of approximate solutions of Problem 4 (dots) to the exact one (continuous line) for $N = 16$.

Table 5
Comparison of the maximum absolute errors (MAE) on Problem 5.

h	MAE with 2BF	MAE in [19]
$\frac{4}{10}$	6.25766×10^{-4}	3.66300×10^{-3}
$\frac{4}{20}$	1.87062×10^{-5}	2.64200×10^{-4}
$\frac{4}{40}$	4.07756×10^{-7}	1.77700×10^{-5}
$\frac{4}{80}$	7.49040×10^{-9}	1.12500×10^{-6}
$\frac{4}{100}$	2.02945×10^{-9}	4.61700×10^{-7}

5.6. *Problem 6*

This problem corresponds to Example 4.3 appeared in [2] (p. 140). It was included there as an example of stiff problem, with the presence of rapidly increasing and non-increasing of fundamental modes. Although in [2] it was presented by a system of two first order equations, as our method is intended for directly solving second order problems, we have reformulated it as

$$y''(x) = \lambda^2 y(x) - \frac{\pi\left(\lambda^2 + 4\pi^2\right)}{\lambda}\, \sin(2\pi x)\,,$$

with Dirichlet conditions

$$y(0) = \frac{e^{-\lambda} - 1}{e^{-\lambda} + 1}\,, \quad y(1) = \frac{1 - e^{-\lambda}}{e^{-\lambda} + 1}\,.$$

The exact solution is given by

$$y(x) = \frac{e^{\lambda(x-1)} - e^{-\lambda x}}{1 + e^{-\lambda}} + \frac{\pi}{\lambda}\, \sin(2\pi x)\,.$$

We have taken $\lambda = 50$ as in [2], where it was shown the bad performance of the shooting method due to the exponential growth of the error. Table 6 presents the maximum absolute errors with the proposed method when $s = 1$ for different stepsizes, showing the good performance of the $2BF$ approach. Fig. 3 shows the exact and discrete solutions taking $N = 32$.

Table 6
Maximum absolute errors (MAE) with the proposed method for Problem 6.

h	MAE with $2BF$
$\frac{1}{32}$	2.23714×10^{-4}
$\frac{1}{64}$	4.40660×10^{-6}
$\frac{1}{128}$	6.91612×10^{-8}
$\frac{1}{256}$	1.08359×10^{-9}
$\frac{1}{512}$	1.69079×10^{-11}

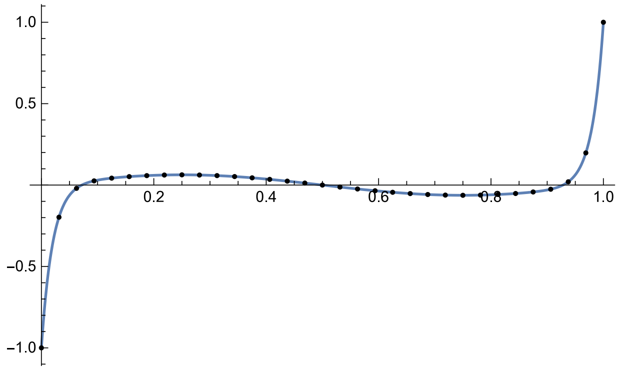

Fig. 3. Exact and discrete solutions of Problem 6 (for $N = 32$).

5.7. Problem 7

In the next problem we consider a linear system of second order BVPs of ordinary differential equations given by

$$\begin{aligned} y_1''(x) + (2x-1)y_1'(x) + \cos(\pi x)y_2'(x) &= f_1(x), \\ y_2''(x) + x y_1(x) &= f_2(x), \end{aligned}$$

subject to the boundary conditions

$$\begin{aligned} &y_1(0) = y_1(1) = 0, \\ &y_2(0) = y_2(1) = 0, \end{aligned}$$

where $0 \le x \le 1$ and

$$\begin{aligned} &f_1(x) = -\pi^2 \sin(\pi x) + (2x-1)\pi\cos(\pi x) + (2x-1)\cos(\pi x), \\ &f_2(x) = 2 + x\sin(\pi x)\,. \end{aligned}$$

The exact solution of the problem is given by:

$$y_1(x) = \sin(\pi x), \qquad y_2(x) = x^2 - x.$$

This problem was also solved by Caglar and Caglar [5], Lu [21] and Dehghan and Nikpour [11]. In Table 7 we can observe the good performance of the proposed method $2BF$. The other approach considered for comparison was the B-spline method in [5]. Our method shows better accuracy compared with the existing method in [5].

5.8. Problem 8

Finally we consider a nonlinear system of second order BVPs of ordinary differential equations given by

$$y_1''(x) + x y_1'(x) + \cos(\pi x)y_2'(x) = f_1(x),$$

Table 7
Comparison of the Maximum Absolute Errors (MAE) on Problem 6.

h	$y_1(x)$ with $2BF$	$y_2(x)$ with $2BF$	$y_2(x)$ in [5]	$y_2(x)$ in [5]
$\frac{1}{21}$	1.09056×10^{-9}	5.56843×10^{-11}	1.89579×10^{-3}	9.60501×10^{-5}
$\frac{1}{41}$	1.97582×10^{-11}	1.00642×10^{-12}	4.74623×10^{-4}	2.40866×10^{-5}
$\frac{1}{61}$	1.82310×10^{-12}	9.27314×10^{-14}	2.10999×10^{-4}	1.07113×10^{-5}

Table 8
Comparison of the Maximum Absolute Errors (MAE) on Problem 7.

h	$y_1(x)$ with $2BF$	$y_2(x)$ with $2BF$
$\frac{1}{21}$	1.04966×10^{-12}	8.54872×10^{-14}
$\frac{1}{41}$	1.88183×10^{-14}	1.55431×10^{-15}
$\frac{1}{61}$	1.77636×10^{-15}	1.66533×10^{-16}
h	$y_1(x)$ with N-LMQDQ in [11]	$y_2(x)$ with N-LMQDQ in [11]
$\frac{1}{21}$	3.8393×10^{-5}	5.5775×10^{-5}
$\frac{1}{41}$	9.5835×10^{-6}	1.3892×10^{-5}
$\frac{1}{61}$	4.2655×10^{-6}	6.0484×10^{-6}

$$y_2''(x) + xy_1(x) + xy_2'(x) = f_2(x),$$

subject to the boundary conditions

$$y_1(0) = y_1(1) = 0,$$
$$y_2(0) = y_2(1) = 0,$$

where $0 \le x \le 1$ and

$$f_1(x) = \sin(x) + (x^2 - x + 2)\cos(x) + (1 - 2x)\cos(\pi x),$$
$$f_2(x) = -2 + x\sin(x) + (x^2 - x)\cos(x) + x(1 - 2x)^2 .$$

The exact solution of the problem is given by:

$$y_1(x) = (x - 1)\sin(x), \qquad y_2(x) = x - x^2.$$

This problem was also solved by Dehghan and Nikpour [11]. In Table 8 we can observe the good performance of the proposed method $2FB$. The other approach considered for comparison was the method based on the Local Radial Basis Function Differential Quadrature (LRBFDQ) technique for approximating the derivative, in [11]. Our method shows better accuracy compared with the existing method in [11].

Fig. 4 shows the exact and discrete solutions provided by $2BF$ (for $N = 21$) for $y_1(x)$ and $y_2(x)$.

6. Conclusions

In this manuscript, a 2-step Falkner-type block method ($2BF$) has been developed for the direct solution of general two-point boundary value problems for ordinary differential systems. The main and additional formulas are obtained from the continuous scheme developed through interpolation and collocation procedure. Numerical solutions obtained using the proposed method show that it is adequate and efficient for solving different kinds of problems. We conclude that the new method proposed in this article is more accurate and can compete favourably with some existing numerical methods for solving the problem in (1).

Acknowledgement

The authors are much grateful to a reviewer for his/her suggestions and constructive comments.

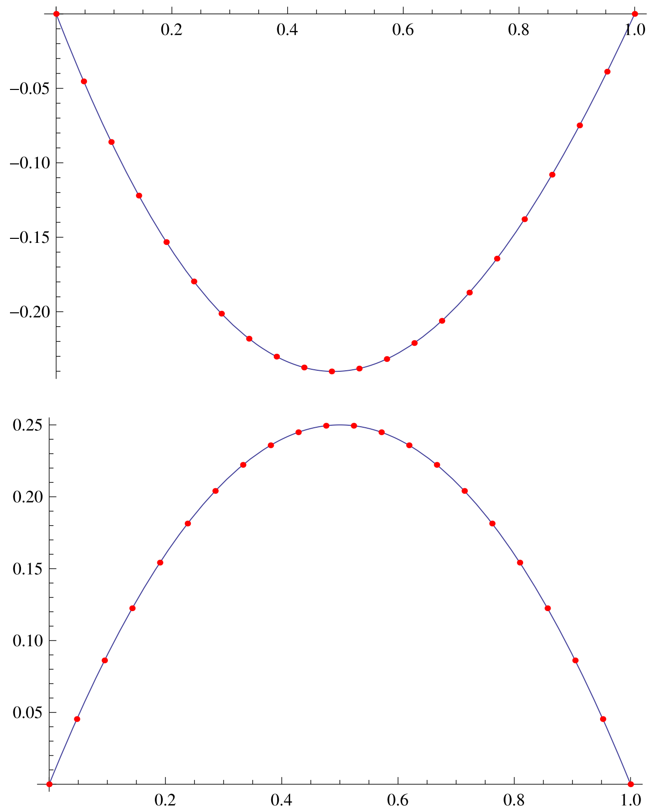

Fig. 4. Exact and discrete solutions (for $N = 21$) of $y_1(x)$ (top) and $y_2(x)$ (bottom).

References

[1] P. Amodio, I. Segura, High-order finite difference schemes for the solution of second-order BVPs, J. Comput. Appl. Math. 176 (2005) 59–76.

[2] U.M. Ascher, R.M.M. Mattheij, R.D. Russel, Numerical Solution of Boundary Value Problems for Ordinary Differential Equations, Prentice Hall, Englewood Cliffs, New York, 1988.

[3] D.O. Awoyemi, E.A. Adebile, A.O. Adesanya, T.A. Anake, Modified block method for the direct solution of second order ordinary differential equations, Int. J. Appl. Math. Comput. 3 (2011) 181–188.

[4] L. Brugnano, D. Trigiante, Solving Differential Problems by Multistep Initial and Boundary Value Methods, Gordon and Breach Science Publishers, 1998.

[5] N. Caglar, H. Caglar, B-spline method for solving linear system of second-order boundary value problems, Comput. Math. Appl. 57 (2009) 757–762.

[6] M. Carla, F. Mazzia, A. Sestini, H. Speleers, BS2 Methods for semi-linear second order boundary value problems, J. Appl. Math. Comput. 255 (2015) 147–156.

[7] 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.

[8] 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.

[9] E.A. Coddington, N. Levinson, Theory of Ordinary Differential Equations, in: International Series in Pure and Applied Mathematics, McGraw-Hill, New York, 1955.

[10] S. Cuomo, A. Marasco, A numerical approach to nonlinear two-point boundary value problems for ODEs, Comput. Math. Appl. 55 (2008) 2476–2489.

[11] M. Dehghan, A. Nikpour, Numerical solution of the system of second-order boundary value problems using the local radial basis functions based differential quadrature collocation method, Appl. Math. Model. 37 (2013) 8578–8599.

[12] S.O. Fatunla, Block methods for second order odes, Int. J. Comput. Math. 41 (1991) 55–63.

[13] D. Greenspan, V. Casulli, Numerical Analysis for Applied Mathematics, Science and Engineering, Addison-Wesley, 1988.

[14] S.N. Ha, A nonlinear shooting method for two-point boundary value problems, Comput. Math. Appl. 42 (2001) 1411–1420.

[15] Siraj-ul Islam, Imran Aziz, Bozidar Sarler, The numerical solution of second-order boundary-value problems by collocation method with the Haar wavelets, Math. Comput. Modelling 52 (2010) 1577–1590.
[16] M.K. Jain, T. Aziz, Cubic spline solution of two-point boundary value problems with signifigant first derivatives, Comput. Methods Appl. Mech. Engrg. 39 (1983) 83–91.
[17] H.B. Keller, Numerical Methods for Two-Point Boundary-Value Problems, Blaisdell, Waltham, Massachusetts, 1968.
[18] M. Lakestani, M. Dehgan, The solution of a second-order nonlinear differential equation with Neumann boundary conditions using semi-orthogonal B-spline wavelets, Int. J. Comput. Math. 83 (2006) 685–694.
[19] Feng-Gong Lang, Xiao-Ping Xu, Quintic B-spline collocation method for second order mixed boundary value problem, Comput. Phys. Comm. 183 (2012) 913–921.
[20] A. Lomtatidze, L. Malaguti, On a two-point boundary value problem for the second order ordinary differential equations with singularities, Nonlinear Anal. 52 (2003) 1553–1567.
[21] J. Lu, Variational iteration method for solving a nonlinear system of second-order boundary value problems, Comput. Math. Appl. 54 (2007) 1133–1138.
[22] A. Marasco, A. Romano, Scientific Computing with Mathematica: Mathematical Problems for Ordinary Differential Equations, in: Modeling and Simulation in Science, Engineering and Technology, Birkhäuser Boston, Inc., Boston, MA, 2001.
[23] H. Ramos, C. Lorenzo, Review of explicit Falkner methods and its modifications for solving special second-order I.V.P.s, Comput. Phys. Comm. 181 (2010) 1833–1841.
[24] H. Ramos, S. Mehta, J. Vigo-Aguiar, A unified approach for the development of k-step block Falkner-type methods for solving general second-order initial-value problems in ODEs, J. Comput. Appl. Math. 318 (2017) 550–564.
[25] H. Ramos, M.A. Rufai, Third derivative modification of k-step block Falkner methods for the numerical solution of second order initial-value problems, Appl. Math. Comput. 333 (2018) 231–245.
[26] H. Ramos, G. Singh, A note on variable step-size formulation of a Simpson's-type second derivative block method for solving stiff systems, Appl. Math. Lett. 64 (2017) 101–107.
[27] H. Ramos, G. Singh, A tenth order A-stable two-step hybrid block method for solving initial value problems of ODEs, Appl. Math. Comput. 310 (2017) 75–88.
[28] H. Ramos, G. Singh, V. Kanwar, S. Bhatia, An efficient variable step-size rational Falkner-type method for solving the special second-order IVP, Appl. Math. Comput. 291 (2016) 39–51.
[29] M.A. Rufai, E.A. Areo, An efficient one-eight step hybrid block method for solving second order initial value problems of ODEs, Int. J. Differ. Equ. Appl. 15 (2) (2016) 117–139.
[30] 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.
[31] R.A. Usmani, A method of high-order accuracy for the numerical integration of boundary value problems, BIT 13 (1973) 458–469.