Egy. J. Pure & Appl. Sci. 2024; 62(2):1-11

Egyptian Journal of Pure and Applied Science

A Reliable Numerical Treatment of Differential Equations via Hybrid Bernstein and Improved Block-Pulse Functions

Mohamed A. Ramadan [1*], Heba S. Osheba [1]

[1] Mathematics and Computer Science Department, Faculty of Science, Menoufia University, Egypt.

ARTICLE INFO

Received 31 December 2023
Accepted 22 January 2024

Keywords

Numerical application,
Bernstein polynomial,
Differential equation,
Improved block-pulse function,
Accuracy.

Correspondence

Mohamed A. Ramadan

E-mail*

(Corresponding Author)

ramadanmohamed13@yahoo.com

ABSTRACT

Many branches of practical mathematics rely heavily on numerical solutions to initial value problems, boundary value problems, and eigenvalue problems for ordinary and partial differential equations. Recently, the authors attempted to solve integral equations using hybrid Bernstein functions and improved block pulse functions. However, this is the first study to present a technical coupling between hybrid Bernstein and improved block-pulse functions for solving differential equations. The current method transforms differential equations into an algebraic system that can be solved with conventional methods. To validate the new method, certain numerical examples are supplied. The findings demonstrated that the method is both promising and highly accurate. The numerical findings reveal that the suggested hybrid approach outperforms the use of Bernstein polynomials, multi derivative hybrid block methods, block-pulse function, and other methods indicated in the numerical section in terms of accuracy. The proposed method can be implemented for more kinds of differential equations. The proposed method is applicable to a broader range of differential equations. The numerical results show that the proposed method are highly accurate and effective.

1. Introduction

Since ordinary differential equations (ODEs) are used to simulate a wide range of physical, biological, and social events, solving ODEs is crucial in many branches of research and engineering. We can forecast how these occurrences will change over time and comprehend their behavior by solving ODEs.

The ability to analyses a system's underlying dynamics even with sparse or insufficient data is one of the main advantages of solving ODEs. Solving ODEs may show how various variables interact and change over time in a system by capturing the relationships between them. Direct extraction of this information from experimental data is frequently challenging or impossible.

ODE solutions can also be used to control systems or forecast the results of certain operations. ODEs are utilized, for instance, in engineering to develop control systems that manage the actions of machinery or processes. ODEs are used in medicine to simulate the transmission of diseases and assess the efficacy of various therapies. In conclusion, the significance of solving ODEs lies in their ability to shed light on complicated system behavior that cannot be discovered from data collection alone. It enables us to assess the results of various actions, optimize systems, and forecast how a system will change over time. Despite the fact that multiple orthogonal and non-orthogonal polynomials have been created and applied to linear and nonlinear differential equations such as Chebyshev, Legendre, Jacobi, Bernstein, Bernoulli, Genocchi, Lucas, Laguerre, Hermite, and Bell.

In recent years, several researchers have proposed direct approaches for solving higher-order ODEs. Majid et al. [1] utilised Jocobi iteration and direct methods with varying step sizes to solve second-order ODEs. Awoyemi and Idowu [2] devised a type of hybrid collocation direct approach for solving third-order ODEs. Olabode and Yusuph [3] used power series collocation and interpolation to develop a three-step block technique for solving third-order ODEs. Waeleh et al. [4] devised a block technique based on numerical integration and interpolation to solve higher-order ODEs, and they gave fourth- and fifth-order approximations. Olabode and Alabi [5] proposed a linear multistep technique for solving fourth-order ODEs that employs interpolation and the collocation of power series approximation solutions.

There have been several studies on using Bernstein polynomials to solve differential equations. Yousefi and Behroozifar [6] employed operational matrices and Bernstein polynomials to solve differential equations. Pandey and Kumar [7] used Bernstein operational matrices to tackle Emden-type problems. Alshbool et al. [8] proposed Bernstein polynomial-based approximations for singular differential equations. Chen et al. proposed a numerical solution to the variable order linear cable equation using Bernstein polynomials [9]. Bellucci [10] introduced the orthonormal Bernstein polynomials, which can be used in a generalised Fourier series to approximate surfaces and curves.

Alshbool et al. [12] solved fractional-order differential equations, whereas Asgari and Ezzati [11] used an operational matrix of two-dimensional Bernstein polynomials to solve fractional-order integral equations. The Hybrid Bernstein polynomials (BPs) and Improved Block-Pulse (IBPFs) functions are introduced in [13]. An efficient computational method for finding a solution of different kind equation such as linear and nonlinear Fredholm Integral Equations [24], Second Kind Fuzzy Fredholm Integral Equations [25], Volterra–Fredholm integral equations [26] and system of linear Fredholm integral equations [27]. In [28-30] investigates some numerical methods for solving the integrated forms of third- and fifth-order differential equations.

In this article, we first show the relationship between the Bernstein polynomials and the enhanced block pulse basis. Using this relation, we created the operational matrix for integration and product of B-polynomials. They are used for solving ordinary differential equations. The current method turns an ordinary differential equation to a series of algebraic equations. We used the proposed method on three test issues and compared the results to the precise solutions and other ways, demonstrating that it is very effective and convenient. Then we utilise them to solve differential equations

$$\sum_{j=0}^{s} \rho_j(x) y^{(j)}(x) = g(x), \quad 0 \leq x \leq 1, \qquad (1.1)$$

With the initial condition

$$y^{(k)}(0) = b_k, \qquad , \quad 0 \leq k \leq s-1, \qquad (1.2)$$

Where $g(x)$ and $\rho_j(x), j = 0, \ldots, s$ are given functions and $y(x)$ is the unknown function to be determined. The key feature of this technique is that it lowers these equations to those of an easily solvable algebraic equation, significantly simplifying them.

The paper is structured as follows: Section 2 presents a combination of the Bernstein and Improved Block-Pulse functions. Section 3 presents an approach for numerically approximating linear and nonlinear differential equations using the HBIBP basis. The accuracy and applicability are determined by solving several linear and nonlinear differential equations. Finally, the last part summarizes the findings.

2. Hybrid Bernstein improved block-pulse functions (HBIBPFs) [13]

Definition 2.1: $HBIBP_{i,j}(x)$ is a combination of Bernstein polynomials and Improved Block-Pulse functions where all are orthogonal and complete, and then the set is a complete orthogonal system. $HBIBP_{i,j}(x)$ where $j = 0,1,\dots,M, i = 1,2,\dots,N+1$, $HBIBP_{i,j}(x)$ have two arguments i and j are the order of IBPFs and degree of BPs, respectively. $HBIBP(x)$ defined on the interval [0, 1) as follows:

$$HBIBP_{i,j}(x) = \begin{cases} B_{j,M}\left(\frac{2x}{h}\right) & , x \in \left[0, \frac{h}{2}\right), \\ 0 & , \text{ otherwise}, \end{cases} \quad for\ i = 1, j = 0,1,\dots,M. \tag{2.1}$$

$$HBIBP_{i,j}(x) = \begin{cases} B_{j,M}\left(\frac{x}{h}+\frac{3}{2}-i\right) & , x \in \left[(i-2)h+\frac{h}{2}, (i-1)h+\frac{h}{2}\right) \\ 0 & , \text{ otherwise}, \end{cases} for\ i = 2,3,\dots,N, j = 0,1,\dots,M. \tag{2.2}$$

$$HBIBP_{i,j}(x) = \begin{cases} B_{j,M}\left(\frac{2x}{h}-\frac{2}{h}+1\right) & , x \in \left[1-\frac{h}{2}, 1\right), \\ 0 & , \text{ otherwise}, \end{cases} for\ i = N+1, j = 0,1,\dots,M. \tag{2.3}$$

Thus, our new basis is $\{HBIBP_{1,0}, HBIBP_{1,1}, \dots, HBIBP_{N+1,M}\}$ and we can approximate function to the base function, where N is an arbitrary positive integer and $h = \frac{1}{N}$. In the next section we really are dealing with the problem of approximating such functions.

2.1. Function approximation

Using the HBIBP basis, a function u(x) can be represented as:

$$u(x) = \sum_{i=1}^{N+1}\sum_{j=0}^{M} c_{i,j}.HBIBP_{i,j}(x) = C^T HBIBP(x), \tag{2.4}$$

where

$$HBIBP(x) = [HBIBP_{1,0}, HBIBP_{1,1}, \dots, HBIBP_{N+1,M}]^T, \tag{2.5}$$

and

$$C = [c_{1,0}, c_{1,1}, \dots, c_{N+1,M}]^T, \tag{2.6}$$

we have

$$C^T < HBIBP(x), HBIBP(x) > = < u(x), HBIBP(x), \tag{2.7}$$

then

$$C = L^{-1} < u(x), HBIBP >, \tag{2.8}$$

where $\langle .,. \rangle$ is the standard inner product and L is an $((N+1)(M+1) \times (N+1)(M+1))$ matrix that is said the dual matrix that is

$$L = < HBIBP(x), HBIBP(x) > = \int_0^1 HBIBP(x).HBIBP^T(x)dx$$

$$= \begin{pmatrix} L_1 & 0 & 0 & \cdots & 0 \\ 0 & L_2 & 0 & \cdots & 0 \\ 0 & 0 & L_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & L_{n+1} \end{pmatrix}, \tag{2.9}$$

$L_i (i = 1,2,\dots,n+1)$ is defined as follows,

$$(L_1)_{i+1,j+1} = \int_0^{\frac{h}{2}} B_{i,M}\left(\frac{2x}{h}\right) B_{j,M}\left(\frac{2x}{h}\right)dx = \frac{h}{2}\int_0^1 B_{i,M}(x)B_{j,M}(x)dx = \frac{h\binom{M}{i}\binom{M}{j}}{2(2M+1)\binom{2M}{i+j}}, \quad for\ i,j = 0,\dots,M, \tag{2.10}$$

$$(L_r)_{i+1,j+1} = \int_{(i-2)h+\frac{h}{2}}^{(i-1)h+\frac{h}{2}} B_{i,M}\left(\frac{x}{h}+\frac{3}{2}-i\right) B_{j,M}\left(\frac{x}{h}+\frac{3}{2}-i\right)dx, \quad for\ r = 2,\dots,N$$

$$= h\int_0^1 B_{i,M}(x)B_{j,M}(x)dx = \frac{h\binom{M}{i}\binom{M}{j}}{(2M+1)\binom{2M}{i+j}}, \quad for\ i,j = 0,\dots,M, \tag{2.11}$$

$$(L_{N+1})_{i+1,j+1} = \int_{1-\frac{h}{2}}^{1} B_{i,M}\left(\frac{2x}{h}-\frac{2}{h}+1\right) B_{j,M}\left(\frac{2x}{h}-\frac{2}{h}+1\right)dx$$

$$= \frac{h}{2}\int_0^1 B_{i,M}(x)B_{j,M}(x)dx = \frac{h\binom{M}{i}\binom{M}{j}}{2(2M+1)\binom{2M}{i+j}}, \quad for\ i,j = 0,\dots,M, \tag{2.12}$$

We can also approximate the function $k(x,t) \in L^2([0,1]\times[0,1])$ as follow:

$$k(x,t) = HBIBP^T(x).K.HBIBP(t)$$

where K is an $(M+1)(N+1)$ matrix that we can obtain as follows:

$K = L^{-1}\langle HBIBP(x), \langle k(x,t), HBIBP(t)\rangle\rangle L^{-1}$.

2.2. Operational product matrix

Suppose that $C^T = [C_1^T, C_2^T, \dots, C_{N+1}^T]$ is an arbitrary $1\times(N+1)(M+1)$ matrix which C_i^T is $1\times(M+1)$ matrix for $i = 1,2,\dots,N+1$, then $\hat{C}$ is $(N+1)(M+1)\times(N+1)(M+1)$ operational matrix of product whenever

$$C^T HBIBP(x)HBIBP(x)^T \simeq HBIBP(x)^T\hat{C} \tag{2.13}$$

We know

$$C^T B(x)B(x)^T \simeq B(x)^T\hat{C}_i, \quad i = 1,2,\dots,N+1,$$

which $\hat{C}_i$ is operational matrix of product of Bernstein polynomials presented in [14, 15], then

$$C^T HBIBP(x)HBIBP(x)^T =$$

$$C^T\begin{bmatrix} HBIBP_{1,m}(x)HBIBP_{1,m}(x)^T & \bar{0} & \cdots & \bar{0} \\ \bar{0} & HBIBP_{2,m}(x)HBIBP_{2,m}(x)^T & \cdots & \bar{0} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{0} & \bar{0} & \cdots & HBIBP_{N+1,m}(x)HBIBP_{N+1,m}(x)^T \end{bmatrix}$$

$$= \begin{bmatrix} C_1 HBIBP_{1,m}(x)HBIBP_{1,m}(x)^T & \bar{0} & \cdots & \bar{0} \\ \bar{0} & C_2 HBIBP_{2,m}(x)HBIBP_{2,m}(x)^T & \cdots & \bar{0} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{0} & \bar{0} & \cdots & C_{N+1}HBIBP_{N+1,m}(x)HBIBP_{N+1,m}(x)^T \end{bmatrix}$$

$$= \begin{bmatrix} HBIBP_{1,m}(x)^T\hat{C}_1 & \bar{0} & \cdots & \bar{0} \\ \bar{0} & HBIBP_{2,m}(x)^T\hat{C}_2 & \cdots & \bar{0} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{0} & \bar{0} & \cdots & HBIBP_{N+1,m}(x)^T\hat{C}_{N+1} \end{bmatrix}$$

$$= HBIBP(x)^T\hat{C},$$

for $m = 0,\dots,M$, where

$$\hat{C}=\begin{bmatrix}\hat{C}_1 & \bar{0} & \dots & \bar{0}\\ \bar{0} & \hat{C}_2 & \dots & \bar{0}\\ \vdots & \vdots & \ddots & \vdots\\ \bar{0} & \bar{0} & \dots & \hat{C}_{N+1}\end{bmatrix},$$

with $\bar{0}$ is $(m+1)\times(m+1)$ matrix.

2.3. Operational Integration Matrix

HBIBP functions can be integrated using the coefficient matrix $\bar{P}$. The operational matrix for integration $\bar{P}$ is given by

$$\int_0^x HBIBP(t)dt \simeq \bar{P}HBIBP(x), \quad 0\le x\le 1 \tag{2.14}$$

where $\bar{P}$ is $(N+1)(M+1)$ square matrix and $HBIBP(x)$ is defined in Eq. (2.1)-(2.3). It is easy to see that:

$$\int_0^1 B_{i,m}(x)dx=\frac{1}{m+1}\ ,\ i=0,1,\dots,m$$

Then

$$\int_0^1 B_{i,m}(kx)dx=\frac{1}{k(m+1)}\ ,\ i=0,1,\dots,m.$$

On the other hand we know

$$\int_0^x B(t)dt\simeq PB(x)$$

which P is the operational matrix of integration of Bernstein function ($B(x)$) and Information on obtaining this matrix are given in [14] and [15].

$$\int_0^x HBIBP_{i,j}(t)dt=\left[\frac{P}{2N},\ \frac{\bar{1}}{2N(m+1)},\ \cdots\ \frac{\bar{1}}{2N(m+1)}\right]HBIBP(x),$$
$$for\ i=1, j=0,1,\dots,M$$

$$\int_0^x HBIBP_{i,j}(t)dt=\left[\underbrace{\bar{0},\ \cdots\ \bar{0},}_{i\ times}\ \frac{P}{N},\ \frac{\bar{1}}{N(m+1)},\ \cdots\ \frac{\bar{1}}{N(m+1)}\right]HBIBP(x),$$
$$for\ i=2,3,\dots,N, j=0,1,\dots,M\ .$$

$$\int_0^x HBIBP_{i,j}(t)dt=\left[\underbrace{\bar{0},\ \cdots\ \bar{0},}_{i\ times}\ \frac{P}{2N}\right]HBIBP(x)\,,$$
$$for\ i=N+1, j=0,1,\dots,M\,,$$

where $\bar{1}$ is a matrix $(M+1)\times(M+1)$ that all of its elements is one and $\bar{0}$ is the zero matrix of size $(M+1)\times(M+1)$.

Therefore, the operational matrix of integration P is obtained as follows:

Assume $A_1=\frac{P}{2N}$, $A_2=\frac{P}{N}$, $B_1=\frac{\bar{1}}{2N(m+1)}$ and $B_2=\frac{\bar{1}}{N(m+1)}$

$$\bar{P}=\begin{pmatrix}A_1 & B_1 & B_1 & \cdots & B_1\\ \bar{0} & A_2 & B_2 & \cdots & B_2\\ \bar{0} & \bar{0} & A_2 & \cdots & B_2\\ \vdots & \vdots & \vdots & \ddots & B_2\\ \bar{0} & \bar{0} & \bar{0} & \cdots & A_1\end{pmatrix}$$

2.4. Operational differentiation matrix

The operational matrix of differentiation $\bar{D}$ is given by

$$\frac{d(HBIBP(x))}{dx}=\bar{D}HBIBP(x)$$

We have

$$\frac{dB(x)}{dx}=DB(x).$$

Which D is the operational matrix of differentiation of $B(x)$ and information on obtaining this matrix is given in [14, 15].

$$\frac{dHBIBP_{i,j}(x)}{dx}=[2ND,\ \bar{0},\ \cdots\ \bar{0}]HBIBP(x),$$
$$for\ i=1, j=0,1,\dots,M$$

$$\frac{dHBIBP_{i,j}(x)}{dx}=\left[\underbrace{\bar{0},\ \cdots\ \bar{0},}_{i-1\ times}\ ND,\ \bar{0},\ \cdots\ \bar{0}\right]HBIBP(x),$$
$$for\ i=2,3,\dots,N, j=0,1,\dots,M.$$

$$\frac{dHBIBP_{i,j}(x)}{dx}=\left[\underbrace{\bar{0},\ \cdots\ \bar{0},}_{i-1\ times}\ 2ND\right]HBIBP(x)\,,$$
$$for\ i=N+1, j=0,1,\dots,M\,,$$

where $\bar{0}$ is a matrix $(M+1)\times(M+1)$ that all of its elements is zero.

So

$$\bar{D}=N\begin{pmatrix}2D & \bar{0} & \bar{0} & \dots & \bar{0}\\ \bar{0} & D & \bar{0} & \dots & \bar{0}\\ \bar{0} & \bar{0} & D & \dots & \bar{0}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ \bar{0} & \bar{0} & \bar{0} & \cdots & 2D\end{pmatrix}.$$

3. Solution Methodology and Illustrative examples

3.1. Solution Methodology

This section presents the derivation of the method for solving the linear differential equation (1.1) with the initial conditions Eq. (1.2). If we approximate $g(x), \rho_j(x),\ j=0,\dots,s$ and $y^{(s)}(x)$ as follows:

$$g(x)=G^T HBIBP(x) \tag{3.1}$$
$$\rho_j(x)=P_j{}^T HBIBP(x),\quad j=0,\dots,s,$$
$$y^{(s)}(x)=C^T HBIBP(x) \tag{3.2}$$

Where $G, P_j,\ j=0,\dots,s$ and C are the coefficients which are defined similarly to Eq. (2.8). With s-times integrating

from Eq. (3.2) with respect to x between $x=0$ to $x=x$, using Eq.(2.14) and the initial conditions Eq. (1.2), we will have

$$\begin{aligned} y^{(s-1)}(x) &= b_{s-1} + C^T \bar{P}\, HBIBP(x) \\ y^{(s-2)}(x) &= b_{s-2} + b_{s-1}x + C^T \bar{P}^{\,2} HBIBP(x) \\ &\vdots \\ y'(x) &= b_1 + b_2 x + \frac{b_3}{2!}x^2 + \cdots + \frac{b_{s-1}}{(s-2)!}x^{(s-2)} + C^T \bar{P}^{\,s-1} HBIBP(x) \\ y(x) &= b_0 + b_1 x + \frac{b_2}{2!}x^2 + \cdots + \frac{b_{s-1}}{(s-1)!}x^{(s-1)} + C^T \bar{P}^{\,s} HBIBP(x) \end{aligned} \tag{3.3}$$

Let

$$\begin{aligned} x^i &= d_i^T HBIBP(x)\,, \qquad i = 1,2,\dots,s-1, \\ b_{s-i} &= b_{s-i} E^T HBIBP(x)\,, \quad i = 1,2,\dots,s, \end{aligned} \tag{3.4}$$

Where

$$1 = E^T HBIBP(x), \tag{3.5}$$

Substituting Eq. (3.4) into Eq. (3.3), we have

$$\begin{aligned} y^{(s)}(x) &= C^T HBIBP(x) = Q_s^T HBIBP(x)\,, \\ y^{(s-1)}(x) &= (b_{s-1}E^T + C^T \bar{P}\,) HBIBP(x) = Q_{s-1}^T HBIBP(x), \\ y^{(s-2)}(x) &= (b_{s-2}E^T + b_{s-1}d_1^T + C^T \bar{P}^{\,2}) HBIBP(x) = Q_{s-2}^T HBIBP(x), \\ &\vdots \end{aligned} \tag{3.6}$$

$$\begin{aligned} y'(x) &= \left(b_1 E^T + b_2 d_1^T + \frac{b_3}{2!}d_2^T + \cdots + \frac{b_{s-1}}{(s-2)!}d_{s-2}^T + C^T \bar{P}^{\,s-1}\right) HBIBP(x) = Q_1^T HBIBP(x) \\ y(x) &= \left(b_0 E^T + b_1 d_1^T + \frac{b_2}{2!}d_2^T + \cdots + \frac{b_{s-1}}{(s-1)!}d_{s-1}^T + C^T \bar{P}^{\,s}\right) HBIBP(x) = Q_0^T HBIBP(x) \end{aligned} \tag{3.7}$$

Replacing Eq. (3.6) and Eq. (3.7) into (1.1), we obtain

$$\sum_{j=0}^{s} P_j^T\, HBIBP(x) HBIBP^T(x) Q_j = G^T HBIBP(x), \tag{3.8}$$

Using Eq. (2.13), we have

$$\sum_{j=0}^{s} HBIBP^T(x) \hat{P}_j Q_j = HBIBP^T(x) G\,, \tag{3.9}$$

Therefore, we get

$$\sum_{j=0}^{s} \hat{P}_j Q_j = G, \tag{3.10}$$

The unknown Vector C can be obtained by solving Eq. (3.10). once C is known, $y(x)$can be calculated from Eq. (3.7).

3.2. Illustrative examples

Example 1: Application to the Bessel differential equation [14].

Consider the following zero-order Bessel differential equation (O'Neil 1987).

$$xy'' + y' + xy = 0$$

with initial conditions $y(0) = 1,\ y'(0) = 0$.

The exact solution known as the Bessel function of the first kind of order zero denoted by $J_0(x)$

$$J_0(x) = \sum_{q=0}^{\infty} \frac{(-1)^q}{(q!)^2}\left(\frac{x}{2}\right)^{2q}$$

First, we approximate the unknown function $y''(x)$ by

$$y''(x) \simeq \sum_{i=1}^{N+1}\sum_{j=0}^{M} c_{i,j}.HBIBP_{i,j}(x) = C^T HBIBP(x)\,, \tag{3.11}$$

Using Eq. (2.14) and initial conditions we have

$$\begin{aligned} y'(x) &\simeq C^T \bar{P}\, HBIBP(x)\,, \\ y(x) &\simeq C^T \bar{P}^2\, HBIBP(x) + Z^T HBIBP(x) \end{aligned} \tag{3.12}$$

Where $1 \simeq Z^T HBIBP(x)$ and we can express function x as

$$x \simeq E^T HBIBP(x) \tag{3.13}$$

Substituting Eq. (3.11) – (3.13) in the Bessel differential equation we obtain

$$\begin{aligned} E^T HBIBP(x)\,.\,C^T HBIBP(x) &+ C^T \bar{P}\, HBIBP(x) \\ &+ E^T HBIBP(x)\,.\,[C^T \bar{P}^2\, HBIBP(x) \\ &+ Z^T HBIBP(x)] = 0 \end{aligned} \tag{3.14}$$

$$\begin{aligned} E^T HBIBP(x)\,.\,C^T HBIBP(x) &+ C^T \bar{P}\, HBIBP(x) \\ &+ E^T HBIBP(x)\,.\,C^T \bar{P}^2\, HBIBP(x) \\ &+ E^T HBIBP(x).\, Z^T HBIBP(x) \\ &= 0 \end{aligned} \tag{3.15}$$

Using Eq. (2.13) we have

$$E^T HBIBP(x) HBIBP(x)^T \simeq HBIBP(x)^T \hat{E} \tag{3.16}$$

Replacing Eq. (3.16) in Eq. (3.15) we get

$$\begin{aligned} HBIBP(x)^T \hat{E}\,.\,C + C^T \bar{P}\, HBIBP(x) \ &+\ HBIBP(x)^T \hat{E}\,.\,\bar{P}^{2^T} C \\ &+ HBIBP(x)^T \hat{E}\,.\,Z = 0 \end{aligned} \tag{3.17}$$

Or

$$\begin{aligned} HBIBP(x)^T \hat{E}\,.\,C + HBIBP(x)^T \bar{P}^{\,2} C & \\ +\ HBIBP(x)^T \hat{E}\,.\,\bar{P}^{2^T} C & \\ + HBIBP(x)^T \hat{E}\,.\,Z = 0 & \end{aligned} \tag{3.18}$$

Then

$$\hat{E}\,.\,C + \bar{P}^{\,2} C\ +\ \hat{E}\,.\,\bar{P}^{2^T} C + \hat{E}\,.\,Z = 0 \tag{3.19}$$

thus we have

$$\mathrm{C} = -\left(\hat{E}\ + \bar{P}^{\,2} +\ \hat{E}\,.\,\bar{P}^{2^T}\right)^{-1} \hat{E}\,.\,Z$$

Here, we solve the same problem using Hybrid Bernstein improved block-pulse functions (HBIBPFs), with $M = 3, N = 3$. In Table 1, a comparison is made between the approximate values using the present approach together with the exact solution of $J_0(x)$ and Bernstein-polynomials approach [14]. It is noted that the maximum error for this problem, obtained in [16] is 10^{-6} and in [14] is 10^{-14} $for\ \ m = 10$.

Example 2: Consider the nonlinear boundary layer equation [17, 18, 19]

$$2y'''(x) + y''(x)y(x) = 0, \tag{3.20}$$

with initial conditions $y(0) = 0, y'(0) = 0, y''(0) = A$.

This equation is well-known as the Blasius equation. Solving the Blasius equation yields the value y'''(0), which is used to evaluate shear stress at the plate. Blasius' equation has been solved using several approaches, such as series expansions, Runge Kutta, differential transformation, and others..

First, we approximate the unknown function $y'''(x)$ by

$$y'''(x) \simeq \sum_{i=1}^{N+1}\sum_{j=0}^{M} c_{i,j}.\,HBIBP_{i,j}(x) = C^T HBIBP(x)\,, \tag{3.21}$$

Using Eq. (2.14) and initial conditions we have

$$\begin{aligned} y''(x) &\simeq C^T \bar{P}\, HBIBP(x) + A \\ &= C^T \bar{P}\, HBIBP(x) + a^T HBIBP(x) \\ &= G^T HBIBP(x) \end{aligned} \tag{3.22}$$

Where we approximate A $\simeq a^T HBIBP(x)$ and

$$y'(x) \simeq C^T \bar{P}^2\, HBIBP(x) + a^T \bar{P}\, HBIBP(x) = Z^T HBIBP(x) \qquad (3.23)$$

$$y(x) \simeq C^T \bar{P}^3\, HBIBP(x) + a^T \bar{P}^2 HBIBP(x) = R^T HBIBP(x) \qquad (3.24)$$

Where $G = \bar{P}^T C + a$, $Z = \bar{P}^{2^T} C + \bar{P}^T a$ and $R = \bar{P}^{3^T} C + \bar{P}^{2^T} a$, then

Substituting Eq. (3.21) – (3.24) in the Blasius equation we obtain

$$2y'''(x) + y''(x)y(x) = 0,$$

$$2C^T HBIBP(x) + G^T HBIBP(x).R^T HBIBP(x) = 0$$

$$2C^T HBIBP(x) + G^T HBIBP(x).HBIBP(x)^T R = 0$$

Using Eq. (2.13) we have

$$G^T HBIBP(x) HBIBP(x)^T \simeq HBIBP(x)^T \hat{G} \ .$$

Then we get

$$2C^T HBIBP(x) + HBIBP(x)^T \hat{G} R = 0,$$

$$2HBIBP(x)^T C + HBIBP(x)^T \hat{G} R = 0,$$

thus we have

$$C = \frac{-1}{2} \hat{G} R.$$

Our methodology's findings are widely acknowledged (see Table 2) when compared to the hybrid block approach [17], the Bernstein polynomials method [18], and the Bernoulli polynomials operational matrix [19] for the constant A = 1.

Example 3: Next consider the following problem [20, 21 and 22]

$$y'''(x) + 5y''(x) + 7y'(x) + 3y = 0, \qquad (3.25)$$

with initial conditions $y(0) = 1, y'(0) = 0, y''(0) = -1$. The exact solution is $y = e^{-x} + xe^{-x}$

First, we approximate the unknown function $y'''(x)$ by

$$y'''(x) \simeq \sum_{i=1}^{N+1} \sum_{j=0}^{M} c_{i,j}. HBIBP_{i,j}(x) = C^T HBIBP(x) , \qquad (3.26)$$

Using Eq. (2.14) and initial conditions we have

$$y''(x) \simeq C^T \bar{P}\, HBIBP(x) - 1 = C^T \bar{P}\, HBIBP(x) - Z^T HBIBP(x) \qquad (3.27)$$

Where we approximate $1 \simeq Z^T HBIBP(x)$ and

$$y'(x) \simeq C^T \bar{P}^2\, HBIBP(x) - Z^T \bar{P}\, HBIBP(x) \qquad (3.28)$$

$$y(x) \simeq C^T \bar{P}^3\, HBIBP(x) - Z^T \bar{P}^2 HBIBP(x) + 1 = C^T \bar{P}^3\, HBIBP(x) - Z^T \bar{P}^2 HBIBP(x) + Z^T HBIBP(x) \qquad (3.29)$$

Substituting Eq. (3.26) – (3.29) in Eq. (3.25) we obtain

$$y'''(x) + 5y''(x) + 7y'(x) + 3y = 0,$$

$$C^T HBIBP(x) + 5\big(C^T \bar{P}\, HBIBP(x) - Z^T HBIBP(x)\big) + 7\big(C^T \bar{P}^2\, HBIBP(x) - Z^T \bar{P}\, HBIBP(x)\big) + 3\big(C^T \bar{P}^3\, HBIBP(x) - Z^T \bar{P}^2 HBIBP(x) + Z^T HBIBP(x)\big) = 0,$$

$$C^T(I + 5\bar{P} + 7\bar{P}^2 + 3\bar{P}^3) = 4Z^T + 7Z^T\bar{P} + 3Z^T\bar{P}^2$$

$$\left(I + 5\bar{P}^T + 7\bar{P}^{2^T} + 3\bar{P}^{3^T}\right) \mathrm{C} = 4Z^T + 7Z^T\bar{P} + 3Z^T\bar{P}^2$$

$$\mathrm{C} = \left(I + 5\bar{P}^T + 7\bar{P}^{2^T} + 3\bar{P}^{3^T}\right)^{-1} (4Z^T + 7Z^T\bar{P} + 3Z^T\bar{P}^2)$$

Here, we solve the same problem using Hybrid Bernstein improved block-pulse functions (HBIBPFs), with $M = 3, N = 3$. In Table 4, a comparison is made between Absolute Errors of the approximate values using the present approach Linear Multistep Hybrid Method [20], Bernstein polynomials [21] and Hybrid Block method [22].

Example 4: Consider the following problem [22, 23]

$$y'''(x) - y''(x) + y'(x) - y = 0, \qquad (3.30)$$

with initial conditions $y(0) = 1, y'(0) = 0, y''(0) = -1$. The exact solution is $y(x) = \cos(x)$.

First, we approximate the unknown function $y'''(x)$ by

$$y'''(x) \simeq \sum_{i=1}^{N+1} \sum_{j=0}^{M} c_{i,j}. HBIBP_{i,j}(x) = C^T HBIBP(x) , \qquad (3.31)$$

Using Eq. (2.14) and initial conditions we have

$$y''(x) \simeq C^T \bar{P}\, HBIBP(x) - 1 = C^T \bar{P}\, HBIBP(x) - Z^T HBIBP(x) \qquad (3.32)$$

Where we approximate $1 \simeq Z^T HBIBP(x)$ and

$$y'(x) \simeq C^T \bar{P}^2\, HBIBP(x) - Z^T \bar{P}\, HBIBP(x) \qquad (3.33)$$

$$y(x) \simeq C^T \bar{P}^3\, HBIBP(x) - Z^T \bar{P}^2 HBIBP(x) + 1 = C^T \bar{P}^3\, HBIBP(x) - Z^T \bar{P}^2 HBIBP(x) + Z^T HBIBP(x) \qquad (3.34)$$

Substituting Eq. (3.31) – (3.34) in Eq. (3.30) we obtain

$$y'''(x) - y''(x) + y'(x) - y = 0,$$

$$C^T HBIBP(x) - \big(C^T \bar{P}\, HBIBP(x) - Z^T HBIBP(x)\big) + \big(C^T \bar{P}^2\, HBIBP(x) - Z^T \bar{P}\, HBIBP(x)\big) - \big(C^T \bar{P}^3\, HBIBP(x) - Z^T \bar{P}^2 HBIBP(x) + Z^T HBIBP(x)\big) = 0,$$

$$C^T(I - \bar{P} + \bar{P}^2 - \bar{P}^3) = Z^T\bar{P} - Z^T\bar{P}^2$$

$$\left(I - \bar{P}^T + \bar{P}^{2^T} - \bar{P}^{3^T}\right) \mathrm{C} = Z^T\bar{P} - Z^T\bar{P}^2$$

$$\mathrm{C} = \left(I - \bar{P}^T + \bar{P}^{2^T} - \bar{P}^{3^T}\right)^{-1} (Z^T\bar{P} - Z^T\bar{P}^2)$$

Here, we solve this problem using Hybrid Bernstein improved block-pulse functions (HBIBPFs), with $M = 3, N = 3$. In Table 4, a comparison is made between Absolute Errors of the approximate values using the present approach, multi derivative hybrid block methods [23] and Hybrid Block method [22].

Table 1. Comparison of the absolute error for numerical results for Example 1 of the given approach with $M = 3$, $N = 3$, Bernstein polynomials method [14], and the exact solution of $J_0(x)$.

x	Exact solution	Bernstein-polynomials [14] for $m = 10$	Presented Method	Absolute Error	
				Bernstein- polynomials [14]	Presented Method
0	1.00000000000000	1.0000000000000038	1.00000000000000	3.774758284e-15	9.7144514655e-17
0.1	0.99750156206604	0.9975015620660412	0.99750156206604	1.221245327e-15	9.1593399532e-17
0.2	0.99002497223958	0.9900249722395755	0.99002497223958	4.440892099e-15	8.6042284408e-17
0.3	0.97762624653830	0.9776262465382969	0.97762624653830	3.108624469e-15	8.0491169285e-17
0.4	0.96039822665956	0.960398226659563	0.96039822665956	2.997602166e-15	7.4940054162e-17
0.5	0.93846980724081	0.9384698072408127	0.93846980724081	2.775557562e-15	6.9388939039e-17
0.6	0.91200486349721	0.9120048634972114	0.91200486349721	1.443289932e-15	6.3837823916e-17
0.7	0.88120088860741	0.8812008886074042	0.88120088860741	5.773159728e-15	5.8286708793e-17
0.8	0.84628735275048	0.8462873527504811	0.84628735275048	1.110223025e-15	5.2735593670e-17
0.9	0.80752379812255	0.8075237981225438	0.80752379812255	6.217248938e-15	4.7184478547e-17
1.0	0.76519768655797	0.7651976865579627	0.76519768655797	7.327471963e-15	3.9677949039e-17

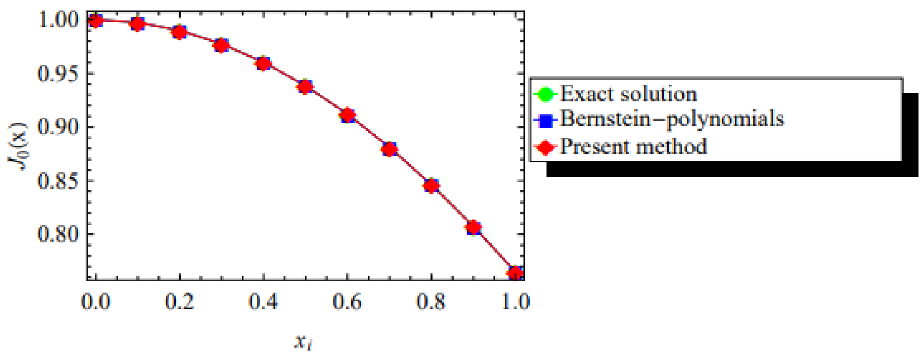

Fig. 1 Graphical solutions of Example 1

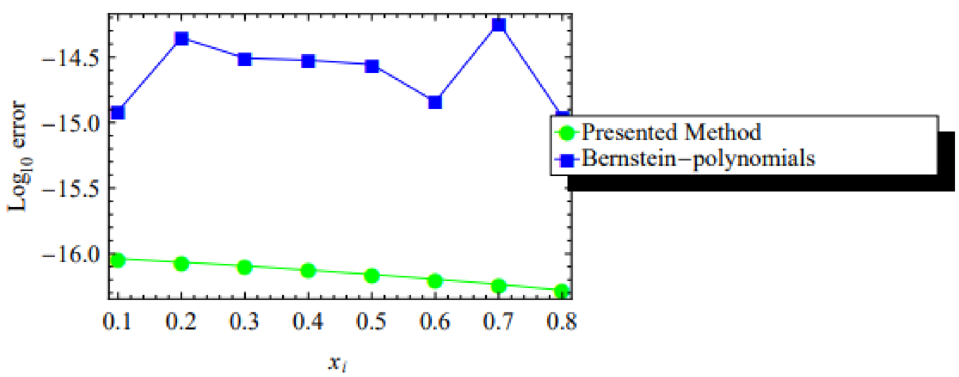

Fig. 2 Comparison of the absolute error for numerical results for Example1 for the presented method with Bernstein-polynomials presented in [14].

Table 2. Comparison of the approximate solution for Example 2 of the given approach with $M = 3$, $N = 3$, Hybrid block method [17], Bernstein polynomials method [18], and Bernoulli polynomials [19].

x	**Hybrid block method** [17]	**Bernstein polynomials** [18]	**Bernoulli polynomials** [19]	**Presented Method**
0.1	0.004999979	0.004999958	0.005	0.0049999583341723
0.2	0.019998667	0.019998667	0.02	0.0199986668419935
0.3	0.044998481	0.044989879	0.045	0.0449898794745896
0.4	0.079991467	0.079957378	0.08	0.0799573779857994
0.5	0.124967454	0.124870058	0.125	0.1248700575229549
0.6	0.179902837	0.179677141	0.18	0.179677141245484
0.7	0.244755068	0.244303617	0.245	0.244303616982151
0.8	0.319454501	0.318646009	0.32	0.318646009310246
0.9	0.403894871	0.402568621	0.405	0.402568620552525
1.0	0.497922483	0.495900383	0.5	0.495900382783151

Table 3. Comparison of the absolute error for numerical results for Example 3 for the presented method with $\mathrm{M} = 3$, $\mathrm{N} = 3$.with other three numerical methods presented in [20, 21, 22]

x	**Exact solution**	**Absolute Error**			
		Linear Multistep Hybrid Method [20]	**Bernstein polynomials** [21]	**Hybrid Block method** [22]	**Presented method**
0.1	1.000000000	1.00×10^{-10}	7.42×10^{-17}	6.4300E-08	$9.7144514655 \times 10^{-17}$
0.2	0.995321160	3.00×10^{-10}	3.01×10^{-16}	2.7200E-08	$9.1593399532 \times 10^{-17}$
0.3	0.982476904	7.00×10^{-10}	6.11×10^{-16}	3.0500E-08	$8.6042284408 \times 10^{-17}$
0.4	0.963063687	7.00×10^{-10}	9.48×10^{-16}	8.9800E-08	$8.0491169285 \times 10^{-17}$
0.5	0.938448064	6.00×10^{-10}	1.26×10^{-15}	4.4260E-07	$7.4940054162 \times 10^{-17}$
0.6	0.909795990	2.00×10^{-10}	1.42×10^{-15}	7.7260E-07	$6.9388939039 \times 10^{-17}$
0.7	0.878098618	9.00×10^{-10}	1.11×10^{-16}	1.9523E-06	$6.3837823916 \times 10^{-17}$
0.8	0.844195016	2.80×10^{-9}	5.78×10^{-16}	1.0274E-06	$5.8286708793 \times 10^{-17}$
0.9	0.808792135	5.40×10^{-9}	5.93×10^{-15}	1.3509E-06	$5.2735593670 \times 10^{-17}$
1.0	0.772482354	3.50×10^{-9}	2.02×10^{-14}	1.3470E-05	$4.47184478547 \times 10^{-17}$

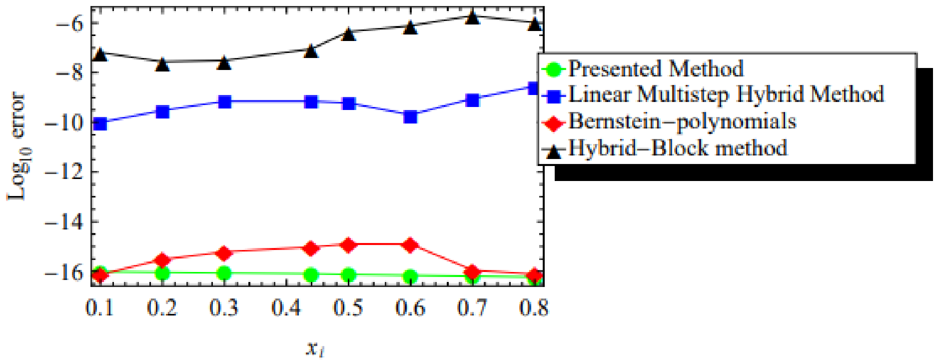

Fig. 3 Comparison of the absolute error for numerical results for Example 3 for the presented method with other three numerical methods presented in [20, 21, 22].

Table 4. Comparison of the absolute error for numerical results for Example 4 for the presented method with $M = 3$, $N = 3$. With other three numerical methods presented in [22, 23].

x	Exact solution	Absolute Error		
		Multi derivative Hybrid block methods [23]	Hybrid Block method [22]	Presented method
0.01	0.999950000	1.60168E-05	6.72000E-07	2.305072e-13
0.02	0.999800007	1.100991E-04	1.34410E-06	2.279087e-13
0.03	0.999550034	5.567153E-04	2.01700E-06	1.351249e-13
0.04	0.999200107	1.6332403E-03	2.68840E-06	5.823823e-14
0.05	0.998750260	3.62018361E-03	3.35940E-06	4.585688e-13

x	Exact solution	Absolute Error			Relative error		
		Multi derivative Hybrid block methods [23]	Hybrid Block method [22]	Presented method	Multi derivative Hybrid block methods [23]	Hybrid Block method [22]	Presented method
0.01	0.999950000	1.601682E-05	6.72000E-07	2.305072e-13	1.6018e-05	6.7203e-07	2.3052e-13
0.02	0.999800007	1.100991E-04	1.34410E-06	2.279087e-13	1.1012e-04	1.3444e-06	2.2795e-13
0.03	0.999550034	5.567153E-04	2.01700E-06	1.351249e-13	5.5697e-04	2.0179e-06	1.3519e-13
0.04	0.999200107	1.633243E-03	2.68840E-06	5.823823e-14	1.63324e-03	2.6906e-06	5.8285e-14
0.05	0.998750260	3.620183E-03	3.35940E-06	4.585688e-13	3.62018e-03	3.3636e-06	4.5914e-13

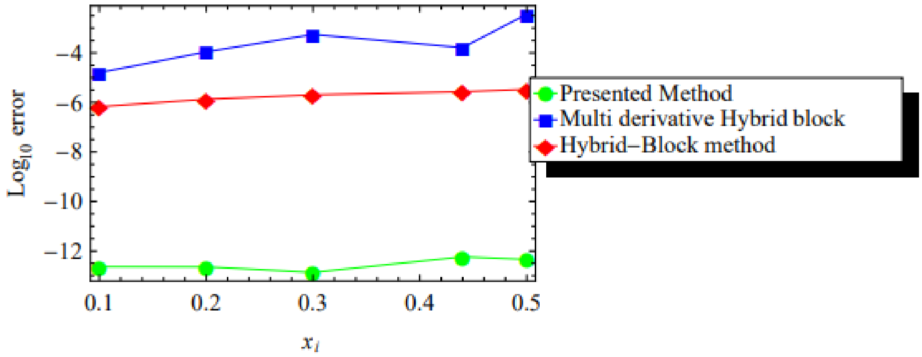

Fig. 4 Comparison of the absolute error for numerical results for Example 4 for the presented method with $M = 3,\ N = 3$. With other three numerical methods presented in [22, 23].

4. Conclusion

To address several second and third order initial value issues of ODEs, the hybrid Bernstein and enhanced block-pulse functions method introduced by Ramadan and Osheba [9] is employed and developed. The current method transforms differential equations into an algebraic system that can be solved using standard methods. The numerical results show that the derived method outperforms some of the other methods addressed in this paper. Based on the numerical results, the method appears to be very promising for managing more general equations that the authors are investigating, such as nonlinear differential equations and oscillator differential equations, which play critical roles in natural and physical simulations.

5. References

1. **M Z.A., Majid., N.A., Azmi., and M., Suleiman. (2009).** Solving second order ordinary differential equations using two point four step direct implicit block method. Eur. J. Sci. Res**. 31, 29–36.**
2. **D., Awoyemi., and O., Idowu. (2005).** A class of hybrid collocation methods for third-order ordinary differential equations. Int. J. Comput. Math. **82, 1287–1293.**
3. **B., Olabode., and Y., Yusuph. (2009).** A new block method for special third order ordinary differential equations. J. Math. Stat. **5, 167–170.**
4. **N., Waeleh., Z., Abdul Majid., F., Ismail., and M. Suleiman. (2011).** Numerical solution of higher order ordinary differential equations by direct block code. J. Math. Stat. **8, 77–81.**
5. **B., T.Olabode., T., Alabi. (2013).** Direct block predictor-corrector method for the solution of general fourth order odes. J. Math. Res. **5, 26–33.**
6. **S., Yousefi., and M., Behroozifar. (2010).** Operational matrices of Bernstein polynomials and their applications. Int. J. Syst. Sci. **41, 709–716.**
7. **R.K., Pandey., and N., Kumar. (2012).** Solution of Lane-Emden type equations using Bernstein operational matrix of differentiation. New Astron. **17, 303–308.**
8. **M., Alshbool., A., Bataineh., I., Hashim., and O.R., Isik. (2015).** Approximate solutions of singular differential equations with estimation error by using Bernstein polynomials. Int J Pure Appl Math. **100, 109–125.**
9. **Y., Chen., L., Liu., B., Li., and Y., Sun. (2014).** Numerical solution for the variable order linear cable equation with Bernstein polynomials. Appl. Math. Comput. **238, 329–341.**
10. **M. A., Bellucci. (2014).** On the explicit representation of orthonormal Bernstein polynomials. arXiv preprint arXiv:1404.2293. Bibliographic Code: arXiv: 1404.2293 [math.CA], Cornell University Library. 13 pages.
11. **M., Asgari., and R., Ezzati. (2017).** Using operational matrix of two-dimensional Bernstein polynomials for solving two-dimensional integral equations of fractional order. Appl. Math. Comput. **307, 290–298.**
12. **M.H.T., Alshbool., A.S., Bataineh., I., Hashim., and O.R., Isik. (2017).** Solution of fractional order differential equations based on the operational matrices of new fractional Bernstein functions. J. King Saud. Univ. Sci. **29, 1–18.**

13. **M. A., Ramadan., and H. S., Osheba. (2020).** A new hybrid orthonormal Bernstein and improved block-pulse functions method for solving mathematical physics and engineering problems, Alexandria Engineering Journal. **59(5) 3643-3652.**
14. **S. A., Yousefi., and M., Behroozifar. (2010).** Operational matrices of Bernstein polynomials and their applications, International Journal of Systems Science, **Vol.41, No.6, June, 709-716.**
15. **M., Behroozifar., and S. A., Yousefi. (2013).** Numerical solution of delay differential equations via operational matrices of hybrid of block-pulse functions and Bernstein polynomials, Computational Methods for Differential Equations **1.2, 78-95.**
16. **M., Razzaghi., and S., Yousefi. (2001).** The Legendre Wavelets Operational Matrix of Integration, International Journal of Systems Science, **32(4), 495–502.**
17. **A.O., Adesanya., D.M., Udoh., and A.M., Ajileye. (2013).** A new hybrid block method for the solution of general third order initial value problems of ordinary differential equations. International Journal of Pure and Applied Mathematics, **Vol. 86 No. 2, 365-375.**
18. **SN., Khataybeh., I., Hashim., and M., Alshbool. (2018).** Solving directly third-order ODEs using operational matrices of Bernstein polynomials method with applications to fluid flow equations. J King Saud Univ Sci: **1–5.**
19. **D., Rani., and V., Mishra. (2020).** Numerical inverse Laplace transform based on Bernoulli polynomials operational matrix for solving nonlinear differential equations, Results in Physics **16: 102836.**
20. **A.M., Sagir. (2014).** On the approximate solution of continuous coefficients for solving third order ordinary differential equations, International Journal of Mathematical, Computational Science and Engineering, **8(3), 39- 43.**
21. **Hashim., Ishak., and Mohammed., Alshbool. (2019).** Solving directly third-order ODEs using operational matrices of Bernstein polynomials method with applications to fluid flow equations. Journal of King Saud University-Science **31.4: 822-826.**
22. **Mohammed., Umaru., and Raphael Babatunde., Adeniyi. (2014).** A three step implicit hybrid linear multistep method for the solution of third order ordinary differential equations.
23. **A.M., Badmus., and Y.A., Yahaya. (2009).** Some multi derivative hybrid block methods for solution of general third order ordinary differential equations, Nigerian Journal of Scientific Research, **8, 103-107.**
24. **Mohamed A., Ramadan., and Heba S., Osheba. (2020).** A New Hybrid Orthonormal Bernstein and Improved Block-Pulse Functions Method for Solving Mathematical Physics and Engineering Problems, Alexandria Engineering Journal Volume **59**, Issue **5**, October, Pages **3643-3652,** https://doi.org/10.1016/j.aej.2020.06.014.
25. **Praveen., Agarwal., Mohamed A., Ramadan., Heba S., Osheba., Yu-Ming., Chu. (2020).** Study of Hybrid Orthonormal Functions Method for solving Second Kind Fuzzy Fredholm Integral Equations, Advances in Difference Equations, Published: 29 September 2020, **533,** https://doi.org/10.1186/s13662-020-02985-3.
26. **Mohamed A., Ramadan., Heba S., Osheba., and Adel R., Hadhoud. (2023).** A numerical method based on Hybrid Orthonormal Bernstein and Improved Block-Pulse Functions for Solving Volterra–Fredholm integral equations, Numerical Methods for Partial Differential Equations, accepted Dec**.**
27. **Ji-Huan., He., Mahmoud H., Taha., Mohamed A., Ramadan., and Galal M., Moatimid. (2022).** A combination of Bernstein and improved block-pulse functions for solving a system of linear Fredholm integral equations, Mathematical Problems in Engineering, accepted Feb.
28. **E.H., Doha., W. M., Abd-Elhameed., and Y. H., Youssri. (2015).** New algorithms for solving third-and fifth-order two point boundary value problems based on nonsymmetric generalized Jacobi Petrov–Galerkin method. *Journal of Advanced Research* **6.5: 673-686.**
29. **W.M., Abd-Elhameed., E.H., Doha., and Y.H., Youssri. (2013).** Efficient spectral-Petrov-Galerkin methods for third-and fifth-order differential equations using general parameters generalized Jacobi polynomials. *Quaestiones Mathematicae*, ***36*(1), 15-38.**
30. **H.E., Doha., W.M., Abd-Elhameed., and Y. H., Youssri. (2012).** Efficient spectral-Petrov–Galerkin methods for the integrated forms of third-and fifth-order elliptic differential equations using general parameters generalized Jacobi polynomials. *Applied Mathematics and Computation* **218.15: 7727-7740.**