

Article

Spectral Solutions of Even-Order BVPs Based on New Operational Matrix of Derivatives of Generalized Jacobi Polynomials

Waleed Mohamed Abd-Elhameed ^{1,*}, Badah Mohamed Badah ², Amr Kamel Amin ^{3,*}
and Muhammad Mahmoud Alsuyuti ⁴

¹ Department of Mathematics, Faculty of Science, Cairo University, Giza 12613, Egypt

² Department of Mathematics, College of Science, University of Jeddah, Jeddah 23218, Saudi Arabia

³ Department of Basic Sciences, Adham University College, Umm AL-Qura University, Makkah 28653, Saudi Arabia

⁴ Department of Basic Science, Egyptian Academy for Engineering and Advanced Technology Affiliated to Ministry of Military Production, Cairo 32511, Egypt

* Correspondence: waleed@cu.edu.eg (W.M.A.-E.); akgadelrab@uqu.edu.sa (A.K.A.)

Abstract: The primary focus of this article is on applying specific generalized Jacobi polynomials (GJPs) as basis functions to obtain the solution of linear and non-linear even-order two-point BVPs. These GJPs are orthogonal polynomials that are expressed as Legendre polynomial combinations. The linear even-order BVPs are treated using the Petrov–Galerkin method. In addition, a formula for the first-order derivative of these polynomials is expressed in terms of their original ones. This relation is the key to constructing an operational matrix of the GJPs that can be used to treat the non-linear two-point BVPs. In fact, a numerical approach is proposed using this operational matrix of derivatives to convert the non-linear differential equations into effectively solvable non-linear systems of equations. The convergence of the proposed generalized Jacobi expansion is investigated. To show the precision and viability of our suggested algorithms, some examples are given.

Keywords: Legendre polynomials; generalized Jacobi polynomials; operational matrix; Petrov–Galerkin method; collocation method

MSC: 65M70; 11B83; 35L02



Citation: Abd-Elhameed, W.M.; Badah, B.M.; Amin, A.K.; Alsuyuti, M.M. Spectral Solutions of Even-Order BVPs Based on New Operational Matrix of Derivatives of Generalized Jacobi Polynomials. *Symmetry* **2023**, *15*, 345. <https://doi.org/10.3390/sym15020345>

Academic Editor: Juan Luis García Guirao

Received: 30 December 2022

Revised: 21 January 2023

Accepted: 23 January 2023

Published: 26 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The numerous applications of special functions and orthogonal polynomials in a variety of fields have made studying these polynomials more crucial. They occur in the study of differential and integral equations; for an illustration, see for example [1–4]. Furthermore, orthogonal polynomials have been shown to be significant in both mathematical statistics and quantum physics. Many theoretical investigations regarding special functions were performed, see for example [5–8]. The classical orthogonal polynomials, which involve the Hermite, Laguerre, and Jacobi polynomials, are the orthogonal polynomials that are the most utilized polynomials, see for example [9–11]. The Jacobi polynomials are among the most significant orthogonal polynomials used in numerical analysis. The most significant class of Jacobi polynomials is known as the Gegenbauer polynomials class, which also includes the classes of Legendre and Chebyshev polynomials of the first and second kinds as special classes. The four kinds of Chebyshev polynomials are all special Jacobi polynomials. The first and second kinds of Chebyshev polynomials are ultraspherical polynomials, whereas the third and fourth-kind Chebyshev polynomials are not ultraspherical polynomials since they are special cases of certain non-symmetric Jacobi polynomials, see for example [12,13].

BVPs play important roles since they may be found everywhere in the field of applied sciences, from engineering to fluid mechanics to optimization theory. For some applications, one can consult [14]. High-order BVPs can be used to describe a variety of real-world phenomena. The even-order two-point BVPs appear in a variety of problems. A fourth-order ordinary differential equation [15] governs the free vibration analysis of beam structures. An ordinary differential equation of sixth-order controls the vibrational activity of rings, see [16]. For some other applications to even-order BVPs, one can refer to [17]. From a numerical perspective, there are several numerical algorithms utilized to solve different types of even-order BVPs. For example, the authors in [18] treated both linear and non-linear two-point BVPs of any even order using the generalized third-kind Chebyshev polynomials. The linear equations were solved via the Galerkin approach, while the non-linear even-order BVPs were treated by applying the standard collocation technique, based on a matrix of derivatives of the generalized third-kind Chebyshev polynomials. Some other algorithms in the literature were developed to treat such types of polynomials. Among these methods are the differential transform methods in [19], perturbation and homotopy perturbation methods in [20], and the matrix method in [21].

The numerical analysis relies heavily on spectral methods. Numerical solutions to differential and integral equations have been obtained successfully using these methods. The required approximate solution can be expressed in terms of certain combinations of orthogonal polynomials that can serve as basis functions by using spectral methods. Tau, collocation, and Galerkin methods are three well-known spectral methods. The Galerkin approach relies on selecting orthogonal polynomial combinations that meet the underlying conditions and after that enforcing the residual of the equation to be orthogonal to the set of test functions that coincide with the set of trial functions, see for example [10,22–25]. The “Petrov–Galerkin” method is a variation of the Galerkin method. The primary distinction between the Galerkin and Petrov–Galerkin techniques is that, in contrast to the Galerkin method, the two sets of trial and test functions in the Petrov–Galerkin approach are not the same. Therefore, the Galerkin approach is less adaptable than the Petrov–Galerkin method. In comparison to the Galerkin and Petrov–Galerkin techniques, the tau method is more widely used, for an example, see [26,27]. This is because of the flexibility with which the basis and trial functions can be chosen. The collocation method, which can handle all different types of differential equations, is the most popular approach, see for instance, [28–31]. A survey of spectral techniques and their uses can be found in [2,32].

Two papers written by Shen [33,34] addressed the issue of combining orthogonal polynomials in order to deal with different types of differential equations. In [33], using the spectral Galerkin method, the author numerically treated the second and fourth-order two-point BVPs by using orthogonal combinations of Legendre polynomials. The fundamental benefit of choosing such combinations is that it allows one to transform the differential equations governed by their underlying conditions into sets of algebraic systems that are specially structured. Additionally, it has been demonstrated that some types of differential equations can be transformed into diagonal systems, which naturally considerably reduces the computational effort needed to solve these particular types of differential equations, see for example [24].

In the context of numerical analysis, the utilization of operational matrices of derivatives and integrals is beneficial. They are used to numerically solve almost all types of differential equations. For example, in [35], Napoli and Abd-Elhameed used harmonic numbers operational matrices of derivatives to treat the non-linear high-order initial value problems. A wide range of fractional differential equations can be solved using operational derivative matrices, see for example [36,37].

The principal objective of the current article is to employ the GJPs as basis functions to obtain spectral solutions of the linear and non-linear even-order BVPs. Two different approaches are utilized for proposing spectral solutions for these equations. The Petrov–Galerkin method is applied to treat the linear even-order BVPs. For the non-linear BVPs, the operational matrix of derivatives of such polynomials is first introduced, and after

that, it is used to transform the differential equations governed by its governing boundary conditions (BCs) into systems of algebraic equations that can be efficiently solved.

The paper is organized as follows. The next section presents some preliminary information and some fundamental properties concerning the Legendre and GJPs. Section 3 develops new formulas concerned with the GJPs and their shifted ones. In Section 4, a numerical algorithm built on the application of the Petrov–Galerkin method is designed to solve the even-order two-point BVPs. Section 5 is devoted to presenting a matrix approach for handling the non-linear two-point BVPs. This algorithm is basically built on the application of the spectral collocation method. The convergence analysis of the proposed shifted generalized Jacobi polynomials is investigated in Section 6. Some illustrative examples are displayed in Section 7. Some concluding remarks are given in Section 8.

2. Preliminaries and Some Interesting Formulas

In this section, we discuss the basic characteristics of Legendre polynomials, their shifted polynomials, and a special class of polynomials called GJPs, which are represented as specific combinations of Legendre polynomials. We also introduce the shifted generalized Jacobi polynomials that are used later on.

2.1. An Account on Legendre Polynomials and the GJPs

Legendre polynomials are well-known to form an orthogonal set of polynomials on $[-1, 1]$ with regard to the unit weight function. Its orthogonality relation is given as

$$\int_{-1}^1 P_m(x) P_n(x) dx = \begin{cases} \frac{2}{2n + 1}, & m = n, \\ 0, & m \neq n. \end{cases}$$

These polynomials can be represented as ([38]):

$$P_n(x) = \frac{1}{2^n} \sum_{m=0}^{\lfloor \frac{n}{2} \rfloor} \frac{(-1)^m (2n - 2m)!}{m! (n - 2m)! (n - m)!} x^{n-2m},$$

and the formula for its inversion takes the form

$$x^\ell = 2^{-\ell} \sqrt{\pi} \ell! \sum_{m=0}^{\lfloor \frac{\ell}{2} \rfloor} \frac{\frac{1}{2} + \ell - 2m}{m! \Gamma(\frac{3}{2} + \ell - m)} P_{\ell-2m}(x), \quad \ell \geq 0,$$

where $\lfloor z \rfloor$ represents the largest number that is less than or equal to z .

The shifted Legendre polynomials on $[a, b]$ can be defined as follows:

$$P_n^*(x) = P_n\left(\frac{2x - a - b}{b - a}\right).$$

It is obvious that $P_n^*(x)$ are orthogonal to $[a, b]$ in the following way:

$$\int_a^b P_m^*(x) P_n^*(x) dx = \begin{cases} \frac{b - a}{2n + 1}, & m = n, \\ 0, & m \neq n. \end{cases} \tag{1}$$

Guo et al. in [39] constructed kinds of orthogonal polynomials that are considered the natural basis of certain even-order two-point BVPs. He called them generalized Jacobi polynomials—“GJPs”. It was shown in [39] that these polynomials are combinations of the Legendre polynomials. Doha et al. in [24] found an explicit expression for these polynomials in terms of Legendre polynomials. More recently, Abd-Elhameed in [40] studied these polynomials from a theoretical point of view, and he established several formulas concerned with these polynomials and their connections with different orthogonal

polynomials. In this paper, we employ these polynomials to treat the linear and non-linear even-order two-point BVPs numerically.

Let ℓ and m be any two integers and define the SGJPs as follows (see [39]):

$$G_r^{(\ell,m)}(x) = \begin{cases} (1-x)^{-\ell} (1+x)^{-m} R_{r-r_0}^{(-\ell,-m)}(x), & r_0 = -(\ell+m), \ell, m \leq -1, \\ (1-x)^{-\ell} R_{\ell-r_0}^{(-\ell,m)}(x), & r_0 = -\ell, \ell \leq -1, m > -1, \\ (1+x)^{-m} R_{\ell-r_0}^{(\ell,-m)}(x), & r_0 = -m, \ell > -1, m \leq -1, \\ R_{\ell-r_0}^{(\ell,m)}(x), & r_0 = 0, \ell, m > -1, \end{cases} \tag{2}$$

where $R_m^{(\delta,\mu)}(x)$ are the normalized Jacobi polynomials defined as ([40]):

$$R_m^{(\delta,\mu)}(x) = \frac{m!}{(\delta+1)_m} P_m^{(\delta,\mu)}(x),$$

where $P_m^{(\delta,\mu)}(x)$ represents the well-known classical Jacobi polynomials.

From the definition in (2), it is clear that the polynomials defined as

$$G_k^n(x) = G_k^{(-n,-n)}(x) = (1-x^2)^n R_k^{(n,n)}(x), \tag{3}$$

fulfill the $(2n)$ BCs:

$$D^q G_k^n(\pm 1) = 0, \quad q = 0, 1, \dots, n-1.$$

Remark 1. The polynomials defined in (3) are those defined in [40], but they differ only in a factor.

In terms of Legendre polynomials, the following lemma shows how the GJPs can be written.

Lemma 1. The GJPs defined in (3) can be expressed in terms of Legendre polynomials as ([24])

$$G_k^n(x) = \frac{1}{2} n! \sum_{j=0}^n \frac{(-1)^j (1+4j+2k) \binom{n}{j} \Gamma\left(\frac{1}{2} + j + k\right)}{\Gamma\left(\frac{3}{2} + j + k + n\right)} P_{k+2j}(x). \tag{4}$$

Now, from the orthogonality relation of the normalized symmetric Jacobi polynomials (ultraspherical polynomials) ([40]), it is clear that the orthogonality relation of the polynomials $G_k^n(x)$ over $[-1, 1]$ with regard to the weight function $w(x) = (1-x^2)^{-n}$ is given by

$$\int_{-1}^1 w(x) G_j^n(x) G_k^n(x) dx = \begin{cases} \frac{2^{2n+1} k! (n!)^2}{(2k+2n+1)(k+2n)!}, & j = k, \\ 0, & j \neq k. \end{cases}$$

2.2. The Shifted Generalized Jacobi Polynomials

The GJPs that are defined in (3) can be extended to be defined on a general interval $[a, b]$. In this respect, the shifted generalized Jacobi polynomials (SGJPs) on $[a, b]$ can be defined as follows:

$$SG_j^n(x) = (b-x)^n (x-a)^n \tilde{R}_j^{(n,n)}(x), \tag{5}$$

where $\tilde{R}_j^{(n,n)}(x)$ are the shifted normalized symmetric Jacobi polynomials on $[a, b]$, defined as

$$\tilde{R}_j^{(n,n)}(x) = R_j^{(n,n)}\left(\frac{2x-a-b}{b-a}\right).$$

The orthogonality relation of the GJPs on $[-1, 1]$ can be easily transformed to give the corresponding orthogonality relation for the SGJPs on $[a, b]$. More definitely, the following orthogonality relation holds for the shifted polynomials $SG_j^n(x)$:

$$\int_a^b \tilde{w}(x) SG_j^n(x) SG_k^n(x) dx = \begin{cases} \frac{(b-a)^{2n+1} k! (n!)^2}{(2k+2n+1)(k+2n)!}, & j = k, \\ 0, & j \neq k, \end{cases} \tag{6}$$

where $\tilde{w}(x)$ is given by $\tilde{w}(x) = (b-x)^{-n} (x-a)^{-n}$.

The following corollary exhibits the expression of the SGJPs in terms of the shifted Legendre polynomials.

Corollary 1. *In terms of shifted Legendre polynomials, the SGJPs can be represented as*

$$SG_k^n(x) = \frac{n!(b-a)^{2n}}{2^{2n+1}} \sum_{j=0}^n \frac{(-1)^j (4j+2k+1) \binom{n}{j} \Gamma(j+k+\frac{1}{2})}{\Gamma(j+k+n+\frac{3}{2})} P_{k+2j}^*(x). \tag{7}$$

Proof. Formula (7) can be deduced from Formula (4) only by replacing x by $\frac{2x-a-b}{b-a}$. \square

3. Some New Formulas Concerned With the GJPs and Their Shifted Ones

This section is devoted to establishing some new formulas concerned with the GJPs and their shifted polynomials, which will be very useful to derive our two proposed algorithms to handle even-order linear and non-linear two-point BVPs.

In the first theorem, we state and prove an expression of the power form representation of the GJPs.

Theorem 1. *The analytic form of $G_j^n(x)$ is given by*

$$G_j^n(x) = \frac{(2n+1)!}{\Gamma(\frac{3}{2}+n)} \sum_{p=0}^{\lfloor \frac{j}{2} \rfloor + n} \frac{(-1)^{n-p} 2^{j-2p-1} \Gamma(\frac{1}{2}+j+n-p)}{p!(j+2n-2p)!} x^{j+2n-2p}, \quad j \geq 0.$$

Proof. The analytic form of $R_j^{(n,n)}(x)$ (see [40]) allows one to write

$$R_j^{(n,n)}(x) = \frac{j!(2n+1)!}{(j+2n)! \Gamma(\frac{3}{2}+n)} \sum_{r=0}^{\lfloor \frac{j}{2} \rfloor} \frac{(-1)^r 2^{j-2r-1} \Gamma(\frac{1}{2}+j+n-r)}{(j-2r)! r!} x^{j-2r},$$

and therefore, we have

$$G_j^n(x) = \frac{j!(2n+1)!}{(j+2n)! \Gamma(\frac{3}{2}+n)} \sum_{r=0}^{\lfloor \frac{j}{2} \rfloor} \frac{(-1)^r 2^{j-2r-1} \Gamma(\frac{1}{2}+j+n-r)}{(j-2r)! r!} (1-x^2)^n x^{j-2r}.$$

The previous formula is transformed into the following form by using the binomial theorem:

$$G_j^n(x) = \frac{j!(2n+1)!}{(j+2n)! \Gamma(\frac{3}{2}+n)} \sum_{r=0}^{\lfloor \frac{j}{2} \rfloor} \frac{(-1)^r 2^{j-2r-1} \Gamma(\frac{1}{2}+j+n-r)}{(j-2r)! r!} \sum_{\ell=0}^n (-1)^\ell \binom{n}{\ell} x^{j+2\ell-2r}. \tag{8}$$

Some algebraic computations on (8) lead to the following formula:

$$G_j^n(x) = \frac{j!(2n+1)!}{(j+2n)!\Gamma(\frac{3}{2}+n)} \sum_{p=0}^{\lfloor \frac{j}{2} \rfloor + n} (-1)^{n-p} \sum_{\ell=0}^p \frac{2^{j-2\ell-1} \binom{n}{\ell+n-p} \Gamma(\frac{1}{2}+j-\ell+n)}{(j-2\ell)!\ell!} x^{j+2n-2p}.$$

The second right-hand sum in the previous formula can now be written as

$$\sum_{\ell=0}^p \frac{2^{j-2\ell-1} \binom{n}{\ell+n-p} \Gamma(\frac{1}{2}+j-\ell+n)}{(j-2\ell)!\ell!} = \frac{2^{j-1} \binom{n}{n-p} \Gamma(j+n+\frac{1}{2})}{j!} {}_3F_2 \left(\begin{matrix} -p, \frac{1}{2} - \frac{j}{2}, -\frac{j}{2} \\ \frac{1}{2} - j - n, 1 + n - p \end{matrix} \middle| 1 \right). \tag{9}$$

The ${}_3F_2(1)$ that appears in (9) can be reduced using the identity of Pfaff–Saalschutz (see, [41]) to give

$${}_3F_2 \left(\begin{matrix} -p, \frac{1}{2} - \frac{j}{2}, -\frac{j}{2} \\ \frac{1}{2} - j - n, 1 + n - p \end{matrix} \middle| 1 \right) = \frac{(2n+j-2p+1)_{2p}}{2^{2p} (j+n-p+\frac{1}{2})_p (n-p+1)_p},$$

and as a result, the power form representation of $G_j^n(x)$ shown below can be obtained:

$$G_j^n(x) = \frac{(2n+1)!}{\Gamma(\frac{3}{2}+n)} \sum_{p=0}^{\lfloor \frac{j}{2} \rfloor + n} \frac{(-1)^{n-p} 2^{j-2p-1} \Gamma(\frac{1}{2}+j+n-p)}{p!(j+2n-2p)!} x^{j+2n-2p}.$$

Theorem 1 is now proved. \square

Our next objective is to present and prove a significant theorem that gives a Legendre-based expression for the high-order derivatives of $G_j^n(x)$.

Theorem 2. Let j and q be two non-negative integers with $j+2n \geq q$. The q th-derivative of the GJPs has the following Legendre expression:

$$D^q G_j^n(x) = (-1)^n 2^q n! \Gamma\left(\frac{1}{2}+j+n\right) \times \sum_{r=0}^{\lfloor \frac{j-q}{2} \rfloor + n} \frac{(-1)^r \binom{j+2n-2r-q}{r} (1+n-r-q)_r}{r! \Gamma(\frac{3}{2}+j+2n-r-q) \left(\frac{1}{2}+j+n-r\right)_r} P_{j+2n-q-2r}(x). \tag{10}$$

Proof. The analytic form of $G_j^n(x)$ in Theorem 1 allows one to write

$$D^q G_j^n(x) = \frac{n!}{\sqrt{\pi}} \sum_{m=0}^{\lfloor \frac{j}{2} \rfloor + n} \frac{(-1)^{n-m} 2^{j-2m+2n} \Gamma(\frac{1}{2}+j-m+n)}{m!(j-2m+2n-q)!} x^{j-2m+2n-q}. \tag{11}$$

The application of the inversion formula of the Legendre polynomials turns Formula (11) into the following formula:

$$D^q G_j^n(x) = 2^q n! \sum_{m=0}^{\lfloor \frac{j}{2} \rfloor + n} \frac{(-1)^{n-m} \Gamma(\frac{1}{2}+j-m+n)}{m!} \times \sum_{r=0}^{\lfloor \frac{j-q}{2} \rfloor + n - m} \frac{\binom{j-2m+2n-q-2r}{r}}{r! \Gamma(\frac{3}{2}+j-2m+2n-q-r)} P_{j-2m+2n-q-2r}(x),$$

which can be rewritten as follows:

$$D^q G_j^n(x) = 2^{q-1} n! \sum_{r=0}^{\lfloor \frac{j-q}{2} \rfloor + n} (1 + 2j + 4n - 4r - 2q) \times \sum_{\ell=0}^r \frac{(-1)^{-\ell+n} \Gamma\left(\frac{1}{2} + j - \ell + n\right)}{\ell! (r - \ell)! \Gamma\left(\frac{3}{2} + j - \ell + 2n - r - q\right)} P_{j-q-2r+2n}(x).$$

Now, noting the identity

$$\sum_{\ell=0}^r \frac{(-1)^{-\ell+n} \Gamma\left(\frac{1}{2} + j - \ell + n\right)}{\ell! (r - \ell)! \Gamma\left(\frac{3}{2} + j - \ell + 2n - r - q\right)} = \frac{(-1)^n \Gamma\left(j + n + \frac{1}{2}\right)}{r! \Gamma\left(\frac{1}{2}(3 + 2j + 4n - 2r - 2q)\right)} {}_2F_1\left(-r, -\frac{1}{2} - j - 2n + r + q \mid \frac{1}{2} - j - n \mid 1\right),$$

then it is easy using the Chu–Vandermonde identity ([41]) to express $D^q G_j^n(x)$ as

$$D^q G_j^n(x) = (-1)^n 2^q n! \Gamma\left(\frac{1}{2} + j + n\right) \times \sum_{r=0}^{\lfloor \frac{j-q}{2} \rfloor + n} \frac{(-1)^r \left(\frac{1}{2} + j + 2n - 2r - q\right) (1 + n - r - q)_r}{r! \Gamma\left(\frac{3}{2} + j + 2n - r - q\right) \left(\frac{1}{2} + j + n - r\right)_r} P_{j+2n-q-2r}(x).$$

This proves Theorem 2. □

Now, the shifted Legendre polynomials can be used to represent the high-order derivatives of the $SG_j^n(x)$ as in the next corollary.

Corollary 2. *Let j and q be two non-negative integers with $j + 2n \geq q$. The q th-derivative of the $SG_j^n(x)$ can be represented in terms of the shifted Legendre polynomials as*

$$D^q SG_j^n(x) = 2^{2q-2n} (b - a)^{2n-q} n! \Gamma\left(j + n + \frac{1}{2}\right) \times \sum_{r=0}^{\lfloor \frac{j-q}{2} \rfloor + n} \frac{(-1)^{n+r} \left(j + 2n - q - 2r + \frac{1}{2}\right) (n - q - r + 1)_r}{r! \Gamma\left(j + 2n - q - r + \frac{3}{2}\right) \left(j + n - r + \frac{1}{2}\right)_r} P_{j+2n-q-2r}^*(x). \tag{12}$$

Proof. Formula (12) can be deduced as a direct consequence of Formula (10) by replacing x by $\frac{2x - a - b}{b - a}$. □

4. Treating Linear High Even-Order Differential Equations via the Petrov–Galerkin Method

The focus of this section is on thoroughly analyzing a spectral solution to the high-even order BVPs presented below:

$$(-1)^n D^{2n} u(x) + \sum_{q=1}^{2n-1} \zeta_q d_q D^q u(x) + d_0 u(x) = g(x), \quad x \in I = [a, b], \quad n \geq 1, \tag{13}$$

governed by the BCs:

$$D^k u(a) = D^k u(b) = 0, \quad k = 0, 1, \dots, n - 1, \tag{14}$$

where $\{d_q, q = 0, 1, \dots, 2n - 1\}$ are arbitrary real constants and ζ_q are defined as

$$\zeta_q = \begin{cases} (-1)^{\frac{q}{2}}, & q \text{ even,} \\ (-1)^{\frac{q+1}{2}}, & q \text{ odd.} \end{cases}$$

Now, we are willing to solve (13)–(14) utilizing the SGJPs that are defined in (5) as basis functions. So, we select the following basis functions:

$$\psi_{j,n}(x) = SG_j^n(x) = (b - x)^n (x - a)^n \bar{R}_j^{(n,n)}(x), \quad j = 0, 1, 2, \dots \tag{15}$$

It should be noticed that $\{\psi_{j,n}(x) : j = 0, 1, 2, \dots\}$ are linearly independent and orthogonal with respect to the weight function $\tilde{w}(x) = (b - x)^{-n} (x - a)^{-n}$.

Consider the Sobolev spaces to be $H^n(I) (n = 0, 1, 2, \dots)$, where the inner product is denoted by $(\cdot, \cdot)_n$, and the norm by $\|\cdot\|_n$ (see, [42]), and consider the following space

$$H_0^n(I) = \{u \in H^n(I) : u^{(q)}(a) = u^{(q)}(b) = 0, \quad 0 \leq q \leq n - 1\}, \tag{16}$$

where $u^{(q)}(x) = \frac{d^q u}{dx^q}$. Now, define the following subspace of $H_0^n(I)$

$$V_N = \text{span}\{\psi_{0,n}(x), \psi_{1,n}(x), \dots, \psi_{N,n}(x)\}.$$

4.1. Function Approximation

Assume now that the function $u(x) \in H_0^n(I)$ (defined in (16)) can be expanded in terms of the polynomials $\psi_{j,n}(x)$ as

$$u(x) = \sum_{j=0}^{\infty} c_j \psi_{j,n}(x),$$

where

$$c_j = \frac{(2j + 2n + 1)(j + 2n)!}{(b - a)^{2n+1} j! (n!)^2} \int_a^b \frac{u(x) \psi_{j,n}(x)}{(x - a)^n (b - x)^n} dx.$$

Additionally, assume an approximate solution $u_N(x) \in V_N(x)$ to (13)–(14) that can be expanded as

$$u_N(x) = \sum_{j=0}^N c_j \psi_{j,n}(x). \tag{17}$$

In the following subsection, we present how to obtain the proposed numerical solution $u_N(x)$ using a suitable spectral method.

4.2. Petrov–Galerkin Approximation to (13)–(14)

In this section, we are interested in applying the shifted generalized Jacobi Petrov–Galerkin method (SGJPGM) to solve (13)–(14). One should follow the concept of picking two distinct sets of trial and test functions. The test functions are selected at random, while the trial functions are selected to fulfill the BCs (14) and after that make the residual of (13) orthogonal to the selected test functions. It is appropriate to select the set of shifted Legendre polynomials as the test functions since it is evident from Formula (7) that the

basis functions are given in terms of the shifted Legendre polynomials. In this regard, we pick the following test functions:

$$\chi_k(x) = P_k^*(x). \tag{18}$$

To apply the SGJPGM to solve (13)–(14), we have to find $u_N(x) \in V_N(x)$ such that

$$\left((-1)^n D^{2n} u_N(x), P_k^*(x) \right) + \sum_{q=1}^{2n-1} \xi_q d_q (D^q u_N(x), P_k^*(x)) + d_0 (u_N(x), P_k^*(x)) = (g(x), P_k^*(x)), \tag{19}$$

where $(u(x), F(x)) = \int_a^b u(x) F(x) dx$ is the scalar inner product in the space $L^2[a, b]$.

Now, if we denote

$$\begin{aligned} g_k &= (g(x), P_k^*(x)), & \mathbf{g} &= (g_0, g_1, \dots, g_N), \\ B^{(2n)} &= (b_{kj}^{(2n)})_{0 \leq k, j \leq N}, & B^{(q)} &= (b_{kj}^{(q)})_{0 \leq k, j \leq N}, \quad 1 \leq q \leq 2n - 1, \\ B^{(0)} &= (b_{kj}^{(0)})_{0 \leq k, j \leq N}, \end{aligned}$$

then the matrix form corresponding to (19) is

$$\left(B^{(2n)} + \sum_{q=1}^{2n-1} d_q B^{(q)} + d_0 B^{(0)} \right) \mathbf{c} = \mathbf{g}, \tag{20}$$

where the nonzero elements of the matrices $B^{(2n)}, B^{(q)}, 1 \leq q \leq 2n - 1$ and $B^{(0)}$ are given explicitly in the following theorem.

Theorem 3. *If the trial and test basis functions $\psi_{j,n}(x)$ and $\chi_k(x)$ are as selected in (15) and (18), and if we denote $b_{kj}^{(2n)} = (-1)^n (D^{2n} \psi_{j,n}(x), P_k^*(x))$, $b_{kj}^{(q)} = \xi_q (D^q \psi_{j,n}(x), P_k^*(x))$, $1 \leq q \leq 2n - 1$, $b_{kj}^{(0)} = (\psi_{j,n}(x), P_k^*(x))$, then the nonzero elements of the matrices $B^{(2n)}, B^{(q)}, 1 \leq q \leq 2n - 1$ and $B^{(0)}$ are given explicitly as follows:*

$$b_{kj}^{(2n)} = \frac{2^{2n-1} n (b-a) \left(\frac{j-k}{2} + n - 1 \right)! \Gamma \left(\frac{1}{2} (j+k+1) + n \right)}{\left(\frac{j-k}{2} \right)! \Gamma \left(\frac{1}{2} (j+k+3) \right)}, \quad j \geq k, (j+k) \text{ even}, \tag{21}$$

$$\begin{aligned} b_{kj}^{(q)} &= \xi_q \frac{(-1)^n 2^q n! \left(\frac{b-a}{2} \right)^{2n-q+1} \left(\frac{1}{2} (j-k+q) - 1 \right)! \Gamma \left(\frac{1}{2} (1+j+k+q) \right)}{\left(\frac{1}{2} (j-k+2n-q) \right)! \Gamma \left(\frac{1}{2} (3+j+k+2n-q) \right) (q-n-1)!}, \\ &(j+k+q) \text{ even}, (j-2n+q) \geq k, 1 \leq q \leq 2n - 1, \end{aligned} \tag{22}$$

$$b_{kj}^{(0)} = \frac{(-1)^{\frac{k-j}{2}} \left(\frac{b-a}{2} \right)^{2n+1} \binom{n}{\frac{k-j}{2}} n! \Gamma \left(\frac{1}{2} (1+j+k) \right)}{\Gamma \left(\frac{1}{2} (3+j+k) + n \right)}, \quad 0 \leq k-j \leq 2n. \tag{23}$$

Proof. To compute the elements of the matrices $B^{(2n)}, B^{(q)}, 1 \leq q \leq 2n - 1$ and $B^{(0)}$, it is required to compute $(D^q \psi_{j,n}(x), P_k^*(x))$. Based on Formula (12), $D^q \psi_{j,n}(x)$ has the following expression:

$$D^q \psi_{j,n}(x) = \sum_{r=0}^{\lfloor \frac{j-q}{2} \rfloor + n} A_{r,j,q} P_{j+2n-q-2r}^*(x), \tag{24}$$

where $A_{r,j,q}$ take the form

$$A_{r,j,q} = \frac{(-1)^{n+r} 2^q n! \left(\frac{b-a}{2}\right)^{2n-q} \Gamma\left(\frac{1}{2} + j + n\right) \left(\frac{1}{2} + j + 2n - 2r - q\right) (1 + n - r - q)_r}{r! \Gamma\left(\frac{3}{2} + j + 2n - r - q\right) \left(\frac{1}{2} + j + n - r\right)_r}.$$

Now, Formula (24) along with the well-known orthogonality property of the shifted Legendre polynomials in (1) will yield

$$(D^q \psi_{j,n}(x), P_k^*(x)) = \sum_{r=0}^{\lfloor \frac{j-q}{2} \rfloor + n} A_{r,j,q} \delta_{j+2n-q-2r,k} h_k, \tag{25}$$

where $h_k = \frac{b-a}{2k+1}$ and $\delta_{j,k}$ is the well-known Kronecker delta function.

It is easy to see that Formula (25) reduces to the following formula:

$$(D^q \psi_{j,n}(x), P_k^*(x)) = \begin{cases} \frac{(-1)^n 2^q n! \left(\frac{b-a}{2}\right)^{2n-q+1} \left(\frac{1}{2}(j-k+q) - 1\right)! \Gamma\left(\frac{1}{2}(1+j+k+q)\right)}{\left(\frac{1}{2}(j-k+2n-q)\right)! \Gamma\left(\frac{1}{2}(3+j+k+2n-q)\right) (q-n-1)!}, & (j-k+q) \text{ even, } j-2n+q \geq k, \\ 0, & \text{otherwise.} \end{cases} \tag{26}$$

Now, regarding the entries of the matrix $B^{(2n)}$, it is clear from Formula (26) that

$$b_{kj}^{(2n)} = (-1)^n (D^{2n} \psi_{j,n}(x), P_k^*(x)) = \begin{cases} \frac{2^{2n-1} n (b-a) \left(\frac{j-k}{2} + n - 1\right)! \Gamma\left(\frac{1}{2}(j+k+1) + n\right)}{\left(\frac{j-k}{2}\right)! \Gamma\left(\frac{1}{2}(j+k+3)\right)}, & j \geq k, (j+k) \text{ even,} \\ 0, & \text{otherwise,} \end{cases}$$

and this proves Formula (21).

Now, it can be seen that Formula (22) is an immediate consequence of Formula (26). Formula (23) is an immediate result of Formula (7) together with the orthogonality relation (6). □

Remark 2. Equation (13) governed by the non-homogeneous BCs turns into homogeneous ones using a suitable transformation, see [18].

Remark 3. Some of the characteristics and advantages of the SGJPGM that are used to solve (13)–(14) can be listed as follows:

- The Petrov–Galerkin method is used to turn the linear even-order BVPs in (13)–(14) into linear systems of equations (20) that can be efficiently solved.
- For some particular two-point BVPs (13), and in particular for $d_q = 0$, the system in (20) reduces to an upper triangular system that can be easily solved. This, of course, gives an advantage when applying this method to these types of equations.

Remark 4. In the following Algorithm 1, we write the steps required to obtain the solution of (13)–(14) using SGJPGM.

Algorithm 1: Required steps to solve (13)–(14) by the SGJPGM

- Step 1.** Choose the basis functions $\psi_{j,n}(x)$ and $\chi_k(x)$ as in (15) and (18).
 - Step 2.** Apply the Petrov-Galerkin method to (13)–(14) to obtain the variational formulation in (19).
 - Step 3.** Convert (19) into its corresponding matrix system (20).
 - Step 4.** Compute the unknown vector c in (20) by a suitable numerical solver.
 - Step 5.** Obtain the numerical solution: $u_N(x) = \sum_{j=0}^N c_j \psi_{j,n}(x)$.
-

5. Treatment of the Non-Linear Differential Equations via a Matrix Approach

This section is devoted to analyzing and implementing in detail the algorithm designed to solve the non-linear even-order two-point BVPs. Now, consider the following non-linear, $2n$ th-order BVPs:

$$Z^{(2n)}(x) = F\left(x, Z(x), Z^{(1)}(x), Z^{(2)}(x), \dots, Z^{(2n-1)}(x)\right), \quad x \in I, \tag{27}$$

governed by the homogenous BCs:

$$Z^{(i)}(a) = Z^{(i)}(b) = 0, \quad i = 0, 1, \dots, n - 1. \tag{28}$$

We consider an approximation to the solution $Z(x)$ in Equation (27) in the form:

$$Z(x) \simeq Z_N(x) = \sum_{i=0}^N c_i \psi_i(x) = \mathbf{C}^T \mathbf{\Psi}(x), \tag{29}$$

where

$$\mathbf{C}^T = [c_0, c_1, \dots, c_N], \quad \mathbf{\Psi}(x) = [\psi_{0,n}(x), \psi_{1,n}(x), \dots, \psi_{N,n}(x)]^T.$$

We tackle (27)–(28) by expressing the derivatives in terms of their original ones. This can be done via the operational matrix method. Thus, the main idea behind the derivation of our algorithm is based on utilizing the operational matrix of the SGJPs that will be employed as basis functions. In the following subsection, we derive this new operational matrix of derivatives.

5.1. New Operational Matrix of Derivatives Based on the SGJPs

This section is devoted to introducing a new operational matrix of derivatives of the SGJPs. This matrix serves to solve the linear and non-linear even-order two-point BVPs using a unified approach.

We now present and demonstrate the fundamental theorem, which enables the introduction of a new operational derivative matrix.

Theorem 4. *If the polynomials $\psi_{i,n}(x)$ are selected as in (15), then the following relation holds for all $i \geq 1$,*

$$D\psi_{i,n}(x) = \frac{2}{b-a} \sum_{\substack{j=0 \\ (i+j) \text{ odd}}}^{i-1} (2n + 2j + 1) \psi_{j,n}(x) + \rho_i(x), \tag{30}$$

with

$$\rho_i(x) = n(b-x)^{n-1}(x-a)^{n-1} \begin{cases} a+b-2x, & i \text{ even,} \\ a-b, & i \text{ odd.} \end{cases} \tag{31}$$

Proof. Without any loss of generality, we consider the case corresponds to $I \equiv [-1, 1]$. In this case, Formula (30) turns into the following formula:

$$D\phi_{i,n}(x) = \sum_{\substack{j=0 \\ (i+j) \text{ odd}}}^{i-1} (2n + 2j + 1) \phi_{j,n}(x) + \theta_i(x), \tag{32}$$

where $\phi_{i,n}(x)$ is given by

$$\phi_{i,n}(x) = (1 - x^2)^n R_i^{(n,n)}(x),$$

and $\theta_i(x)$ is given by

$$\theta_i(x) = 2n(-1)^n (x^2 - 1)^{n-1} \begin{cases} x, & i \text{ even,} \\ 1, & i \text{ odd.} \end{cases}$$

This formula was stated and proved in [40], but by taking Remark 1 into consideration.

Now, replacing x by $\left(\frac{2x - a - b}{b - a}\right)$ in Formula (32) yields the desired result. \square

Now, and with the aid of Theorem 4, one can deduce that $\frac{d\Psi(x)}{dx}$ has the following expression:

$$\frac{d\Psi(x)}{dx} = S \Psi(x) + \rho(x), \tag{33}$$

where $\rho(x) = (\rho_0(x), \rho_1(x), \dots, \rho_N(x))^T$, and $S = (s_{ij})_{0 \leq i, j \leq N}$ is a $(N + 1) \times (N + 1)$ matrix whose nonzero entries can be explicitly determined from relation (30) as

$$s_{ij} = \begin{cases} \frac{2}{b - a} (2n + 2j + 1), & i > j, (i + j) \text{ odd,} \\ 0, & \text{otherwise.} \end{cases} \tag{34}$$

As an example, for $N = 6$ and $n = 4$, we have

$$S = \frac{2}{b - a} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 9 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 11 & 0 & 0 & 0 & 0 & 0 \\ 9 & 0 & 13 & 0 & 0 & 0 & 0 \\ 0 & 11 & 0 & 15 & 0 & 0 & 0 \\ 9 & 0 & 13 & 0 & 17 & 0 & 0 \\ 0 & 11 & 0 & 15 & 0 & 19 & 0 \end{pmatrix}_{7 \times 7}.$$

Corollary 3. The q th-derivative of the vector $\Psi(x)$ is given by

$$\frac{d^q \Psi(x)}{dx^q} = S^q \Psi(x) + \sum_{m=0}^{q-1} S^{q-m-1} \frac{d^m \rho(x)}{dx^m}. \tag{35}$$

Proof. The repeated application of Formula (33) yields the expression in (35). \square

5.2. Our Proposed Matrix Method for Treating (27)–(28)

This section is confined to presenting a matrix approach for treating (27)–(28). The key of our proposed approach is based on employing the operational matrix of derivatives of the SGJPs that are derived in Section 5.1. More definitely, the shifted generalized Jacobi operational matrix method (SGJOMM) is employed to treat (27)–(28).

Now, if $Z(x)$ is approximated as in (29), then based on Formula (33), the derivatives of the approximate solution $Z_N(x)$ given by (29) can be represented as

$$\left. \begin{aligned} Z'_N(x) &= C^T (S \Psi(x) + \rho(x)), \\ Z''_N(x) &= C^T (S^2 \Psi(x) + \rho_2(x)), \\ &\vdots \\ Z_N^{(2n-1)}(x) &= C^T (S^{2n-1} \Psi(x) + \rho_{2n-1}(x)), \\ Z_N^{(2n)}(x) &= C^T (S^{2n} \Psi(x) + \rho_{2n}(x)), \end{aligned} \right\} \tag{36}$$

where S is the operational matrix of derivatives whose elements are given explicitly in (34), and the vector $\rho_q(x)$ is given by the following formula:

$$\rho_q(x) = \sum_{m=0}^{q-1} S^{q-m-1} \frac{d^m \rho(x)}{dx^m}, \quad 2 \leq q \leq 2n,$$

where (31) provides the components of the vector $\rho(x)$.

To find the residual of Equation (27), we use the representations in (36) to give

$$\begin{aligned} R(x) &= C^T (S^{2n} \Psi(x) + \rho_{2n}(x)) \\ &\quad - F(x, C^T \Psi(x), C^T (S \Psi(x) + \rho(x)), C^T (S^2 \Psi(x) + \rho_2(x)), \dots, C^T (S^{2n-1} \Psi(x) + \rho_{2n-1}(x))). \end{aligned}$$

The philosophy behind the application of the typical collocation method to obtain the desired numerical solution $Z_N(x)$ is to enforce the residual vanish at suitably selected $(N + 1)$ points, say $x_i, 0 \leq i \leq N$; that is, we get

$$\begin{aligned} C^T (S^{2n} \Psi(x_i) + \rho_{2n}(x_i)) &= \\ F(x_i, C^T \Psi(x_i), C^T (S \Psi(x_i) + \rho(x_i)), C^T (S^2 \Psi(x_i) + \rho_2(x_i)), \dots, C^T (S^{2n-1} \Psi(x_i) + \rho_{2n-1}(x_i))). \end{aligned} \tag{37}$$

There are numerous ways to choose these collocation points. A numerical approximation $Z_N(x)$ is generated for each option. These are some options for these collocation points:

1. The $(N + 1)$ zeros of the shifted Legendre polynomials $P_{N+1}^*(x)$.
2. The $(N + 1)$ zeros of the shifted Chebyshev polynomial of the first kind $T_{N+1}^*(x)$.
3. The $(N + 1)$ zeros of the shifted Chebyshev polynomial of the second kind $U_{N+1}^*(x)$.

It is evident that a set of non-linear equations in the expansion coefficients, c_i , are produced for every choice of collocation points, with $(N + 1)$ being the number of non-linear equations. Newton’s iterative approach, widely used to solve non-linear systems, can be applied here, and the associated approximate solution $Z_N(x)$ can then be acquired.

Remark 5. We mention here the advantages of the SGJOMM derived in Section 5 to solve (27)–(28). They can be listed as follows:

- The simplicity of the SGJOMM as well as its high efficiency.
- Multiple solutions can be obtained using different collocation points.
- The derived operational matrix approach can be used to solve both linear and non-linear two-point even-order BVPs.

Remark 6. In Algorithm 2, We list the required steps to obtain the solution of the non-linear equation using SGJOMM.

Algorithm 2: Required steps to solve (27)–(28) by the SGJOMM

- Step 1.** Choose the basis functions $\psi_{j,n}$ as in (15).
 - Step 2.** Consider an approximate solution to (27) as: $Z_N(x) = \sum_{i=0}^N c_i \psi_{i,n}(x)$.
 - Step 3.** Represent $Z_N(x)$ and $D^q Z_N(x)$, $1 \leq q \leq 2n$ as in (29) and (36), respectively.
 - Step 4.** Apply the collocation method to obtain the system in (37).
 - Step 5.** Solve the system (37) by using Newton’s iterative method.
 - Step 6.** Obtain the approximate solution $Z_N(x)$.
-

6. Discussion on the Convergence Analysis

Here, we provide an extensive analysis of the proposed generalized Jacobi expansion, including a study of its convergence and examining the resulting global error. Two theorems are given and proved in this context. The first shows that the generalized Jacobi expansion of a function $u_N(x) = \sum_{j=0}^N c_j \psi_{j,n}(x)$ converges uniformly to $u(x)$. In addition, a global error estimate upper bound is found in the second theorem. The following four lemmas are useful in the sequel.

Lemma 2 ([43]). *The shifted Legendre polynomials $P_j^*(x)$, $x \in I$ satisfy the inequality*

$$\sqrt{\sin \vartheta} \left| P_j^*(\cos \vartheta) \right| < \sqrt{\frac{2}{\pi j}}, \quad \vartheta \in [0, \pi].$$

Lemma 3 ([38]). *For a non-zero real number α and any non-negative integers j , one has*

$$\Gamma(j + \alpha) = \mathcal{O}(j^{\alpha-1}j!).$$

Lemma 4 ([38]). *The shifted polynomials $\tilde{R}_j^{(n,n)}(x)$ satisfy the following inequality:*

$$\left| \tilde{R}_j^{(n,n)}(x) \right| \leq 1, \quad x \in I.$$

Lemma 5 ([18]). *If the m times repeated integrations of $P_j^*(x)$ are denoted by*

$$I_j^{(m)}(x) = \overbrace{\int \int \cdots \int}^{m \text{ times}} P_j^*(x) \overbrace{dx dx \cdots dx}^{m \text{ times}},$$

then, $I_j^{(m)}(x)$ can be expressed as

$$I_j^{(m)}(x) = \sum_{k=0}^m \frac{(-1)^k 4^{-m} (b-a)^m \binom{m}{k} \left(j + m - 2k + \frac{1}{2} \right)}{\left(j - k + \frac{1}{2} \right)_{m+1}} P_{j+m-2k}^*(x) + \tilde{\pi}_{m-1}(x),$$

where $\tilde{\pi}_{m-1}(x)$ is a polynomial of degree at most $(m - 1)$.

Theorem 5. *If $u(x) = (b - x)^n (x - a)^n g(x)$ is expanded in infinite series of the basis functions $\psi_{j,n}(x)$ defined in (15), i.e.,*

$$u(x) = \sum_{j=0}^{\infty} c_j \psi_{j,n}(x), \quad x \in I. \tag{38}$$

Then this series converges uniformly to $u(x)$, and the expansion coefficients c_j satisfy the following inequality:

$$|c_j| < \frac{M}{j^{2m-4n-2} (j!)^2}, \quad \forall j > m, m > 2n + 1,$$

with $|g^{(m)}(x)| \leq M$, where M is a positive constant.

Proof. The assumption (38) and the orthogonality relation (6) allow one to write

$$c_j = \frac{1}{h_j} \int_a^b \tilde{w}(x) u(x) \psi_{j,n}(x) dx.$$

Now, if we assume that $u(x) = (b - x)^n(x - a)^n g(x)$, and using (15), then we have

$$c_j = \frac{1}{h_j} \int_a^b g(x) (b - x)^n(x - a)^n \tilde{R}_j^{(n,n)}(x) dx,$$

In virtue of Formula (7), we can write

$$c_j = \frac{1}{h_j} \sum_{i=0}^n \eta_{i,j,n} \int_a^b g(x) P_{2i+j}^*(x) dx, \tag{39}$$

where $\eta_{i,j,n}$ take the following form:

$$\eta_{i,j,n} = \frac{(-1)^i n!(b - a)^{2n} (4i + 2j + 1) \binom{n}{i} \Gamma\left(i + j + \frac{1}{2}\right)}{2^{2n+1} \Gamma\left(i + j + n + \frac{3}{2}\right)}.$$

The properties of the shifted Legendre polynomials are used with Lemma 5 to enable us to rewrite (39), after integrating m times by parts, in the form

$$c_j = \frac{1}{h_j} \sum_{i=0}^n \eta_{i,j,n} \int_a^b g^{(m)}(x) P_{2i+j}^{*(m)}(x) dx.$$

If we use the change of variable $x = \frac{(b - a)}{2} \cos \vartheta + \frac{(b + a)}{2}$, $\vartheta = \cos^{-1}\left(\frac{2x - a - b}{b - a}\right)$, then we have

$$c_j = -\frac{1}{h_j} \sum_{i=0}^n \eta_{i,j,n} \int_0^\pi g^{(m)}\left(\frac{\pi}{2}(1 - \cos \vartheta)\right) I_{2i+j}^{(m)}\left(\frac{\pi}{2}(1 - \cos \vartheta)\right) \sin \vartheta d\vartheta.$$

Taking into consideration the assumption $\left|g^{(m)}\left(\frac{\pi}{2}(1 - \cos \vartheta)\right)\right| \leq M$, we get

$$|c_j| < \frac{M}{h_j} \sum_{i=0}^n |\eta_{i,j,n}| \int_0^\pi \left|I_{2i+j}^{(m)}\left(\frac{\pi}{2}(1 - \cos \vartheta)\right)\right| |\sin \vartheta| d\vartheta.$$

After some manipulation, the following inequality can be produced from Lemmas 2, 3, and 5:

$$|c_j| < \frac{M}{j^{2m-4n-2} (j!)^2}, \quad \forall j > m, m > 2n + 1,$$

which proves the theorem. \square

Theorem 6. Let $u(x)$ satisfy the same assumptions of Theorem 5, and consider the approximate solution $u_N(x)$ as in (17). The following error estimate holds:

$$\|u(x) - u_N(x)\|_\infty < \frac{M(b - a)^{2n}}{N^{2m-4n-1} (N!)^2}.$$

Proof. From relation (17), we have $\forall x \in I$

$$|u(x) - u_N(x)| = \left| \sum_{j=0}^{\infty} c_j \psi_{j,n}(x) - \sum_{j=0}^N c_j \psi_{j,n}(x) \right|,$$

and hence

$$|u(x) - u_N(x)| = \left| \sum_{j=N+1}^{\infty} c_j \psi_{j,n}(x) \right|,$$

by virtue of Formula (15), we can write

$$|u(x) - u_N(x)| \leq \sum_{j=N+1}^{\infty} |c_j| |(x-a)^n (b-x)^n| \left| \tilde{R}_j^{(n,n)}(x) \right|.$$

With the aid of Lemma 4 and Theorem 5, the following inequality can be obtained:

$$|u(x) - u_N(x)| < \frac{M(b-a)^{2n}}{N^{2m-4n-1} (N!)^2},$$

and accordingly, we have

$$\|u(x) - u_N(x)\|_{\infty} < \frac{M(b-a)^{2n}}{N^{2m-4n-1} (N!)^2}.$$

Theorem 6 is now proved. \square

7. Illustrative Problems and Comparisons

Examples of both linear and non-linear BVPs are presented here to demonstrate the usefulness and precision of our two suggested techniques. The shifted generalized Jacobi Petrov–Galerkin method (SGJPGM) is used to solve the linear differential equations, while the shifted generalized Jacobi operational matrix method (SGJOMM) is used to solve the non-linear differential equations in the examples below. First, we define the maximum absolute errors (MAEs) and the errors resulting from the least squares method (LSM), i.e., L^{∞} - and L^2 -errors for the function $u(x)$ defined on I . They can be defined, respectively, by

$$\text{MAEs}(L^{\infty} - \text{errors}) = \max_{x \in I} |u(x) - u_N(x)|,$$

$$\text{LSM}(L^2 - \text{errors}) = \sqrt{\int_a^b |u(x) - u_N(x)|^2 dx},$$

where $u_N(x)$ is the approximate solution of $u(x)$.

Example 1. Consider the following eighth-order BVP [44–47]:

$$u^{(8)}(x) - u(x) = -8(2x \cos x + 7 \sin x), \quad x \in [0, 1], \tag{40}$$

$$\begin{aligned} u(0) &= 0, & u'(0) &= 1, & u''(0) &= 0, & u'''(0) &= -3, \\ u(1) &= 0, & u'(1) &= -e, & u''(1) &= -4e, & u'''(1) &= -9e, \end{aligned}$$

with the exact solution: $u(x) = x(1-x)e^x$.

We apply our algorithm—namely, SGJPGM—to solve problem (40). Table 1 illustrates a comparison of L^{∞} - and L^2 -errors resulting from our algorithm with distinct N . Table 2 compares L^{∞} -errors resulting from our algorithm with those resulting from the application of the following methods:

- The spline method (SM) in [44].
- The non-polynomial spline method (NPSM) in [45].
- The reproducing kernel method (RKM) in [46].
- The Legendre matrix method (LMM) in [47].

Furthermore, Figure 1 shows the MAEs of our algorithm at $N = 7$, while Figure 2 shows the $\text{Log}_{10}(L^\infty\text{-errors})$ and $\text{Log}_{10}(L^2\text{-errors})$ of our algorithm with distinct N .

Table 1. Comparison of L^∞ - and L^2 -errors of our algorithm with distinct N for Example 1.

N	$L^\infty\text{-Errors}$	$L^2\text{-Errors}$
1	$4.48011 \cdot 10^{-7}$	$2.44031 \cdot 10^{-7}$
2	$4.30610 \cdot 10^{-9}$	$2.13197 \cdot 10^{-9}$
3	$1.65350 \cdot 10^{-9}$	$8.88121 \cdot 10^{-10}$
4	$1.02447 \cdot 10^{-11}$	$5.64667 \cdot 10^{-12}$
5	$1.35607 \cdot 10^{-12}$	$7.03195 \cdot 10^{-13}$
6	$7.02861 \cdot 10^{-15}$	$3.89473 \cdot 10^{-15}$
7	$8.40090 \cdot 10^{-16}$	$3.46545 \cdot 10^{-16}$

Table 2. Comparison of L^∞ -errors of our algorithm with those of methods in [44–47] for Example 1.

Error	SM in [44]	NPSM in [45]	RKM in [46]	LMM in [47]			SGJPGM
				$N = 8$	$N = 10$	$N = 7$	
E	$1.20 \cdot 10^{-5}$	$1.02 \cdot 10^{-8}$	$4.90 \cdot 10^{-9}$	$8.95 \cdot 10^{-14}$	$2.85 \cdot 10^{-16}$	$8.40 \cdot 10^{-16}$	

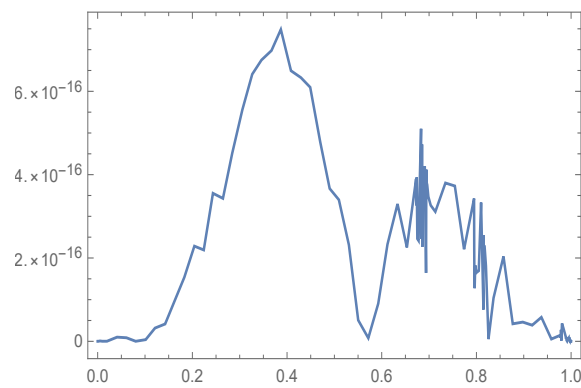


Figure 1. MAEs of our algorithm at $N = 7$ for Example 1.

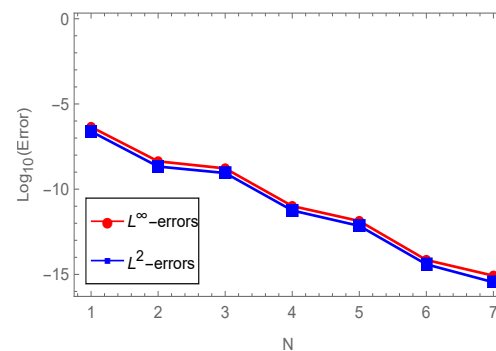


Figure 2. $\text{Log}_{10}(L^\infty\text{-errors})$ and $\text{Log}_{10}(L^2\text{-errors})$ of our algorithm with distinct N for Example 1.

Example 2. Consider the following BVP: [44–46]:

$$u^{(8)}(x) - u(x) = -8(2x \cos x + 7 \sin x), \quad x \in [-1, 1], \tag{41}$$

$$u(-1) = 0, \quad u'(-1) = 2 \sin(1), \quad u''(-1) = -4 \cos(1) - 2 \sin(1), \quad u'''(-1) = 6 \cos(1) - 6 \sin(1),$$

$$u(1) = 0, \quad u'(1) = 2 \sin(1), \quad u''(1) = 4 \cos(1) + 2 \sin(1), \quad u'''(1) = 6 \cos(1) - 6 \sin(1),$$

with the exact solution: $u(x) = (x^2 - 1) \sin x$.

Our proposed method (SGJPGM) is applied to solve problem (41). Table 3 illustrates a comparison of L^∞ - and L^2 -errors of SGJPGM for various values of N . Table 4 compares the L^∞ -errors resulting from our algorithm with those obtained if the methods in [44–46] are applied. In addition, Figure 3 displays the MAEs of our algorithm for $N = 9$, while Figure 4 shows the $\text{Log}_{10}(L^\infty\text{-errors})$ and $\text{Log}_{10}(L^2\text{-errors})$ of our algorithm with distinct N .

Table 3. Comparison of L^∞ and L^2 -errors of our algorithm with distinct N for Example 2.

N	$L^\infty\text{-Errors}$	$L^2\text{-Errors}$
1	$3.72783 \cdot 10^{-6}$	$3.16549 \cdot 10^{-6}$
3	$4.44508 \cdot 10^{-8}$	$3.73237 \cdot 10^{-8}$
5	$1.28602 \cdot 10^{-10}$	$1.05002 \cdot 10^{-10}$
7	$1.10634 \cdot 10^{-13}$	$8.23261 \cdot 10^{-14}$
9	$5.00745 \cdot 10^{-14}$	$3.73238 \cdot 10^{-14}$

Table 4. Comparison of L^∞ -errors of our algorithm with those of methods in [44–46] for Example 2.

Error	SM in [44] $h = 1/32$	NPSM in [45] $h = 1/32$	RKM in [46] $N = 10$	SGJPGM $N = 9$
E	$1.20 \cdot 10^{-5}$	$1.02 \cdot 10^{-8}$	$4.90 \cdot 10^{-9}$	$5.01 \cdot 10^{-14}$

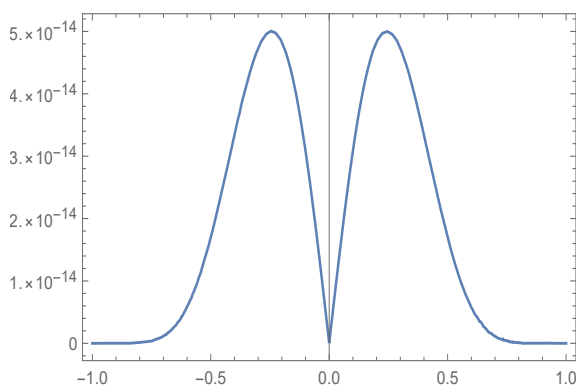


Figure 3. MAEs of our algorithm at $N = 9$ for Example 2.

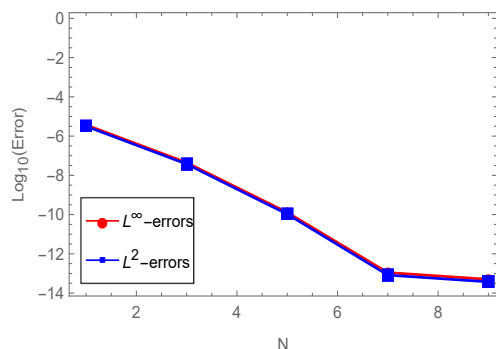


Figure 4. Log₁₀(L[∞] – errors) and Log₁₀(L² – errors) of our algorithm with distinct N for Example 2.

Example 3. Consider the BVP [48,49]:

$$Z^{(8)}(x) = 7! \left(e^{-8Z(x)} - \frac{2}{(x+1)^8} \right), \quad x \in [0, e^{1/2} - 1], \tag{42}$$

subject to the BCs

$$\begin{aligned} Z(0) &= 0, & Z'(0) &= 1, & Z''(0) &= -1, & Z'''(0) &= 2, \\ Z(e^{1/2} - 1) &= \frac{1}{2}, & Z'(e^{1/2} - 1) &= e^{-1/2}, & Z''(e^{1/2} - 1) &= -e^{-1}, & Z'''(e^{1/2} - 1) &= 2e^{-3/2}, \end{aligned}$$

with the exact solution: $Z(x) = \ln(x + 1)$.

Our proposed method—namely, SGJOMM—is applied to solve problem (42). Table 5 illustrates a comparison of L[∞]- and L²-errors of SGJOMM for various values of N. Table 6 compares the L[∞]-errors resulting from our algorithm with those obtained by the following two methods:

- The Galerkin septic B-splines method (GSBSM) in [48].
- The Sinc–Galerkin method (SGM) in [49].

In addition, Figure 5 displays the MAEs of our algorithm at N = 11, while Figure 6 shows the Log₁₀(L[∞]-errors) and Log₁₀(L²-errors) of our algorithm with distinct N.

Table 5. Comparison of L[∞]- and L²-errors of our algorithm with distinct N for Example 3.

N	L [∞] -Errors	L ² -Errors
1	6.49854 · 10 ⁻⁷	2.84248 · 10 ⁻⁷
2	6.84114 · 10 ⁻⁸	2.77812 · 10 ⁻⁸
3	3.70217 · 10 ⁻⁸	1.58289 · 10 ⁻⁸
4	3.01981 · 10 ⁻⁹	1.17252 · 10 ⁻⁹
5	6.13366 · 10 ⁻¹⁰	2.45326 · 10 ⁻¹⁰
6	4.58591 · 10 ⁻¹¹	1.65191 · 10 ⁻¹¹
7	2.76063 · 10 ⁻¹²	9.38161 · 10 ⁻¹³
8	2.24598 · 10 ⁻¹³	6.82216 · 10 ⁻¹⁴
9	1.78746 · 10 ⁻¹⁴	5.54010 · 10 ⁻¹⁵
10	1.27676 · 10 ⁻¹⁵	4.29149 · 10 ⁻¹⁶
11	5.55112 · 10 ⁻¹⁶	7.43891 · 10 ⁻¹⁷

Table 6. Comparison of L^∞ -errors of our algorithm with those of methods in [48,49] for Example 3.

Error	GSBSM in [48]	SGM in [49]	SGJOMM
$N = 11$			
E	$8.508563 \cdot 10^{-5}$	$1.2303 \cdot 10^{-11}$	$5.55112 \cdot 10^{-16}$

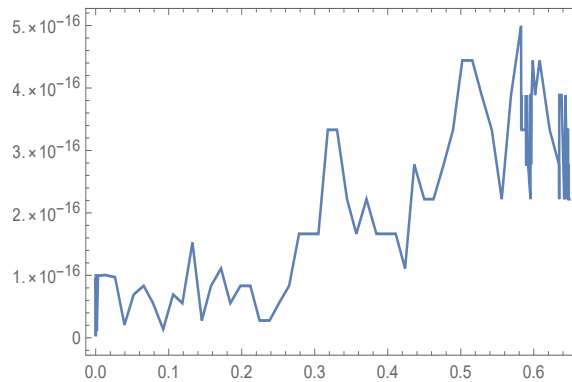


Figure 5. MAEs of our algorithm at $N = 11$ for Example 3.

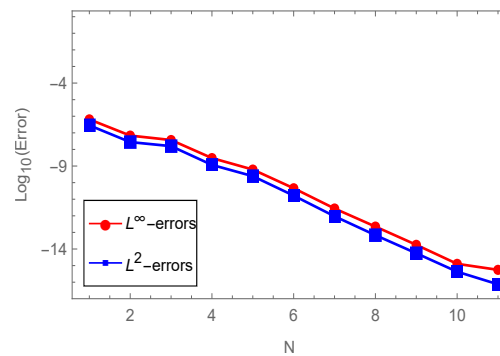


Figure 6. $\text{Log}_{10}(L^\infty\text{-errors})$ and $\text{Log}_{10}(L^2\text{-errors})$ of our algorithm with distinct N for Example 3.

Example 4. Consider the following non-linear two-point BVP [50,51]:

$$Z^{(4)}(x) + (Z''(x))^2 = \sin x + \sin^2 x, \quad x \in [0, 1], \tag{43}$$

subject to the BCs:

$$\begin{aligned} Z(0) &= 0, & Z'(0) &= 1, \\ Z(1) &= \sin(1), & Z'(1) &= \cos(1), \end{aligned}$$

with the following exact solution: $Z(x) = \sin x$.

SGJOMM is applied to solve problem (43). Table 7 illustrates a comparison of L^∞ - and L^2 -errors of SGJOMM for various values of N using the following three types of collocation points:

- The zeros of the shifted Legendre polynomials.
- The zeros of the shifted Chebyshev polynomials of the first kind.
- The zeros of the shifted symmetric Jacobi polynomials $\tilde{R}_j^{(2,2)}(x)$.

Table 8 compares the L^∞ -errors resulting from our algorithm with those obtained by the following two methods:

- The variational iteration method (VIM) in [50].
- The double decomposition method (DDM) in [51].

In addition, Figure 7 shows a comparison of $\text{Log}_{10}(L^\infty\text{-errors})$ of our algorithm by using Legendre, Chebyshev, and symmetric Jacobi polynomials with distinct N , while Figure 8 shows a comparison of $\text{Log}_{10}(L^2\text{-errors})$ of our algorithm by using (Legendre, Chebyshev, and symmetric Jacobi) with distinct N . using the same zeros.

Table 7. Comparison of L^∞ - and L^2 -errors of our algorithm by using Legendre, Chebyshev, and $\tilde{R}_j^{(2,2)}(x)$ with distinct N for Example 4.

N	Legendre		Chebyshev		$\tilde{R}_j^{(2,2)}(x)$	
	$L^\infty\text{-Errors}$	$L^2\text{-Errors}$	$L^\infty\text{-Errors}$	$L^2\text{-Errors}$	$L^\infty\text{-Errors}$	$L^2\text{-Errors}$
1	$3.099 \cdot 10^{-5}$	$1.917 \cdot 10^{-5}$	$1.814 \cdot 10^{-5}$	$1.099 \cdot 10^{-5}$	$1.594 \cdot 10^{-6}$	$8.994 \cdot 10^{-7}$
2	$8.634 \cdot 10^{-7}$	$5.173 \cdot 10^{-7}$	$5.665 \cdot 10^{-7}$	$3.354 \cdot 10^{-7}$	$8.568 \cdot 10^{-8}$	$4.853 \cdot 10^{-8}$
3	$4.630 \cdot 10^{-8}$	$2.536 \cdot 10^{-8}$	$1.747 \cdot 10^{-8}$	$8.598 \cdot 10^{-9}$	$1.416 \cdot 10^{-9}$	$7.770 \cdot 10^{-10}$
4	$1.040 \cdot 10^{-9}$	$5.920 \cdot 10^{-10}$	$4.779 \cdot 10^{-10}$	$2.553 \cdot 10^{-10}$	$6.677 \cdot 10^{-11}$	$3.720 \cdot 10^{-11}$
5	$1.177 \cdot 10^{-11}$	$5.603 \cdot 10^{-12}$	$5.177 \cdot 10^{-12}$	$2.726 \cdot 10^{-12}$	$9.008 \cdot 10^{-13}$	$4.917 \cdot 10^{-13}$
6	$3.471 \cdot 10^{-13}$	$1.820 \cdot 10^{-13}$	$1.712 \cdot 10^{-13}$	$9.264 \cdot 10^{-14}$	$3.603 \cdot 10^{-14}$	$1.983 \cdot 10^{-14}$
7	$3.886 \cdot 10^{-15}$	$1.757 \cdot 10^{-15}$	$2.442 \cdot 10^{-15}$	$9.810 \cdot 10^{-16}$	$4.996 \cdot 10^{-16}$	$2.212 \cdot 10^{-16}$
8	$2.220 \cdot 10^{-16}$	$5.449 \cdot 10^{-17}$	$2.220 \cdot 10^{-16}$	$3.080 \cdot 10^{-17}$	$2.220 \cdot 10^{-16}$	$7.638 \cdot 10^{-18}$
9	$2.220 \cdot 10^{-16}$	$4.844 \cdot 10^{-19}$	$2.220 \cdot 10^{-16}$	$3.224 \cdot 10^{-19}$	$2.220 \cdot 10^{-16}$	$8.795 \cdot 10^{-20}$
10	$2.220 \cdot 10^{-16}$	$3.847 \cdot 10^{-20}$	$2.220 \cdot 10^{-16}$	$3.685 \cdot 10^{-20}$	$2.220 \cdot 10^{-16}$	$4.775 \cdot 10^{-20}$

Table 8. Comparison of L^∞ -errors of our algorithm by using Legendre, Chebyshev, and $\tilde{R}_j^{(2,2)}(x)$ at $N = 7$ with those of methods in [50,51] for Example 4.

x	VIM in [50]	DDM in [51]	Our Method		
			Legendre	Chebyshev	$\tilde{R}_j^{(2,2)}(x)$
0.0	$9.5923 \cdot 10^{-14}$	$1.6490 \cdot 10^{-25}$	$0.0000 \cdot 10^{-00}$	$0.0000 \cdot 10^{-00}$	$0.0000 \cdot 10^{-00}$
0.1	$7.7856 \cdot 10^{-8}$	$1.2030 \cdot 10^{-10}$	$6.1932 \cdot 10^{-16}$	$3.2266 \cdot 10^{-16}$	$1.7019 \cdot 10^{-16}$
0.2	$2.7231 \cdot 10^{-7}$	$4.0080 \cdot 10^{-10}$	$5.2855 \cdot 10^{-17}$	$6.7811 \cdot 10^{-16}$	$2.5175 \cdot 10^{-16}$
0.3	$5.2489 \cdot 10^{-7}$	$7.3035 \cdot 10^{-10}$	$2.6620 \cdot 10^{-15}$	$1.1722 \cdot 10^{-15}$	$2.0676 \cdot 10^{-16}$
0.4	$7.7730 \cdot 10^{-7}$	$1.0007 \cdot 10^{-9}$	$3.8641 \cdot 10^{-16}$	$2.2400 \cdot 10^{-16}$	$1.8735 \cdot 10^{-16}$
0.5	$9.7145 \cdot 10^{-7}$	$1.1173 \cdot 10^{-9}$	$3.7557 \cdot 10^{-15}$	$2.3187 \cdot 10^{-15}$	$4.1785 \cdot 10^{-16}$
0.6	$1.0502 \cdot 10^{-6}$	$1.0304 \cdot 10^{-9}$	$2.0600 \cdot 10^{-17}$	$2.7539 \cdot 10^{-17}$	$2.4112 \cdot 10^{-16}$
0.7	$9.6286 \cdot 10^{-7}$	$7.6427 \cdot 10^{-10}$	$2.8821 \cdot 10^{-15}$	$1.1441 \cdot 10^{-15}$	$8.8471 \cdot 10^{-17}$
0.8	$6.8407 \cdot 10^{-7}$	$4.1663 \cdot 10^{-10}$	$1.9657 \cdot 10^{-16}$	$9.5746 \cdot 10^{-16}$	$3.7036 \cdot 10^{-16}$
0.9	$2.7069 \cdot 10^{-7}$	$1.2106 \cdot 10^{-10}$	$6.6153 \cdot 10^{-16}$	$2.7111 \cdot 10^{-16}$	$3.0602 \cdot 10^{-16}$
1.0	$1.5676 \cdot 10^{-13}$	$1.4649 \cdot 10^{-23}$	$0.0000 \cdot 10^{-00}$	$0.0000 \cdot 10^{-00}$	$0.0000 \cdot 10^{-00}$

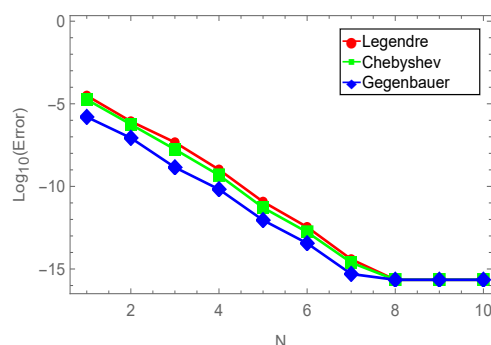


Figure 7. Comparison of $\text{Log}_{10}(L^\infty\text{-errors})$ of our algorithm by using Legendre, Chebyshev, and $\tilde{R}_j^{(2,2)}(x)$ with distinct N for Example 4.

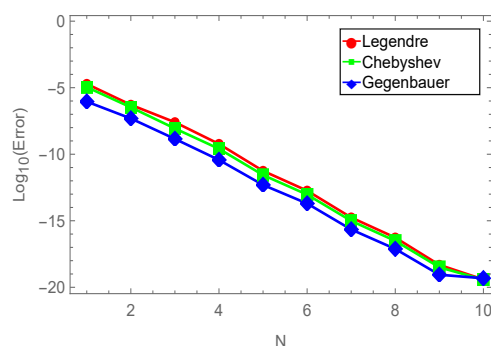


Figure 8. Comparison of $\text{Log}_{10}(L^2\text{-errors})$ of our algorithm by using Legendre, Chebyshev, and $\tilde{R}_j^{(2,2)}(x)$ with distinct N for Example 4.

8. Conclusions

A kind of orthogonal polynomial—namely, SGJPs—was utilized to treat both linear and non-linear BVPs. The linear ones were handled by applying the Petrov–Galerkin spectral method, while the non-linear ones were treated with the aid of the typical collocation method. To treat the linear BVPs, the expression for the derivatives of the SGJPs in terms of the shifted Legendre polynomials was established and utilized, while the algorithm designed for treating the non-linear BVPs was built on the utilization of the operational matrix of derivatives of the SGJPs. The two proposed numerical algorithms were tested via illustrative examples. These examples show the high performance and accuracy of the suggested algorithms. We also mention that the codes were written and debugged using Mathematica version 12 software using a PC machine, with Intel(R) Core(TM) i5-8500 CPU @ 3.00 GHz, 12.00 GB of RAM.

Author Contributions: W.M.A.-E. contributed to conceptualization, methodology, software, validation, formal analysis, investigation, Writing—Original draft, Writing—review & editing, and Supervision. B.M.B. contributed to methodology, validation and investigation, A.K.A. contributed to validation and funding Acquisition. M.M.A. contributed to methodology, software, formal analysis, investigation, Writing—review & editing. All authors have read and agreed to the published version of the manuscript.

Funding: The third author, Amr Kamel Amin (akgadelrab@uqu.edu.sa), is funded by the Deanship for Research and Innovation, Ministry of Education in Saudi Arabia through the project number IFP22UQU4331287DSR038.

Data Availability Statement: Not applicable.

Acknowledgments: The authors extend their appreciation to the Deanship for Research and Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project number IFP22UQU4331287DSR038.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mason, J.C.; Handscomb, D.C. *Chebyshev Polynomials*; CRC Press: Boca Raton, FL, USA, 2002.
2. Boyd, J.P. *Chebyshev and Fourier Spectral Methods*; Courier Corporation: North Chelmsford, MA, USA, 2001.
3. Shen, J.; Tang, T.; Wang, L.L. *Spectral Methods: Algorithms, Analysis and Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2011; Volume 41.
4. Hesthaven, J.S.; Gottlieb, S.; Gottlieb, D. *Spectral Methods for Time-Dependent Problems*; Cambridge University Press: Cambridge, UK, 2007; Volume 21.
5. Raza, N.; Zainab, U.; Araci, S.; Esi, A. Identities involving 3-variable Hermite polynomials arising from umbral method. *Adv. Differ. Equ.* **2020**, *2020*, 640. [[CrossRef](#)]
6. Boussayoud, A.; Kerada, M.; Araci, S.; Acikgoz, M.; Esi, A. Symmetric functions of binary products of Fibonacci and orthogonal polynomials. *Filomat* **2019**, *33*, 1495–1504. [[CrossRef](#)]
7. Khan, W.A.; Araci, S.; Acikgoz, M.; Esi, A. Laguerre-based Hermite-Bernoulli polynomials associated with bilateral series. *Tbil. Math. J.* **2018**, *11*, 111–121.
8. Albosaily, S.; Khan, W.A.; Araci, S.; Iqbal, A. Fully degenerating of Daehee numbers and polynomials. *Mathematics* **2022**, *10*, 2528. [[CrossRef](#)]
9. Yari, A. Numerical solution for fractional optimal control problems by Hermite polynomials. *J. Vib. Control* **2021**, *27*, 698–716. [[CrossRef](#)]
10. Sun, Q.; Zhang, R.; Hu, Y. Domain decomposition-based discontinuous Galerkin time-domain method with weighted Laguerre polynomials. *IEEE Trans. Antennas Propag.* **2021**, *69*, 7999–8002. [[CrossRef](#)]
11. Abd-Elhameed, W.M.; Doha, E.H.; Alsuyuti, M.M. Numerical treatment of special types of odd-order boundary value problems using nonsymmetric cases of Jacobi polynomials. *Prog. Fract. Differ. Appl.* **2022**, *8*, 305–319.
12. Sakran, M.R.A. Numerical solutions of integral and integro-differential equations using Chebyshev polynomials of the third kind. *Appl. Math. Comput.* **2019**, *351*, 66–82. [[CrossRef](#)]
13. Doha, E.H.; Abd-Elhameed, W.M.; Bassuony, M.A. On using third and fourth kinds Chebyshev operational matrices for solving Lane-Emden type equations. *Rom. J. Phys.* **2015**, *60*, 281–292.
14. Agarwal, R.P. *Boundary Value Problems for Higher Order Differential Equations*; World Scientific: Singapore, 1986.
15. Tomar, S. A computationally efficient iterative scheme for solving fourth-order boundary value problems. *Int. J. Appl. Comput. Math.* **2020**, *6*, 111. [[CrossRef](#)]
16. Baldwin, P. Asymptotic estimates of the eigenvalues of a sixth-order boundary-value problem obtained by using global phase-integral methods. *Philos. Trans. R. Soc. Lond. Ser. A* **1987**, *322*, 281–305.
17. Chandrasekhar, S. *Hydrodynamic and Hydromagnetic Stability*; Courier Corporation: North Chelmsford, MA, USA, 2013.
18. Abd-Elhameed, W.M.; Alkenedri, A.M. Spectral solutions of linear and nonlinear BVPs using certain Jacobi polynomials generalizing third-and fourth-kinds of Chebyshev polynomials. *CMES Comput. Model. Eng. Sci.* **2021**, *126*, 955–989. [[CrossRef](#)]
19. Islam, S.U.; Haq, S.; Ali, J. Numerical solution of special 12th-order boundary value problems using differential transform method. *Commun. Nonlinear Sci. Numer. Simul.* **2009**, *14*, 1132–1138. [[CrossRef](#)]
20. Golbabai, A.; Javidi, M. Application of homotopy perturbation method for solving eighth-order boundary value problems. *Appl. Math. Comp.* **2007**, *191*, 334–346. [[CrossRef](#)]
21. Izadi, M.; Srivastava, H. A novel matrix technique for multi-order pantograph differential equations of fractional order. *Proc. R. Soc.* **2021**, *477*, 20210321. [[CrossRef](#)]
22. Alsuyuti, M.M.; Doha, E.H.; Ezz-Eldien, S.S. Galerkin operational approach for multi-dimensions fractional differential equations. *Commun. Nonlinear Sci. Numer. Simul.* **2022**, *114*, 106608. [[CrossRef](#)]
23. Alsuyuti, M.M.; Doha, E.H.; Ezz-Eldien, S.S.; Bayoumi, B.I.; Baleanu, D. Modified Galerkin algorithm for solving multitype fractional differential equations. *Math. Methods Appl. Sci.* **2019**, *42*, 1389–1412. [[CrossRef](#)]
24. Doha, E.H.; Abd-Elhameed, W.M.; Bhrawy, A.H. New spectral-Galerkin algorithms for direct solution of high even-order differential equations using symmetric generalized Jacobi polynomials. *Collect. Math.* **2013**, *64*, 373–394. [[CrossRef](#)]
25. Abdelhakem, M.; Alaa-Eldeen, T.; Baleanu, D.; Alshehri, M.G.; El-Kady, M. Approximating real-life BVPs via Chebyshev polynomials' first derivative pseudo-Galerkin method. *Fractal Fract.* **2021**, *5*, 165. [[CrossRef](#)]
26. Abd-Elhameed, W.M. Novel expressions for the derivatives of sixth-kind Chebyshev polynomials: Spectral solution of the non-linear one-dimensional Burgers' equation. *Fractal Fract.* **2021**, *5*, 53. [[CrossRef](#)]
27. Tu, H.; Wang, Y.; Lan, Q.; Liu, W.; Xiao, W.; Ma, S. A Chebyshev-Tau spectral method for normal modes of underwater sound propagation with a layered marine environment. *J. Sound Vib.* **2021**, *492*, 115784. [[CrossRef](#)]
28. Singh, H. Jacobi collocation method for the fractional advection-dispersion equation arising in porous media. *Numer. Methods Partial. Differ. Equations* **2022**, *38*, 636–653. [[CrossRef](#)]

29. Rashidinia, J.; Eftekhari, T.; Maleknejad, K. Numerical solutions of two-dimensional nonlinear fractional Volterra and Fredholm integral equations using shifted Jacobi operational matrices via collocation method. *J. King Saud Univ. Sci.* **2021**, *33*, 101244. [[CrossRef](#)]
30. Khader, M.M.; Saad, K.M.; Hammouch, Z.; Baleanu, D. A spectral collocation method for solving fractional KdV and KdV-Burgers equations with non-singular kernel derivatives. *Appl. Numer. Math.* **2021**, *161*, 137–146. [[CrossRef](#)]
31. Kumbinarasaiah, S.; Preetham, M.P. Applications of the Bernoulli wavelet collocation method in the analysis of MHD boundary layer flow of a viscous fluid. *J. Umm Al-Qura Univ. Appl. Sci.* **2022**. [[CrossRef](#)]
32. Trefethen, L. *Spectral Methods in MATLAB*; SIAM: Philadelphia, PA, USA, 2000.
33. Shen, J. Efficient spectral-Galerkin method I. Direct solvers of second-and fourth-order equations using Legendre polynomials. *SIAM J. Sci. Comput.* **1994**, *15*, 1489–1505. [[CrossRef](#)]
34. Shen, J. Efficient spectral-Galerkin method II. Direct solvers of second-and fourth-order equations using Chebyshev polynomials. *SIAM J. Sci. Comput.* **1995**, *16*, 74–87. [[CrossRef](#)]
35. Napoli, A.; Abd-Elhameed, W.M. An innovative harmonic numbers operational matrix method for solving initial value problems. *Calcolo* **2017**, *54*, 57–76. [[CrossRef](#)]
36. Shloof, A.M.; Senu, N.; Ahmadian, A.; Salahshour, S. An efficient operation matrix method for solving fractal–fractional differential equations with generalized Caputo-type fractional–fractal derivative. *Math. Comput. Simul.* **2021**, *188*, 415–435. [[CrossRef](#)]
37. Youssri, Y.H. Orthonormal ultraspherical operational matrix algorithm for fractal–fractional Riccati equation with generalized Caputo derivative. *Fractal Fract.* **2021**, *5*, 100. [[CrossRef](#)]
38. Rainville, E.D. *Special Functions*; The Maximalan Company: New York, NY, USA, 1960.
39. Guo, B.Y.; Shen, J.; Wang, L.L. Optimal spectral-Galerkin methods using generalized Jacobi polynomials. *J. Sci. Comput.* **2006**, *27*, 305–322. [[CrossRef](#)]
40. Abd-Elhameed, W.M. Novel formulae of certain generalized Jacobi polynomials. *Mathematics* **2022**, *10*, 4237. [[CrossRef](#)]
41. Andrews, G.E.; Askey, R.; Roy, R. *Special Functions*; Cambridge University Press: Cambridge, UK, 1999; Volume 71.
42. Canuto, C.; Hussaini, M.Y.; Quarteroni, A.; Zang, T.A. *Spectral Methods in Fluid Dynamics*; Springer: Berlin/Heidelberg, Germany, 1988.
43. Giordano, C.; Laforgia, A. On the Bernstein-type inequalities for ultraspherical polynomials. *J. Comput. Appl. Math.* **2003**, *153*, 243–248. [[CrossRef](#)]
44. Siddiqi, S.S.; Twizell, E.H. Spline solutions of linear eighth-order boundary-value problems. *Comput. Methods Appl. Mech. Eng.* **1996**, *131*, 309–325. [[CrossRef](#)]
45. Siddiqi, S.S.; Akram, G. Solution of eighth-order boundary value problems using the non-polynomial spline technique. *Int. J. Comput. Math.* **2007**, *84*, 347–368. [[CrossRef](#)]
46. Akram, G.; Rehman, H.U. Numerical solution of eighth order boundary value problems in reproducing Kernel space. *Numer. Algorithms* **2013**, *62*, 527–540. [[CrossRef](#)]
47. Napoli, A.; Abd-Elhameed, W.M. Numerical solution of eighth-order boundary value problems by using Legendre polynomials. *Int. J. Comput. Methods* **2018**, *15*, 1750083. [[CrossRef](#)]
48. Ballem, S.; Viswanadham, K.N.S.K. Numerical solution of eighth order boundary value problems by Galerkin method with septic B-splines. *Procedia Eng.* **2015**, *127*, 1370–1377. [[CrossRef](#)]
49. El-Gamel, M.; Abdrabou, A. Sinc-Galerkin solution to eighth-order boundary value problems. *SeMA J.* **2019**, *76*, 249–270. [[CrossRef](#)]
50. Noor, M.A.; Mohyud-Din, S.T. Variational iteration method for fifth-order boundary value problems using He’s polynomials. *Math. Probl. Eng.* **2008**, *2008*, 954794. [[CrossRef](#)]
51. AL-Zaid, N.; AL-Refaidi, A.; Bakodah, H.; AL-Mazmumy, M. Solution of second-and higher-order nonlinear two-point boundary-value problems using double decomposition method. *Mathematics* **2022**, *10*, 3519. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.