

**TCHEBYCHEV POLYNOMIAL APPROXIMATIONS  
FOR  $m^{th}$  ORDER BOUNDARY VALUE PROBLEMS**

Samir Kumar Bhowmik

Department of Mathematics and Statistics

College of Science

Al Imam Mohammad Ibn Saud Islamic University (IMSIU)

P.O. BOX 90950, 11623 Riyadh, KINGDOM OF SAUDI ARABIA

**Abstract:** Higher order boundary value problems (BVPs) play an important role modeling various scientific and engineering problems. A general recipe for a  $m^{th}$  order BVPs is sparse (for any natural number  $m \geq 2$ ).

In this article we develop an efficient numerical scheme for linear  $m^{th}$  order BVPs. First we convert the higher order BVP to a system of first order BVPs. Then we use Tchebychev polynomials to approximate the solution of the BVP as a weighted sum of polynomials. We collocate at Tchebychev clustered grids to generate a system of equations to approximate the weights for the polynomials. The excellency of the numerical scheme is illustrated through some examples.

**AMS Subject Classification:** 65M12, 65M15

**Key Words:** Tchebychev polynomial; spectral collocation method; boundary value problem; numerical approximation

## 1. Introduction

Boundary value problems are an interesting topic in the field of applied mathematics, physics and engineering [2, 3, 4, 6, 7]. Higher order BVPs arise in

modeling various physical, chemical and biological realities. As for example, the Schrödinger equation can be presented by a second order BVP [20]; the free vibration analysis of beam structures is governed by a 4<sup>th</sup> order differential equation; and that of ring structures by a 6<sup>th</sup> order differential equation; an ordinary convection yields a 10<sup>th</sup> order BVP; an ordinary overstability yields a 12<sup>th</sup> order BVP [19]. Also an historically important example of a 4<sup>th</sup> order BVP is the Orr Sommerfeld equation from the field of hydrodynamic stability [16]. Numerical approximations for these problems are challenging because of the higher order derivatives and boundary conditions involving higher order derivatives of the unknown function.

In this study we consider the  $m^{\text{th}}$  order non-homogeneous linear differential equation

$$y^{(m)}(t) + a_1(t)y^{(m-1)}(t) + a_2(t)y^{(m-2)}(t) + \cdots + a_{m-1}(t)y^{(1)}(t) + a_m(t)y(t) = f(t),$$

$$t \in (a, b), \quad (1)$$

with a set of  $m$  boundary conditions. Here  $a_i(t)$ ,  $i = 1, 2, \dots, m$  and  $f(t)$  are real valued functions. Here we consider (1) with two exemplary set of boundary conditions though the scheme is not limited to them.

1. i.  $y^{(2k)}(a) = \alpha_{2k}$ ,  $k = 0, 1, 2, \dots, m/2 - 1$ ,  
ii.  $y^{(2k)}(b) = \beta_{2k}$ ,  $k = 0, 1, 2, \dots, m/2 - 1$ ,  $m$  even,
2. i.  $y^{(2k)}(a) = \alpha_{2k}$ ,  $k = 0, 1, 2, \dots, (m-1)/2$   
ii.  $y^{(2k)}(b) = \beta_{2k}$ ,  $k = 0, 1, 2, \dots, (m-1)/2 - 1$ ,  $m$  odd.

or

1. i.  $y^{(k)}(a) = \alpha_k$ ,  $k = 0, 1, 2, \dots, m/2 - 1$ ,  
ii.  $y^{(k)}(b) = \beta_k$ ,  $k = 0, 1, 2, \dots, m/2 - 1$ ,  $m$  even,
2. i.  $y^{(k)}(a) = \alpha_k$ ,  $k = 0, 1, 2, \dots, (m-1)/2$   
ii.  $y^{(k)}(b) = \beta_k$ ,  $k = 0, 1, 2, \dots, (m-1)/2 - 1$ ,  $m$  odd.

The existence and uniqueness of analytic solutions of the higher order BVP (1) have been well studied [3, 11, 12] and many references therein. The numerical approximation on the solution of the  $m^{\text{th}}$ -order BVP (1) is sparse. So we are not interested in discussing the existence and uniqueness issues rather we focus on an efficient numerical scheme for (1).

The authors concentrate on numerical methods for 4<sup>th</sup> order BVPs in [5, 17]. A low-order numerical method is outlined in [17]. Taher et. al [15] study 4<sup>th</sup>

order Sturm-Liouville BVPs. They use Tchebychev differentiation matrices to approximate the model. They present some examples to demonstrate the scheme.

A detailed study about a  $6^{th}$  order BVPs can be found in [10]. They update various existing MATLAB codes to solve the  $6^{th}$  order differential equation with various boundary conditions using finite difference based schemes. A good discussion about the existence and uniqueness of solutions of such BVPs can be found in this article.

A locally supported Lagrange polynomial approximation of higher order BVPs can be found in [19]. They approximate a  $6^{th}$  order differential equation and an  $8^{th}$  order differential equation, respectively to illustrate the proposed method with two different set of boundary conditions.

Recently Adomian decomposition methods for  $4^{th}$  order BVPs are used by [1]. Shi and Cao [21] approximates a higher order eigenvalue problem using Haar wavelets. Whereas [18] applies a differential quadrature method for a  $4^{th}$  order BVP.

The famous and popular texts on higher order BVPs using spectral method are [16, 20]. In both of the study the authors consider Tchebychev nodes to approximate the higher order derivatives in the MATLAB programming environment. They present numerical solutions of some  $2^{nd}$  and  $4^{th}$  order BVPs and eigenvalue problems. Both the authors use some differentiation matrices to approximate the higher order derivatives. Then they discuss issues regarding boundary conditions. They discuss some difficulties of handling boundary conditions, specially boundary conditions related to the higher order derivatives. The limitation of the scheme is that nonuniqueness of approximate solutions may occur [20] because of the boundary conditions.

From all the discussions above and the references therein we experience that the main difficulty of using these schemes is to implement/handle boundary conditions. Studying all these articles we notice that most techniques are developed focusing some particular boundary conditions. Most of the articles are restricted to  $4^{th}$  order problems. It is also noticed that some articles discuss schemes for even order BVPs and some for odd order BVPs. Thus a rigorous study to develop a high accurate numerical scheme a general higher order BVPs is missing. Also, in most cases, the authors compute Tchebychev differentiation matrices (TDM) to approximate all higher order derivatives. The implementation of the boundary conditions in TDM is complicated [16, 20]. Here we develop a scheme where we do not need to compute TDMs, handling boundary conditions is easy and is computationally high accurate.

Here we propose a simple and efficient scheme for a general higher order

BVP (1) for any natural number  $m \geq 2$ . The efficient recipe is based on the Tchebychev collocation method. In this scheme one can handle boundary conditions without any complication. Actually we convert the higher order BVP to a first order BVP. Here all boundary conditions involving higher order derivatives have been converted to boundary conditions for the respective transformed functions. We approximate the resulting first order system of differential equations using Tchebychev polynomials defined in a bounded interval  $[a, b]$ . We present the unknown functions and the boundary values by weighted sums of truncated Tchebychev polynomials. As a result the scheme gives us a system of linear equations for the Tchebychev weights/coefficients which can be solved by using a standard linear algebra solver. We do not have to compute Tchebychev differentiation matrices for higher order derivatives here. Thus the scheme becomes simple and efficient. The proposed scheme preserves geometric convergence rate which is desired from an efficient numerical scheme. Here along with the solution of (1) the proposed scheme generates an approximation of the first  $m - 1$  derivatives of the unknown function which is also important in the study of some physical realities.

The rest of the article is organised as follows: In Section 2 we convert the higher order differential equation (1) to a set of first order differential equations. We discuss some basic results on Tchebychev polynomials and recurrence relations to support our study in Section 3. The proposed scheme for the resulting system of first order differential equations has been developed in Section 4. We finish with some numerical examples and discussions in Section 5.

## 2. Reduction of Order for the BVP (1)

Any higher order differential equation can be converted into a system of first order differential equations. Thus a higher order BVP can be converted into a first order system of differential equations with the same boundary conditions. Here in this study we aim to develop a scheme to approximate higher order BVPs after converting them to first order system of differential equations. To that end using the transformations

$$y = y_1, \quad \frac{dy}{dt} = y_2, \quad \frac{d^2y}{dt^2} = y_3, \quad \dots, \quad \frac{d^{m-1}y}{dt^{m-1}} = y_m, \quad (2)$$

we rewrite the  $m^{\text{th}}$  order linear BVPs (1) as a system of ordinary differential equations

$$\begin{aligned} \frac{dy_1}{dt} &= y_2, \quad \frac{dy_2}{dt} = y_3, \quad \dots, \quad \frac{dy_{m-1}}{dt} = y_m \\ \frac{dy_m}{dt} &= -\sum_{i=1}^{m-1} a_i y_{m-i} + f(t). \end{aligned} \quad (3)$$

The system of equations (3) can be written as

$$\frac{d\underline{y}}{dt} = \underline{A}\underline{y} + \underline{f}(t), \quad t \in (a, b), \quad (4)$$

where

$$\underline{A} = \begin{pmatrix} 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 0 & 1 \\ -a_m(t) & -a_{m-1}(t) & \dots & \dots & -a_2(t) & -a_1(t) & 0 \end{pmatrix},$$

$$\underline{y}(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_m(t) \end{pmatrix}, \quad \underline{f}(t) = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ f(t) \end{pmatrix}.$$

The boundary condition for the  $k^{\text{th}}$  derivative of  $y(t)$  is now transformed into the boundary condition of the unknown function  $y_k(t)$  for all  $k = 0, 1, 2, \dots, m-1$ . So in the similar fashion the transformed boundary conditions can be written as

$$\begin{aligned} \text{for } m \text{ even} & \begin{cases} y_{2k+1}(a) = \alpha_{2k}, \quad k = 0, 1, 2, \dots, m/2 - 1, \\ y_{2k+1}(b) = \beta_{2k}, \quad k = 0, 1, 2, \dots, m/2 - 1, \end{cases} \\ \text{for } m \text{ odd} & \begin{cases} y_{2k+1}(a) = \alpha_{2k}, \quad k = 0, 1, 2, \dots, (m-1)/2, \\ y_{2k+1}(b) = \beta_{2k}, \quad k = 0, 1, 2, \dots, (m-1)/2 - 1, \end{cases} \end{aligned} \quad (5)$$

or

$$\begin{aligned} \text{for } m \text{ even} & \begin{cases} y_{k+1}(a) = \alpha_k, \quad k = 0, 1, 2, \dots, m/2 - 1, \\ y_{k+1}(b) = \beta_k, \quad k = 0, 1, 2, \dots, m/2 - 1, \end{cases} \\ \text{for } m \text{ odd} & \begin{cases} y_{k+1}(a) = \alpha_k, \quad k = 0, 1, 2, \dots, (m-1)/2, \\ y_{k+1}(b) = \beta_k, \quad k = 0, 1, 2, \dots, (m-1)/2 - 1. \end{cases} \end{aligned} \quad (6)$$

From now on we consider the system of first order differential equations (4) with the set of transforms boundary conditions (5) and (6).

### 3. Tchebychev Polynomial Approximation

Before moving onto our main study we discuss some basic properties to support our scheme. In this article we are concerned with approximating solutions of (4) using Tchebychev polynomials as trials functions and Tchebychev nodes for collocation. Polynomial interpolations based on Tchebychev nodes are often used to approximate smooth function. The pseudospectral methods perform well, in cases where both solutions and coefficients are not smooth [8, 9] as well. In general, Spectral methods [16] are a class of spatial discretizations for differential equations. They can be categorised as Galerkin, tau and collocation(or pseudospectral) spectral methods. Galerkin and tau work with the coefficients of a global expansion whereas pseudospectral work with the values at collocation points. There are two key components for the formulation of Spectral methods:

- Trial function, which is also called the expansion or approximation functions.
- Test function, which is also known as weight functions.

Let  $L_n y(t)$  be the  $n^{th}$  degree polynomial approximation of  $y(t)$ . We state an upper bound for the solutions generated by pseudospectral approximation.

**Definition 3.1.** An approximation scheme converges to the function approximated if

$$\lim_{n \rightarrow \infty} \|L_n y(t) - y(t)\| = 0$$

for all  $t \in \Omega$ , where  $\Omega$  is the domain for  $y(t)$ , and

$$L_n y(t) = \sum_{j=0}^n a_j \phi_j(t), \quad (7)$$

where  $\phi_j(t)$  are orthogonal polynomials defined in  $\Omega$ .

**Theorem 3.1.** [14] Let  $y(t)$  be finite at every point of the finite interval  $(a, b)$  and such that  $\int_a^b y^2(t) dt$  exists. Then

$$\lim_{n \rightarrow \infty} \int_a^b |y(t) - L_n(y)|^r d\psi = 0, \quad 0 < r \leq 2,$$

if  $\{\phi_n(t)\}$  represents any set of orthogonal polynomials corresponding to  $(a, b)$ , and  $\psi$  is a weight function.

*Proof.* For exact details see [14].  $\square$

Now a days, a lot of attention has been grown to the study of the Tchebychev orthogonal polynomial approximation focusing various real life models. The efficiency of the method is very important, and have been well studied by several authors [16, 20]. The goal of this section is to recall some properties of the Tchebychev polynomials, state some known results, and derive useful formulas that are important for this study.

The Tchebychev points are unequally spaced points over  $[-1, 1]$ . These points are the horizontal coordinates of a unit circle with center  $(0, 0)$ . It is to note that they are numbered from right to left. These points are the extreme points of the  $n$ th degree Tchebychev polynomial of the first kind. These points are defined by

$$t_i = \cos\left(\frac{i\pi}{n}\right), \quad i = 0, 1, 2, \dots, n.$$

Tchebychev polynomials can be defined over  $[-1, 1]$  and are obtained by expanding the following generating formula

$$T_r(t) = \cos(r \cos^{-1}(t)), \quad r = 0, 1, 2, \dots, \quad -1 \leq t \leq 1.$$

This polynomials satisfy the following properties:

1.  $T_0(t) = 1$ ,  $T_1(t) = t$  and  $T_r(t) = 2tT_{r-1}(t) - T_{r-2}(t)$ ,  $r \geq 2$ .

- 2.

$$\int_{-1}^1 \frac{T_r T_p}{\sqrt{1-t^2}} dt = \begin{cases} 0, & r \neq p \\ \frac{\pi}{2}, & r > 0 \\ \pi, & r = 0. \end{cases}$$

Any analytic function  $y(t)$  can be approximated by a truncated Tchebychev series as

$$y(t) = \sum_{r=0}^n c_r T_r(t), \quad -1 \leq t \leq 1,$$

which can be written as

$$y(t) = T(t)c,$$

where

$$T(t) = [T_0(t), T_1(t), \dots, T_n(t)],$$

and

$$c = [c_0, c_1, \dots, c_n]'$$

Here, following the above convergence result we see that if the function  $u$  belongs to  $C^\infty$  class, the produced error of approximation as  $n$  tends to infinity, approaches zero with exponential rate ( $\mathcal{O}(e^{-c^+n})$ , for some  $c^+ > 0$ ). This phenomenon is usually referred to as “spectral accuracy”.

Assuming the function is differentiable the derivatives of  $y(t)$  can be written as [13]

$$y^{(k)}(t) = \sum_{j=0}^n c_j^{(k)} T_r(t), \quad -1 \leq t \leq 1, k \geq 0. \quad (8)$$

It is well known that the coefficients  $c_r^{(k)}$  and  $c_r^{(k+1)}$  of  $u^{(k)}(t)$  and  $u^{(k+1)}(t)$  can be written by the following three term recurrence relation

$$2rc_r^{(k)} = c_{r-1}^{(k+1)} - c_{r+1}^{(k+1)}, \quad r \geq 1,$$

can be simplified into the following relation [13]

$$c_r^{(k+1)} = 2 \sum_{j=0}^{\infty} (r + 2j + 1) c_{r+2j+1}^{(k)},$$

where  $c_r^{(0)} = c_r$ . Accordingly, the truncated system  $c^{(k)}$  can be computed as

$$c^{(k)} = 2^k \mathcal{M}^k c, \quad (9)$$

where

$$\text{when } n \text{ is odd, } \mathcal{M} = \begin{pmatrix} 0 & \frac{1}{2} & 0 & \frac{3}{2} & 0 & \dots & \frac{n}{2} \\ 0 & 0 & 2 & 0 & 4 & 0 \dots & 0 \\ 0 & 0 & 0 & 3 & 0 & 5 \dots & n \\ \vdots & \vdots & \vdots & \vdots & & & \vdots \\ 0 & 0 & 0 & & \dots & & n \\ 0 & 0 & 0 & 0 & 0 \dots & 0 & 0 \end{pmatrix},$$

and

$$\text{when } n \text{ is even, } \mathcal{M} = \begin{pmatrix} 0 & \frac{1}{2} & 0 & \frac{3}{2} & 0 & \dots & 0 \\ 0 & 0 & 2 & 0 & 4 & 0 \dots & n \\ 0 & 0 & 0 & 3 & 0 & 5 \dots & 0 \\ 0 & 0 & 0 & & \dots & & n \\ 0 & 0 & 0 & 0 & 0 \dots & 0 & 0 \end{pmatrix}.$$



### Tchebychev Polynomials on $[a, b]$

Now the Tchebychev polynomials can be defined over an interval  $[a, b]$  by

$$T_r^*(t) = \cos\left(r \cos^{-1}\left(\frac{2}{b-a}\left(t - \frac{a+b}{2}\right)\right)\right), \quad r = 0, 1, 2, \dots, \quad a \leq t \leq b. \quad (10)$$

The collocation points in  $[a, b]$  can be defined by

$$t_j = \frac{a-b}{2} \left( \frac{a+b}{b-a} + \cos\left(\frac{j\pi}{n}\right) \right), \quad j = 0, 1, 2, \dots, n. \quad (11)$$

Also using the transformed Tchebychev polynomials a unknown function  $v^*(t)$  and the derivatives can be approximated by

$$v^{*(k)}(t) = \sum_{j=0}^n C_j^{*(k)} T_j^*(t), \quad t \in [a, b], \quad k = 0, 1, 2, \dots,$$

where

$$C^{*(k)} = \left( \frac{4}{b-a} \right)^k \mathcal{M}^k C^*, \quad (12)$$

$$C^* = [C_0^* \ C_1^* \ C_2^* \ \dots \ C_n^*]'$$

#### 4. Tchebychev Polynomial Approximation of the System of BVPs

(4)

In this section we motivate ourselves to solve the system of differential equations (4) with a given set of boundary conditions. To that end we recall (4) with solutions as a truncated series of Tchebychev polynomials given by

$$y_i(t) = \sum_{j=0}^n C_{i,j} T_j^*(t), \quad a \leq t \leq b, \quad i = 1, 2, 3, \dots, m \quad (13)$$

which can be expressed as

$$y_i(t) = T^*(t) C_i,$$

where

$$T^*(t) = [T_0^*(t) \ T_1^*(t) \ \dots \ T_n^*(t)],$$

and

$$C_i = [C_{i,0} \ C_{i,1} \ C_{i,2} \ \dots \ C_{i,n}]'; \quad i = 1, 2, \dots, m,$$

and  $T_j^*(t)$  are given by (10) in  $[a \ b]$ . It is easy to see that  $y_i(t)$  approximates  $y^{(i-1)}(t)$  for  $i = 1, 2, 3, \dots, m$ . Here we aim to compute  $C_{i,j}$  values so that (13) approximates (4) satisfying all the boundary conditions. We write the derivatives of the unknown function  $y_i(t)$  as

$$y_i^{(k)}(t) = T^*(t)C_i^{(k)}. \quad (14)$$

Combining (13), (14) and (12) the derivatives involved in (4) can be expressed as

$$y_i^{(k)}(t) = \left(\frac{4}{b-a}\right)^k T^*(t)\mathcal{M}^k C_i, \quad k = 0, 1.$$

Thus  $\underline{y}(t)$  and all the derivatives in (4) can be approximated by

$$\underline{y}^{(k)}(t) = \left(\frac{4}{b-a}\right)^k \Psi(t)\mathbf{M}^k C, \quad k = 0, 1,$$

where

$$\Psi(t) = \begin{pmatrix} T^*(t) & & & \\ & T^*(t) & & \\ & & \ddots & \\ & & & T^*(t) \end{pmatrix}_{m \times m},$$

$$C = \begin{pmatrix} C_1 \\ C_2 \\ \vdots \\ C_m \end{pmatrix}_{m \times 1},$$

and

$$\mathbf{M}^k = \begin{pmatrix} \mathcal{M}^k & & & \\ & \mathcal{M}^k & & \\ & & \ddots & \\ & & & \mathcal{M}^k \end{pmatrix}_{m \times m}.$$

Using the above notations the system of ODEs (4) gives us a system of algebraic equations for the Tchebychef weights  $C$  as

$$\underline{y}^{(1)}(t) - A\underline{y}^{(0)}(t) = \underline{f}(t), \quad (15)$$

where

$$\underline{y}^{(0)}(t) = \Psi(t)C,$$

and

$$\underline{y}^{(1)}(t) = \frac{4}{b-a} \Psi(t) \mathbf{M} \mathbf{C}.$$

To solve the system for  $C_{i,j}$  we collocate (15) at translated Tchebychev nodes (11). Collocating at the prescribed grid points (15) yields

$$P_1 \underline{Y}^{(1)} - P_0 \underline{Y}^{(0)} = \underline{F}, \quad (16)$$

where

$\underline{Y}^{(i)}$ ,  $i = 0, 1$ , are  $m(n+1)$  column vectors defined on the Tchebychev nodes,

and

$$\underline{F} = [\underline{f}(t_0) \ \underline{f}(t_1) \ \cdots \ \underline{f}(t_n)]', \text{ a } m(n+1) \text{ column vector,}$$

$$P_1 = \begin{pmatrix} I_{m \times m} & & & \\ & I_{m \times m} & & \\ & & \ddots & \\ & & & I_{m \times m} \end{pmatrix},$$

a  $m(n+1) \times m(n+1)$  matrix, and

$$P_0 = \begin{pmatrix} A_{m \times m} & & & \\ & A_{m \times m} & & \\ & & \ddots & \\ & & & A_{m \times m} \end{pmatrix},$$

a  $m(n+1) \times m(n+1)$  matrix. Thus (16) gives a full discrete system of equation of the form

$$\mathcal{W} \underline{C} = \underline{F}, \quad (17)$$

where  $\mathcal{W}$  is the resulting discrete matrix,  $\underline{C}$  is the column of unknowns,  $\underline{F}$  is the discrete right hand side vector. (17) needs to be solved for  $m(n+1)$  unknowns  $\underline{C}$  after imposing boundary conditions.

Now we discuss the implementation of boundary conditions. We have  $m$  boundary conditions. Depending on the physical conditions modelled the  $k^{th}$  left and right boundary conditions (5) or (6) (or BCs of any type) can be presented by

$$\sum_{j=0}^n C_{k,j} T_j^*(t_n) = \alpha_k, \text{ and } \sum_{j=0}^n C_{k,j} T_j^*(t_0) = \beta_k,$$

which can be written in vector form as

$$T^*(t_n)C_k = \alpha_k, \text{ and } T^*(t_0)C_k = \beta_k. \quad (18)$$

So we need to combine the system of equations (17) and (18) to solve them for  $C_{i,j}$ . Now (17) and (18) involve the Tchebychev weights  $C_{i,j}$  for the unknown functions  $y_i$  at  $t = t_0$  and  $t = t_n$ . We need to apply boundary conditions (18) in (17) before we attempt to solve them for  $C_{i,j}$ . Here we use the following steps to solve the system for the weights so that the resulting polynomial satisfies the boundary conditions.

**Right Boundary Conditions** The first  $m$  rows of  $\mathcal{W}$ ,  $F$  and  $\Psi$  have been computed at  $t = t_0$ . Let us assign them as submatrices  $W_1$ ,  $F_1$ , and  $\Psi_1$ . Now the  $k^{th}$  row of  $W_1$ , and  $F_1$  are assigned for the unknown function  $y_k(t)$ ,  $k = 1, 2, \dots, m$  at  $t = t_0$ . Thus we replace the  $k^{th}$  row of  $W_1$  by the  $k^{th}$  row of  $\Psi_1$  ( $W_1(k, :) = \Psi_1(k, :)$ ), and respective  $k^{th}$  element of  $F_1$  by the boundary value  $\beta_k$  ( $F_1(k, 1) = \beta_k$ ) for all boundary conditions at  $t_0 = b$ .

**Left Boundary Conditions** The last  $m$  rows of  $\mathcal{W}$ ,  $F$  and  $\Psi$  have been computed at  $t = t_n$ . Let us assign them as submatrices  $W_n$ ,  $F_n$ , and  $\Psi_n$ . Now the  $k^{th}$  row of  $W_n$ , and  $F_n$  are assigned for the unknown function  $y_k(t)$ ,  $k = 1, 2, \dots, m$  at  $t = t_n$ . Thus we replace the  $k^{th}$  row of  $W_n$  by the  $k^{th}$  row of  $\Psi_n$  ( $W_n(k, :) = \Psi_n(k, :)$ ), and respective  $k^{th}$  element of  $F_n$  by the boundary value  $\alpha_k$  ( $F_n(k, 1) = \alpha_k$ ) for all boundary conditions at  $t_n = a$ .

Thus replacing the first  $m$  rows of  $\mathcal{W}$  by the rearranged  $W_1$ , the last  $m$  rows of  $\mathcal{W}$  by the rearranged  $W_n$ , first  $m$  elements of  $F$  by the rearranged  $F_1$ , and the last  $m$  elements of  $F$  by the rearranged  $F_n$  we get the system of equations of the form

$$\tilde{\mathcal{W}}\underline{C} = \tilde{\underline{F}}, \quad (19)$$

where

$$\tilde{\mathcal{W}} = \begin{bmatrix} W_1 \\ \mathcal{W}((m+1) : mn, 1 : m(n+1)) \\ W_n \end{bmatrix}, \text{ and } \tilde{\underline{F}} = \begin{bmatrix} F_1 \\ \mathcal{W}((m+1) : mn, 1) \\ F_n \end{bmatrix}.$$

The  $m(n+1) \times m(n+1)$  system of equations (19) can be solved for  $C$  by using any standard linear system solver. Here

- $C_{1,j}$ ,  $j = 0, 1, 2, \dots, n$  are the Tchebychev weights for the solution  $y(t)$  of (1),

- $C_{2,j}$ ,  $j = 0, 1, 2, \dots, n$  are the Tchebychev weights for  $y'(t)$  of (1),
- $C_{3,j}$ ,  $j = 0, 1, 2, \dots, n$  are the Tchebychev weights for  $y''(t)$  of (1),
- $\vdots$
- $C_{m,j}$ ,  $j = 0, 1, 2, \dots, n$  are the Tchebychev weights for  $y^{(m-1)}(t)$  of (1).

Thus we get a simple algorithm to solve the higher order linear differential equation (1) with any set of boundary conditions. In the next section we inspect some examples to show the efficiency of the scheme.

## 5. Numerical Experiments and Discussions

In this section we present numerical solutions of some higher order BVPs using the method outlined in the previous section. First define the set of clustered nodes (11). We use the transformation (2) to convert the higher order BVPs to a system of first order BVPs. Here we consider some examples (BVPs) with exact solutions so that we can present accuracy of our scheme numerically.

**Example 1.** We consider the following BVP

$$y^{(2)} + y = 1, \quad y(0) = 0, \quad y(1) = 1.$$

The exact solution is

$$y(t) = 1 - \cos t + \cot 1 \sin t.$$

The following Figure 1 demonstrates the numerical solution of the 2<sup>nd</sup> order BVP. Here from the error computation we observe that the accuracy of the scheme is of an exponential order and we need 8 nodes for a 10 digit accurate solution.

**Example 2.** We consider the following BVP

$$y^{(4)} + 2y^{(2)} + y = 1, \quad y(0) = 0, \quad y'(0) = 0, \quad y(1) = 0, \quad y'(1) = 0.$$

The exact solution is

$$y(x) = \frac{1}{1 + \sin(1)} (1 - t \cos(1-t) - \cos(t) + t \cos(t) + \sin(1) - \sin(1-t) - \sin(t)).$$

The following Figure 2 demonstrates the numerical solution of the 4<sup>th</sup> order BVP. Here from the error computation we observe that the accuracy of the scheme is of an exponential order. Here we notice that we need 9 nodes for a 10 digit accurate solution.

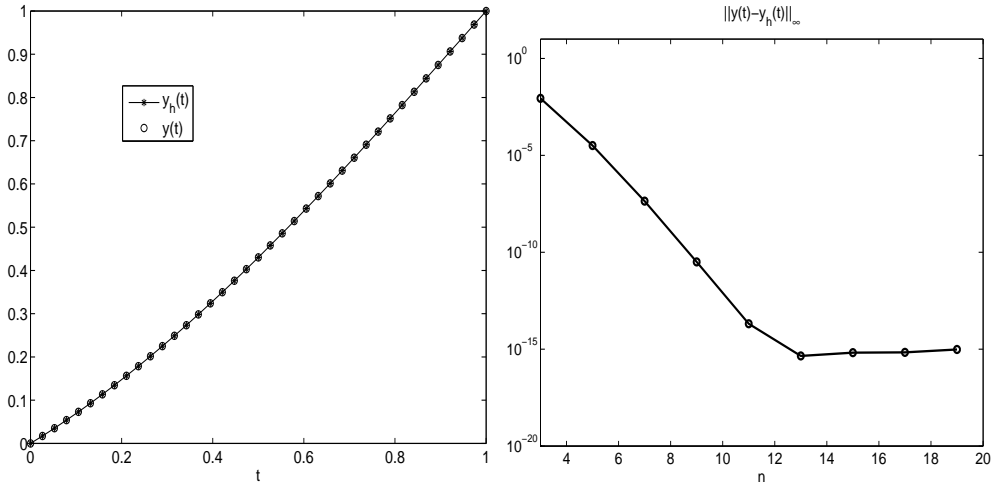


Figure 1: Solution of the second order BVP. In the left figure we plot the exact and the approximate solutions. In the right figure we plot the computational error.

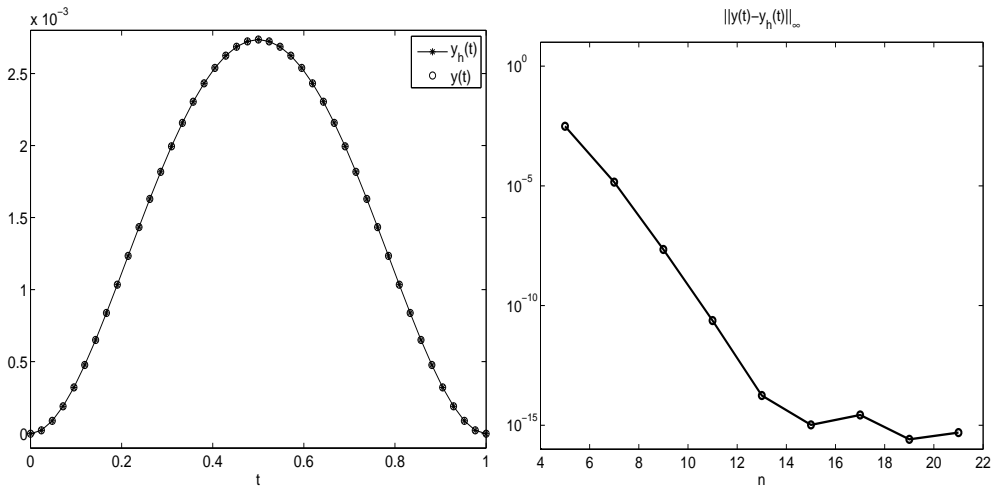


Figure 2: Solution of the fourth order BVP. In the left figure we plot the exact and the approximate solutions. In the right figure we plot the computational error.

**Example 3.** We consider the following BVP

$$y^{(5)} - y = -15e^t - 10te^t, \quad y(0) = 0, \quad y'(0) = 1, \quad y''(0) = 0, \quad y(1) = 0, \quad y'(1) = -e.$$

The exact solution is

$$y(t) = t(t-1)e^t.$$

The following Figure 3 demonstrates the numerical solution of the 5<sup>th</sup> order BVP. Here from the error computation we observe that the accuracy of the scheme is of an exponential order. Here we need 11 nodes for a 10 digit accurate solution.

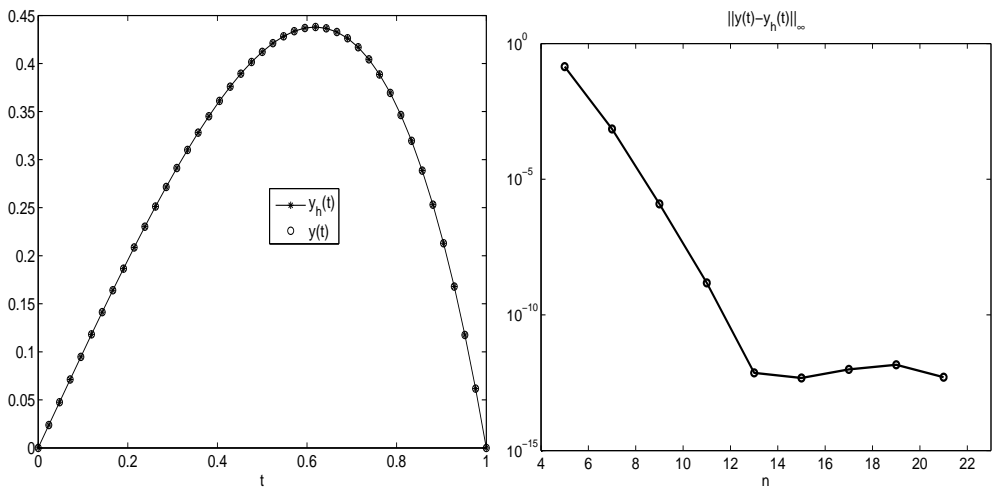


Figure 3: Solution of the fifth order BVP. In the left figure we plot the exact and the approximate solutions. In the right figure we plot the computational error.

**Example 4.** Consider the BVP

$$y^{(6)} - y = -6e^t,$$

with  $y(0) = 1$ ,  $y'(0) = 0$ ,  $y^{(2)}(0) = -1$ ,  $y(1) = 0$ ,  $y'(1) = -e$ ,  $y^{(2)}(1) = -2e$ . The exact solution is

$$y(t) = (1-t)e^t.$$

The following Figure 4 demonstrates the numerical solution of the 6<sup>th</sup> order BVP. Here from the error computation we observe that the accuracy of the

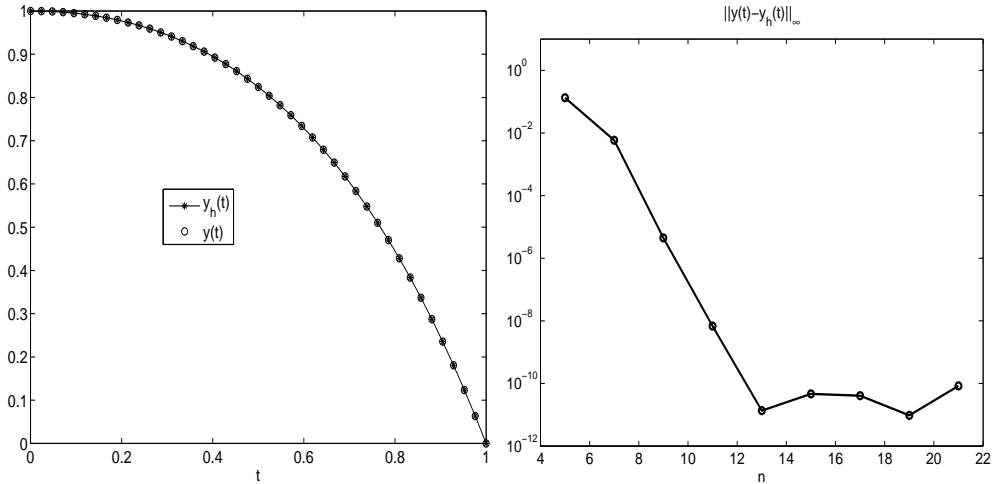


Figure 4: Solution of the sixth order BVP. In the left figure we plot the exact and the approximate solutions. In the right figure we plot the computational error.

scheme is of an exponential order and 11 nodes are enough for a 10 digit accurate solution.

**Example 5.** Consider the BVP

$$y^{(9)} - y = -9e^t,$$

with  $y(0) = 1$ ,  $y'(0) = 0$ ,  $y^{(2)}(0) = -1$ ,  $y^{(3)}(0) = -2$ ,  $y^{(4)}(0) = -3$ ,  $y(1) = 0$ ,  $y'(1) = -e$ ,  $y^{(2)}(1) = -2e$ ,  $y^{(3)}(1) = -3e$ . The exact solution is

$$y(t) = (1 - t)e^t.$$

The following Figure 5 demonstrates the numerical solution of the 9<sup>th</sup> order BVP. Here from the error computation we observe that the accuracy of the scheme is of an exponential order. Here we see that we need 13 nodes for a 8 digit accurate solution.

We discuss in detail the formulations of the Tchebychev collocation method for a general  $m^{th}$  order BVPs using Tchebychev polynomials as basis functions. From these computations we notice that the result is minimum 8 – 15 digit accurate when  $n > 10$  which agrees with other published works available in the literature. Here the advantage of using this scheme is that we can handle boundary conditions simply and efficiently. Also analyzing the error plots



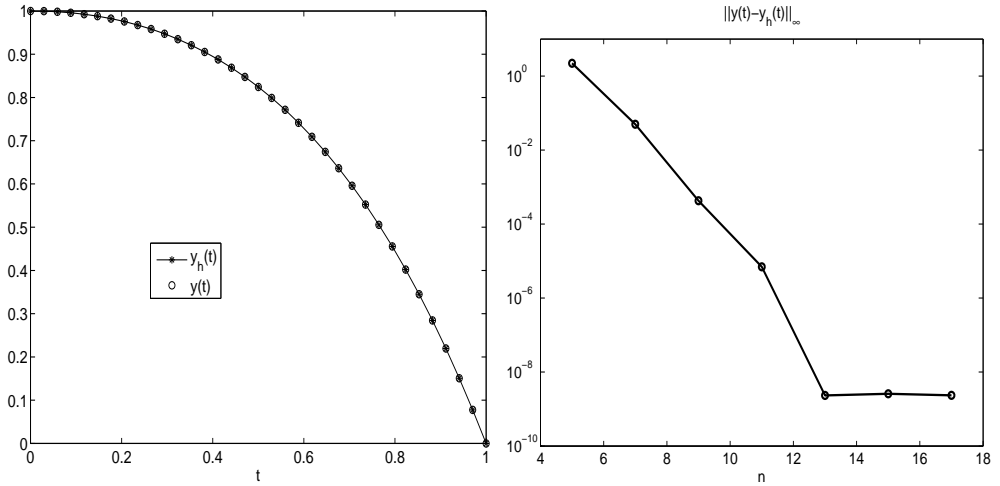


Figure 5: Solution of the ninth order BVP. In the left figure we plot the exact and the approximate solutions. In the right figure we plot the computational error.

(Figures 1- 5) we notice that the convergence is supergeometric. Therefore, we conclude that the Tchebychev polynomial approach for the transformed system of first order BVPs produces much accurate results than all other previously published lower order works. From these computations we see that our method compete very well with other methods cited in this article.

There are some benefits of using the transformed scheme for (1). Here we do not need to compute higher order derivatives of the unknown function. In this presented scheme, by converting the higher order differential equation (1) to the first order system of differential equations (4) we avoid computing Tchebychev differentiation matrices (what one needs while using Tchebychev polynomials to approximate higher order derivatives for a direct scheme for (1)[16]). All boundary conditions  $y^{(i)}(a) = \alpha_i$  and  $y^{(i)}(b) = \beta_i$  have been converted to boundary conditions  $y_i(a) = \alpha_i$ , and  $y_i(b) = \beta_i$  respectively, which are simple to handle in our proposed set up. Whereas applying  $y^{(i)}(a)$  and  $y^{(i)}(b)$  in the Tchebychev differentiation matrices is really complicated (see [16, 20] for the exact detail). Thus the scheme we present here is simply efficient for solving the higher order BVP (1), and preserves exponential/supergeometric order of convergence.

## References

- [1] B. Attili and D. Lesnic. An Efficient Method for Computing Eigenelements of Sturm-Liouville Fourth-Order Boundary Value Problems. *Applied Mathematics and Computation*, 182(2):1247–1254, 2006.
- [2] A. Yildirim. Solution of BVPs for fourth order integro-differential equations by using homotopy perturbation method. *Computers and Mathematics with Applications*, 56(12):3175–3180, December 2008.
- [3] R. P. Agarwa. *Boundary value problems for higher order differential equations*. World Scientific, Singapore, 1986.
- [4] A. H. Bhrawy and T. M. Taha. An operational matrix of fractional integration of the laguerre polynomials and its application on a semi-infinite interval. *Mathematical Sciences 2012*, 6:41, 2012.
- [5] M. M. Chawla and C. P. Katti. Finite difference methods for two-point boundary value problems involving high order differential equations. *j-BIT*, 19(1):27–33, mar 1979.
- [6] E. H. Doha, A. H. Bhrawy, and M. A. Saker. Integrals of bernstein polynomials: An application for the solution of high even-order differential equations. *Appl. Math. Lett.*, 24(4):559–565, 2011.
- [7] El-Gamel, Mohamed and Cannon, John R. and Zayed, Ahmed I. Sinc Galerkin method for solving linear sixth order boundary value problems. *Math. Comput.*, 73(247):1325–1343, 2004.
- [8] B Fornberg. *A Practical Guide to Pseudo Spectral Methods*. Cambridge monographs on Applied and Computational Mathematics, 1996.
- [9] W Gautschi. *Orthogonal Polynomials, Computation and Approximation*. Oxford Science Publications, 2004.
- [10] N. P. HALE. A sixth-order extension to the matlab bvp4c software of j. kierzenka and l. shampine. *MS thesis, Imperial College London*, June 2006.
- [11] X. Hao and L. Liu. Multiple monotone positive solutions for higher order differential equations with integral boundary conditions. *Boundary Value problems*, 2014:74, 2014.

- [12] M. D. Raisinghania. *Integral Equations and Boundary Value Problems*. S Chand & Company PVT. LTD, 2007.
- [13] M. Sezer and M. Kayanak. Chebyshev polynomial solutions of linear differential equations. *Int. J. Math. Educ. Sci. Technol.*, 27(4):607–618, 1996.
- [14] J. Shohat. On the convergence properties of Lagrange interpolation based on the zeros of orthogonal Tchebycheff polynomials. *The Annals of Mathematics*, 38(4):758–769, Oct 1937.
- [15] A. H. S Taher, A. Malek, and S. H. M Masuleh. Chebychev differentiation matrices for efficient computation of the eigenvalues of fourth order sturm-liouville problems. *Applied Mathematical Modeling*, 37:4634–4642, 2013.
- [16] Lloyd N. Trefethen. *Spectral Methods in Matlab*. Thomson, 2000.
- [17] E.H. Twizell. *Numerical Methods in the Biomedical Sciences*. Prentice Hall PTR, 1988.
- [18] U. Ycel and K. Boubaker. Differential Quadrature Method (DQM) and Boubaker Polynomials Expansion Scheme (BPES) for Efficient Computation of the Eigenvalues of Fourth-Order Sturm-Liouville Problems. *Applied Mathematical Modeling*, 36(1):4020–4026, 2012.
- [19] Y. Wanga, Y.B. Zhaoa, and G.W. Weia. A note on the numerical solution of high-order differential equations. *Journal of Computational and Applied Mathematics*, 159:387–398, June 2003.
- [20] J. Andre Weideman and Satish C. Reddy. A matlab differentiation matrix suite. *ACM Trans. Math. Softw.*, 26(4):465–519, 2000.
- [21] Z. Shi and Y. Cao. Application of Haar Wavelet Method to Eigenvalue Problems of High Order Differential Equations. *Applied Mathematical Modeling*, 36(9):4020–4026, 2012.

