A Novel Pseudo-Spectral Method to Solve Initial and Boundary Value Problems

Achamyelesh Amare Aligaz (achamyelesh@aims.ac.za)
አቻምየለሽ አማረ አሊጋዝ
African Institute for Mathematical Sciences (AIMS)

Supervised by: Professor Kailash C. Patidar
University of the Western Cape, South Africa

22 May 2014

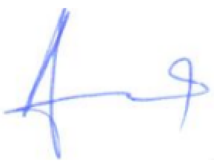
Submitted in partial fulfillment of a structured masters degree at AIMS South Africa

Abstract

In this research project, a new Fibonacci-type pseudo-spectral method for solving initial and boundary value problems is discussed. The idea of this method is to describe the solution as a linear combination of Fibonacci polynomials. Derivation of this method is done by collocating the proposed linear combination into the given initial and/or boundary value problem and then by making the residual zero at some collocation points. The applicability of this method is shown by an initial value problem and two boundary value problems, which are regular problems, and one singular perturbation problem. By these examples, it is shown that the proposed method is very effective and more accurate than the standard finite difference methods.

Declaration

I, the undersigned, hereby declare that the work contained in this research project is my original work, and that any work done by others or by myself previously has been acknowledged and referenced accordingly.



ALIGAZ Achamyelesh Amare, 22 May 2014

Contents

List of Tables

List of Figures

1. Introduction

Many ordinary differential equations with initial and/or boundary conditions can rarely be solved in closed form. For a given ordinary differential equations of the form

$$\sum_{k=0}^{m} q_k(x)y^{(k)}(x) = g(x), \tag{1.0.1}$$

with boundary or initial conditions

$$y^{(l)}(c_j) = \lambda_j, \ \ j = 0,\ 1,\ \dots,\ M. \tag{1.0.2}$$

where c_j are boundary or initial points in the given domain and $y^{(l)}$ is the l^{th} derivative of y, some numerical techniques including finite element method, finite difference method and spectral methods can be used. As mentioned in Fornberg (1996), the oldest and the simplest numerical method is the Euler's method which is derived by Leonhard Euler in 1768.

In this research project, a special type of spectral method known as Fibonacci type pseudo-spectral (collocation) method for a differential equation of the form (1.0.1) - (1.0.2) is discussed. In spectral methods, the numerical solution is a linear combination of the polynomials in the basis set of the form

$$\sum_{k=0}^{N} a_k\phi_k(x), \tag{1.0.3}$$

where the set $\{\phi_k(x)\}$ is a basis function and $a_k's$ are constant coefficients.

Scalar coefficients, a_k, can be determined by different types of spectral methods like the Tau method, Galerkin method and Pseudo-spectral method.

As explained in Boyd (1989), the choice of an appropriate set of basis is a fundamental issue in spectral methods. For instance, Fourier expansion $\{1, cos(nx), sin(nx), \dots\}$ are all periodic; thus, we use it for solutions of problems with periodic behaviours. On the other hand, Chebyshev polynomials of the first kind and Legendre polynomials are orthogonal in the interval [-1,1]; hence, expansion of these functions are useful for solutions of non-periodic problems on the range of [-1,1]. However, when the problem is defined on an unbounded interval, other ways of solving the problem are developed; these includes domain truncation, see Li et al. (2003) and choosing a basis function which are defined on an infinite interval such as Sinc functions, see Parand et al. (2009) and Hermite functions, see Yalcinbas et al. (2011). And when the problem is defined on semi-infinite interval, rational Chebyshev and Laguerre can be used, see Boyd (1989).

Recently, a set of Fibonacci polynomials is found to be the most effective basis function for the pseudo-spectral (collocation) method of solving boundary and initial value problems on any subinterval of the real axis requiring no domain translation. Fibonacci polynomials are the set of polynomials, $\{F_k(x)\}$ for $k = 0,\ 1,\ \dots$ which is given by

$$F_{k+2}(x) - xF_{k+1}(x) - F_k(x) = 0, \tag{1.0.4}$$

with the initial conditions $F_0(x) = 0$ and $F_1(x) = 1$.

As the word 'collocation' indicates, pseudo-spectral (collocation) method refers to collocating (putting) the linear combination of the basis functions of the form (1.0.3) into a differential equation of the form

(1.0.1) - (1.0.2) for the unknown function. The scalar coefficients will be determined by evaluating (1.0.1) and (1.0.2) at some collocation and initial or boundary points, respectively.

Fibonacci type pseudo-spectral (collocation) method for a differential equation of the form (1.0.1) - (1.0.2) refers to collocating a linear combination of Fibonacci polynomials, of the form

$$\sum_{k=1}^{N} a_k F_k(x), \tag{1.0.5}$$

where the set $\{F_k(x)\}$ is a sequence of Fibonacci polynomials and $a_k{}'s$ are constant coefficients, into (1.0.1) - (1.0.2) for the unknown function. After collocation, the scalar coefficients, a_k, will be determined by evaluating (1.0.1) and (1.0.2) at some discrete points called collocation points and initial or boundary points, respectively. Thus, the accuracy of the approximation and the efficiency of its implementation are closely related to the choice of trial functions and the wise use of initial or boundary conditions.

The outline of this research project goes as follows: in Chapter 2, important concepts relevant for the proposed method, with a focus on spectral methods, are discussed. Definition, some properties and applications of Fibonacci polynomials, which are relevant for the discuss of Fibonacci-type pseudo-spectral methods are discussed in Chapter 3. In Chapter 4, implementation of the proposed method is presented. The applicability of the proposed method is shown by 4 different examples and these results are presented in Chapter 5. Finally, discussion and conclusion on those results are presented in Chapter 6.

2. Important concepts relevant for the proposed method

The most common numerical techniques include finite element, finite volume, finite difference and spectral methods. As stated in Fornberg (1996), the choice of an appropriate numerical technique depends on the complexity of the domain and the required levels of accuracy. Finite element methods are particularly preferable to problems in very complex geometries (e.g. 3-D engineering structures), whereas spectral methods are cost efficient and can offer best accuracies mainly in simple geometries such as boxes and spheres. Finite difference methods perform well over a broad range of accuracy requirements and moderately complex domains. Both finite element and finite volume methods are closely related to finite difference methods. Finite element methods can frequently be seen as very convenient ways to generate and administer complex finite difference schemes and to obtain results with relatively sharp-error estimates.

Finite difference methods approximate derivatives of a function by local arguments; i.e.,

$$\frac{dy(x)}{dx} \approx \frac{y(x+h) - y(x-h)}{2h},$$

where h is a small grid spacing; these methods are typically designed to be exact for polynomials of low order. This approach focuses on local property of a function, that is a function need not to be smooth. For the comparison purpose, finite difference method will be used in this research project.

However, spectral methods focuses on global property of a function; that is, the approximated solution of the given differential equation should be a linear combination of very smooth basis functions of the form (1.0.3). This approach is powerful for many cases in which both solutions and variable coefficients are non-smooth or even discontinuous. However, some factors, like boundary conditions and irregular domains can cause difficulties or inefficiencies when using spectral methods. These methods are highly successful in several areas: turbulence modelling, weather prediction, non-linear waves and seismic modelling, see Fornberg (1996).

Important terms in spectral methods

Under this title, definitions and some properties of important terms, that usually exist in spectral methods are stated. These definitions are taken from Luke (1982).

Definition 2.1. The Dirac delta function is a areal valued function which is denoted by δ and given as

$$\delta(x) = \begin{cases} 1, & \text{if } x = 0, \\ 0, & \text{if } x \neq 0. \end{cases}$$

Definition 2.2. Let P be the space of real polynomials, we define an inner product (related to the weight $w(x)$) of the two polynomials p and q in P as

$$\langle p(x), q(x) \rangle = \int_a^b p(x)q(x)w(x)dx.$$

The norm of p is defined as

$$\|p\| = \left(\int_a^b p(x)^2 w(x) dx \right)^{1/2},$$

where w is a fixed non-negative function which does not depend on the indices m and n.

Definition 2.3. A system of real function $f_n(x)$, $n = 0,\ 1,\ 2,\ 3,\ \ldots$ is said to be orthogonal with respect to inner product defined in Definition (2.2), on the interval $[a, b]$ if

$$\langle f_m(x), f_n(x) \rangle = \begin{cases} 0, & \text{for } m \neq n, \\ \lambda_n, & \text{for } m = n. \end{cases} \tag{2.3.1}$$

Example 2.4. Chebyshev polynomials, $T_n(x) = \cos(n\theta)$, $\theta = \arccos x$; $n = 0,\ 1,\ 2,\ 3,\ \ldots$ are orthogonal with weight 1 on the interval $[0, \pi]$, since

$$\int_0^\pi \cos(nx)\cos(mx) dx = 0 \text{ if } m \neq n.$$

Example 2.5. Legendre polynomials, Hermite polynomials and Laguerre polynomials are also examples of orthogonal polynomials.

Remark 2.6. Orthogonal polynomials are the integrals of differential equations of a simple form, and can be defined as the coefficients in expansion in powers of t of suitably chosen functions $w(x, t)$, called generating functions.

Example 2.7. The function $w(x, t) = (1 - 2xt + t^2)^{1/2}$ is the generating function of the Legendre polynomials $P_n(x)$, i.e., the expansion

$$(1 - 2xt + t^2)^{1/2} = \sum_0^\infty P_n(x) t^n$$

holds for sufficiently small $|t|$.

Definition 2.8. Trial or basis function is an element of a particular basis for a function space such that every continuous function in the function space can be represented as a linear combination of basis functions.

Definition 2.9. Test functions are family of functions $(\chi_n), n = 0,\ 1,\ 2,\ \ldots,\ N$ that are used to define the smallness of the residual R , by means of the Hilbert space scalar product; i.e.,

$$(\chi_n, R) = 0, \quad \forall n \in \{0,\ 1,\ 2,\ \ldots,\ N\}.$$

Note that, a numerical solution for (1.0.1) - (1.0.2), $\tilde{y}$, is a function which satisfies (1.0.2) and makes the residual small, where

$$R := \sum_{k=0}^{m} q_k(x) \tilde{y}^{(k)}(x) - g(x). \tag{2.9.1}$$

Spectral methods

Spectral methods are one of spatial discretization methods for differential equations. Formulation of these methods for numerical solutions of ordinary differential equations is introduced by Lanczos, see Lanczos (1938). Their popularity for partial differential equations dates back to the early 1970s, see (Fornberg, 1996).

Trial functions and test functions are the fundamental components for formulation of spectral methods. The trial functions, which are linear combinations of suitable trial basis functions, are used to provide the approximated representation of the solution. The test functions are used to ensure that the differential equation and some boundary or initial conditions are satisfied as closely as possible by the truncated series expansion. This is achieved by minimizing, with respect to a suitable norm, the residual produced by using the truncated expansion instead of the exact solution. The residual accounts for the differential equation and sometimes the boundary or initial conditions, either explicitly or implicitly. An equivalent requirement is that the residual satisfy a suitable orthogonality condition with respect to each of the test functions, see Canuto (2006).

For the following discussion, recall that numerical solutions in spectral methods is a linear combination of the polynomials in the basis set of the form (1.0.3),

$$\tilde{y}(x) = \sum_{k=0}^{N} a_k \phi_k(x),$$

where the set $\{\phi_k(x)\}$ is a basis function and $a_k's$ are constant coefficients.

In spectral methods, we need to consider the following procedures, see Fornberg (1996).

(i) We choose a finite set of trial or basis functions, ϕ_k for $k = 0,\ 1,\ 2,\ \ldots,\ N$ which are easy to compute; such that

- The approximated solution, (1.0.3), must converge rapidly (at least for reasonably smooth functions);
- For given coefficients a_k, it should be easy to determine b_k such that

$$\frac{d}{dx}\left(\sum_{k=0}^{N} a_k \phi_k(x)\right) = \sum_{k=0}^{N} b_k \phi_k(x).$$

- Any solution can be represented to arbitrarily high accuracy by taking the truncation N to be sufficiently large.

(ii) We choose a set of test functions, $\chi_n(x)$ for $n = 1,\ 2,\ \ldots,\ N$, such that the residual,

$$R := \sum_{k=0}^{m} q_k(x)(\tilde{y})^k(x) - g(x); \tag{2.9.2}$$

where $\tilde{y}$ is the approximated solution of the form (1.0.3), is small.

For all spectral methods, trial functions (ϕ_k) are globally smooth functions. Whereas, the choice of test functions distinguishes between the three earliest types of spectral schemes, namely, the Galerkin, Tau and Pseudo-spectral (Collocation) methods. Some of the following ideas are taken from Fornberg (1996).

Galerkin method: In this method, test functions are the same as trial functions. Each ϕ_k satisfy the boundary condition. Hence, for differential equation of the form

$$Ly(x) = g(x),\ x \in \Omega, \tag{2.9.3}$$
$$By(c_j) = \lambda_j,\ j \in \{0,\ 1,\ \dots,\ M\} \tag{2.9.4}$$

where L and B are linear differential operators and c_j are initial or boundary conditions,

$$B\phi_k(c_j) = \lambda_j.$$

Since $\chi_n = \phi_n$, the smallness condition for the residual for all $n \in \{0,\ 1,\ 2,\ \dots,\ N\}$ is

$$\begin{aligned}
&(\phi_n,\ R) = 0 \iff (\phi_n,\ L\tilde{y} - g) = 0,\\
&\iff \left(\phi_n,\ L\sum_{k=0}^{N} a_k\phi_k\right) - (\phi_n,\ g) = 0,\\
&\iff \sum_{k=0}^{N} a_k\,(\phi_n,\ L\phi_k) - (\phi_n,\ g) = 0,\\
&\iff \sum_{k=0}^{N} L_{nk}a_k = (\phi_n,\ g)\,, \qquad (2.9.5)
\end{aligned}$$

where $L_{nk} := (\phi_n,\ L\phi_k)$.

Solving for the linear system (2.9.5) leads to the $N+1$ coefficients a_k of ϕ_k.

Tau method: Similar to Galerkin method, test functions are the same as trial functions. But, trial functions do not satisfy the boundary conditions. It requires that a_k be selected so that the boundary conditions are satisfied and make the residual orthogonal to as many of the basis functions as possible. The tau method was first used by Lanczos, see Lanczos (1938).

For differential equation of the form (2.9.3) - (2.9.4), $\chi_k = \phi_k$; but, the ϕ_k's do not satisfy the boundary condition in (2.9.4). So, for $M+1 < N+1$ initial or boundary conditions in (2.9.4) becomes

$$B\tilde{y}(c_j) = \lambda_j,\ j \in \{0,\ 1,\ \dots,\ M\}. \tag{2.9.6}$$

Hence, the system of linear equations for the $N+1$ coefficients a_k is then taken to be the $N-M$ first rows of the Galerkin system (2.9.5) plus the $M+1$ equations from (2.9.6); i.e.,

$$\begin{aligned}
&\sum_{k=0}^{N} L_{nk}a_k = (\phi_k, g)\,,\ 0 \le n \le N-M-1 \text{ and}\\
&B\tilde{y}(c_j) = \lambda_j,\ 0 \le j \le M.
\end{aligned}$$

The solution (a_k) of this system gives rise to a function

$$\tilde{y} = \sum_{k=0}^{N} a_k\phi_k.$$

Pseudo-spectral (Collocation) method: In this method, test functions are delta functions at collocation points, x_n; i.e., $\chi_n(x) = \delta(x - x_n)$ and ϕ_k does not satisfy boundary conditions. Similar to the tau method, it requires that a_k be selected so that the boundary conditions are satisfied, but make the residual zero at as many (suitably chosen) spatial points as possible. That is, for differential equation of the form (2.9.3) - (2.9.4),

$$\begin{aligned} & 0 = (\chi_n(x_n), R(x_n)) = (\delta(x_n - x_n), R(x_n)), \\ & \Longleftrightarrow R(x_n) = 0, \\ & \Longleftrightarrow L\tilde{y}(x_n) = g(x_n), \\ & \Longleftrightarrow \sum_{k=0}^{N} L\phi_k(x_n)a_k = g(x_n). \end{aligned} \tag{2.9.7}$$

The boundary or initial conditions are imposed as in the tau method. Hence, we drop $M + 1$ rows in the linear system (2.9.7) and solve the system

$$\begin{aligned} & \sum_{k=0}^{N} L\phi_k(x_n)a_k = g(x_n), \quad 0 \leq n \leq N - M - 1 \text{ and} \\ & B\tilde{y}(c_j) = \lambda_j, \quad 0 \leq j \leq M. \end{aligned}$$

Therefore, pseudo-spectral (collocation) method for differential equation of the form (1.0.1) - (1.0.2) refers to collocating a linear combination of basis functions, of the form (1.0.3), for the unknown function into (1.0.1) - (1.0.2) and evaluating it at some discrete points called collocation points and initial or boundary points, respectively. The accuracy of the approximation and the efficiency of its implementation are closely related to the choice of trial functions and the appropriate use of initial or boundary conditions.

For periodic problems we choose trigonometric function as basis functions. For non-periodic problems, orthogonal polynomials of Jacobi type with Chebyshev and Legendre polynomials are commonly used, see Fornberg (1996). Recently, Fibonacci polynomials are found to be the most effective basis functions for any problems on any bounded intervals, which is the main concern of this research project.

For linear problems, the systems of equations that arise from the Tau and Galerkin methods can sometimes be solved rapidly by favourable sparsity patterns. However, variable coefficients and non-linearities are often difficult to handle; in this case, pseudo-spectral method is applicable.

3. Fibonacci polynomials and their applications

In this chapter, definitions, some properties and applications of Fibonacci polynomials, which are vital for the implementation of the proposed method, are discussed.

Definition and some properties of Fibonacci polynomials

The following two definitions are taken from Koç et al. (2013).

Definition 3.1. For any positive real number k, the k-Fibonacci sequence, $\{F_{k,n}\}$, is defined recurrently by

$$F_{k,n+1} = kF_{k,n} + F_{k,n-1}, \text{ for } n \geq 1, \tag{3.1.1}$$

with initial conditions $F_{k,0} = 0$ and $F_{k,1} = 1$.

Definition 3.2. If k is a real variable x in (3.1.1), then $F_{k,n} = F_{x,n}$ and they correspond to the Fibonacci polynomials defined by

$$F_{n+1}(x) = \begin{cases} 1, & \text{if} \quad n = 0 \\ x, & \text{if} \quad n = 1 \\ xF_n(x) + F_{n-1}(x), & \text{if} \quad n > 1. \end{cases} \tag{3.2.1}$$

From (3.2.1), we can see that the first ten polynomials are

$$\begin{aligned}
&F_1(x) = 1, \\
&F_2(x) = x, \\
&F_3(x) = x^2 + 1, \\
&F_4(x) = x^3 + 2x, \\
&F_5(x) = x^4 + 3x^2 + 1, \\
&F_6(x) = x^5 + 4x^3 + 3x, \\
&F_7(x) = x^6 + 5x^4 + 6x^2 + 1, \\
&F_8(x) = x^7 + 6x^5 + 10x^3 + 4x, \\
&F_9(x) = x^8 + 7x^6 + 15x^4 + 10x^2 + 1 \text{ and} \\
&F_{10}(x) = x^9 + 8x^7 + 21x^5 + 20x^3 + 5x.
\end{aligned}$$

The following remark and theorem are taken from Mirzaee and Hoseini (2013).

Remark 3.3. The Fibonacci polynomials have generating function

$$G(x,t) = \frac{t}{1 - t^2 - tx}.$$

This implies that

$$\begin{aligned} G(x,t) &= \sum_{n=1}^{\infty} F_n(x)t^n, \\ &= t + xt^2 + (x^2+1)t^3 + (x^3+2x)t^4 + \dots \quad . \end{aligned}$$

Theorem 3.4. *The explicit sum formula for Fibonacci polynomials is given by*

$$F_{n+1}(x) = \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n-k}{k} x^{n-2k}, \quad n \geq 0,$$

where $\lfloor \frac{n}{2} \rfloor$ *denotes the greatest integer less than or equal to* $\frac{n}{2}$*.*

Proof. From Remark 3.3,

$$\begin{aligned} \sum_{n=0}^{\infty} F_n(x)t^n &= t\left(1 - t^2 - tx\right)^{-1}, \\ &= t\sum_{n=0}^{\infty}(x+t)^n t^n, \\ &= t\sum_{n=0}^{\infty}\sum_{k=0}^{n}\binom{n}{k}x^{n-k}t^k t^n, \\ &= \sum_{n=0}^{\infty}\sum_{k=0}^{n}\binom{n}{k}x^{n-k}t^{n+k+1}, \\ &= \sum_{n=0}^{\infty}\sum_{k=0}^{n}\frac{n!}{k!(n-k)!}x^{n-k}t^{n+k+1}, \\ &= \sum_{n=0}^{\infty}\sum_{k=0}^{\lfloor \frac{n-1}{2} \rfloor}\frac{(n-k-1)!}{k!(n-2k-1)!}x^{n-2k-1}t^n. \end{aligned}$$

Thus, equating the coefficients of t^n, we get

$$\begin{aligned} F_n(x) &= \sum_{k=0}^{\lfloor \frac{n-1}{2} \rfloor}\frac{(n-k-1)!}{k!(n-2k-1)!}x^{n-2k-1}, \quad n \geq 1 \\ &= \sum_{k=0}^{\lfloor \frac{n-1}{2} \rfloor}\binom{n-k-1}{k}x^{n-2k-1}, \quad n \geq 1. \end{aligned}$$

Which is equivalent to

$$F_{n+1}(x) = \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n-k}{k} x^{n-2k}, \quad n \geq 0.$$

□

The following two propositions are taken from Falcon and Plaza (2009).

Proposition 3.5.

$$F_n(x) = \frac{1}{n}\left[F'_{n+1}(x) + F'_{n-1}(x)\right], \ \forall n \in \mathbb{N},$$

and it follows that

$$\begin{aligned}\int_0^x F_n(x)dx &= \frac{1}{n}\left[F_{n+1}(x) + F_{n-1}(x) - F_{n+1}(0) - F_{n-1}(0)\right] \\ &= \begin{cases} \frac{1}{n}\left[F_{n+1}(x) + F_{n-1}(x)\right], & \text{if n is odd,} \\ \\ \frac{1}{n}\left[F_{n+1}(x) + F_{n-1}(x) - 2\right], & \text{if n is even.}\end{cases}\end{aligned}$$

Proof. From Theorem 3.4, we have

$$F_{n+1} = \sum_{i=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n-i}{i} x^{n-2i}, \quad \text{for} \quad n \geq 0 \quad \text{and} \tag{3.5.1}$$

$$F_{n-1} = \sum_{i=0}^{\lfloor \frac{n-2}{2} \rfloor} \binom{n-2-i}{i} x^{n-2-2i}, \quad n \geq 2. \tag{3.5.2}$$

From (3.5.1) and (3.5.2),

$$\begin{aligned} F_{n+1}(x) + F_{n-1}(x) &= x^n + \sum_{i=1}^{\lfloor \frac{n}{2} \rfloor} \binom{n-i}{i} x^{n-2i} + \sum_{i=0}^{\lfloor \frac{n-2}{2} \rfloor} \binom{n-2-i}{i} x^{n-2-2i}, \\ &= x^n + \sum_{i=1}^{\lfloor \frac{n}{2} \rfloor} \left[\binom{n-i}{i} + \binom{n-1-i}{i-1}\right] x^{n-2i}, \end{aligned} \tag{3.5.3}$$

where

$$\begin{aligned} \binom{n-i}{i} + \binom{n-1-i}{i-1} &= \frac{(n-i)!}{i!(n-2i)!} + \frac{(n-1-i)!}{(i-1)!(n-2i)!}, \\ &= \frac{(n-i)(n-i-1)!}{i(i-1)!(n-2i)!} + \frac{(n-1-i)!}{(i-1)!(n-2i)!}, \\ &= \left(\frac{(n-i-1)!}{(i-1)!(n-2i)!}\right)\left(\frac{n-i}{i} + 1\right), \\ &= \binom{n-1-i}{i-1}\left(\frac{n}{i}\right). \end{aligned} \tag{3.5.4}$$

Substituting (3.5.4) into (3.5.3), we get

$$F_{n+1}(x) + F_{n-1}(x) = x^n + n \sum_{i=1}^{\lfloor \frac{n}{2} \rfloor} \binom{n-1-i}{i-1} \frac{1}{i} x^{n-2i}. \tag{3.5.5}$$

Differentiating both sides of (3.5.5) yields

$$F'_{n+1}(x) + F'_{n-1}(x) = nx^{n-1} + n \sum_{i=1}^{\lfloor \frac{n}{2} \rfloor} \binom{n-1-i}{i-1} \frac{n-2i}{i} x^{n-1-2i},$$

which is equivalent to

$$\frac{F'_{n+1}(x)+F'_{n-1}(x)}{n} = \sum_{i=0}^{\lfloor\frac{n-1}{2}\rfloor}\binom{n-1-i}{i}x^{n-1-2i}$$
$$= F_n(x).$$

□

Proposition 3.6.

$$F_1'(x)=0 \quad \text{and} \quad F_n'(x)=\sum_{i=1}^{n-1}F_i(x)F_{n-i}(x) \quad \text{for} \quad n>1.$$

Proof. We prove it by induction. Thus

(i) When $n=2$, $F_2'(x)=F_1(x)F_1(x)=1$.

(ii) Assume it is true for $k\le n$; i.e.,

$$F_k=\sum_{i=1}^{k-1}F_i(x)F_{k-i}(x).$$

(iii) We show that it is true for $n+1$. We have that

$$\begin{aligned}
F'_{n+1} &= F_n(x)+xF_n'(x)+F_{n-1}'(x),\\
&= F_n(x)+x\sum_{i=1}^{n-1}F_i(x)F_{n-i}(x)+\sum_{i=1}^{n-2}F_i(x)F_{n-1-i}(x) \quad \text{by assumption (ii)},\\
&= F_n(x)+xF_{n-1}(x)F_1(x)+\sum_{i=1}^{n-2}xF_i(x)F_{n-i}(x)+\sum_{i=1}^{n-2}F_i(x)F_{n-1-i}(x),\\
&= F_n(x)+xF_{n-1}(x)+\sum_{i=1}^{n-2}F_i(x)\left[xF_{n-i}(x)+F_{n-1-i}(x)\right],\\
&= F_n(x)F_1(x)+F_{n-1}(x)F_2(x)+\sum_{i=1}^{n-2}F_i(x)F_{n+1-i}(x),\\
&= \sum_{i=1}^{n}F_i(x)F_{n+1-i}(x),
\end{aligned}$$

hence the proof is.

□

Approximation of continuous functions by Fibonacci polynomials

Suppose $y(x)$ is a continuous function that can be expressed in terms of the Fibonacci polynomials, $F_r(x)$ for $r = 1, 2, \dots$; i.e.,

$$y(x) = \sum_{r=1}^{\infty} a_r F_r(x), \tag{3.6.1}$$

where $a_r's$ are constant coefficients.

Then, a truncated expansion of N Fibonacci polynomials can be written

$$y(x) \approx \sum_{r=1}^{N} a_r F_r(x) = F(x)A, \tag{3.6.2}$$

where $F(x) = [F_1(x) \;\; F_2(x) \;\; \dots \;\; F_N(x)]$ and $A = [a_1 \;\; a_2 \;\; \dots \;\; a_N]^T$, see Koç et al. (2013).

Approximation of derivatives of continuous functions by Fibonacci polynomials

From Proposition 3.5, it follows that the k^{th} order derivative of the function in (3.6.1) can be written as

$$y^{(\mathsf{k})} = \sum_{r=1}^{\infty} a_r^{(k)} F_r(x), \quad k \geq 1. \tag{3.6.3}$$

When the infinite sum is truncated to N terms, we get

$$y^{(\mathsf{k})} \approx \sum_{r=1}^{N} a_r^{(k)} F_r(x) = F(x)A^{(k)}, \tag{3.6.4}$$

where $a_r^{(0)} = a_r, \quad y^{(0)}(x) = y(x)$ and $A^{(k)} = \left[a_1^{(k)} \quad a_2^{(k)} \quad \dots \quad a_N^{(k)}\right]^T$, see Koç et al. (2013).

The following proposition and corollary are taken from Koç et al. (2013).

Proposition 3.7. Let $y(x)$ and it's k^{th} order derivative be the functions given in (3.6.2) and (3.6.4), respectively. Then, there exists a relation between the Fibonacci coefficient matrices as

$$A^{(k+1)} = D^{k+1}A, \quad k = 0, \dots, m-1,$$

where D is a $N \times N$ operational matrix for the derivative defined by

$$D = [d_{i,j}] = \begin{cases} i \sin\left(\frac{(j-i)\pi}{2}\right), \text{ for } i < j, \\ \\ 0, \text{ for } i \geq j. \end{cases}$$

Proof.

$$\begin{aligned}
\sum_{r=1}^{\infty} a_r^{(k)} F_r(x) &= \sum_{r=1}^{\infty} a_r^{(k)} \left(\frac{1}{r} \left[F'_{r+1}(x) + F'_{r-1}(x) \right] \right), \quad \text{by Proposition 3.5,} \\
&= \sum_{r=1}^{\infty} a_r^{(k)} \left(\frac{1}{r} F'_{r+1}(x) \right) + \sum_{r=1}^{\infty} a_r^{(k)} \left(\frac{1}{r} F'_{r-1}(x) \right), \\
&= \sum_{r=2}^{\infty} a_{r-1}^{(k)} \left(\frac{1}{r-1} F'_r(x) \right) + \sum_{r=0}^{\infty} a_{r+1}^{(k)} \left(\frac{1}{r+1} F'_r(x) \right), \\
&= \sum_{r=2}^{\infty} a_{r-1}^{(k)} \left(\frac{1}{r-1} F'_r(x) \right) + \sum_{r=2}^{\infty} a_{r+1}^{(k)} \left(\frac{1}{r+1} F'_r(x) \right) \quad \text{since} \quad F_0'(x) = F_1'(x) = 0, \\
&= \sum_{r=2}^{\infty} \left(\frac{a_{r-1}^{(k)}}{r-1} + \frac{a_{r+1}^{(k)}}{r+1} \right) F_r'(x), \\
&= \sum_{r=2}^{\infty} \left(\frac{a_{r-1}^{(k+1)}}{r-1} + \frac{a_{r+1}^{(k+1)}}{r+1} \right) F_r(x).
\end{aligned}$$

Thus, equating the coefficients of $F_r(x)$, we get

$$a_r^{(k)} = \frac{a_{r-1}^{(k+1)}}{r-1} + \frac{a_{r+1}^{(k+1)}}{r+1} \quad \text{for} \quad r \geq 2. \tag{3.7.1}$$

From (3.7.1), we get the following recursive relations

$$a_{r+1}^{(k)} = \frac{a_r^{(k+1)}}{r} + \frac{a_{r+2}^{(k+1)}}{r+2} \quad \text{for} \ \ r \geq 1, \tag{3.7.2}$$

$$a_{r+3}^{(k)} = \frac{a_{r+2}^{(k+1)}}{r+2} + \frac{a_{r+4}^{(k+1)}}{r+4}, \tag{3.7.3}$$

$$a_{r+5}^{(k)} = \frac{a_{r+4}^{(k+1)}}{r+4} + \frac{a_{r+6}^{(k+1)}}{r+6}, \tag{3.7.4}$$

$$a_{r+m}^{(k)} = \frac{a_{r+m-1}^{(k+1)}}{r+m-1} + \frac{a_{r+m+1}^{(k+1)}}{r+m+1}, \quad \text{for} \quad m \geq 6. \tag{3.7.5}$$

From (3.7.2), we get

$$\begin{aligned}
\frac{a_r^{(k+1)}}{r} &= a_{r+1}^{(k)} - \frac{a_{r+2}^{(k+1)}}{r+2}, \\
&= a_{r+1}^{(k)} - \left(a_{r+3}^{(k)} - \frac{a_{r+4}^{(k+1)}}{r+4} \right) \quad \text{by (3.7.3)}, \\
&= a_{r+1}^{(k)} - a_{r+3}^{(k)} + \left(\frac{a_{r+4}^{(k+1)}}{r+4} \right), \\
&\ \ \vdots \\
&= a_{r+1}^{(k)} - a_{r+3}^{(k)} + a_{r+5}^{(k)} - a_{r+7}^{(k)} + \dots \quad .
\end{aligned}$$

Collectively, we can write

$$a_r^{(k+1)} = r\sum_{i=0}^{\infty}(-1)^i a_{r+2i+1}^k, \quad r = 1,\ 2,\ 3,\ \dots,\ N \tag{3.7.6}$$

Assuming that $a_r^{(k)} = 0$ for $r > N$, the system (3.7.6) can be transformed into the matrix form,

$$A^{k+1} = DA^{(k)}, \quad k = 0,\ 1,\ 2,\ \dots,\ m-1, \tag{3.7.7}$$

where $m < N$ is the order of a given differential equation.

Using (3.7.7), we write

$$\begin{aligned}
A^{(1)} &= DA, \\
A^{(2)} &= DA^{(1)} = D^2A, \\
A^{(3)} &= DA^{(2)} = D^3A, \\
&\vdots \\
A^{(k)} &= D^kA, \\
A^{(k+1)} &= D^{k+1}A,
\end{aligned}$$

where $A^{(0)} = A$. □

Corollary 3.8. The k^{th} order derivative of function of the form (3.6.2) can be expressed in terms of Fibonacci coefficients; i.e.,

$$y^{(k)}(x) = F(x)D^kA, \quad k = 0,\ 1,\ \dots,\ m,$$

where $D^0 = 1$.

Proof. From (3.6.1), we have

$$\begin{aligned}
y^{(k)}(x) &= \sum_{r=1}^{\infty} a_r^{(k)} F_r(x), \\
&\approx \sum_{r=1}^{N} a_r^{(k)} F_r(x) = F(x)A^{(k)}, \quad \text{from} \quad (3.6.4), \\
&= F(x)D^kA, \quad \text{from Proposition 3.7},
\end{aligned}$$

where $A^{(k)} = \begin{bmatrix} a_1^{(k)} & a_2^{(k)} & \dots & a_N^{(k)} \end{bmatrix}^T$.

We also note that $a_r^{(0)} = a_r$ and $y(x)^{(0)}(x) = y(x)$. □

4. Implementation of the proposed method

The idea of Fibonacci-type pseudo-spectral (collocation) method is to find an approximated solution as a linear combination of Fibonacci polynomials for differential equations of the form (1.0.1) - (1.0.2). In this chapter derivation of this method is presented.

The approximated solution of the proposed method is of the form

$$\tilde{y} = \sum_{r=1}^{N} a_r F_r(x) = F(x)A, \tag{4.0.1}$$

where $F_r(x)$ is the r^{th} Fibonacci polynomial, a_r is the scalar coefficient of $F_r(x)$,
$F(x) = [F_1(x) \quad F_2(x) \quad \dots \quad F_N(x)]$ and $A = [a_1 \quad a_2 \quad \dots \quad a_N]^T$.
From (3.8) and (4.0.1), it follows that

$$\tilde{y}^{(k)} = F(x)A^{(k)} = F(x)D^k A. \tag{4.0.2}$$

Then, substituting $\tilde{y}$ for y in (1.0.1) and evaluate the resulting differential equation at N collocation points,

$$x_i = a + \frac{b-a}{N-1}(i-1), \quad i = 1,\ 2,\ \dots,\ N,\ \ a \le x_i \le b,$$

we get

$$\left.\begin{aligned}
&q_0(x_1)\tilde{y}(x_1) + q_1(x_1)\tilde{y}^{(1)}(x_1) + \cdots + q_m(x_1)\tilde{y}^{(m)}(x_1) = g(x_1),\\
&q_0(x_2)\tilde{y}(x_2) + q_1(x_2)\tilde{y}^{(1)}(x_2) + \cdots + q_m(x_2)\tilde{y}^{(m)}(x_2) = g(x_2),\\
&q_0(x_3)\tilde{y}(x_3) + q_1(x_3)\tilde{y}^{(1)}(x_3) + \cdots + q_m(x_3)\tilde{y}^{(m)}(x_3) = g(x_3),\\
&\qquad\vdots\\
&q_0(x_N)\tilde{y}(x_N) + q_1(x_N)\tilde{y}^{(1)}(x_N) + \cdots + q_m(x_N)\tilde{y}^{(m)}(x_N) = g(x_N).
\end{aligned}\right\} \tag{4.0.3}$$

Equivalently, (4.0.3) can be re-written as

$$\begin{pmatrix} q_0(x_1) & 0 & \dots & 0 \\ 0 & q_0(x_2) & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & q_0(x_N) \end{pmatrix} \begin{pmatrix} \tilde{y}(x_1) \\ \tilde{y}(x_2) \\ \vdots \\ \tilde{y}(x_N) \end{pmatrix} + \begin{pmatrix} q_1(x_1) & 0 & \dots & 0 \\ 0 & q_1(x_2) & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & q_1(x_N) \end{pmatrix} \begin{pmatrix} \tilde{y}^{(1)}(x_1) \\ \tilde{y}^{(1)}(x_2) \\ \vdots \\ \tilde{y}^{(1)}(x_N) \end{pmatrix} +$$

$$\dots \quad + \begin{pmatrix} q_m(x_1) & 0 & \dots & 0 \\ 0 & q_m(x_2) & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & q_m(x_N) \end{pmatrix} \begin{pmatrix} \tilde{y}^{(m)}(x_1) \\ \tilde{y}^{(m)}(x_2) \\ \vdots \\ \tilde{y}^{(m)}(x_N) \end{pmatrix} = \begin{pmatrix} g(x_1) \\ g(x_2) \\ \vdots \\ g(x_N) \end{pmatrix}. \tag{4.0.4}$$

From (4.0.1) and (4.0.2), (4.0.4) becomes

$$\begin{pmatrix} q_0(x_1) & 0 & \dots & 0 \\ 0 & q_0(x_2) & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & q_0(x_N) \end{pmatrix} \begin{pmatrix} F(x_1)A \\ F(x_2)A \\ \vdots \\ F(x_N)A \end{pmatrix} + \begin{pmatrix} q_1(x_1) & 0 & \dots & 0 \\ 0 & q_1(x_2) & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & q_1(x_N) \end{pmatrix} \begin{pmatrix} F(x_1)DA \\ F(x_2)DA \\ \vdots \\ F(x_N)DA \end{pmatrix} +$$

$$\dots + \begin{pmatrix} q_m(x_1) & 0 & \dots & 0 \\ 0 & q_m(x_2) & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & q_m(x_N) \end{pmatrix} \begin{pmatrix} F(x_1)D^mA \\ F(x_2)D^mA \\ \vdots \\ F(x_N)D^mA \end{pmatrix} = \begin{pmatrix} g(x_1) \\ g(x_2) \\ \vdots \\ g(x_N) \end{pmatrix} := G. \tag{4.0.5}$$

Let

$$q_k = \begin{pmatrix} q_k(x_1) & 0 & \dots & 0 \\ 0 & q_k(x_2) & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & q_k(x_N) \end{pmatrix}, \text{ for } k = 0,\ 1,\ \dots,\ m \text{ and}$$

$$F = \begin{pmatrix} F(x_1) \\ F(x_2) \\ \vdots \\ F(x_N) \end{pmatrix},$$

where $F(x_k) = [F_1(x_k) \quad F_2(x_k) \quad \dots \quad F_N(x_k)]$. Then (4.0.5) can be re-written as

$$\sum_{k=0}^{m} \left(q_k F D^k\right) A =: G \tag{4.0.6}$$

Assuming that

$$W = \sum_{k=0}^{m} \left(q_k F D^k\right), \tag{4.0.7}$$

equation (4.0.6) can be written as

$$WA =: G, \tag{4.0.8}$$

which is the matrix representation of the system of linear equations in (4.0.3).

The augmented matrix of (4.0.8) will be

$$[W : G]. \tag{4.0.9}$$

Next, we incorporate the initial and (or) boundary conditions

$$y^{(l)}(c_j) = F(c_j)D^{(l)}A = \lambda_j, \quad j = 0,\ 1,\ \dots,\ M. \tag{4.0.10}$$

where c_j are boundary or initial points in the given domain and $l = 0,\ 1,\ \ldots,\ m$.

Let

$$F(c_j)D^{(l)} = [u_{j1} \quad u_{j2} \quad \ldots \quad u_{jN}]\,. \tag{4.0.11}$$

Therefore, the augmented matrix of (4.0.10) will be

$$[u_{j1} \quad u_{j2} \quad \ldots \quad u_{jN} : \lambda_j]\,, \ \ j = 0,\ 1,\ \ldots,\ M. \tag{4.0.12}$$

Then, we replace the first and the last rows of (4.0.9) by the rows of (4.0.12). As a result, we get a new augmented matrix $[W^* : G^*]$; provided that det$(W^*) \neq 0$. Finally, the vector A (vector of the coefficients, a_r) is determined by solving $[W^* : G^*]$. So that the approximated solution will be the product of $F(x)$ and A; i.e.,

$$\begin{aligned}
\tilde{y} &= F(x) * A, \\
&= [F_1(x) \quad F_2(x) \quad \ldots \quad F_N(x)] * [a_1 \quad a_2 \quad \ldots \quad a_N]^T, \\
&= \sum_{r=1}^{N} a_r F_r(x),
\end{aligned}$$

which is of the form (4.0.1).

5. Numerical results

In this chapter, some examples are given to illustrate the applicability of the method and all of them are performed using SAGE software. The absolute point-wise errors in tables are the values of absolute values of exact solution minus numerical solution at selected points and the point-wise errors are approximated to three decimal places. For the sake of comparison, the standard finite difference method (FDM) with spline interpolation is used for the first three examples. Similar to the previous chapters, N denotes the number of collocation points; whereas, I denotes the number of subintervals. In fact, $I = N - 1$. The first three and the last examples are taken from Koç et al. (2013) and Patidar (2005), respectively.

Example 1

Consider the linear non-homogeneous problem

$$y^{''}(x) + xy^{'}(x) - 2y(x) = x\cos(x) - 3\sin(x), \quad x \in [-1, 1]$$

with given conditions

$$y(0) = 0 \quad \text{and} \quad y^{'}(0) = 1,$$

for which the exact solution is $y^e(x) = \sin(x)$.

Figure 5.1 shows plots of the exact and the numerical solution of Example 1 when $N = 7$. Table 5.1 - Table 5.4 show point-wise errors observed when the proposed method is used. Table 5.5 and Table 5.6 show point-wise error observed when FDM is used.

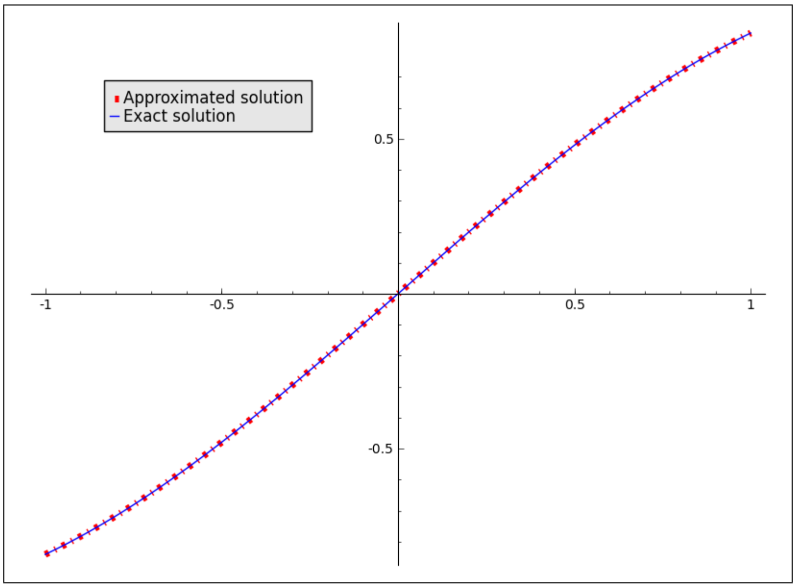


Figure 5.1: Numerical and exact solutions for Example 1 when $N = 7$.

Table 5.1: Point-wise errors of the proposed method for Example 1 at collocation points when $N = 8$.

x	-1	-5/7	-3/7	-1/7	1/7	3/7	5/7	1
Error	3.91E-7	1.07E-8	2.77E-9	1.22E-10	1.22E-10	2.77E-9	1.07E-8	3.91E-7

Table 5.2: Point-wise errors of the proposed method for Example 1 at collocation points when $N = 16$.

x	-1	-13/15	-11/15	-3/5	-7/15	-1/3	-1/5	-1/15
Error	5.55E-16	5.55E-16	3.33E-16	3.33E-16	2.22E-16	2.22E-16	1.11E-16	4.16E-17

x	1/15	1/5	1/3	7/15	3/5	11/15	13/15	1
Error	4.16E-17	1.67E-16	2.22E-16	5.55E-16	3.33E-16	5.55E-16	3.33E-16	5.55E-16

Table 5.3: Point-wise errors of the proposed method for Example 1.

x	$I = 4$	$I = 8$	$I = 16$
-0.8	8.08E-4	2.37E-8	1.11E-16
-0.6	8.33E-4	3.3E-9	1.11E-16
-0.4	3.52E-4	5.05E-9	0
-0.2	5.2E-5	1.71E-9	2.78E-17
0	0	0	0
0.2	5.2E-5	1.71E-9	2.78E-17
0.4	3.52E-4	5.05E-9	1.11E-16
0.6	8.33E-4	3.3E-9	1.11E-16
0.8	8.08E-4	2.37E-8	1.11E-16

Table 5.4: Maximum errors of the proposed method for Example 1.

Number of sub-intervals (I)	$I = 4$	$I = 8$	$I = 16$
Maximum error	8.33E-4	2.37E-8	1.11E-16

Table 5.5: Point-wise errors of the FDM for Example 1.

x	$I = 125$	$I = 250$	$I = 500$	$I = 1000$
-0.8	4.83E-7	1.21E-7	3.02E-8	7.54E-9
-0.6	7.35E-7	1.84E-7	4.59E-8	1.15E-8
-0.4	7.03E-7	1.76E-7	4.39E-8	1.1E-8
-0.2	4.23E-7	1.06E-7	2.64E-8	6.6E-9
0	0	0	0	0
0.2	4.23E-7	1.06E-7	2.64E-8	6.6E-9
0.4	7.03E-7	1.76E-7	4.39E-8	1.1E-8
0.6	7.35E-7	1.84E-7	4.59E-8	1.15E-8
0.8	4.83E-7	1.21E-7	3.02E-8	7.54E-9

Table 5.6: Maximum errors of the FDM for Example 1.

Number of sub-intervals (I)	$I = 125$	$I = 250$	$I = 500$	$I = 1000$
Maximum error	7.35E-7	1.84E-7	4.59E-8	1.15E-8

Example 2

Consider the linear homogeneous boundary value problem

$$-y''(x) = (2 - 4x^2)y, \quad x \in [0, 1]$$

with boundary conditions

$$y'(0) = 0 \quad \text{and} \quad y'(1) = -2/e,$$

for which the exact solution is $y^e(x) = e^{-x^2}$.

Figure 5.2 shows plots of the exact and the numerical solution of Example 2 when $N = 7$. Table 5.7 - Table 5.10 show point-wise errors observed when the proposed method is used. Table 5.11 and Table 5.12 show point-wise error observed when FDM is used.

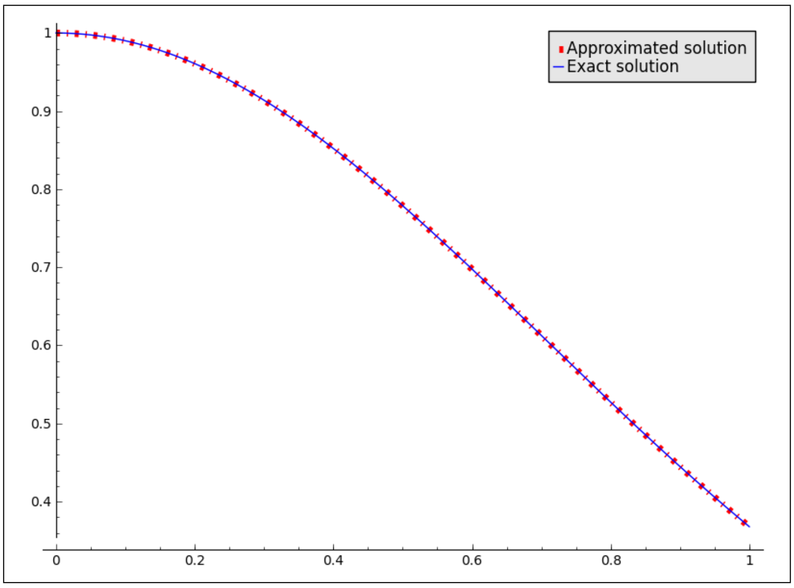


Figure 5.2: Numerical and exact solutions for Example 2 when $N = 7$.

Table 5.7: Point-wise errors of the proposed method for Example 2 at collocation points when $N = 8$.

x	0	1/7	2/7	3/7	4/7	5/7	6/7	1
Error	5.85E-4	5.84E-4	5.64E-4	5.25E-4	4.72E-4	4.13E-4	3.54E-4	3.12E-4

Table 5.8: Point-wise errors of the proposed method for Example 2 at collocation points when $N = 16$.

x	0	1/15	2/15	1/5	4/5	1/3	2/5	7/15
Error	6.27E-10	6.36E-10	6.42E-10	6.43E-10	6.39E-10	6.29E-10	6.16E-10	5.99E-10

x	8/15	3/5	2/3	11/15	4/5	13/15	14/15	1
Error	5.78E-10	5.56E-10	5.32E-10	5.07E-10	4.83E-10	4.6E-10	4.4E-10	4.25E-10

Table 5.9: Point-wise errors of the proposed method for Example 2.

x	$I = 4$	$I = 8$	$I = 16$
0.1	1.07E-2	5.0E-5	1.69E-11
0.2	1.43E-2	5.85E-5	1.77E-11
0.3	1.82E-2	6.56E-5	1.81E-11
0.4	2.15E-2	7.16E-5	1.82E-11
0.5	2.44E-2	7.67E-5	1.81E-11
0.6	2.7E-2	8.11E-5	1.78E-11
0.7	2.97E-2	8.5E-5	1.74E-11
0.8	3.26E-2	8.89E-5	1.69E-11
0.9	3.52E-2	9.34E-5	1.66E-11

Table 5.10: Maximum errors of the proposed method for Example 2.

Number of sub-intervals (I)	$I = 4$	$I = 8$	$I = 16$
Maximum error	3.52E-2	9.34E-5	1.82E-11

Table 5.11: Point-wise errors of the FDM for Example 2.

x	$I = 125$	$I = 250$	$I = 500$	$I = 1000$
0.1	1.23E-6	3.07E-7	7.68E-8	1.92E-8
0.2	1.83E-6	4.58E-7	1.14E-7	2.86E-8
0.3	1.89E-6	4.72E-7	1.18E-7	2.95E-8
0.4	1.54E-6	3.84E-7	9.6E-8	2.4E-8
0.5	9.48E-7	2.37E-7	5.93E-8	1.48E-8
0.6	3.08E-7	7.70E-8	1.93E-8	4.82E-9
0.7	2.17E-7	5.43E-8	1.36E-8	3.39E-9
0.8	4.95E-7	1.24E-7	3.1E-8	7.74E-9
0.9	4.38E-7	1.09E-7	2.74E-8	6.84E-9

Table 5.12: Maximum errors of the FDM for Example 2.

Number of sub-intervals (I)	$I = 125$	$I = 250$	$I = 500$	$I = 1000$
Maximum error	1.89E-6	4.72E-7	1.18E-7	2.95E-8

Example 3

Consider the linear non-homogeneous boundary value problem

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

with boundary conditions

$$y^{'}(0) = 0 \quad \text{and} \quad y^{'}(1) = \sin(2)$$

for which the exact solution is $y^e(x) = \sin^2(x)$.

Figure 5.3 shows plots of the exact and the numerical solution of Example 3 when $N = 7$. Table 5.13 - Table 5.16 show point-wise errors observed when the proposed method is used. Table 5.17 and Table 5.18 show point-wise error observed when FDM is used.

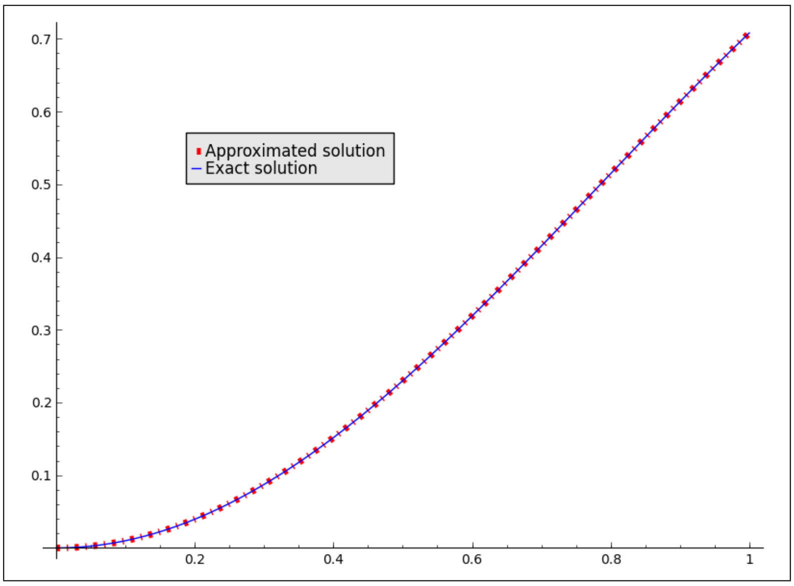


Figure 5.3: Numerical and exact solutions for Example 3 when $N = 7$.

Table 5.13: Point-wise errors of the proposed method for Example 3 at collocation points when $N = 8$.

x	0	1/7	2/7	3/7	4/7	5/7	6/7	1
Error	4.84E-6	7.97E-6	1.14E-5	1.39E-5	1.53E-5	1.55E-5	1.43E-5	1.27E-5

Table 5.14: Point-wise errors of the proposed method for Example 3 at collocation points when $N = 16$.

x	0	1/15	2/15	1/5	4/5	1/3	2/5	7/15
Error	4.26E-14	4.13E-14	3.9E-14	3.61E-14	3.26E-14	2.86E-14	2.41E-14	1.91E-14

x	8/15	3/5	2/3	11/15	4/5	13/15	14/15	1
Error	1.4E-14	8.72E-15	3.16E-15	2.16E-15	7.77E-15	1.3E-14	1.8E-14	2.14E-14

Table 5.15: Point-wise errors of the proposed method for Example 3.

x	$I = 4$	$I = 8$	$I = 16$
0.1	1.32E-2	2.58E-6	1.24E-15
0.2	1.09E-2	2.07E-6	1.51E-15
0.3	7.95E-3	1.49E-6	1.72E-15
0.4	4.78E-3	8.45E-7	1.86E-15
0.5	1.53E-3	1.68E-7	1.89E-15
0.6	1.79E-3	5.16E-7	1.94E-15
0.7	5.14E-3	1.18E-6	1.83E-15
0.8	8.37E-3	1.79E-6	1.78E-15
0.9	1.1E-2	2.35E-6	1.55E-15

Table 5.16: Maximum errors of the proposed method for Example 3.

Number of sub-intervals (I)	$I = 4$	$I = 8$	$I = 16$
Maximum error	1.32E-2	2.58E-6	1.94E-15

Table 5.17: Point-wise errors of the FDM for Example 3.

x	$I = 125$	$I = 250$	$I = 500$	$I = 1000$
0.1	1.91E-6	4.77E-7	1.19E-7	2.98E-8
0.2	3.32E-6	8.31E-7	2.08E-7	5.19E-8
0.3	4.22E-6	1.05E-6	2.64E-7	6.59E-8
0.4	4.59E-6	1.15E-6	2.87E-7	7.17E-8
0.5	4.49E-6	1.12E-6	2.8E-7	7.01E-8
0.6	3.98E-6	9.94E-7	2.49E-7	6.21E-8
0.7	3.15E-6	7.88E-7	1.97E-7	4.93E-8
0.8	2.13E-6	5.33E-7	1.33E-7	3.33E-8
0.9	1.04E-6	2.6E-7	6.49E-8	1.62E-8

Table 5.18: Maximum errors of the FDM for Example 3.

Number of sub-intervals (I)	$I = 125$	$I = 250$	$I = 500$	$I = 1000$
Maximum error	4.59E-6	1.15E-6	2.87E-7	7.17E-8

Example 4

Consider the following self-adjoint singularly perturbed two point boundary value problem

$$-\epsilon(a(x)y^{'})^{'} + b(x)y = g(x), \quad x \in [0, 1]$$

with boundary conditions

$$y(0) = y(1) = 0$$

where $0 < \epsilon < 1$ and $g(x)$, $a(x)$ and $b(x)$ are sufficiently smooth functions satisfying the conditions

$$a(x) \geq a \quad b(x) \geq b > 0.$$

For $a(x) = 1$, $b(x) = 1 + x(1-x)$, and
$g(x) = 1 + x(1-x) + (2\sqrt{\epsilon} - x^2(1-x))e^{\frac{-1+x}{\sqrt{\epsilon}}} + (2\sqrt{\epsilon} - x(1-x)^2)e^{\frac{-x}{\sqrt{\epsilon}}}$, the exact solution is given by

$$y^e(x) = 1 + (x-1)e^{\frac{-x}{\sqrt{\epsilon}}} - xe^{\frac{x-1}{\sqrt{\epsilon}}}.$$

Plots of the exact and numerical solutions of Example 4, for different ϵ and different values of I, are shown in Figure 5.4 - Figure 5.6. Table 5.19 shows the Maximum errors observed when the proposed method is used.

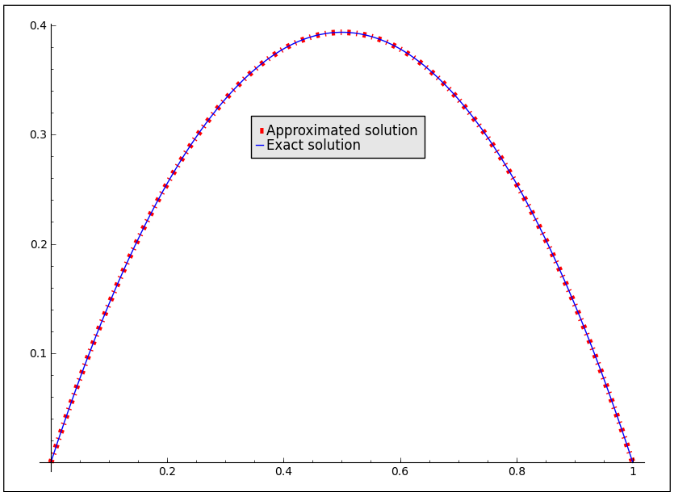


Figure 5.4: Numerical and exact solutions for Example 4 when $I = 4$ and for $\epsilon = 1$.

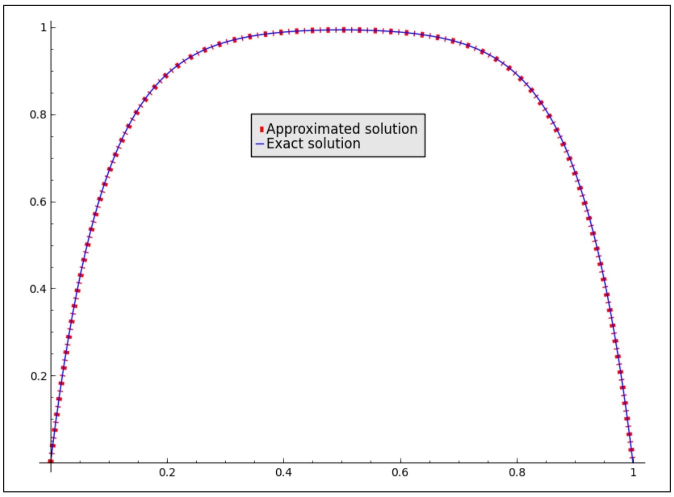


Figure 5.5: Numerical and exact solutions for Example 4 when $I = 8$ and for $\epsilon = 10^{-2}$.

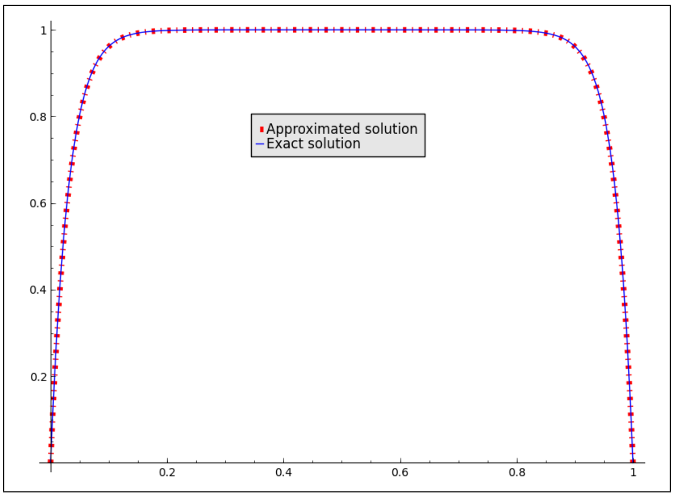


Figure 5.6: Numerical and exact solutions for Example 4 when $I = 16$ and for $\epsilon = 10^{-3}$.

Table 5.19: Maximum errors of the proposed method for Example 4.

ϵ	$I = 5$	$I = 10$
1	3.51E-5	3.9E-13
10^{-1}	3.0E-3	2.82E-8
10^{-3}	1.77E-2	3.69E-5
10^{-6}	1.98E-5	2.76E-6
10^{-9}	1.98E-8	3.97E-9

In the next chapter we discuss the above results.

6. Discussion and Conclusion

Through various numerical simulations, it is shown that the proposed method is very efficient and perform much better than the standard Finite difference method. To be more specific, as shown in Table 5.6, the maximum error obtained for Example 1 by using finite difference method (FDM) for $I = 125$, $I = 250$, $I = 500$ and $I = 1000$ is $7.35E-7$, $1.84E-7$, $4.59E-8$, $1.15E-8$, respectively. Whereas, the maximum error of proposed method for $I = 8$ and $I = 9$ is $2.37E-8$ and $7.16E-11$, respectively. Which implies that taking 8 and 9 subintervals in the proposed method have a better accuracy than taking 500 and 1000 subintervals, respectively in FDM. Furthermore, as shown in Table 5.12, the maximum error obtained for Example 2 by using finite difference method (FDM) for $I = 125$, $I = 250$, $I = 500$, $I = 1000$ is $1.89E-6$, $4.72E-7$, $1.18E-7$ and $2.95E-8$, respectively. Whereas, the maximum error of proposed method for $I = 11$ and $I = 14$ is $1.1E-6$ and $1.13E-9$, respectively. Which implies that taking 11 and 14 subintervals in the proposed method have a better accuracy than taking 125 and 1000 subintervals, respectively in FDM. From Table 5.18, we see that the maximum error obtained for Example 3 by using finite difference method (FDM) for $I = 125$, $I = 250$, $I = 500$, $I = 1000$ is $4.59E-6$, $1.15E-6$, $2.87E-7$ and $7.17E-8$, respectively. Whereas, the maximum error of proposed method for $I = 8$ and $I = 9$ and $I = 10$ is $2.58E-6$, $1.23E-7$ and $1.64E-8$, respectively. Which implies that taking 8, 9 and 10 subintervals in the proposed method have a better accuracy than taking 125, 500 and 1000 subintervals, respectively, in FDM. The results for the last example also shows that the proposed method is much better than the results obtained by Patidar (2005).

From the above discussion, we deduce that the proposed method performs much better than other contemporary methods. However, the derivation of the scheme is very tedious. In future, we will look for ways to derive this methods more efficiently.

Acknowledgements

My special thanks goes to the almighty God for his grace that dwelt in me and for his kindness and love unto me. Next, I would like to thank my husband, Abebe Menedo, for his great thoughts, encouragement and for taking a good care of our son, Jonathan Abebe. I'm also grateful to my sister, Ejigayehu Amare, who is also looking after my son and for her encouragement.

My appreciation also goes to a visionary person Prof. Neil Turock, the founder of AIMS. I would like to thank also Prof. Barry Green, the director of AIMS, and Prof. Jeff Sanders, academic director of AIMS, for their understanding and kindness.

I'm also grateful to my supervisor Prof. Kailash C. Patidar for his guidance and constructive comments that helped me to successfully accomplish my research project. I'm also grateful to Jan Groenewald, IT director of AIMS, and our tutor Evans Doe for their encouragement and assistance in my stay at AIMS. I would like to thank also Frances Aron and our tutor Hilaire Mambo for proofreading my research project.

I would like to thank also my friends especially Amos and Belthasara for their encouragement and being good friends in my stay at AIMS.

References

J. P. Boyd. *Chebyshev and Fourier Spectral Methods*. Springer-Verlag, New York, 1989.

C. Canuto. *Spectral Methods : Fundamentals in Single Domains*. Scientific computation. Springer, Berlin, 2006.

S. Falcon and Á. Plaza. On k-fibonacci sequences and polynomials and their derivatives. *Chaos, Solitons and Fractals*, 39(3):1005 – 1019, 2009.

B. Fornberg. *A Practical Guide to Pseudospectral Methods*. Cambridge University Press, New York, 1996.

A. B. Koç, M. Çakmak, A. Kurnaz1, and K. Uslu. A new fibonacci type collocation procedure for boundary value problems. *a SpringerOpen Journal*, 262(3):1005 – 1019, 2013.

A. Kurt, S. Yalçinbas, and M. Sezer. Fibonacci collocation method for solving high-order linear fredholm integro-differential-difference equations. *International Journal of Mathematics and Mathematical Sciences*, 2013, 2013.

C. Lanczos. Trigonometric interpolation of empirical and analytical functions. *Journal of Mathematical Physics*, 17(3):123–199, 1938.

Li, Huiyuan, and Ma. Shifted chebyshev collocation domain truncation for solving problems on an infinite interval. *Journal of Scientific Computing*, 18(2):191–213, 2003.

Y. Luke. *The Special Functions and Their Approximations*. Number v. 2 in Mathematics in science and engineering. Acad. Press, 1982.

F. Mirzaee and S. F. Hoseini. Solving singularly perturbed differential-difference equations arising in science and engineering with fibonacci polynomials. *Results in Physics*, 3(0):134 – 141, 2013. ISSN 2211-3797.

K. Parand, M. Dehghan, and A. Pirkhedri. Sinc-collocation method for solving the blasius equation. *Physics Letters A*, 373(44):4060 – 4065, 2009.

K. Patidar. High order fitted operator numerical method for self adjoint singular perturbation problems. *Applied Mathematics and Computation*, 171(1):547–566, 2005.

Yalcinbas, Salih, Aynigul, Muge, and M. Sezer. A collocation method using hermite polynomials for approximate solution of pantograph equations. *Journal of The Franklin Institute*, 348(6):1128–1139, 2011.