Turk. J. Math. Comput. Sci.
15(1)(2023) 42–53
© MatDer
DOI : 10.47000/tjmcs.960168

Fibonacci Collocation Method for Solving a Class of Nonlinear Differential Equations

Musa Çakmak[1], Sertan Alkan[2,*]

[1] *Department of Accounting and Tax Applications, Yayladağı Social Sciences Vocational School , Hatay Mustafa Kemal University, 31500, Yayladagi, Hatay, Turkey.*
[2] *Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Iskenderun Technical University, 31200, Iskenderun, Hatay, Turkey.*

Received: 30-06-2021 • Accepted: 24-11-2022

Abstract. In this study, a collocation method based on Fibonacci polynomials is used for approximately solving a class of nonlinear differential equations with initial conditions. The problem is firstly reduced into a nonlinear algebraic system via collocation points, later the unknown coefficients of the approximate solution function are calculated. Also, some problems are presented to test the performance of the proposed method by using error functions. Additionally, the obtained numerical results are compared with exact solutions of the test problems and approximate ones obtained with other methods in literature.

2010 AMS Classification: 34B15, 34G20, 35C11, 65L60, 65L80

Keywords: Fibonacci collocation method, nonlinear differential equations, initial value problems.

1. Introduction

Solving nonlinear differential equations is highly important because of their role in the modeling of scientific phenomena and engineering. Due to the difficulties on obtaining the analytical solutions, several numerical methods are developed to solve those equations approximately. Some of the applied numerical methods on the approximate solutions of nonlinear differential equations are as follows: Variational iteration method [13], operational matrix method based on Bernoulli polynomials [21], optimized decomposition method [19], homotopy analysis method [20].

Additionally, in [11], the authors used to Fictitious time integration method for solving time-fractional telegraph equation. In [1], reproducing kernel method is applied to Thomas-Fermi equation that is a nonlinear differential equation. The paper given by [9] deals with that the application to convective-radiative-conduction fin problem of geometric numerical integration method. In [2], reproducing kernel method is used for the Poisson-Boltzmann equation. In [22], the post-buckling analysis of shear-deformable prismatic columns under uniform compression is studied using the Generalized Beam Theory. In [10], the group preserving scheme and the reproducing kernel method are investigated. In [12], generalized squared remainder minimization method is used for solving multi-term fractional differential equations.

In [14], Fibonacci collocation method is applied to linear differential-difference equations. Similarly, in [15], the high-order linear Fredholm integro-differential-difference equations are solved by using Fibonacci collocation method. In [16], a class of systems of linear Fredholm integro-differential equations is studied by the method. The paper given

*Corresponding Author
Email addresses: musacakmak@mku.edu.tr (M. Çakmak), sertan.alkan@iste.edu.tr (S. Alkan)

by [17] deals with that the application of Fibonacci collocation method to singularly perturbed differential-difference equations. Also, in [18], Fibonacci collocation method is used for approximately solving a class of systems of high-order linear Volterra integro-differential equations.

In this paper, the Fibonacci collocation method is developed for solving the following class of nonlinear differential equation:

$$\sum_{k=0}^{m}\sum_{r=0}^{n} P_{kr}(x)u^{r}\,(x)\,u^{(k)}(x) + \sum_{k=1}^{m}\sum_{r=1}^{n} Q_{kr}(x)u^{(r)}\,(x)\,u^{(k)}(x) = g(x), \text{ for } \quad a \leq x \leq b, \tag{1.1}$$

according to the following initial conditions

$$\sum_{k=0}^{m}\left[a_{jk}u^{(k)}(a) + b_{jk}u^{(k)}(b)\right] = \delta_j, \qquad j = 0, 1, \tag{1.2}$$

where $u^{(0)}(x) = u(x),\ u^{0}(x) = 1$ and $u(x)$ is an unknown function. $P_{kr}(x)$, $Q_{kr}(x)$ and $g(x)$ are given continuous functions on interval $[0, 1]$, a_{jk}, b_{jk} and δ_j are suitable constants. Our goal is to get the approximate solution as the truncated Fibonacci series defined by

$$u\,(x) = \sum_{n=1}^{N+1} c_n F_n(x), \tag{1.3}$$

where $F_n(x)$ denotes the Fibonacci polynomials; c_n $(1 \leq n \leq N + 1)$ are the unknown coefficients for Fibonacci polynomial, and N is any positive integer which possess $N \geq m$.

The paper consists of six sections. In Section 2, and basic properties and definitions related to Fibonacci polynomials are presented. In Section 3, the fundamental matrix forms of Fibonacci collocation method by using fundamental relations of Fibonacci polynomials are constructed to obtain the approximate solutions for the given class of nonlinear differential equations. In section 4, two error estimation functions are formulated. In Section 5, five test problems are presented and the method are tested using error estimation functions. Finally, conclusions are given in Section 6.

2. Properties of Fibonacci Polynomials

The *Fibonacci polynomials* were studied by Falcon and Plaza [4, 5]. The recurrence relation of those polynomials is defined by

$$F_n(x) = xF_{n-1}(x) + F_{n-2}(x),$$

for $n \geqslant 3$, , $F_1(x) = 1$, $F_2(x) = x$. The properties were further investigated by Falcon and Plaza in [4, 5]. The first few Fibonacci polynomials are

$$\begin{aligned}
F_1(x) &= 1, \\
F_2(x) &= x, \\
F_3(x) &= x^2 + 1, \\
F_4(x) &= x^3 + 2x, \\
F_5(x) &= x^4 + 3x^2 + 1, \\
F_6(x) &= x^5 + 4x^3 + 3x, \\
F_7(x) &= x^6 + 5x^4 + 6x^2 + 1, \\
F_8(x) &= x^7 + 6x^5 + 10x^3 + 4x, \\
&\ \ \vdots
\end{aligned} \tag{2.1}$$

3. Fundamental Relations

Let us assume that linear combination of Fibonacci polynomials (1.3) is an approximate solution of Eq (1.1). Our purpose is to determine the matrix forms of Eq (1.1) by using (1.3). Firstly, we can write Fibonacci polynomials (2.1) in the matrix form

$$\mathbf{F}\,(x) = \mathbf{T}\,(x)\,\mathbf{M}, \tag{3.1}$$

where $\mathbf{F}(x) = [F_1(x)\ F_2(x)\cdots F_{N+1}(x)]$, $\mathbf{T}(x) = \left(1\ x\ x^2\ x^3 ... x^N\right)$, $\mathbf{C} = (c_1\ c_2\ \cdots c_{N+1})^T$ and

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & ... \\ 0 & 1 & 0 & 2 & 0 & 3 & 0 & 4 & 0 & ... \\ 0 & 0 & 1 & 0 & 3 & 0 & 6 & 0 & 10 & ... \\ 0 & 0 & 0 & 1 & 0 & 4 & 0 & 10 & 0 & ... \\ 0 & 0 & 0 & 0 & 1 & 0 & 5 & 0 & 15 & ... \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 6 & 0 & ... \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 7 & ... \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & ... \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ... & ... \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

The matrix form of (1.3) by a truncated Fibonacci series is given by ´

$$u(x) = \mathbf{F}(x)\,\mathbf{C}. \tag{3.2}$$

By using (3.1) and (3.2), the matrix relation is expressed as

$$\begin{aligned} u(x) &\cong u_N(x) = \mathbf{T}(x)\,\mathbf{MC}, \\ u'(x) &\cong u'_N(x) = \mathbf{T}(x)\mathbf{BMC}, \\ u''(x) &\cong u''_N(x) = \mathbf{T}(x)\,\mathbf{B}^2\mathbf{MC}, \\ &... \\ u^{(k)}(x) &\cong u_N^{(k)}(x) = \mathbf{T}^k(x)\,\mathbf{B}^k\mathbf{MC}. \end{aligned} \tag{3.3}$$

Also, the relations between the matrix $\mathbf{T}(x)$ and its derivatives $\mathbf{T}'(x)$, $\mathbf{T}''(x)$,...,$\mathbf{T}^{(k)}(x)$ are

$$\begin{aligned} \mathbf{T}'(x) &= \mathbf{T}(x)\,\mathbf{B},\ \mathbf{T}''(x) = \mathbf{T}(x)\,\mathbf{B}^2, \\ \mathbf{T}'''(x) &= \mathbf{T}(x)\,\mathbf{B}^3, ..., \mathbf{T}^{(k)}(x) = \mathbf{T}(x)\,\mathbf{B}^k, \end{aligned} \tag{3.4}$$

where

$$\mathbf{B} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 5 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & N \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \mathbf{B}^0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 \end{bmatrix},$$

$$\mathbf{T} = \begin{bmatrix} \mathbf{T}(x_0) \\ \mathbf{T}(x_1) \\ \vdots \\ \mathbf{T}(x_N) \end{bmatrix} = \begin{bmatrix} 1 & x_0 & ... & x_0^N \\ 1 & x_1 & ... & x_1^N \\ 1 & \vdots & ... & \vdots \\ 1 & x_N & ... & x_N^N \end{bmatrix}.$$

By using (3.3) and (3.4), we obtain the following relation

$$u^{(k)}(x) = \mathbf{T}(x)\,\mathbf{B}^k\mathbf{MC}. \tag{3.5}$$

By substituting the Fibonacci collocation points given by

$$x_i = a + \frac{(b-a)\,i}{N},\ i = 0, 1, ...N \tag{3.6}$$

into Eq(3.5), we obtain

$$u^{(k)}(x_i) = \mathbf{T}(x_i)\,\mathbf{B}^k\mathbf{MC},\ k = 0, 1, ..., m \tag{3.7}$$

and the compact form of the relation (3.7) becomes

$$\mathrm{U}^{(k)} = \mathbf{TB}^k\mathbf{MC},\ k = 0, 1, ..., m. \tag{3.8}$$

Here,

$$\mathbf{U}^{(k)} = \begin{bmatrix} u^{(k)}(x_0) \\ u^{(k)}(x_1) \\ \vdots \\ u^{(k)}(x_N) \end{bmatrix}.$$

In addition, we can obtain the matrix forms $\left(\hat{\mathbf{U}}\right)^r \mathbf{U}^{(k)}$and $\left(\hat{\mathbf{U}}\right)^{(r)} \mathbf{U}^{(k)}$ which appears in the nonlinear part of Eq. (1.1), by using Eq. (3.3) as

$$\begin{aligned} \left(\hat{\mathbf{U}}\right)^r \mathbf{U}^{(k)} &= \begin{bmatrix} u^r(x_0)\,u^{(k)}(x_0) \\ u^r(x_1)\,u^{(k)}(x_1) \\ \vdots \\ u^r(x_N)\,u^{(k)}(x_N) \end{bmatrix} \qquad (3.9) \\ &= \begin{bmatrix} u^r(x_0) & 0 & \dots & 0 \\ 0 & u^r(x_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & u^r(x_N) \end{bmatrix} \begin{bmatrix} u^{(k)}(x_0) \\ u^{(k)}(x_1) \\ \vdots \\ u^{(k)}(x_N) \end{bmatrix}, \\ \left(\hat{\mathbf{U}}\right)^{(r)} \mathbf{U}^{(k)} &= \begin{bmatrix} u^{(r)}(x_0)\,u^{(k)}(x_0) \\ u^{(r)}(x_1)\,u^{(k)}(x_1) \\ \vdots \\ u^{(r)}(x_N)\,u^{(k)}(x_N) \end{bmatrix} \\ &= \begin{bmatrix} u^{(r)}(x_0) & 0 & \dots & 0 \\ 0 & u^{(r)}(x_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & u^{(r)}(x_N) \end{bmatrix} \begin{bmatrix} u^{(k)}(x_0) \\ u^{(k)}(x_1) \\ \vdots \\ u^{(k)}(x_N) \end{bmatrix}, \end{aligned}$$

where

$$\hat{\mathbf{U}} = \hat{\mathbf{T}}\,\hat{\mathbf{M}}\,\hat{\mathbf{C}} \text{ and } \left(\hat{\mathbf{U}}\right)^{(r)} = \hat{\mathbf{T}}\left(\hat{\mathbf{B}}\right)^r \hat{\mathbf{M}}\,\hat{\mathbf{C}}, \tag{3.10}$$

$$\hat{\mathbf{T}} = \begin{bmatrix} \mathbf{T}(x_0) & 0 & \dots & 0 \\ 0 & \mathbf{T}(x_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{T}(x_N) \end{bmatrix}, \hat{\mathbf{B}} = \begin{bmatrix} \mathbf{B} & 0 & \dots & 0 \\ 0 & \mathbf{B} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{B} \end{bmatrix}, \hat{\mathbf{M}} = \begin{bmatrix} \mathbf{M} & 0 & \dots & 0 \\ 0 & \mathbf{M} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{M} \end{bmatrix}, \hat{\mathbf{C}} = \begin{bmatrix} \mathbf{C} & 0 & \dots & 0 \\ 0 & \mathbf{C} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{C} \end{bmatrix}.$$

Substituting the collocation points ($t_i = a + (b-a)i/N, i = 0, 1, \cdots, N$) into Eq. (3.9), gives the system of equations

$$\sum_{k=0}^{m}\sum_{r=0}^{n} P_{kr}(x_i)u^r(x_i)\,u^{(k)}(x_i) + \sum_{k=1}^{m}\sum_{r=1}^{n} Q_{kr}(x_i)u^{(r)}(x_i)\,u^{(k)}(x_i) = g(x_i),$$

which can be expressed with the aid of Eqs. (3.7) and (3.9) as

$$\sum_{k=0}^{m}\sum_{r=0}^{n} \mathbf{P}_{kr}\left(\hat{\mathbf{U}}\right)^r \mathbf{U}^{(k)} + \sum_{k=1}^{m}\sum_{r=1}^{n} \mathbf{Q}_{kr}\left(\hat{\mathbf{U}}\right)^{(r)} \mathbf{U}^{(k)} = \mathbf{G}, \tag{3.11}$$

where

$$\begin{aligned} \mathbf{P}_{kr} &= diag\left[P_{kr}(x_0)\;\; P_{kr}(x_1)\;\; \dots\;\; P_{kr}(x_N)\right], \\ \mathbf{Q}_{kr} &= diag\left[Q_{kr}(x_0)\;\; Q_{kr}(x_1)\;\; \dots\;\; Q_{kr}(x_N)\right], \\ \text{and } \mathbf{G} &= \begin{bmatrix} g(x_0) & g(x_1) & \dots & g(x_N) \end{bmatrix}^T. \end{aligned}$$

By substituting the relations (3.8) and (3.10) into Eq. (3.11), the fundamental matrix equation is attained as

$$\left\{\sum_{k=0}^{m}\sum_{r=0}^{n} \mathbf{P}_{kr}\left(\hat{\mathbf{T}}\,\hat{\mathbf{M}}\,\hat{\mathbf{C}}\right)^r \mathbf{T}\mathbf{B}^k\mathbf{M} + \sum_{k=1}^{m}\sum_{r=1}^{n} \mathbf{Q}_{kr}\hat{\mathbf{T}}\left(\hat{\mathbf{B}}\right)^r \hat{\mathbf{M}}\,\hat{\mathbf{C}}\;\mathbf{T}\mathbf{B}^k\mathbf{M}\right\}\mathbf{C} = \mathbf{G}. \tag{3.12}$$

Briefly, Eq. (3.12) can also be shown as,

$$\mathbf{WC} = \mathbf{G} \quad \text{or } [\mathbf{W}; \mathbf{G}]. \tag{3.13}$$

Here,

$$\mathbf{W} = \sum_{k=0}^{m} \sum_{r=0}^{n} \mathbf{P}_{kr} \left(\hat{\mathbf{T}}\, \hat{\mathbf{M}}\, \hat{\mathbf{C}} \right)^{r} \ \mathbf{T}\mathbf{B}^{k}\mathbf{M} + \sum_{k=1}^{m} \sum_{r=1}^{n} \mathbf{Q}_{kr} \hat{\mathbf{T}} \ \left(\hat{\mathbf{B}} \right)^{r} \ \hat{\mathbf{M}}\, \hat{\mathbf{C}} \ \ \mathbf{T}\mathbf{B}^{k}\mathbf{M}.$$

Here, Eq. (3.13) is a system containing $(N+1)$ nonlinear algebraic equations with the $(N+1)$ unknown Fibonacci coefficients. Using Eq. (3.8) at the points a and b, the matrix representation of the conditions in Eq. (1.2) is given by

$$\left\{ \sum_{k=0}^{m-1} \left[a_{jk} \mathbf{T}(a) + b_{jk} \mathbf{T}(b) \right] (\mathbf{B})^{(k)} \mathbf{M} \right\} \mathbf{C} = \delta_j, \ \ j = 0, 1, 2, ..., m-1$$

or, we can write as

$$\mathbf{V}_j\, \mathbf{C} = \left[\delta_j \right] \qquad \text{or} \qquad \left[\mathbf{V}_j; \delta_j \right]; \quad j = 0, 1, 2, ..., m-1. \tag{3.14}$$

Here,

$$\mathbf{V}_j = \sum_{k=0}^{m-1} \left[a_{jk} \mathbf{T}(a) + b_{jk} \mathbf{T}(b) \right] (\mathbf{B})^{(k)} \mathbf{M} = \left[v_{j0} \ \ v_{j1} \ \ v_{j2} \ ... \ v_{jN} \right].$$

Therefore, by replacing the condition matrices in (3.14) by the m rows of the augmented matrix (3.13), the new augmented matrix will be

$$\left[\hat{\mathbf{W}}; \hat{\mathbf{G}} \right] = \begin{bmatrix} w_{00} & w_{01} & w_{02} & \cdots & w_{0N} & ; & g(x_0) \\ w_{10} & w_{11} & w_{12} & \cdots & w_{1N} & ; & g(x_1) \\ w_{20} & w_{21} & w_{22} & \cdots & w_{2N} & ; & g(x_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots & ; & \vdots \\ w_{(N-m)0} & w_{(N-m)1} & w_{(N-m)2} & \cdots & w_{(N-m)N} & ; & g(x_{N-m}) \\ v_{00} & v_{01} & v_{02} & \cdots & v_{0N} & ; & \delta_0 \\ v_{10} & v_{11} & v_{12} & \cdots & v_{1N} & ; & \delta_1 \\ v_{20} & v_{21} & v_{22} & \cdots & v_{2N} & ; & \delta_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & ; & \vdots \\ v_{(m-1)0} & v_{(m-1)1} & v_{(m-1)2} & \cdots & v_{(m-1)N} & ; & \delta_{m-1} \end{bmatrix}. \tag{3.15}$$

In this way, the unknown Fibonacci coefficients $c_n, \ n = 1, 2, ..., N+1$ are obtained by solving the system in (3.15). Then, these coefficients are substituted into (1.3), and the approximate solution is obtained.

4. Error Estimation

In this section, to test the accuracy of the proposed method, it is presented that estimate error function $\tilde{E}_N(x)$ and actual error function $E_N(x)$ that is the absolute error. The function $E_N(x)$ is given by

$$E_N(x) = |u_N(x) - u(x)|, \tag{4.1}$$

where $u_N(x)$ and $u(x)$ are the approximate and exact solutions of Eq.(1.1), respectively. For $x_k \in [a, b]$, the function $\tilde{E}_N(x)$ is given by

$$\tilde{E}_N(x_k) = |L[u_N(x_k)] - g(x_k)| \cong 0 \tag{4.2}$$

and $\tilde{E}_N \leq 10^{-t_k}$ (t_k any positive constant).

5. Illustrative Examples

In this section, four numerical examples are presented to illustrate the efficient of the proposed method. On these problems, the method is tested by using the error functions given by (4.1) and (4.2). The obtained numerical results are presented with tables and graphics.

Example 1. Let us consider the following second order nonlinear differential equation

$$u''(x) + xu'(x) - 2u^2(x) + x^2u(x) = -x^4 + 2x^2 + 2 \tag{5.1}$$

with the initial conditions

$$u(0) = u'(0) = 0.$$

The exact solution of Eq. (5.1) is $u(x) = x^2$. Then, the approximate solution $u(x)$ determined by the Fibonacci polynomials is

$$u(x) = \sum_{n=1}^{N+1} c_n F_n(x),$$

where $N = 2$, $P_{20}(x) = 1$, $P_{10}(x) = x$, $P_{02}(x) = -2$, $P_{01}(x) = x^2$ and $g(x) = -x^4 + 2x^2 + 2$. Thus, for $N = 2$ the set of collocation points obtained by (3.6) are computed as

$$x_0 = 0,\ \ x_1 = \frac{1}{2},\ x_2 = 1.$$

From Eq. (3.12), we obtain

$$\left\{\mathbf{P}_{20}\mathbf{T}\mathbf{B}^2\mathbf{M} + \mathbf{P}_{10}\mathbf{T}\mathbf{B}\mathbf{M} + \mathbf{P}_{02}\hat{\mathbf{T}}\,\hat{\mathbf{M}}\,\hat{\mathbf{C}}\,\mathbf{T}\mathbf{M} + \mathbf{P}_{01}\mathbf{T}\mathbf{M}\right\}\mathbf{C} = \mathbf{G},$$

where

$$\mathbf{W} = \mathbf{P}_{20}\mathbf{T}\mathbf{B}^2\mathbf{M} + \mathbf{P}_{10}\mathbf{T}\mathbf{B}\mathbf{M} + \mathbf{P}_{02}\hat{\mathbf{T}}\,\hat{\mathbf{M}}\,\hat{\mathbf{C}}\,\mathbf{T}\mathbf{M} + \mathbf{P}_{01}\mathbf{T}\mathbf{M},$$

$$\mathbf{P}_{20} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},\ \mathbf{P}_{10} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix},\ \mathbf{P}_{02} = \begin{bmatrix} -2 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -2 \end{bmatrix},$$

$$\mathbf{P}_{01} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 \\ 0 & 0 & 1 \end{bmatrix},\ \mathbf{T} = \begin{bmatrix} \mathbf{T}(0) \\ \mathbf{T}\left(\frac{1}{2}\right) \\ \mathbf{T}(1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{4} \\ 1 & 1 & 1 \end{bmatrix},$$

$$\hat{\mathbf{T}} = \begin{bmatrix} \mathbf{T}(0) & 0 & 0 \\ 0 & \mathbf{T}\left(\frac{1}{2}\right) & 0 \\ 0 & 0 & \mathbf{T}(1) \end{bmatrix},\ \mathbf{M} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},\ \hat{\mathbf{M}} = \begin{bmatrix} M & 0 & 0 \\ 0 & M & 0 \\ 0 & 0 & M \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix},\ \hat{\mathbf{B}} = \begin{bmatrix} \mathbf{B} & 0 & 0 \\ 0 & \mathbf{B} & 0 \\ 0 & 0 & \mathbf{B} \end{bmatrix}\ \hat{\mathbf{C}} = \begin{bmatrix} \mathbf{C} & 0 & 0 \\ 0 & \mathbf{C} & 0 \\ 0 & 0 & \mathbf{C} \end{bmatrix},\ \mathbf{G} = \begin{bmatrix} 2 \\ \frac{39}{16} \\ 3 \end{bmatrix}.$$

From Eq. (3.14), the matrix form for initial condition is

$$[\mathbf{V}_0;\ \delta_0] = [1\ \ 0\ \ 1\ \ ;\ 0],\ [\mathbf{V}_1;\ \delta_1] = [0\ \ 1\ \ 0\ \ ;\ 0].$$

Therefore, the new augmented matrix $[\hat{\mathbf{W}};\hat{\mathbf{G}}]$ of the problem is yielded. After solving this system, the Fibonacci coefficients matrix is determined as

$$\mathbf{C} = [-1\quad 0\quad 1]^T$$

for $N = 2$, the approximate solution obtained with the Fibonacci polynomials is

$$u_2(x) = x^2.$$

Example 2. [7,8] Assume that the following Abel differential equation

$$u(x)u'(x) + xu(x) + u^2(x) + x^2u^3(x) = g(x); \quad u(0) = 1, \tag{5.2}$$

TABLE 1. Numerical results of different methods for *Example 2* for $N = 5$

x	Taylor Matrix met. [7]	Shifted Chebyshev col. met [8]	Exact sol.	The proposed met.	Absolute error
0	1	1	1	1	0
0.1	0.9048374167	0.9048374178	0.9048374180	0.9048369518	4.66185×10^{-7}
0.2	0.8187306667	0.8187307453	0.8187307531	0.8187300727	6.80311×10^{-7}
0.3	0.7408172500	0.7408181410	0.7408182207	0.7408177925	4.28085×10^{-7}
0.4	0.6703146667	0.6703196344	0.6703200460	0.6703198170	2.28988×10^{-7}
0.5	0.6065104167	0.6065292082	0.6065306597	0.6065303361	3.23535×10^{-7}
0.6	0.5487520000	0.5488076309	0.5488116361	0.5488112328	4.03214×10^{-7}
0.7	0.4964369167	0.4965759540	0.4965853038	0.4965852913	1.24184×10^{-8}
0.8	0.4490026667	0.4493096539	0.4493289641	0.4493294056	4.41570×10^{-7}
0.9	0.4059167500	0.4065333712	0.4065696597	0.4065677881	1.87159×10^{-6}
1	0.3666666667	0.3678160915	0.3678794412	0.3678651778	1.42633×10^{-5}

TABLE 2. Numerical results of the error function E_N at the different values of N for *Example 2*

x	E_4	E_7	E_{10}
0.2	1.41831×10^{-5}	9.29864×10^{-10}	2.87548×10^{-14}
0.4	5.30231×10^{-6}	6.80168×10^{-10}	2.15383×10^{-14}
0.6	5.34811×10^{-6}	3.24082×10^{-10}	1.32117×10^{-14}
0.8	7.23260×10^{-6}	2.52861×10^{-10}	2.10942×10^{-15}
1	2.30924×10^{-4}	3.49407×10^{-8}	1.81444×10^{-12}

where

$$g(x) = xe^{-x} + x^2 e^{-3x}.$$

The exact solution of Eq.(5.2) is given by $u(x) = e^{-x}$. Table 1 presents values of error function given in Eq.(4.1) and a numerical comparison of proposed method with Taylor and Chebyshev methods for Eq.(5.2) when $N = 5$. Additionally, in Table 2, numerical results of the error function in Eq.(4.1) for $N = 4, 7, 10$ are presented. In Figure 1, it is presented that graphical comparison of approximate and exact solutions obtained by the proposed method for $N = 2, 3$ and 4. According to the Figure 1, it is seen that the approximate solution converges to the exact solution when the iteration N increases.

Example 3. [3] Consider following the classical Van der Pol equation

$$u''(x) + \epsilon(u^2(x) - 1)u'(x) + u(x) = 0; \tag{5.3}$$

with the initial conditions

$$u(0) = \alpha, u'(0) = 0,$$

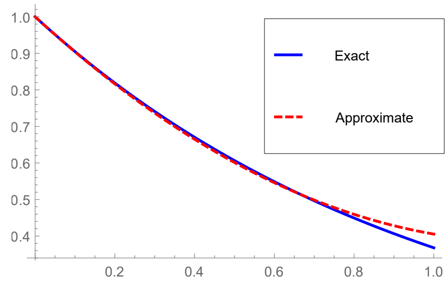
where $\epsilon = 0.1$ and $\alpha = 0.1$. Table 3 presents values of estimate error function given in Eq.(4.2) for Eq.(5.3) when $N = 4, 7, 10$. Also, in Table 4, numerical results of approximate solutions obtained for $N = 4, 7, 10$ are presented. In Figure 2, it is presented that graphics of estimate error function for $N = 4, 7$ and 10. According to the tables and graphics, it is seen that the error $\tilde{E}_N$ decreases when the iteration N increases.

Example 4. [6] Consider the following differential equation

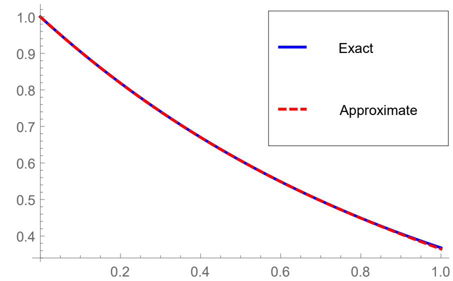
$$u^{(4)}(x)u''(x) - (u'''(x))^2 = 0; \quad 0 \leq x \leq 1 \tag{5.4}$$

with the initial conditions

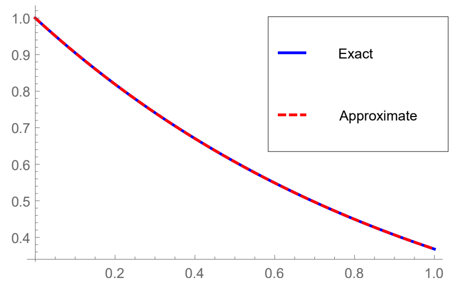
$$u(0) = 2, u'(0) = -1, u''(0) = 3, u'''(0) = 1.$$



(A) $N = 2$



(B) $N = 3$



(C) $N = 4$

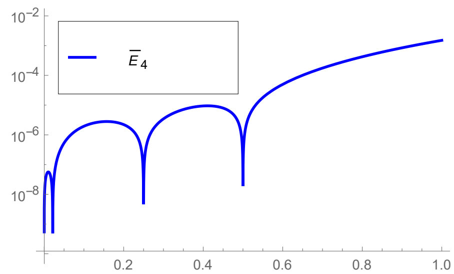
FIGURE 1. Graphical comparison of the exact and approximate solutions when $N = 2, 3, 4$ for Example 2

TABLE 3. Numerical results of the error function $\tilde{E}_N$ at the different values of N for *Example 3*

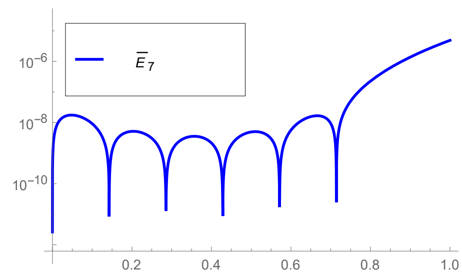
x	$\tilde{E}_4$	$\tilde{E}_7$	$\tilde{E}_{10}$
0.2	2.21363×10^{-6}	5.15027×10^{-9}	4.46691×10^{-17}
0.4	9.37006×10^{-6}	2.14311×10^{-9}	6.50521×10^{-18}
0.6	5.00360×10^{-5}	5.63117×10^{-9}	5.03070×10^{-17}
0.8	4.22641×10^{-4}	2.26743×10^{-7}	3.72966×10^{-17}
1	1.50762×10^{-3}	4.88836×10^{-6}	2.9407×10^{-9}

TABLE 4. Numerical results of the approximate solution function u_N at the different values of N for *Example 3*

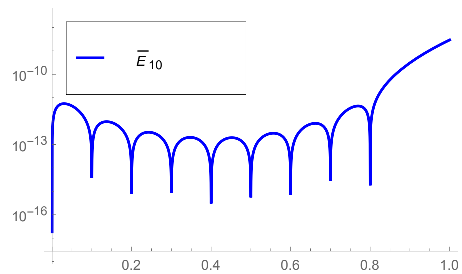
x	u_4	u_7	u_{10}
0.2	0.0979934643	0.0979934437	0.0979934435
0.4	0.0920011494	0.0920010922	0.0920010918
0.6	0.0821842205	0.0821842938	0.0821842931
0.8	0.0688641457	0.0688613200	0.0688613192
1	0.0525226965	0.0524976639	0.0524976848



(a) $N = 4$



(b) $N = 7$



(c) $N = 10$

Figure 2. Graphics of error functions $\tilde{E}_N$ when the values of N are 4, 7 and 10 for Example 3

Table 5. Numerical results of the error function $\tilde{E}_N$ at the different values of N for *Example 4*

N	Bernstein pol. method [6] max E_N	Proposed method max E_N
4	2.88×10^{-3}	9.79×10^{-4}
7	2.85×10^{-6}	9.76×10^{-9}
10	9.01×10^{-10}	3.81×10^{-14}

The exact solution of Eq.(5.4) is

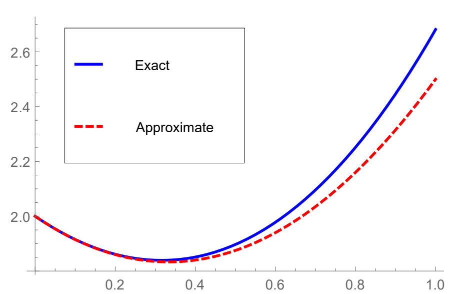
$$u(x) = 27e^{\frac{x}{3}} - 10x - 25.$$

Table 5 presents a comparison of maximum absolute errors of proposed method with Bernstein polynomial method for Eq.(5.4) when $N = 4, 7, 10$. In Figure 3, it is presented that graphics of approximate and exact solutions obtained by the proposed method in the interval $(0, 1)$ for $N = 2, 3$ and 4. According to the Figure 3, it is seen that the approximate solution converges to the exact solution when the iteration N increases.

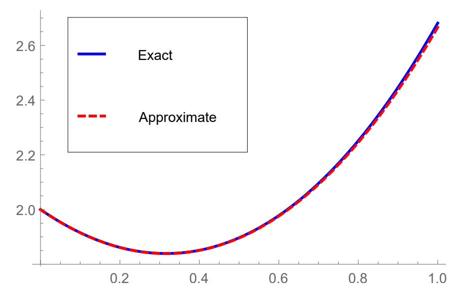
Example 5. Lastly, consider the following differential equation

$$u''(x) + u(x) + u^2(x) = 0; \quad u(0) = 0, u'(0) = 0; \quad 0 \leq x \leq 10. \tag{5.5}$$

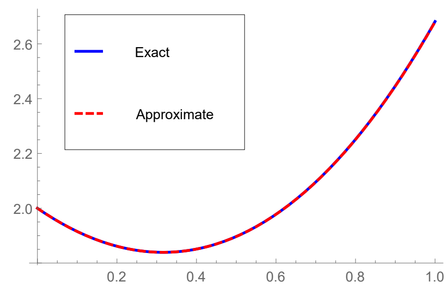
The exact solution of Eq.(5.5) is unknown. Table 6 presents values of error function given in Eq.(4.2) for $N = 4, 5, 6$. In Figure 4, it is presented that graphics of estimate error function for $N = 4, 5$ and 6. According to the tables and graphics, it is seen that the error $\tilde{E}_N$ decreases when the iteration N increases.



(A) $N = 2$



(B) $N = 3$



(C) $N = 4$

FIGURE 3. Graphical comparison of exact and approximate solutions when $N = 2, 3, 4$ for Example 4

TABLE 6. Numerical results of the error function $\tilde{E}_N$ at the different values of N for *Example 5*

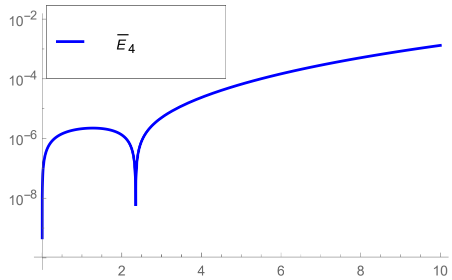
x	$\tilde{E}_4$	$\tilde{E}_5$	$\tilde{E}_6$
2	1.32141×10^{-6}	4.39950×10^{-7}	7.44000×10^{-9}
4	2.37012×10^{-5}	4.27248×10^{-9}	6.03580×10^{-8}
6	1.46829×10^{-4}	1.52821×10^{-5}	4.83065×10^{-8}
8	5.08061×10^{-4}	1.13663×10^{-4}	6.14407×10^{-8}
10	1.31622×10^{-3}	4.62630×10^{-4}	9.01390×10^{-7}

6. CONCLUSIONS

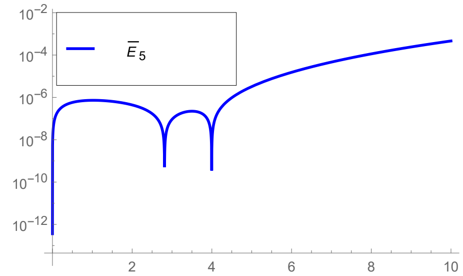
In this paper, the Fibonacci collocation Method was used for solving a class of nonlinear differential equations. The efficiency and accuracy of the method with five different examples is shown. The obtained approximate and error results are compared with ones obtained with the Taylor matrix method, Shifted Chebyshev collocation method, and Bernstein polynomial method. As a result of these comparisons, it can be said that the method is very effective to obtain an approximate solution of nonlinear differential equations. The results in the tables and figures that are given by test problems show that the solution accuracies improve when N is increased. The other advantage of the method is that all the computations can be calculated in a short time with computer software. In future studies, the method is planned to be applied to systems of the nonlinear differential equation.

CONFLICTS OF INTEREST

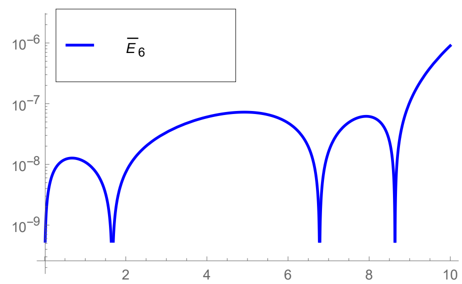
The authors declare that there are no conflicts of interest regarding the publication of this article.



(A) $N = 4$



(B) $N = 5$



(c) $N = 6$

Figure 4. Graphics of error functions $\tilde{E}_N$ when the values of N are 4, 5 and 6 for Example 5

Authors Contribution Statement

All authors jointly worked on the results and they have read and agreed to the published version of the manuscript.

References

[1] Akgul, A., Hashemi, M.S., Inc, M., Raheem, S.A., *Constructing two powerful methods to solve the Thomas-Fermi equation*, Nonlinear Dynamics, **87**(2)(2016), 1435–1444.

[2] Akgul, A., Inc, M., Hashemi, M.S., *Group preserving scheme and reproducing kernel method for the Poisson-Boltzmann equation for semiconductor devices*, Nonlinear Dynamics, **88**(4)(2017), 2817–2829.

[3] Chaharborj, S.S., See, P.P., *Application of Chebyshev neural network to solve Van der Pol equations*, International Journal of Basic and Applied Sciences, **10**(1)(2021), 7–19.

[4] Falcon, S., Plaza, A., *The k-Fibonacci sequence and the Pascal 2-triangle*, Chaos, Solitons & Fractals, **33**(1)(2007), 38–49.

[5] Falcon, S., Plaza, A., *On k-Fibonacci sequences and polynomials and their derivatives*, Chaos, Solitons & Fractals, **39**(3)(2009), 1005–1019.

[6] Gospodarczyk, P., Wozny, P., *An iterative approximate method of solving boundary value problems using dual Bernstein polynomials*, Tech. Rep. 2018-03-01, University of Wroclaw Institute of Computer Science, arXiv:1709.02162 (2018).

[7] Guler, C., *A new numerical algorithm for the Abel equation of the second kind*, International Journal of Computer Mathematics, **84**(1)(2007), 109–119.

[8] Gulsu, M., Ozturk, Y., Sezer, M., *On the solution of the Abel equation of the second kind by the shifted Chebyshev polynomials*, Applied Mathematics and Computation, **217**(9)(2011), 4827–4833.

[9] Hashemi, M.S., *Constructing a new geometric numerical integration method to the nonlinear heat transfer equations*, Communications in Nonlinear Science and Numerical Simulation, **22**(1-3)(2015), 990–1001.

[10] Hashemi, M.S., Akgul, A., *On the MHD boundary layer flow with diffusion and chemical reaction over a porous flat plate with suction/blowing: two reliable methods*, Engineering with Computers, **37**(2)(2019), 1147–1158.

[11] Hashemi, M.S., Baleanu, D., *Numerical approximation of higher-order time-fractional telegraph equation by using a combination of a geometric approach and method of line*, Journal of Computational Physics, **316**(2016), 10–20.

[12] Hashemi, M.S., Inc, M., Hajikhah, S., *Generalized squared remainder minimization method for solving multi-term fractional differential equations*, Nonlinear Analysis: Modelling and Control, **26**(1)(2021), 57–71.

[13] He, J.H., Latifizadeh, H., *A general numerical algorithm for nonlinear differential equations by the variational iteration method*, International Journal of Numerical Methods for Heat & Fluid Flow, **30**(11)(2020), 4797–4810.

[14] Kurt, A., Yalcinbas, S., Sezer, M., *Fibonacci collocation method for solving linear differential-difference equations*, Mathematical and Computational Applications, **18**(3)(2013), 448–458.

[15] Kurt, A., Yalcinbas, S., Sezer, M., *Fibonacci collocation method for solving high-order linear Fredholm integro-differential-difference equations*, International Journal of Mathematics and Mathematical Sciences, **2013**(2013), 1–9.

[16] Mirzaee, F., Hoseini, S.F., *Solving systems of linear Fredholm integro-differential equations with Fibonacci polynomials*, Ain Shams Engineering Journal, **5**(1)(2014), 271–283.

[17] Mirzaee, F., Hoseini, S.F., *Solving singularly perturbed differential-difference equations arising in science and engineering with Fibonacci polynomials*, Results in Physics, **3**(2013), 134–141.

[18] Mirzaee, F., Hoseini, S.F., *A new collocation approach for solving systems of high-order linear Volterra integro-differential equations with variable coefficients*, Applied Mathematics and Computation, **311**(2017), 272–282.

[19] Odibat, Z., *An optimized decomposition method for nonlinear ordinary and partial differential equations*, Physica A: Statistical Mechanics and its Applications, **541**(2020), 123323.

[20] Odibat, Z., *An improved optimal homotopy analysis algorithm for nonlinear differential equations*, Journal of Mathematical Analysis and Applications, **488**(2)(2020), 124089.

[21] Rani, D., Mishra, V., *Numerical inverse Laplace transform based on Bernoulli polynomials operational matrix for solving nonlinear differential equations*, Results in Physics, **16**(2020), 102836.

[22] Simao, P.D., *Post-buckling analysis of nonlinear shear-deformable prismatic columns using a GBT consistent energy formulation*, Computers and Structures, **190**(2017), 186–204.