Iranian Journal of Numerical Analysis and Optimization
Vol. 10, No. 1, (2020), pp 63–79
DOI:10.22067/ijnao.v10i1.81828

Research Article

The Fibonacci polynomials solution for Abel's integral equation of second kind

H. Deilami Azodi*

Abstract

We suggest a convenient method based on the Fibonacci polynomials and the collocation points for solving approximately the Abel's integral equation of second kind. Initially, the solution is supposed in the form of the Fibonacci polynomials truncated series with the unknown coefficients. Then, by placing this series into the main problem and collocating the resulting equation at some points, a system of algebraic equations is obtained. After solving it, the unknown coefficients and so the solution of main problem are determined. The error analysis is discussed elaborately. Also, the reliability of the method is quantified through numerical examples.

AMS(2010): Primary 45D05; Secondary 11B39, 65N35.

Keywords: Abel's integral equation; Fibonacci polynomials; Collocation points; Error analysis.

1 Introduction

It is well known that the Abel's integral equation is an applied Volterra integral equation with a singular kernel. Some applications of it appear in the plasma spectroscopy [3], thermoelectricity [8], and spectrographic data [9].

Solving the integral equations with singularity feature is often contestable because of the complexity and the heavy computations. Thus, various methods especially in recent years have been proposed for solving the Abel's integral equation. Some of them include Bernstein polynomials [2], Taylor expansion [10], quadrature and mechanical quadrature methods [11, 12], Laplace transform [16], Block-pulse functions [24], modified Homotopy and Adomian

*Corresponding author
Received 9 July 2019; revised 16 August 2019; accepted 26 October 2019
Haman Deilami Azodi
Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran. email: haman_azodi@webmail.guilan.ac.ir

methods [25], variational iteration method [26], shifted Legendre collocation method [27], Laguerre polynomials [28], and Legendre wavelets [29].

Recently, the nonorthogonal polynomials have been used widely to solve different problems; see [5, 4, 14, 23]. One of these polynomials is the Fibonacci polynomials. They have been derived successfully to find the approximate solutions of the fractional diffusion equation [6], the boundary value problems [15], the Fredholm-Volterra integral equations [19, 21], the systems of high-order linear Volterra integro-differential equations with variable coefficients [22], the nonlinear stochastic Itô-Volterra integral equations [20], the singularly perturbed differential-difference equations [17], and the systems of linear Fredholm integro-differential equations [18]. Moreover, some advantages of the Fibonacci polynomials rather than orthogonal polynomials have been described in [1].

Throughout this manuscript, the generalized Abel's integral equation of second kind is considered in the following form:

$$u(x) = g(x) + k \int_0^x \frac{J(u(t))}{(x-t)^\alpha} dt, \tag{1}$$

so that $0 < \alpha < 1$, $k \in \mathbb{R}$, $0 \leq x, t \leq b$ for $b > 0$, and g and J are real continuous functions. Note that if $\alpha = \frac{1}{2}$ and $J(u(t)) = u(t)$, then (1) is reduced to the conventional Abel's integral equation of second kind.

To the best of our knowledge, the existing works on the Fibonacci polynomials in the literature have mainly concentrated on the problems with nonsingular kernels. In this paper, we try to assess the application of these polynomials for solving an important family of the integral equations with the singular kernel. In fact, the aim of this research is to present an approximate solution of the problem (1) in the form of

$$u(x) \approx u_N(x) = \sum_{n=1}^{N} a_n F_n(x), \tag{2}$$

where F_n, $n = 1, \ldots, N$ denote to Fibonacci polynomials; a_n, $n = 1, \ldots, N$ are the unknown coefficients; and N is chosen any positive integer. Here, the Fibonacci polynomials are defined by [7, 13]

$$F_n(x) = \sum_{j=0}^{\left[\frac{n-1}{2}\right]} \binom{n-1-j}{j} x^{n-1-2j}, \quad n = 1, \ldots, N. \tag{3}$$

Furthermore, the Fibonacci polynomials can also be identified by the recurrence formula below

$$\begin{cases} F_1(x) = 1, \\ F_2(x) = x, \\ F_n(x) = xF_{n-1}(x) + F_{n-2}(x), \quad n = 3, 4, \ldots. \end{cases}$$

To achieve the aforesaid aim,(2) is substituted into (1) and by collocating the obtained identity at the collocation points $x_j = \frac{b}{N-1}(j-1)$, $j = 1, \ldots, N$, a system of algebraic equations consisting of N unknowns is gained whose solution is $a_1, \ldots, a_N$. Therefore, after solving this system the solution of corresponding problem can be found by applying (2).

In the next section, the Fibonacci collocation method for the problem (1) is formulated by utilizing the matrix forms. The error of suggested method for (1) is discussed in Section 3. Section 4 provides the numerical results of applying the Fibonacci collocation method by four illustrative examples. At the end, a conclusion is given in Section 5.

2 Method of solution

At the beginning, let us exhibit (3) in the matrix form as follows

$$\mathbf{F}(x) = \mathbf{X}(x)\mathbf{D}^T, \tag{4}$$

where

$$\mathbf{F}(x) = \left[F_1(x)\ F_2(x)\ \ldots\ F_N(x)\right], \quad \mathbf{X}(x) = \left[1\ x\ \ldots\ x^{N-1}\right],$$

$$\mathbf{D} = \begin{bmatrix} \binom{0}{0} & 0 & 0 & 0 & \ldots & 0 \\ 0 & \binom{1}{0} & 0 & 0 & \ldots & 0 \\ \binom{1}{1} & 0 & \binom{2}{0} & 0 & \ldots & 0 \\ 0 & \binom{2}{1} & 0 & \binom{3}{0} & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \binom{\frac{N-2}{2}}{\frac{N-2}{2}} & 0 & \binom{\frac{N}{2}}{\frac{N-4}{2}} & 0 & \ldots & 0 \\ 0 & \binom{\frac{N}{2}}{\frac{N-2}{2}} & 0 & \binom{\frac{N+2}{2}}{\frac{N-4}{2}} & \ldots & \binom{N-1}{0} \end{bmatrix}_{N\times N},$$

if N is even, and

$$\mathbf{D}=\begin{bmatrix} \binom{0}{0} & 0 & 0 & 0 & \dots & 0 \\ 0 & \binom{1}{0} & 0 & 0 & \dots & 0 \\ \binom{1}{1} & 0 & \binom{2}{0} & 0 & \dots & 0 \\ 0 & \binom{2}{1} & 0 & \binom{3}{0} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \binom{\frac{N-1}{2}}{\frac{N-3}{2}} & 0 & \binom{\frac{N+1}{2}}{\frac{N-5}{2}} & \dots & 0 \\ \binom{\frac{N-1}{2}}{\frac{N-1}{2}} & 0 & \binom{\frac{N+1}{2}}{\frac{N-3}{2}} & \dots & \dots & \binom{N-1}{0} \end{bmatrix}_{N\times N},$$

if N is odd.

We divide the structure of the proposed method into linear case and nonlinear one.

- Linear case:

  In this case, $J(u(t)) = u(t)$. For the sake of simplicity, (1) may be shown in the form of

$$u(x) = g(x) + kI(x), \tag{5}$$

in which

$$I(x) = \int_0^x \frac{u(t)}{(x-t)^\alpha}dt. \tag{6}$$

For a given N, the desired solution of (1) is supposed in the form of the truncated Fibonacci series (2). So, $u(x)$ can be written in the matrix form

$$u(x) = \mathbf{F}(x)\mathbf{A}; \quad \mathbf{A} = \left[a_1\ a_2\ \dots\ a_N\right]^T,$$

or with the help of (4), it follows that

$$u(x) = \mathbf{X}(x)\mathbf{D}^T\mathbf{A}. \tag{7}$$

By substituting (7) into (6), it is obvious

$$I(x) = \int_0^x \frac{\mathbf{X}(t)\mathbf{D}^T\mathbf{A}}{(x-t)^\alpha}dt = \left(\int_0^x \frac{\mathbf{X}(t)}{(x-t)^\alpha}dt\right)\mathbf{D}^T\mathbf{A}. \tag{8}$$

To acquire a matrix relation for $I(x)$, an explicit formula must be determined for the integral

$$I_{i,\alpha}(x) = \int_0^x \frac{t^i}{(x-t)^\alpha}dt, \quad i = 0, 1, \dots, N-1.$$

For this purpose, changing the variables by $t = xz$ enables one to write

$$dt = xdz, \quad 0 \leq z \leq 1,$$

and hence

$$I_{i,\alpha}(x) = x^{i+1-\alpha} \int_0^1 (1-z)^{-\alpha} z^i dz = \beta(i+1, 1-\alpha) x^{i+1-\alpha}, \tag{9}$$

or equivalently

$$I_{i,\alpha}(x) = \frac{\Gamma(i+1)\Gamma(1-\alpha)}{\Gamma(i-\alpha+2)} x^{i+1-\alpha},$$

where $\Gamma(\cdot)$ and $\beta(\cdot,\cdot)$ refer to the Gamma and Beta functions; see [5].

Lastly, using (8) and (9) results in

$$I(x) = \mathbf{I}_\alpha(x)\mathbf{D}^T\mathbf{A}, \tag{10}$$

where

$$\mathbf{I}_\alpha(x) = \left[\beta(1, 1-\alpha)x^{1-\alpha} \ \beta(2, 1-\alpha)x^{2-\alpha} \ \ldots \ \beta(N, 1-\alpha)x^{N-\alpha}\right].$$

In order to implement the numerical procedure, by substituting (7) and (10) into (5) it is clear that

$$\mathbf{X}(x)\mathbf{D}^T\mathbf{A} = k\mathbf{I}_\alpha(x)\mathbf{D}^T\mathbf{A} + g(x). \tag{11}$$

Now, collocating (11) at the points

$$\{x_1, x_2, x_3, \ldots, x_N\} = \left\{0, \frac{b}{N-1}, \frac{2b}{N-1}, \ldots, b\right\}, \tag{12}$$

the main matrix equation is produced as follows:

$$\left\{\bar{\mathbf{X}}\mathbf{D}^T - k\bar{\mathbf{I}}_\alpha\mathbf{D}^T\right\}\mathbf{A} = \mathbf{G}, \tag{13}$$

in which

$$\bar{\mathbf{X}} = \begin{bmatrix} \mathbf{X}(x_1) \\ \mathbf{X}(x_2) \\ \vdots \\ \mathbf{X}(x_N) \end{bmatrix}, \quad \bar{\mathbf{I}}_\alpha = \begin{bmatrix} \mathbf{I}_\alpha(x_1) \\ \mathbf{I}_\alpha(x_2) \\ \vdots \\ \mathbf{I}_\alpha(x_N) \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} g(x_1) \\ g(x_2) \\ \vdots \\ g(x_N) \end{bmatrix}.$$

It is notable that $\mathbf{A}$ and $\mathbf{G}$ are column vectors with N entries and the dimension of others in (13) is $N \times N$.

Briefly, (13) may be shown as

$$\mathbf{W}\mathbf{A} = \mathbf{G}, \tag{14}$$

so that

$$\mathbf{W} = \bar{\mathbf{X}}\mathbf{D}^T - k\bar{\mathbf{I}}_\alpha\mathbf{D}^T.$$

Unequivocally, (14) is a linear system of algebraic equations whose solution is the coefficients $a_1, a_2, \ldots, a_N$. Finally, the solution of (1) can be uncovered by employing (7).

- Nonlinear case:

  Based on (7), we have

$$u(x) = \mathbf{X}(x)\mathbf{B}, \tag{15}$$

where $\mathbf{B} = \mathbf{D}^T\mathbf{A}$. If $m \in \mathbb{N}$ and $\xi_m = (N-1)m + 1$, then there exists the vector $\mathbf{P}_m = \left[p_1 \; p_2 \; \ldots \; p_{\xi_m}\right]^T$ such that

$$u^m(x) = \mathbf{X}_m(x)\mathbf{P}_m, \tag{16}$$

in which

$$\mathbf{X}_m(x) = \left[1 \; x \; x^2 \; \ldots \; x^{(N-1)m}\right].$$

According to the multi-binomial theorem, the entries of $\mathbf{P}_m$ are expressed in terms of $a_1, a_2, \ldots, a_N$. In the other word, for $1 \le i_m \le \xi_m$, $p_{i_m} = \varphi_{i_m}(a_1, a_2, \ldots, a_N)$, where φ_{i_m} is a real multivariable function for every m and i.

Suppose that J applied in (1) is an $M \in \mathbb{N}$ times differentiable function at $x = 0$. Then,

$$J(x) \approx \sum_{m=0}^{M} \frac{J^{(m)}(0)}{m!} x^m.$$

Therefore,

$$J(u(t)) \approx \sum_{m=0}^{M} \frac{J^{(m)}(0)}{m!} u^m(t). \tag{17}$$

Looking at (16) and (17), we conclude

$$J(u(t)) \approx \sum_{m=0}^{M} \frac{J^{(m)}(0)}{m!} \mathbf{X}_m(t)\mathbf{P}_m. \tag{18}$$

From (1), (15) and (18), we obtain

$$\mathbf{X}(x)\mathbf{B} \approx g(x) + k\sum_{m=0}^{M} \frac{J^{(m)}(0)}{m!}\mathbf{R}_m(x)\mathbf{P}_m. \tag{19}$$

It must be mentioned that

$$\mathbf{R}_m(x) = [r_i]_{1\le i\le \xi_m} = \int_0^x \frac{\mathbf{X}_m(t)}{(x-t)^\alpha}dt,$$

is a row vector and by using (9), one can imply that

$$r_i = \beta(i+1, 1-\alpha)x^{i+1-\alpha}, \quad i = 1, 2, \dots, \xi_m.$$

If we replace $\approx$ by $=$ in (19) and collocating the obtained identity at the points defined by (12), a nonlinear system of algebraic equations is produced which may be solved by any available iterative scheme such as the Newton–Raphson method.

3 A discussion on the error

Throughout this section, the error of the method proposed in the previous section is analyzed. Consider $L^\infty[0, b]$ is the L-infinity space defined on $[0, b]$ with the norm

$$\|h\|_\infty = \sup_{0\le x\le b} |h(x)|,$$

where h is a real function. In addition, for $F_j(x)$, $1 \le j \le N$, suppose that

$$\boldsymbol{\mathcal{F}} = \mathrm{span}\{\mathrm{F_1(x), F_2(x), \dots, F_N(x)}\} \subset \mathrm{L^\infty[0, b]}.$$

Since $\boldsymbol{\mathcal{F}}$ is a finite-dimensional vector space, $h(x)$ has the best approximation such as $h_s(x) \in \boldsymbol{\mathcal{F}}$, namely

$$\text{for all } r(x) \in \boldsymbol{\mathcal{F}} : \|h(x) - h_s(x)\|_\infty \le \|h(x) - r(x)\|_\infty.$$

Due to $h_s(x) \in \boldsymbol{\mathcal{F}}$, there exist the unique coefficients c_k such that

$$h(x) \approx h_s(x) = \sum_{k=1}^{N} c_k F_k(x) = \mathbf{F}(x)\mathbf{C}^T,$$

where $\mathbf{C} = \begin{bmatrix} c_1 \; c_2 \; \dots \; c_N \end{bmatrix}$.

Lemma 1. Let $h : [x_0, b] \to \mathbb{R}$ is N times continuously differentiable for $0 \le x_0 < b$. If $h_s(x)$ is the best approximation to h, then the error bound is declared as follows

$$\|h(x) - h_s(x)\|_\infty \leq \frac{M(b-x_0)^N}{N!}, \tag{20}$$

in which $M = \sup_{0\leq x\leq b} \left|h^{(N)}(x)\right|$.

Proof. Suppose that $\tilde{h}(x)$ is an arbitrary approximation of $h(x)$. We choose this approximation in the form of Taylor series of $h(x)$. Clearly,

$$\begin{aligned}\tilde{h}(x) &= h(x_0) + h'(x_0)(x-x_0) + h''(x_0)\frac{(x-x_0)^2}{2!}\\ &\quad + \cdots + h^{(N-1)}(x_0)\frac{(x-x_0)^{N-1}}{(N-1)!}.\end{aligned}$$

Therefore, there exists $\eta \in (x_0, b)$ such that

$$|h(x) - \tilde{h}(x)| = \left|h^{(N)}(\eta)\frac{(x-x_0)^N}{N!}\right|.$$

Since $h_s(x)$ is the best approximation of h,

$$\begin{aligned}\|h(x) - h_s(x)\|_\infty \leq \|h(x) - \tilde{h}(x)\|_\infty &= \sup_{0\leq x\leq b}|h(x) - \tilde{h}(x)|\\ &= \sup_{0\leq x\leq b}\left|h^{(N)}(\eta)\frac{(x-x_0)^N}{N!}\right|.\end{aligned}$$

Putting $M = \sup_{0\leq x\leq b}\left|h^{(N)}(x)\right|$, (20) is realized. □

The main result of this section is announced in the following theorem.

Theorem 1. Assume that $u(x)$ and $u_N(x)$ for a given $N > 1$ are the exact and the Fibonacci solutions of (1), respectively, on $[0, b]$. If J satisfies in the Lipschitz condition with the constant η and also $1 - \alpha - |k|\eta b^{1-\alpha} > 0$, then one can deduce

$$\|u(x) - u_N(x)\|_\infty \leq \frac{M'(b-x_0)^N}{\lambda N!}, \tag{21}$$

where $\lambda = \frac{1-\alpha-|k|\eta b^{1-\alpha}}{1-\alpha}$ and $M' = \sup_{0\leq x\leq b}\left|g^{(N)}(x)\right|$.

Proof. Suppose that $g(x)$ is expanded in terms of the Fibonacci polynomials; then the obtained solution is an approximated polynomial, $u_N(x)$. We attempt to seek an upper bound for the associated error between the exact solution $u(x)$ and the approximated solution $u_N(x)$ for (1). Consequently,

$$u(x) - u_N(x) = g(x) - g_N(x) + k\int_0^x \frac{J(u(t)) - J(u_N(t))}{(x-t)^\alpha}dt.$$

Applying the triangular inequality, it is concluded

$$|u(x) - u_N(x)| \le |g(x) - g_N(x)| + |k| \left| \int_0^x \frac{J(u(t)) - J(u_N(t))}{(x-t)^\alpha} dt \right|. \tag{22}$$

Besides, one can write

$$\begin{aligned} \left| \int_0^x \frac{J(u(t)) - J(u_N(t))}{(x-t)^\alpha} dt \right| &\le \|J(u(x)) - J(u_N(x))\|_\infty \int_0^x \frac{dt}{(x-t)^\alpha} \\ &= \frac{x^{1-\alpha}}{1-\alpha} \|J(u(x)) - J(u_N(x))\|_\infty. \end{aligned}$$

Thus, for $x \in [0, b]$, we have

$$\left| \int_0^x \frac{J(u(t)) - J(u_N(t))}{(x-t)^\alpha} dt \right| \le \frac{\eta b^{1-\alpha}}{1-\alpha} \|u(x) - u_N(x)\|_\infty. \tag{23}$$

On the other hand, if $M' = \sup \left| g^{(N)}(x) \right|$ for $x \in [0, b]$, based on (20), then

$$|g(x) - g_N(x)| \le \frac{M'(b - x_0)^N}{N!}. \tag{24}$$

Ultimately according to the relations (22)–(24) and the mentioned assumption, (21) is satisfied. □

4 Numerical examples

This section is devoted to four examples for the description of the method's applicability. The corresponding computations are performed by MATLAB R2015a software on a 64-bit PC with 2.20 GHz processor and 8 GB memory.

Example 1. [16] Consider the Abel's integral equation (1) in the form of

$$u(x) = x^2 + \frac{16}{15} x^{\frac{5}{2}} - \int_0^x \frac{u(t)}{(x-t)^{\frac{1}{2}}} dt, \quad 0 \le x \le 1. \tag{25}$$

The exact solution of (25) is $u(x) = x^2$.

Suppose that $N = 3$. By this selection for N, the collocation points become $\left\{0, \frac{1}{2}, 1\right\}$. After some calculations, we have

$$\mathbf{W} = \begin{bmatrix} 1 & 0 & 0 \\ 985/408 & 1155/1189 & 1027/360 \\ 3 & 7/3 & 76/15 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} 0 \\ (2\sqrt{2}/15) + (1/4) \\ 31/15 \end{bmatrix}.$$

By solving the algebraic system $\mathbf{WA} = \mathbf{G}$, one obtains

$$\mathbf{A} = \begin{bmatrix} -1 & -5.67 \times 10^{-16} & 1 \end{bmatrix}^T.$$

Accordingly, $u_3(x) = (-5.67 \times 10^{-16})x + x^2$. This solution concludes that $\|u(x) - u_3(x)\|_\infty = 5.67 \times 10^{-16}$. It is worth to mention that in [16], the maximum absolute error of the Laplace transform method is 5×10^{-7} at the level $n = 25$, which requires large computational effort.

Example 2. [25] Consider the Abel's integral equation (1) as follows:

$$u(x) = 2\sqrt{x} - \int_0^x \frac{u(t)}{(x-t)^{\frac{1}{2}}} dt, \quad 0 \le x \le 1. \tag{26}$$

The exact solution of (26) is $u(x) = 1 - e^{\pi x} erfc(\sqrt{\pi x})$, in which the complementary error function $erfc$ is defined as

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-\tau^2} d\tau.$$

By deriving the presented method for $N = 5$ and $N = 10$, the approximate solutions are

$$\begin{cases} u_5(x) = -5.2570118x^4 + 12.973931x^3 - 11.381689x^2 + 4.3781671x \\ +(2.8421709 \times 10^{-14}), \\ u_{10}(x) = 318.98646x^9 - 1582.0737x^8 + 3371.4195x^7 - 4038.7149x^6 \\ +2986.8902x^5 - 1409.7584x^4 + 424.24302x^3 - 79.328853x^2 + 9.0564174x \\ +(9.3132257 \times 10^{-10}). \end{cases}$$

Table 1 and Figure 1 show the numerical results and the absolute errors of the presented method, respectively. It is seen the results of applying the proposed method are close to the exact solution. One should note that the corresponding values of the exact solution have been calculated with the aid of $erfc(\cdot)$ command of MATLAB software.

Table 1: Numerical results of Example 2

t	Presented method		Exact solution
	$N = 5$	$N = 10$	
0.0	0.00000	0.00000	0.00000
0.2	0.51575	0.51305	0.50835
0.4	0.62595	0.60539	0.60335
0.6	0.65055	0.65762	0.65632
0.8	0.70763	0.69299	0.69184
1.0	0.71340	0.71974	0.71794

Example 3. [24, 29] Consider the following Abel's integral equation

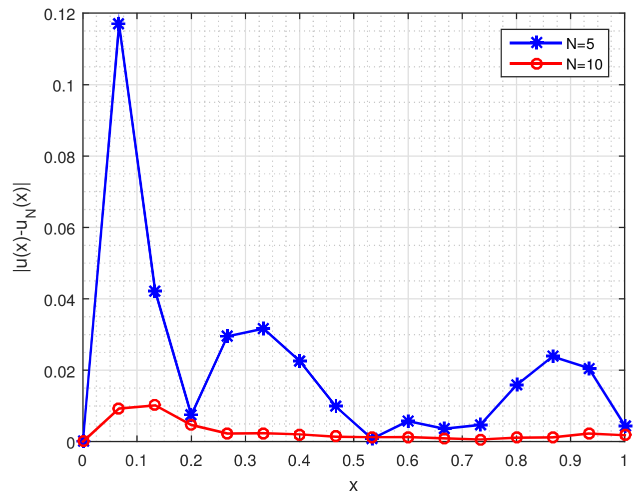

Figure 1: The illustration of absolute error for Example 2

$$u(x) = \frac{\pi}{8} + \frac{1}{\sqrt{x+1}} - \frac{1}{4}\sin^{-1}\left(\frac{1-x}{1+x}\right) - \frac{1}{4}\int_0^x \frac{u(t)}{(x-t)^{\frac{1}{2}}}dt. \tag{27}$$

The exact solution of (27) is $u(x) = \frac{1}{\sqrt{x+1}}$.

The solutions of the Fibonacci collocation method for $N = 4$ and $N = 8$ are identified as follows:

$$\begin{cases} u_4(x) = -0.0852864x^3 + 0.277508x^2 - 0.485193x + 1.0, \\ u_8(x) = -0.0120919x^7 + 0.0615555x^6 - 0.147352x^5 + 0.234167x^4 \\ -0.303006x^3 + 0.373771x^2 - 0.499937x + 1.0. \end{cases}$$

Figure 2 is connected with the absolute error of the approximate solutions in the logarithmic scale. In Table 2, the numerical results of the suggested method have been compared with those of Block-Pulse functions (BPFs) method for 32 and 64 basis functions [24] and the Legendre wavelets (LWs) method for $m = 10$ ones [29]. It seems the presented method provides more accurate results than BPFs and LWs methods by smaller number of basis functions.

Example 4. [30] As the final example, consider that

$$u(x) = \int_0^x \frac{xt}{\sqrt{x-t}} u^2(t)dt + g(x), \tag{28}$$

in which $g(x) = x^3 - \frac{4096}{6435}x^{\frac{17}{2}}$. The exact solution of (28) is $u(x) = x^3$.

Table 2: Numerical results of Example 3

t	BPFs [24]		LWs [29]	Ours		Exact
	$k = 32$	$k = 64$	$m = 10$	$N = 4$	$N = 8$	
0.0	0.999123	0.999993	0.999432	1.000000	1.000000	1.000000
0.2	0.912305	0.912873	0.912320	0.913379	0.912871	0.912871
0.4	0.845156	0.845154	0.845321	0.844866	0.845154	0.845154
0.6	0.790527	0.790562	0.790539	0.790365	0.790569	0.790569
0.8	0.745361	0.745316	0.745342	0.745784	0.745356	0.745356
1.0	0.707120	0.707103	0.707163	0.707029	0.707107	0.707107

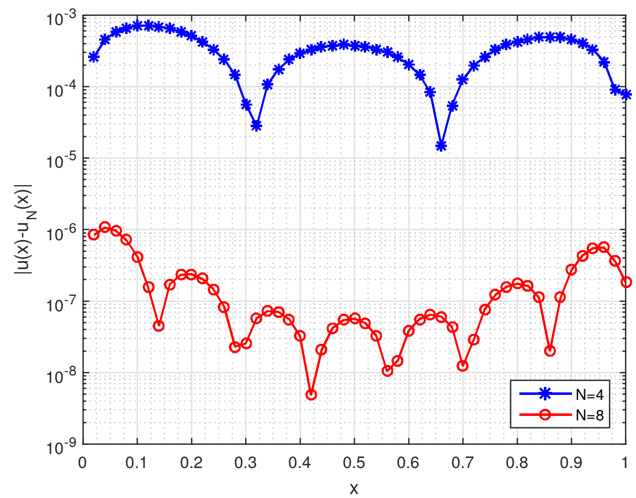

Figure 2: The illustration of absolute error for Example 3

For solving (28) by the presented method, similar to the previous examples assume that the approximate solution is in the form of (7). Subsequently,

$$u^2(x) = \mathbf{X}(x)\mathbf{D}^T\mathbf{A}\mathbf{X}(x)\mathbf{D}^T\mathbf{A} = \mathbf{A}^T\mathbf{D}\mathbf{X}^T(x)\mathbf{X}(x)\mathbf{D}^T\mathbf{A}. \tag{29}$$

Placing (29) into (28), it is obvious that

$$\mathbf{X}(x)\mathbf{D}^T\mathbf{A} = x\mathbf{A}^T\mathbf{D}\left(\int_0^x \frac{\hat{\mathbf{X}}(t)}{\sqrt{x-t}}dt\right)\mathbf{D}^T\mathbf{A} + g(x).$$

Note that $\hat{\mathbf{X}}(t) = t\mathbf{X}^T(t)\mathbf{X}(t) = [c_{ij}]_{1\le i,j\le N}$ is an $N \times N$ matrix so that $c_{ij} = t^{i+j-1}$. With the aid of (9), we can write

$$\mathbf{Q}(x) = [q_{ij}]_{1\le i,j\le N} = x\left(\int_0^x \frac{\hat{\mathbf{X}}(t)}{\sqrt{x-t}}dt\right),$$

in which

$$q_{ij} = \beta\left(i+j, \frac{1}{2}\right)x^{i+j+\frac{1}{2}}.$$

Hence, the following main matrix equation is found

$$\mathbf{X}(x)\mathbf{D}^T\mathbf{A} = \mathbf{A}^T\mathbf{D}\mathbf{Q}(x)\mathbf{D}^T\mathbf{A} + g(x). \tag{30}$$

Collocating (30) at the points defined as (12) generates a nonlinear algebraic system consisting of N equations with N unknowns. We solve this system by use of MATLAB *fsolve* command with the initial guess

$$\mathbf{A}_0^T = \underbrace{[0\ 0\ \ldots\ 0]}_{\mathrm{N}}.$$

Clearly, when $\mathbf{A}$ is detected the solution of (28) can be assigned by (7).

Utilizing the described method for $N = 5$ and $N = 8$, the solutions are acquired as follows:

$$\begin{cases} u_5(x) = -0.000027344159x^4 + 1.0000417x^3 - 0.000019310296x^2 \\ +0.0000026504592x + (3.3881318 \times 10^{-21}), \\ u_8(x) = -0.00019083629x^7 + 0.00055866874x^6 - 0.00064855817x^5 \\ +0.00038192279x^4 + 0.99987881x^3 + 0.000019722856x^2 \\ -0.0000012761595x + (1.0842022 \times 10^{-19}). \end{cases}$$

Figure 3 is related to the absolute error of the obtained solutions in the logarithmic scale.

Table 3 compares the absolute errors results of the proposed scheme for N basis functions with the second kind Chebyshev wavelet method [30] for $m' = 2^{k-1}M$ ones at some points of $[0, 1]$. It is observed that the presented method produces more accurate solutions.

5 Conclusion

In this research, a new method using the Fibonacci polynomials, collocation points, and matrix operations was established for solving the Abel's integral equation of second kind. The useful properties of the proposed method with

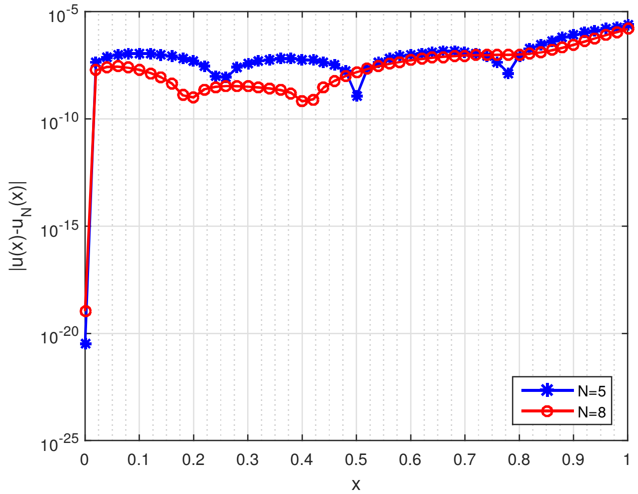

Figure 3: The illustration of absolute error for Example 4

the help of collocation technique converted the problem to a solvable system of algebraic equations. As it was seen, to gain the best approximations, the value of N must be selected large enough.

One of the method's advantages is the matrix forms of the calculations. This style of writing makes the computer programs more comfortable. The examples demonstrated the validity and the correctness of the presented method by some tables and figures. The numerical results and the comparisons with other methods in the literature revealed that the presented method works well in terms of the suitability, impression, accuracy, and computer-orientation.

The method described in this paper may be developed for other problems. Especially, the fractional integro-differential equations with a weakly singular kernel or the systems of Abel's integral equations can be suitable problems for everyone who is interested in these areas.

References

1. Abd-Elhameed, W.M. and Youssri, Y.H. *A novel operational matrix of Caputo fractional derivatives of Fibonacci polynomials: spectral solutions of fractional differential equations*, Entropy 18(10) (2016), 345, 15 pages.

2. Alipour, M. and Rostamy, D. *Bernstein polynomials for solving Abel's integral equation*, J. Math. Computer Sci. 3(4) (2011), 403–412.

Table 3: The absolute errors of Example 4

t	SCW [30]			Presented method	
	$k=3, M=2$	$k=4, M=2$	$k=5, M=2$	$N=5$	$N=8$
0.1	1.1485×10^{-3}	9.6679×10^{-5}	5.8961×10^{-5}	1.1091×10^{-7}	1.9331×10^{-8}
0.2	7.7311×10^{-4}	4.7192×10^{-4}	6.8431×10^{-5}	4.7529×10^{-8}	1.0123×10^{-9}
0.3	1.8040×10^{-3}	7.6218×10^{-4}	9.1859×10^{-5}	3.8377×10^{-8}	3.1911×10^{-9}
0.4	3.8610×10^{-3}	5.2959×10^{-4}	2.5427×10^{-4}	6.0674×10^{-8}	6.6190×10^{-10}
0.5	2.0979×10^{-2}	4.7042×10^{-3}	1.1218×10^{-3}	1.1457×10^{-9}	1.5094×10^{-8}
0.6	7.4657×10^{-3}	4.3467×10^{-4}	4.5196×10^{-4}	1.0197×10^{-7}	5.4143×10^{-8}
0.7	8.9784×10^{-5}	2.7024×10^{-5}	2.3869×10^{-6}	1.3104×10^{-7}	9.0162×10^{-8}
0.8	5.5454×10^{-4}	5.0051×10^{-5}	4.8479×10^{-6}	8.7990×10^{-8}	1.0116×10^{-7}
0.9	7.4323×10^{-2}	7.3236×10^{-3}	2.9982×10^{-3}	7.9713×10^{-7}	2.8504×10^{-7}

3. Andanson, P., Cheminat B. and Halbique, A.M. *Numerical solution of the Abel integral equation: application to plasma spectroscopy*, J. Phys. D Appl. Phys. 11(3) (1978), 209.

4. Azodi, H.D. *Euler polynomials approach to the system of nonlinear fractional differential equations*, Punjab Univ. J. Math. (Lahore) 51(7) (2019), 71–87.

5. Azodi, H.D. and Yaghouti, M.R. *Bernoulli polynomials collocation for weakly singular Volterra integro-differential equations of fractional order*, Filomat 32(10) (2018), 3623–3635.

6. Başi, A.K. and Yilçinbaş, S. *Numerical solutions and error estimations for the space fractional diffusion equation with variable coefficients via Fibonacci collocation method*, Springerplus 5 (2016) 1375.

7. Bicknell, M. *A primer for the Fibonacci numbers VII* Fibonacci Quart. 8 (1970), 407–420.

8. Cimatti, G. *Application of the Abel integral equation to an inverse problem in thermoelectricity*, Eur. J. Appl. Math. 20(6) (2009), 519–529.

9. Cremers, C.J. and Birkebak, R.C. *Application of the Abel integral equation to spectrographic data*, Appl. Opt. 5(6) (1966), 1057–1064.

10. Huang, L., Huang, Y. and Li, X.F. *Approximate solution of Abel integral equation*, Comput. Math. Appl. 56(7) (2008), 1748–1757.

11. Liu, Y.P. and Tao, L. *High accuracy combination algorithm and a posteriori error estimation for solving the first kind Abel integral equations*, Appl. Math. Comput. 178(2) (2006), 441–451.

12. Liu, Y.P. and Tao, L. *Mechanical quadrature methods and their extrapolation for solving first kind Abel integral equations*, J. Comput. Appl. Math. 201(1) (2007), 300–313.

13. Lucas, E. *Theorie de fonctions numeriques simplement periodiques* Amer. J. Math. 1 (1878), 184–240; 289–321.

14. Loh, J.R., Phang, C. and Isah, A. *New operational matrix via Genocchi polynomials for solving Fredholm-Volterra fractional integro-differential equations*, Adv. Math. Phys. 2017, ID 3821870, 12 pages.

15. Koç, A.B., Çakmak, M. and Kurnaz, A. and Uslu, K. *A new Fibonacci type collocation procedure for boundary value problems*, Adv. Differ. Equ. (2013) 2013: 262.

16. Kumar, S., Kumar, A., Kumar, D., Singh, J. and Singh, A. *Analytical solution of Abel integral equation arising in astrophysics via Laplace transform*, J. Egypt. Math. Soc. 23(1) (2015), 102–107.

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

18. Mirzaee, M. and Hoseini, S.F. *Solving systems of linear Fredholm integro-differential equations with Fibonacci polynomials*, Ain Shams Eng. J. 5(1) (2014), 271–283.

19. Mirzaee, M. and Hoseini, S.F. *A Fibonacci collocation method for solving a class of Fredholm-Volterra integral equations in two-dimensional spaces*, Beni-Suef Univ. J. Basic Appl. Sci. 3(2) (2014), 157–163.

20. Mirzaee, M. and Hoseini, S.F. *Numerical approach for solving nonlinear stochastic Itô-Volterra integral equations using Fibonacci operational matrices*, Sci. Iran. 22(6) (2015), 2472–2481.

21. Mirzaee, M. and Hoseini, S.F. *A new collocation approach for solving systems of high-order linear Volterra integro-differential equations with variable coefficients*, Appl. Math. Comput. 273 (2016), 637–644.

22. Mirzaee, M. and Hoseini, S.F. *A new collocation approach for solving systems of high-order linear Volterra integro-differential equations with variable coefficients*, Appl. Math. Comput. 311 (2017), 272–282.

23. Mirzaee, M. and Samadyar, S. *Numerical Solution of Weakly Singular Ito-Volterra Integral Equations via Operational Matrix Method based on Euler Polynomials*, Math. Res. 4(1) (2018), 91–104

24. Nosrati Sahlan, M., Marasi, H.R. and Ghahramani, F. *Block-pulse functions approach to numerical solution of Abel's integral equation*, Cogent Math. 2(1) (2015) Art. ID 1047111, 9 pp.

25. Pandey, R.K., Singh, O.P. and Singh, V.K. *Efficient algorithms to solve singular integral equations of Abel type*, Comput Math Appl. 57(4) (2009), 664–676.

26. Prajapati, R.N. Mohan, R. and Kumar, P. *Numerical solution of generalized Abel's integral equation by variational iteration method*, Am. J. Comput. Math. 2(4) (2012), 312–315.

27. Saadatmandi, A. and Dehghan, M. *A collocation method for solving Abel's integral equations of first and second kinds*, Z. Naturforsch. A 63(12) (2008), 752–756.

28. Setia A. and Pandey, R.K. *Laguerre polynomials based numerical method to solve a system of generalized Abel integral equations*, Procedia Eng. 38 (2012), 1675–1682.

29. Yousefi, S.A. *Numerical solution of Abel's integral equation by using Legendre wavelets*, Appl. Math. Comput. 175(1) (2006), 574–580.

30. Zhu, L. and Wang, Y. *Numerical solutions of Volterra integral equation with weakly singular kernel using SCW method*, Appl. Math. Comput. 260 (2015), 63–70.