International Journal of Mathematical Archive-6(8), 2015, 216-226
Available online through www.ijma.info ISSN 2229 – 5046

SOLUTION OF SECOND KIND VOLTERRA INTEGRAL AND INTEGRO-DIFFERENTIAL EQUATION BY BERNSTEIN POLYNOMIALS METHOD

MOHAMMED. KHALID SHAHOOTH*, OMAR. K. AL-HUSSEINI**

Pure Mathematics, Department of Mathematics,
Faculty Science and Technology, National University of Malaysia.

(Received On: 27-06-15; Revised & Accepted On: 27-07-15)

ABSTRACT

In this paper, we introduce a solution of second kind Volterra integral and integro-differential equations by using Bernstein polynomials method (BPM). First, we introduce the proposed method, then we used it to transform the integral and integro-differential equations to the system of algebraic equations. Finally, the numerical examples illustrate the efficiency and accuracy of this method.

Keywords: *Integral Equations, Integro-differential Equations, Bernstein polynomials method.*

1. INTRODUCTION

Volterra integral equation arise in engineering, physics, chemistry and biological problems such as parabolic boundary value problems, the spatio-temporal development of the epidemic, population dynamics and semi-conductor device. Many initial and boundary value problems associated with the ordinary and partial differential equations can be cast into the Volterra integral equation types. The Volterra integral equation was first used by Vito Volterra [1] in 1884.

Mathematical modelling of real-life problems usually results in functional equations, e.g. partial differential equations, integral and integro-differential equations, stochastic equations and others. Many mathematical formulation of physical phenomena contain integro-differential equations, these equations arises in fluid dynamics, biological models and chemical kinetics [2], for more details see [3,4]. This type of equations was introduced by Volterra for the first time in the early 1900. Volterra investigated the population growth, focussing his study on the hereditary influences, where through his research work the topic of integro-differential equations was established [5].

There are several numerical and analytical methods have been used to solve Volterra integral equations. For Example, A new approach to solve Volterra integral equation by using Bernstein's approximation is employed in [6]. Application of Collocation method on Volterra integral equations are investigated in [7, 8]. Taylor series expansion method is used for second kind Volterra integral equation in [9]. In [10] Chebyshev polynomials is used to find numerical solution of nonlinear Volterra integral equations of the second kind.[11] applied Variational iteration method to solve integral equation. Application of Adomian's decomposition method to solve integral equations are found in [12, 13]. Numerical solution of the second kind Volterra integral equation using an expansion method is employed in [14].

Integro-differential equations are usually difficult to solve analytically so it is required to obtain an efficient approximate solution. So, they have been of great interest by several authors. In literature, there exist many numerical and semi-analytical-numerical techniques to solve Integro-differential equation. For Example, Application of Adomian's decomposition method on Integro-differential equation are investigated in [15, 12, 13]. Comparison between Wavelet Galerkin method and Adomian's decomposition method to solve integro differential equation is found in [16]. The Tau method is applied to the integro-differential equation in [17]. [18] used Taylor polynomials to solve high-order Volterra integro-differential equation. Wavelet Galerkin method (WGM) to solve integro-differential equation can be found in [19]. In [20] Collocation method is used to solve fractional integro-differential equation. Application of He's Homotopy perturbation method to solve Volterra integro-differential equation are found in [21, 22]. Solution of forth-order integro - differential equation using variational iteration method can be found in [2]. In [23] rationalized Haar functions method is applied on system of linear integro-differential equations. In [24, 25] integro-differential equation is studied by using the differential transform method.

Corresponding Author: Mohammed. Khalid. Shahooth, Pure Mathematics,*
Department of Mathematics, Faculty Science and Technology, National University of Malaysia.

Bernstein polynomials method (**BPM)** has been recently used for the solution of integral and integro-differential equations. For example, Bernstein polynomials is applied to find an approximate solution for Fredholm integro-Differential equation and integral equation of the second kind in [26].[27] investigated the application of Bernstein polynomials for deriving the modified Simpson's 3/8, and the composite modified Simpson's 3/8 to solve one dimensional linear Volterra integral equations of the second kind. This method is employed to find an approximate solution of Fredholm integral equation of the second kind in [28].

In this paper, we propose Bernstein polynomials method to solve second kind Volterra integral and integro-differential equations. We have introduced that the **BPM** is very powerful and efficient technique in finding analytical solutions for the second kind Volterra integral and integro-differential equations.

A second kind Volterra integral and integro-differential equations are represented respectively in the form:

$$u(x) = f(x) + \lambda \int_0^x k(x,t)u(t)dt, \tag{1}$$

$$u^{(n)}(x) = f(x) + \lambda \int_0^x k(x,t)u(t)dt, \tag{2}$$

Where $a \le x \le b, \lambda$ are scalar parameters and $f(x)$ is the continuous function, $k(x,t)$ is the kernel of integral equation, $u^{(n)}(x) = \dfrac{d^n u}{dx^n}$ and $u(x)$ is the unknown function to be determine.

2. BERNSTEIN POLYNOMIALS METHOD (BPM)

Polynomials are incredibly useful mathematical tools as they are simply defined, can be calculated quickly on computer systems and represent a tremendous variety of functions. The Bernstein polynomials of degree - *n* are defined by [29]:

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad \text{for} \quad i = 0,1,2,...,n \tag{3}$$

Where $\binom{n}{i} = \dfrac{n!}{i!(n-i)!}$, (*n*) is the degree of polynomials, (*i*) is the index of polynomials and (*t*) is the variable.

The exponents on the (*t*) term increase by one as (*i*) increases, and the exponents on the (*1-t*) term decrease by one as (*i*) increases. The Bernstein polynomials of degree - *n* can be defined by blending together two Bernstein polynomials of degree (*n-1*) That is, the k^{th} n^{th}-degree Bernstein polynomial can be written as [29]:

$$B_k^n(t) = (1-t)B_k^{n-1}(t) + tB_{k-1}^{n-1}(t) \tag{4}$$

Bernstein polynomials of degree (*n*) can be written in terms of the power basis. This can be directly calculated using the equation (3) and the binomial theorem as follows [26]:

$$B_k^n(t) = \binom{n}{k} t^k (1-t)^{n-k} = \sum_{i=k}^{n} (-1)^{i-k} \binom{n}{i}\binom{i}{k} t^i \tag{5}$$

Where the binomial theorem is used to Expand $(1-t)^{n-k}$. The derivatives of the n^{th}-degree Bernstein polynomials are polynomials of degree (*n*-1)

$$\frac{d}{dt}B_k^n(t) = \frac{d}{dt}\binom{n}{k} t^k (1-t)^{n-k} = n\left(B_{k-1}^{n-1}(t) - B_k^{n-1}(t)\right), \quad 0 \le k \le n. \tag{6}$$

3. A MATRIX REPRESENTATION FOR BERNSTEIN POLYNOMIALS

In many applications, a matrix formulation for the Bernstein polynomials is useful. These are straight forward to develop if only looking at a linear combination in terms of dot products. Given a polynomial written as a linear combination of the Bernstein basis functions [26]:

$$B(t) = c_0 B_0^n(t) + c_1 B_1^n(t) + c_2 B_2^n(t) + \cdots + c_n B_n^n(t) \tag{7}$$

It is easy to write this as a dot product of two vectors

$$B(t) = \begin{bmatrix} B_0^n(t) & B_1^n(t) & B_2^n(t) & .\,. & B_n^n(t) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \tag{8}$$

which can be converted to the following form:

$$B(t) = \begin{bmatrix} 1 & t & t^2 & ... & t^n \end{bmatrix} \begin{bmatrix} b_{00} & 0 & 0 & \cdots & 0 \\ b_{10} & b_{11} & 0 & \cdots & 0 \\ b_{20} & b_{21} & b_{22} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{n0} & b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \tag{9}$$

where b_{nn} are the coefficients of the power basis that are used to determine the respective Bernstein polynomials, we note that the matrix in this case lower triangular. The matrix of derivatives of Bernstein polynomials is: [26].

$$B'(t) = \begin{bmatrix} 0 & 1 & 2t & ... & nt^{n-1} \end{bmatrix} \begin{bmatrix} b_{00} & 0 & 0 & \cdots & 0 \\ b_{10} & b_{11} & 0 & \cdots & 0 \\ b_{20} & b_{21} & b_{22} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{n0} & b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \tag{9a}$$

4. SOLUTION FOR VOLTERRA INTEGRAL EQUATIONS OF THE SECOND KIND

In this section Bernstein polynomials method is proposed to find an approximate solution for Volterra integral equations of the second kind. Consider the Volterra integral equation of the second kind in equation (1).

Applying the following equation:

$$u(x) = \begin{bmatrix} B_0^n(x) & B_1^n(x) & B_2^n(x) & \cdots & B_n^n(x) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \tag{10}$$

Substituting (10) into equation (1) we get:

$$\begin{bmatrix} B_0^n(x) & B_1^n(x) & B_2^n(x) & \cdots & B_n^n(x) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}$$

$$= f(x) + \lambda \int_a^x k(x,t) \begin{bmatrix} B_0^n(t) & B_1^n(t) & B_2^n(t) & \cdots & B_n^n(t) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} dt \tag{11}$$

Using the following equation into equation (11) we have:

$$u(t)=\begin{bmatrix}1 & x & x^2 & \cdots & x^n\end{bmatrix}\begin{bmatrix}B_0^n(x) & B_1^n(x) & B_2^n(x) & \cdots & B_n^n(x)\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix} \tag{12}$$

$$\begin{bmatrix}B_0^n(x)\; B_1^n(x)\; B_2^n(x)\cdots B_n^n(x)\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix}$$

$$= f(x)+\lambda\int_a^x k(x,t)\begin{bmatrix}1 & t & t^2\cdots t^n\end{bmatrix}\begin{bmatrix}b_{00} & 0 & 0 & \cdots & 0\\ b_{10} & b_{11} & 0 & \cdots & 0\\ b_{20} & b_{21} & b_{22} & \cdots & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ b_{n0} & b_{n1} & b_{n2} & \cdots & b_{nn}\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix}dt \tag{13}$$

Now to find the Volterra integration in equation (13). Then in order to determine $c_0, c_1, ...c_n$. we need *n* equations. Now choice $x_i, i=1,2,3...,n$ in the interval $[a,b]$, which gives *n* equations. Solve the *n* equations by Gauss elimination to find the values of $c_0, c_1, ...c_n$.

5. SOLUTION FOR VOLTERRA INTEGRO-DIFFERENTIAL EQUATIONS OF THE SECOND KIND

In this section Bernstein polynomials method is used to find the approximate solution for Volterra integro - differential equation of the second kind. Consider the Volterra integro-differential equation of the second kind in equation (2).

$$u^{(n)}(x)=\begin{bmatrix}B_0^n(x) & B_1^n(x) & B_2^n(x) & \cdots & B_n^n(x)\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix} \tag{14}$$

Substituting (14) into equation (2), we get:

$$\left[\begin{bmatrix}B_0^n(x)\; B_1^n(x)\; B_2^n(x)\cdots B_n^n(x)\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix}\right]^n$$

$$= f(x)+\lambda\int_a^x k(x,t)\begin{bmatrix}B_0^n(t)\; B_1^n(t)\; B_2^n(t)\cdots B_n^n(t)\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix}dt \tag{15}$$

Using the following equation into equation (15) we have:

$$u(t)=\begin{bmatrix}1 & x & x^2 & \cdots & x^n\end{bmatrix}\begin{bmatrix}B_0^n(x) & B_1^n(x) & B_2^n(x) & \cdots & B_n^n(x)\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix} \tag{16}$$

$$\left[\begin{bmatrix}B_0^n(x) \; B_1^n(x) \; B_2^n(x)\cdots B_n^n(x)\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix}\right]^n$$

$$= f(x)+\lambda\int_a^x k(x,t)\begin{bmatrix}1tt^2\cdots & t^n\end{bmatrix}\begin{bmatrix}b_{00} & 0 & 0 & \cdots & 0\\ b_{10} & b_{11} & 0 & \cdots & 0\\ b_{20} & b_{21} & b_{22} & \cdots & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ b_{n0} & b_{n1} & b_{n2} & \cdots & b_{nn}\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix}dt \tag{17}$$

Now to find all differentiation here, and the Volterra integration in equation (17). Then in order to determine $c_0,c_1,...c_n$. we need *n* equations. Now choice $x_i, i=1,2,3...,n$ in the interval $[a,b]$, which gives n equations. Solve the *n* equations by Gauss elimination to find the values of $c_0,c_1,...c_n$.. The following algorithm summarizes the steps for finding the solution for the second kind Volterra integral and integro-differential equations of the second kind.
.

6. ALGORITHM (BPM)

Input: $\left(f(x),k(x,t),u(x),a,b,\lambda\right)$

Output: polynomials of degree *n*

Step-1: Choice *n* the degree of Bernstein polynomials

$$B_i^n(t)=\binom{n}{i}t^i(1-t)^{n-i} \quad \text{for} \quad i=0,1,2,...,n$$

Step-2: Put the Bernstein polynomials in the linear Volterra integral and integro-differential equations of the second kind

$$u^{(n)}(x)=\begin{bmatrix}0 \; 1 \; 2t\cdots nt^{n-1}\end{bmatrix}\begin{bmatrix}b_{00} & 0 & 0 & \cdots & 0\\ b_{10} & b_{11} & 0 & \cdots & 0\\ b_{20} & b_{21} & b_{22} & \cdots & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ b_{n0} & b_{n1} & b_{n2} & \cdots & b_{nn}\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix}$$

$$= f(x)+\int_a^x k(x,t)\begin{bmatrix}B_0^n(t) \; B_1^n(t) \; B_2^n(t)\cdots B_n^n(t)\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ \vdots\\ c_n\end{bmatrix}dt$$

$$B_i^n(x) = f(x) + \int_a^x k(x,t) B_i^n(t) dt$$

Step-3: Compute Volterra integral $\int_a^x k(x,t)\begin{bmatrix}1 & t & t^2 & \cdots & t^n\end{bmatrix}\begin{bmatrix} b_{00} & 0 & 0 & \cdots & 0 \\ b_{10} & b_{11} & 0 & \cdots & 0 \\ b_{20} & b_{21} & b_{22} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{n0} & b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix}\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} dt$

Compute $\begin{bmatrix}0 & 1 & 2t & \cdots & nt^{n-1}\end{bmatrix}\begin{bmatrix} b_{00} & 0 & 0 & \cdots & 0 \\ b_{10} & b_{11} & 0 & \cdots & 0 \\ b_{20} & b_{21} & b_{22} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{n0} & b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix}\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}$ and $\int_a^x k(x,t) B_i^n(t) dt$

Step-4: Compute $c_0, c_1, \cdots, c_n$, where $x_i, i = 1, 2, 3, \cdots, n$, $x_i \in [a,b]$

End.

7. NUMERICAL EXPERIMENTS

In this section we apply BPM to solving the linear Volterra integral and integro-differential equations of the second kind. Also we presented here two linear Volterra integral equations and two linear Volterra Integro-differential equations. These four examples, the first two examples are solved by Adomian's decomposition method (ADM) [13]. And the last two examples are solved by Homotopy analysis method (HAM) [5].The computations associated with these examples were performed using Matlab ver.2013a.

Example1: Consider the following Volterra integral equation of the second kind [13].

$u(x) = 1 - \int_0^x u(t) dt$, with the exact solution $u(x) = e^{-x}$.

Here we can noticed that $f(x) = 1$, $\lambda = -1$ and $k(x,t) = 1$.

Table (1): Numerical results for example 1 with exact solution

t	$yexact$	$yapp, n = 1$	$yapp, n = 2$	$yapp, n = 3$
0	1.0000	1.0000	1.0000	1.0000
0.1000	0.9048	1.0000	0.9084	0.9050
0.2000	0.8187	1.0000	0.8231	0.8188
0.3000	0.7408	1.0000	0.7442	0.7407
0.4000	0.6703	1.0000	0.6715	0.6700
0.5000	0.6065	1.0000	0.6052	0.6062
0.6000	0.5488	1.0000	0.5452	0.5486
0.7000	0.4966	1.0000	0.4916	0.4966
0.8000	0.4493	1.0000	0.4442	0.4496
0.9000	0.4066	1.0000	0.4032	0.4069
1.0000	0.3679	1.0000	0.3685	0.3679

Table (2): Absolute error for example 1 by using BPM.

$(yexact-yapp,n=1)^2$	$(yexact-yapp,n=2)^2$	$(yexact-yapp,n=3)^2$
0	0	0
0.0091	0.0000	0.0000
0.0329	0.0000	0.0000
0.0672	0.0000	0.0000
0.1087	0.0000	0.0000
0.1548	0.0000	0.0000
0.2036	0.0000	0.0000
0.2534	0.0000	0.0000
0.3032	0.0000	0.0000
0.3522	0.0000	0.0000
0.3996	0.0000	0.0000

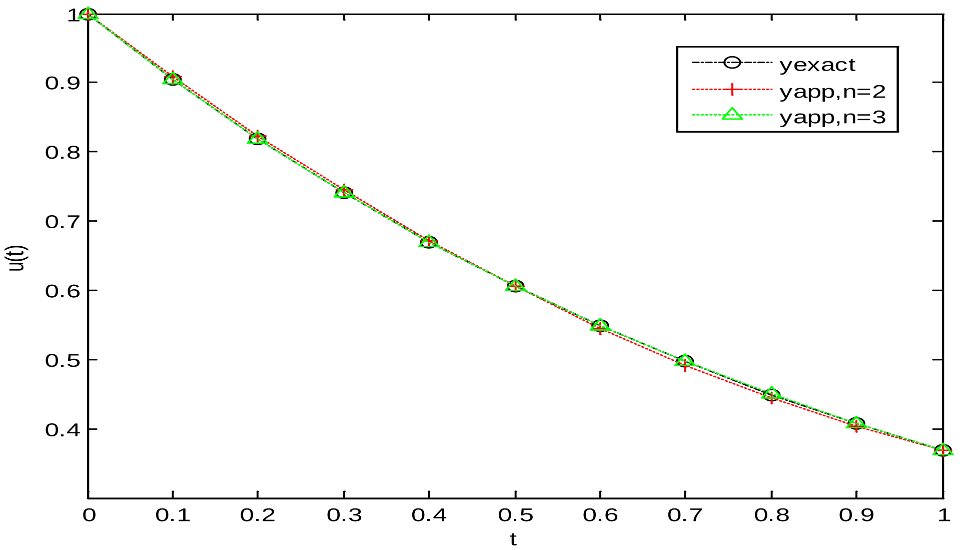

Figure-1: 3rd-order approximate solution by BPM and exact solution.

Example 2: Consider the following Volterra integral equation of the second kind [13].

$$u(x)=1+\int_0^x (t-x)u(t)dt\text{ , with the exact solution } u(x)=\cos(x).$$

Also we can noticed that $f(x)=1, \lambda=1$ and $k(x,t)=(t-x)$.

Table (1): Numerical results for Example 2 with exact solution

t	$yexact$	$yapp,n=1$	$yapp,n=2$	$yapp,n=3$
0	1.0000	1.0000	1.0000	1.0000
0.1000	0.9950	0.9571	0.9928	0.9953
0.2000	0.9801	0.9143	0.9769	0.9802
0.3000	0.9553	0.8714	0.9525	0.9551
0.4000	0.9211	0.8286	0.9195	0.9205
0.5000	0.8776	0.7857	0.8778	0.8770
0.6000	0.8253	0.7429	0.8276	0.8248
0.7000	0.7648	0.7000	0.7688	0.7646
0.8000	0.6967	0.6572	0.7014	0.6968
0.9000	0.6216	0.6143	0.6253	0.6219
1.0000	0.5403	0.5714	0.5407	0.5404

Table (2): The error for Example 2 by using BPM.

$(yexact - yapp, n = 1)^2$	$(yexact - yapp, n = 2)^2$	$(yexact - yapp, n = 3)^2$
0	0	0
0.0014	0.0000	0.0000
0.0043	0.0000	0.0000
0.0070	0.0000	0.0000
0.0086	0.0000	0.0000
0.0084	0.0000	0.0000
0.0068	0.0000	0.0000
0.0042	0.0000	0.0000
0.0016	0.0000	0.0000
0.0001	0.0000	0.0000
0.0010	0.0000	0.0000

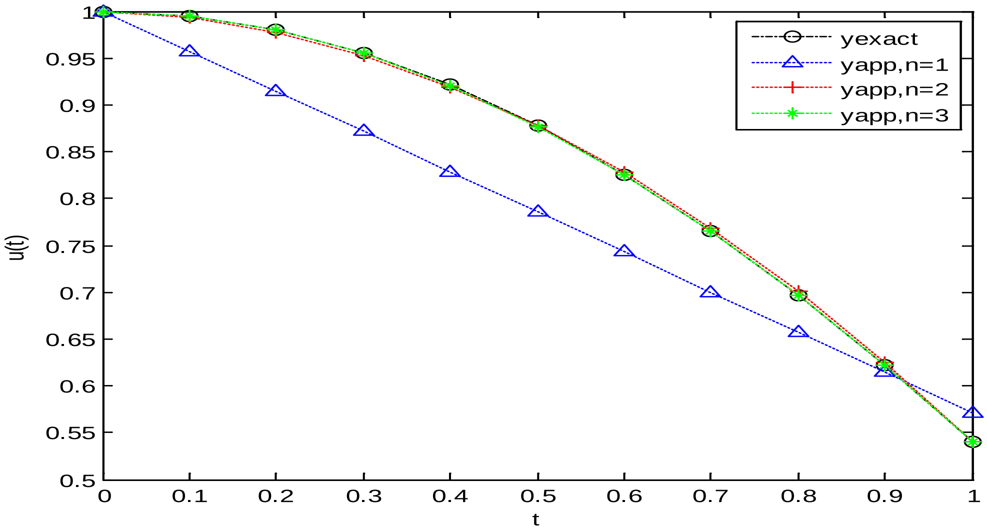

Figure-2: 3rd-order approximate solution by BPM and exact solution.

Example 3: Consider the following Volterra integro-differential equation of the second kind [5].

$u'(x) = 1 - \int_0^x u(t)dt, u(0) = 0$, and the exact solution is $u(x) = \sin(x)$.

Table (1): Numerical results for Example 3 with exact solution

t	$yexact$	$yapp, n=1$	$yapp, n=2$	$yapp, n=3$
0	0	0	0	0
0.1000	0.0998	0.0667	0.1180	0.1009
0.2000	0.1987	0.1333	0.2290	0.2002
0.3000	0.2955	0.2000	0.3329	0.2970
0.4000	0.3894	0.2667	0.4298	0.3906
0.5000	0.4794	0.3333	0.5196	0.4803
0.6000	0.5646	0.4000	0.6023	0.5652
0.7000	0.6442	0.4667	0.6780	0.6445
0.8000	0.7174	0.5333	0.7466	0.7176
0.9000	0.7833	0.6000	0.8082	0.7836
1.0000	0.8415	0.6667	0.8627	0.8417

Table (2): The error for Example 3 by using BPM.

$(yexact - yapp, n=1)^2$	$(yexact - yapp, n=2)^2$	$(yexact - yapp, n=3)^2$
0	0	0
0.0011	0.0003	0.0000
0.0043	0.0009	0.0000
0.0091	0.0014	0.0000
0.0151	0.0016	0.0000
0.0213	0.0016	0.0000
0.0271	0.0014	0.0000
0.0315	0.0011	0.0000
0.0339	0.0009	0.0000
0.0336	0.0006	0.0000
0.0306	0.0005	0.0000

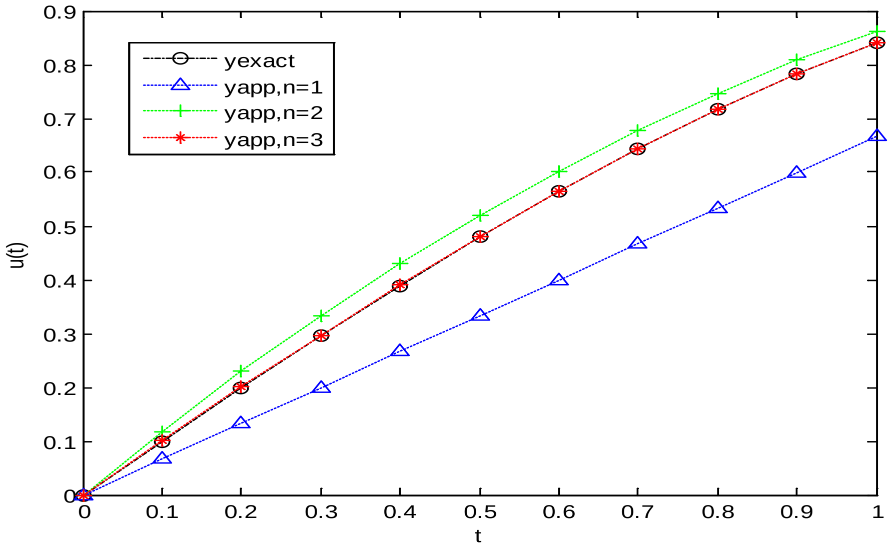

Figure-3: 3rd-order approximate solution by BPM and exact solution.

Example 4: Consider the following Volterra integro-differential equation of the second kind [5].

$$u''(x) = 1 + x + \int_0^x (x-t)u(t)dt, \quad u(0)=1, u'(0)=1$$ and the exact solution is $u(x) = e^x$.

Table (1): Numerical results for Example 4 with exact solution

t	$yexact$	$yapp, n=1$	$yapp, n=2$	$yapp, n=3$
0	1.0000	1.0000	1.0000	1.0000
0.1000	1.1052	1.1000	1.1083	1.1053
0.2000	1.2214	1.2000	1.2330	1.2223
0.3000	1.3499	1.3000	1.3743	1.3528
0.4000	1.4918	1.4000	1.5320	1.4984
0.5000	1.6487	1.5000	1.7063	1.6609
0.6000	1.8221	1.6000	1.8970	1.8420
0.7000	2.0138	1.7000	2.1043	2.0435
0.8000	2.2255	1.8000	2.3280	2.2670
0.9000	2.4596	1.9000	2.5683	2.5143
1.0000	2.7183	2.0000	2.8251	2.7871

Table (2): The error for Example 4 by using BPM.

$(yexact - yapp, n=1)^2$	$(yexact - yapp, n=2)^2$	$(yexact - yapp, n=3)^2$
0	0	0
0.0000	0.0000	0.0000
0.0005	0.0001	0.0000
0.0025	0.0006	0.0000
0.0084	0.0016	0.0000
0.0221	0.0033	0.0001
0.0493	0.0056	0.0004
0.0984	0.0082	0.0009
0.1811	0.0105	0.0017
0.3132	0.0118	0.0030
0.5159	0.0114	0.0047

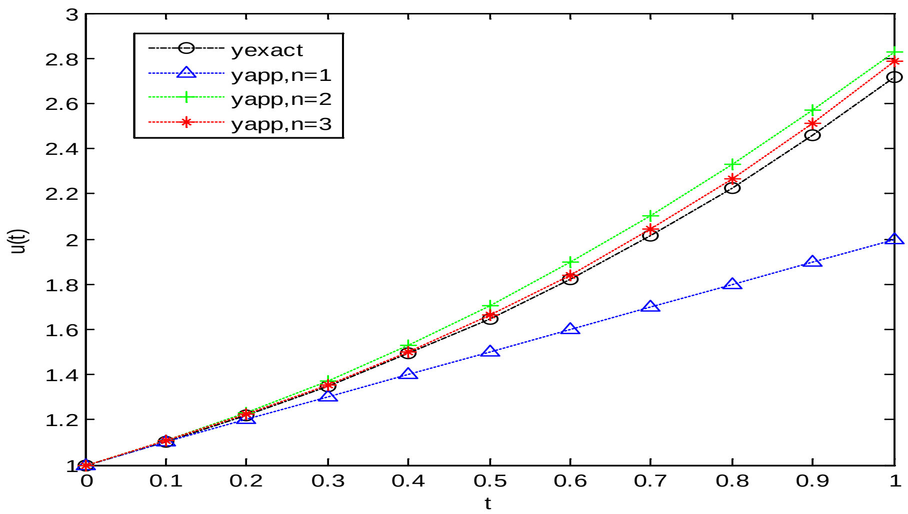

Figure-4: 3rd-order approximate solution by BPM and exact solution.

CONCLUSION

In this paper, we have successfully used BPM for solving Volterra integral and integro-differential equations of the second kind. The integral equations are usually difficult to solve analytically. In many cases, it is required to obtain the numerical solution, for this purpose the presented method can be proposed and it's apparently seen that BPM is a powerful and easy-to-use analytic tool for finding the solutions for integral and integro-differential equations. Numerical experiments in comparison with other methods such as Adomian's decomposition method (ADM) and Homotopy analysis method (HAM). The results shown the efficiency of the Bernstein polynomials method (BPM) for solving this type of equations. Also we noted that when the degree of Bernstein polynomials is increasing the errors decrease to smaller values.

REFERENCES

1. V. Volterra, Theory of functional of integral and integro-differential equations, Dover, New York, (1959).
2. N. Sweilam, Fourth order integro-differential equations using variational iteration method, Compu. Math. Appl., 54 (2007), 1086-1091.
3. P.K. Kythe, P. Puri, Computational methods for linear integral equations, University of New Orlans, New Orlans, 2002.
4. A.M. Wazwaz, A comparison study between the modified decomposition method and the traditional methods for solving nonlinear integral equations, Applied Mathematics and Computation (in press).
5. K. Shah, T. Singh, Solution of second kind Volterra integral and integro-differential equation by Homotopy analysis method, International Journal of Mathematical Archive,6(2015), 49-59.
6. K. Maleknejad, E. Hashemizadeh, R. Ezzati, A new approach to the numerical solution of Volterra integral equations by using Bernstein's approximation, Commu. Nonlinear Sci. Num. Simul., 16 (2011), 647-655.
7. H. Brunner, On the Numerical Solution of Nonlinear Volterra-Fredholm Integral Equations by Collocation Methods, SIAM Jour. Numer. Anal., 27 (1990), 987-1000.
8. H. Brunner, The numerical solution of weakly singular Volterra integral equations by collocation on graded meshes, Appl. Math. Compu., 45 (1985), 417-437.

9. K. Maleknejad, N. Agazadeh, Numerical solution of Volterra integral equations of the second kind with convolution kernel by using Taylor-series expansion method, Appl. Math. Compu., 161 (2005), 915-922.
10. K. Maleknejad, S. Sohrabi, Y. Rostami, Numerical solution of nonlinear Volterra integral equations of the second kind by using Chebyshev polynomials, Appl. Math. Compu., 188 (2007), 123-128. 17.
11. L. Xu, Variational iteration method for solving integral equations, I. Jour. comp. math. appl., 54 (2007), 1071-1078.
12. A. Wazwaz, Two methods for solving integral equations, Appl. Math. Compu., 77 (1996), 79-89.
13. A. Wazwaz, Linear and Nonlinear Integral Equations: Methods and Applications, Higher education press, Springer, (2011).
14. M. Rabbani, K. Maleknejad, N. Aghazadeh, Numerical computational solution of the Volterra integral equations system of the second kind by using an expansion method, Appl. Math. Compu., 187 (2007), 1143-1146.
15. I. Hashim, Adomian decomposition method for solving BVPs for fourth-order integro- differential equations, Jour. Compu. Appl. Math., 193 (2006), 658-664.
16. S. El-Sayed, M. Abdel-Aziz, A comparison of Adomian's decomposition method and Wavelet-Galerkin method for integro-differential equations, Appl. Math. Compu., 136 (2003), 151-159.
17. S. Hosseini, S. Shahmorad, Numerical solution of a class of Integro-Differential equations by the Tau Method with an error estimation, Appl. Math. Compu., 136 (2003), 559-570.
18. K. Maleknejad, Y. Mahmoudi, Taylor polynomial solution of high-order nonlinear Volterra-Fredholm integro differential equations, Appl. Math. Compu., 145 (2003), 641-653.
19. A. Avudainayagam, C. Vani, Wavelet-Galerkin method for integro-differential equations, Appl. Numer. Math., 32 (2000), 247-254.
20. E. Rawashdeh, Numerical solution of fractional integro-differential equation by collocation method, Appl. Math. Compu., 176 (2006), 1-6.
21. M. Dehghan, F. Shakeri, Solution of an integro-differential equation arising in oscillating magnetic field using He's Homotopy Perturbation method, PIER, 78 (2008), 361-376.
22. M. El-Shahed, Application of He's Homotopy Perturbation Method to Volterra Integro-differential equation, Int. Jour. Nonlinear Sci. Num. Simulat., 6 (2005), 163-168. 16.
23. K. Maleknejad, F. Mirzaee, S. Abbasbandy, Solving linear integro-differential equations system by using rationalized Haar function method, Appl. Math. Compu., 155 (2005), 317- 328.
24. A. Arikoglu, I. Ozkol, Solution of boundary value problems for integro-differential equations by using differential transform method, Appl. Math. Compu., 168 (2005), 1145-1158.
25. P. Darnaia, A. Ebadian, A method for the numerical solution of the integro-differential equations, Appl. Math. Compu., 188 (2007), 657-668.
26. A. AL-Juburee, Approximate Solution for linear Fredholm Integro-Differential Equation and Integral Equation by Using Bernstein Polynomials method, Journal of the Faculty of Education .Al- Mustansiriyah University. Iraq, (2010).
27. J. Ahmad, Deriving the Composite Simpson Rule by Using Bernstein Polynomials for Solving Volterra Integral Equations. Baghdad Science Journal, 11(2014).
28. H. Ali, S. Hussain, The Collocation Method for Solving the Linear Fredholm Integral Equation of the Second Kind Using Bernstein Polynomials, Eng. Tech. Journal, 28(2010).
29. K. Joy, Bernstein Polynomials, On-Line Geometric Modeling Notes, 2000.

Source of support: Nil, Conflict of interest: None Declared