Approximate Solution to First-Order Integro-differential Equations Using Polynomial Collocation Approach

Ganiyu Ajileye* and F.A. Aminu

Department of Mathematics and Statistics, Federal University Wukari, Taraba, Nigeria

Abstract

In this study, power series and shifted Chebyshev polynomials are used as basis function for solving first order volterra integro-differential equations using standard collocation method. An assumed approximate solution in terms of the constructed polynomial was substituted into the class of integro-differential equation considered. The resulted equation was collocated at appropriate points within the interval of consideration [0,1] to obtain a system of algebraic linear equations. Solving the system of equations, by inverse multiplication, the unknown coefficients involved in the equations are obtained. The required approximate results are obtained when the values of the constant coefficients are substituted back into the assumed approximate solution. Numerical example are presented to confirm the accuracy and efficiency of the method.

Keywords: Integro-differential equations • Collocation • Power series polynomial • Shifted chebyshev polynomial

Introduction

An Integro-differential equation is any type of equation combining an unknown function involving both integral and derivative operators. It is applicable in several fields, such as engineering and science. These equations are either Fredholm, Volterra, or Fredholm-Volterra integro-differential equations. As a result, a numerical method for solving such equations is required. Some numerical approaches in literature for solving integro-differential equations include: Collocation method [1-5], Hybrid linear multistep method [6-8], Differential transform method [9], Perturbed Method [10], Homotopy Perturbation [11], Bernoulli matrix method [12], The Mellin transform approach [13] and Pseudospectral Method [14]. Shahooth et al. presented a numerical method for solving the linear Volterra-Fredholm integro-differential equations of the second kind [15]. This method is called the Bernestein polynomial method. This technique transforms the integro-differential equations into the system of algebraic equations. Some numerical results are presented to illustrate the efficiency and accuracy of this method.

In this work, we consider Volterra Integro-differential equation of the form:

$$y'(x) = f(x) + \lambda \int_0^x k(x,t) y(t) dt$$

Subject to initial condition;

$$y^j(0) = a_j, j = 0,1$$

Where $y^j = d^j y / dy^j$ y(x) is the unknown function, f(x) is a known function, λ is a known constant and k(x,t) is the integral kernel function.

Basic Definitions

This section defines the basic terms that will be frequently used during the research work.

*Address for Correspondence: *Ganiyu Ajileye, Department of Mathematics and Statistics, Federal University Wukari, Taraba, Nigeria, Email: ajileye@fuwukari.edu.ng*

Definition 1: Standard Collocation Method: This is the method partitioning intervals into different points. For example, let say an interval [a,b], N be the number of collocation points to get the desired collocation points, we use;

$$x_p = a + \frac{b-a}{N} p, where \ p = 1,2,\ldots N$$

Definition 2: Power Series Polynomial: A power series is basically an infinite degree polynomial that represents some function. It is generally written in the form; $y(x) = \sum_{n=0}^{\infty} a_n x^n$ Where an is the coefficient of the polynomials which are real numbers.

Definition 3: Shifted Chebyshev Polynomials: This is a special type of polynomial generated from Chebyshev polynomial. The shifted Chebyshev has interval [0,1], while Chebyshev has an interval of [-1,1]. The shifted Chebyshev is generated using;

$$T_n *(x) = T_n (2x-1), n \ \geq \ 0$$

where $T_n (x)$ is the shifted chebyshev term, while $T_n (x)$ is the chebyshev term.

Few terms of the shifted Chebyshev terms are;

$T_0*(x)=1$

$T_1*(x)=2x-1$

$T_2*(x)=8x^2-8x+1$

$T_3*(x)=32x^3-48x^2+18x-1$

$T_4*(x)=128x^4-256x^3+160x^2+32x+1$

Methodology

In this section, combination of collocation method and power series approximation is employed for the numerical solution of volterra integro-dfferential equations.

Let the solution to (1) and (2) be approximated by (3)

$$y(x) = \sum_{n=0}^{N} a_n x^n, where \ n \geq 0$$

Where an is the real coefficients to be determined, differenting (3) with respect to x, gives

$$y'(x) = \sum_{n}^{N} n a_n x^{n-1}, where \ n \geq 0$$

substituting (3) and (4) into (1), gives

$$\left[\sum_n^N na_n x^{(n-1)}\right] = f(x) + \left[\sum_{n=0}^N \ddot{e}\int_a^x a_n t^n k(x,t)dt\right]$$

simplifying (5) gives,

$$\sum_{n=0}^N a_n\left[nx^{(n-1)} - \ddot{e}\int_a^x t^n k(x,t)dt\right] = f(x)$$

Hence,

$$\sum_{n=0}^N a_n \tau_n(x) = f(x)$$

Where

$$\left[nx^{n-1} - \ddot{e}\int_a^x t^n k(x,t)dt\right] = \tau_n(x)$$

Equation (6) can be written in the form

$$\tau_n(x)\mathbf{A} = f(x)$$

Where

$$\tau_n = \begin{bmatrix}\tau_0(x) & \tau_1(x)\ldots & \tau_n(x)\end{bmatrix}, \mathbf{A} = \begin{bmatrix}a_0 & a_1\ldots & a_n\end{bmatrix}^T$$

Collocating equation (7) using the standard collocation points

Where $x_p = a + \dfrac{b-a}{N}p, p = 0,1,2\ldots\ldots N$, gives

$$\tau(x_p) = \begin{pmatrix} \tau_0 x_0 & \tau_1 x_0 & \cdot & \cdot & \tau_n x_0 \\ \tau_0 x_1 & \tau_1 x_1 & \cdot & \cdot & \tau_n x_0 \\ \cdot & \cdot & & & \cdot \\ \cdot & \cdot & & & \cdot \\ \cdot & \cdot & & & \cdot \\ \tau_0 x_N & \tau_1 x_N & \cdot & \cdot & \tau_n x_N \end{pmatrix}$$

$$f(x_p) = \begin{bmatrix}f(x_0) & f(x_1)\ldots & f(x_N)\end{bmatrix}^T$$

Considering the initial condition,

$$y^{(j)}(a_j) = a_j, j = 0,1,2\ldots n-1$$

Substituting (3) into (9) we have,

$$\frac{d^{(j)}}{dx^j}\left[\sum_{n=0}^N a_n x^n\right] = a_j$$

Using equation(10) and equation (8) gives

$$\tau^*(x_p)A = f^*(x_p)$$

Solving equation (11) using inverse multiplication to get the values of the unknown coefficient a_n, n=1, 2, …N and substituting the value of the coefficients into the approximate solution gives the numerical results. Repeating the above

steps using shifted Chebyshev polynomial as the basis polynomial function, we obtained the approximate solution.

Numerical Examples: In this section, numerical examples with initial conditions are solved to confirm the efficiency and accuracy of the method. Let y_n (x) and y(x) be the approximate and exact solution respectively. Error_N = |y_n (x)-y(x)|.

Example 1: Considering Volterra integro-differential equation

$$y'(x) + y(x) = f(x) - \int_0^x ty(t)dt$$

Where $f(x) = (x^2 + 2x + 1)e^{-x} + 5x + 8$

subject to initial condition y(0)=10

Exact solution: y(x)=10-xe^{-x}

Solution 1: Solving at N = 5 using power series polynomial. Substituting approximate solution (3) into example 1 gives

$$\left[\sum_{n=0}^5 na_n x^{(n-1)}\right] + \sum_{n=0}^5 a_n x^n = f(x) - \left[\sum_{n=0}^5 \int_0^x a_n t^{n+1}dt\right]$$

Collecting the like terms gives

$$\sum_{n=0}^5 a_n\left[nx^{(n-1)} + x^n + \int_0^x t^{n+1}dt\right] = f(x)$$

Equation (12) can be written in the form

$$\tau(x)\mathbf{A} = f(x)$$

Where

$$\tau(x) = \sum_{n=0}^5\left[nx^{(n-1)} + x^n + \int_0^x t^{n+1}dt\right], \mathbf{A} = \begin{bmatrix}a_0 & a_1\ldots & a_n\end{bmatrix}^T$$

collocating at $x_5 = \begin{bmatrix}0 & \dfrac{1}{5} & \dfrac{2}{5} & \dfrac{3}{5} & \dfrac{4}{5} & 1\end{bmatrix}$ and substituting the initial conditions gives

$$\tau^*(x_p)A = f^*(x_p)$$

Solving equation (14), we obtained the following approximate solution

$$y_s = 9.9999999937 - 0.9999952318x + 0.9987005900x^2 - 0.4921341016x^3 + 0.1487825625x^4 - 0.0232281987x^5$$

Using the above steps for solving shifted Chebyshev polynomial as the basis polynomial function, we obtained the approximate solution;

$$y_s = 9.9999999820 - 0.9999901422x + 0.9976550264x^2 - 0.4861629152x^3 + 0.1707549056x^4 - 0.0425249792x^5$$

Example 2: Considering Volterra integro-differential equation

$$y'(x) + y(x) = f(x) + \int_0^x e^{t-x}y(t)dt$$

Where f(x)=0

subject to initial condition y(0)=1

Exact solution: $y(x) = e^{-x}\cosh(x)$

Solution 2: Solving at N = 5 using power series polynomial. Substituting approximate solution (3) into example 2 gives

$$\left[\sum_{n=0}^5 na_n x^{(n-1)}\right] + \sum_{n=0}^5 a_n x^n = f(x) + \left[\sum_{n=0}^5 \int_0^x a_n e^{t-x}t^n dt\right]$$

Collecting the like terms gives

$$\sum_{n=0}^{5} a_n \left[nx^{(n-1)} + x^n - \int_0^x e^{t-x} t^n dt \right] = f(x)$$

Equation (15) can be written in the form

$$\tau(x) \mathbf{A} = f(x)$$

Where

$$\tau(x) = \sum_{n=0}^{5} \left[nx^{(n-1)} + x^n - \int_0^x e^{t-x} t^n dt \right], \mathbf{A} = \begin{bmatrix} a_0 & a_1 \dots & a_n \end{bmatrix}^T$$

collocating at $x_5 = \begin{bmatrix} 0 & \dfrac{1}{5} & \dfrac{2}{5} & \dfrac{3}{5} & \dfrac{4}{5} & 1 \end{bmatrix}$ and substituting the initial conditions gives

$$\tau^*(x_p) A = f^*(x_p)$$

Solving equation (17), we obtained the following approximate solution (Tables 1-4).

Table 1: Exact and approximate values of example 1.

X_i	Exact Solution	Power Series$_{n=5}$	Shifted Chebyshev$_{n=5}$
0	10	9.999999994	9.999999982
0	9.909516258	9.909509989	9.909508005
0	9.836253849	9.836242518	9.836278452
0	9.777754534	9.77774555	9.777945271
0	9.731871982	9.731868389	9.732450172
1	9.69673467	9.696733792	9.69799158
1	9.670713018	9.670710096	9.67297361
1	9.652390287	9.652383354	9.655955045
1	9.640536829	9.640529449	9.645598296
1	9.634087306	9.634086223	9.640618379
1	9.632120559	9.632125613	9.639731878

Table 2. Absolute error for example 1.

X_i	Power Series$_{N=5}$	Shifted Chebyshev$_{n=5}$
0	6.00E-09	1.70E-08
0	6.27E-06	8.25E-06
0	1.13E-05	2.46E-05
0	8.98E-06	1.91E-04
0	3.59E-06	5.78E-04
1	8.78E-07	1.26E-03
1	2.92E-06	2.26E-03
1	6.93E-06	3.56E-03
1	7.38E-06	5.06E-03
1	1.08E-06	6.53E-03
1	5.05E-06	7.61E-03

Table 3. Exact and approximate values of example 2.

X_i	Exact Solution	Power Series$_{n=5}$	Shifted Chebyshev$_{n=5}$
0	1	0.999999938	0.999999916
0.1	0.909365377	0.90934122	0.909281555
0.2	0.835160023	0.835116552	0.834695109
0.3	0.774405818	0.774370716	0.773063213
0.4	0.724664482	0.724648492	0.72176967
0.5	0.68393972	0.683932578	0.678688419
0.6	0.650597106	0.650581523	0.642112506
0.7	0.623298482	0.623267655	0.610683052
0.8	0.600948259	0.600915006	0.583318222
0.9	0.582649444	0.582637242	0.559142197
1	0.567667642	0.567675589	0.537414142

Table 4. Absolute error of example 2.

X_i	Power Series$_{N=5}$	Shifted Chebyshev$_{N=5}$
0	6.24E-08	8.43E-08
0.1	2.42E-05	8.38E-05
0.2	4.35E-05	4.65E-04
0.3	3.51E-05	1.34E-03
0.4	1.60E-05	2.89E-03
0.5	7.14E-06	5.25E-03
0.6	1.56E-05	8.48E-03
0.7	3.08E-05	1.26E-02
0.8	3.33E-05	1.76E-02
0.9	1.22E-05	2.35E-02
1	7.95E-06	3.03E-02

$$y_5 = 0.9999999376 - 0.9999822616 + 0.9949659319x^2 - 0.6356387354x^3 + 0.2600574315x^4 - 0.0517267155x^5$$

Conclusion

The numerical solution of the Volterra integro-differential equation is presented in this study and the results are compared to the exact solution. Power series and shifted Chebyshev polynomials are used as the basis functions.

The numerical results of the examples as shown in tables shows that the results of power series approximation converges to the exact solution than the results of shifted chebyshev approximation. Observation also shows that as the value of x increases i.e. (from 0 to 1), the results of the shifted chebyshev diverges from the exact solutions.

Conflict of Interest

None.

References

1. Olumuyiwa, A. Agbolade and Timothy A. Anak. "Solution of first order Volterra linear integro-differential equations by collocation method." *J Appl Math* 1510267 (2017).

2. Adesanya, A. O., Yahaya Y. A., Ahmed B. and Onsachi R. O. "Numerical Solution of Linear Integral and Integro-Differential Equations Using Boubakar Collocation Method." *Intern J Math Sci Optim Theory Appl* 2019 (2019): 592-598.

3. Gegele, O. A., Evans O. P. and Akoh D. "Numerical Solution of Higher Order Linear Fredholm–Integro–Differential Equations." *Am J Eng Res* 3 (2014): 243-247.

4. Nemati, S., Pedro Lima and Yadollah Ordokhani. "Numerical method for the mixed Volterra-Fredholm integral equations using hybrid Legendre functions." *Appl Math* 2015 (2015): 184-193.

5. Abubakar, A. and Taiwo O. A. "Integral collocation approximation methods for the numerical solution of high-orders linear Fredholm-Volterra integro-differential equations." *J Comput Appl Math* 4 (2014): 111-117.

6. Ajileye, G., Amoo S. A. and Ogwumu O. D. "Hybrid Block Method Algorithms for Solution of first order initial value problems in ordinary differential equations." *J Appl Computat Math* 7 (2018): 2.

7. Mehdiyeva, Galina, Mehriban Imanova and Vagif Ibrahimov. "On the Construction of the Multistep Methods to Solving the Initial-Value Problem for ODE and the Volterra Integro-Differential Equations."

8. Ajileye, G., Okedayo G. T. and Aboiyar T. "Laguerre Collocation Approach for Continuous Hybrid Block Method for Solving Initial Value Problems of Second Order Ordinary Differential Equations." *Trends Food Sci Techn* 2 (2017): 379- 383.

9. Celik, Ercan and Khatereh Tabatabaei. "Solving a class of Volterra integral equation systems by the differential transform method." *Int J Nonlinear Sci* 16 (2013): 87-91.

10. Uwaheren, O. A., Adebisi A. F. and Taiwo O. A. "Perturbed collocation method for solving singular multi-order fractional differential equations of Lane-Emden type." *J Niger Soc Phys Sci* (2020): 141-148.

11. Raftari, Behrouz. "Numerical solutions of the linear Volterra integro-differential equations: Homotopy perturbation method and finite difference method." *World Appl Sci J* 9 (2010).

12. Bhrawy, Ali H., Emran Tohidi and Fazlollah Soleymani. "A new Bernoulli matrix method for solving high-order linear and nonlinear Fredholm integro-differential equations with piecewise intervals." *Appl Math Comput* 219 (2012): 482-497.

13. Fadugba, Sunday Emmanuel. "Solution of fractional order equations in the domain of the Mellin transform." *J Niger Soc Phys Sci* (2019): 138-142.

14. El-Kady, M. and Biomy M. "Efficient Legendre pseudospectral method for solving integral and integro-differential equations." *Commun Nonlinear Sci Numer Simul* 15 (2010): 1724-1739.

15. Shahooth, M. K., Ahmad R., Salman U. and Swidan W., et al. "Approximation solution of solving linear Volterra Fredholm integro differential equations of the second kind by using Berstein polynomial method." *J Appl Computat Math* 5 (2016): 298.