Journal of Computational and Applied Mathematics 311 (2017) 314–323

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Evolutionary computational intelligence in solving a class of nonlinear Volterra–Fredholm integro-differential equations

Bothayna S.H. Kashkaria [a], Muhammed I. Syam [b,*]

[a] Department of Mathematics, Faculty of Science, AL Faisaliah, King Abdulaziz University, Jeddah, Saudi Arabia

[b] Department of Mathematical Sciences, UAE University, College of Science, P. O. Box 17551, Al-Ain, United Arab Emirates

ARTICLE INFO

Article history:
Received 23 May 2016
Received in revised form 15 June 2016

Keywords:
Volterra–Fredholm integro-differential equations
Computational intelligence
Artificial neural network

ABSTRACT

In this paper, a stochastic computational intelligence technique for solving a class of nonlinear Volterra–Fredholm integro-differential equations with mixed conditions is presented. The strength of feed forward artificial neural networks is used to accurately model the integro-equation. Comparisons with the exact solution and other numerical techniques are presented to show the efficiency of the proposed method. Theoretical and numerical results are presented. Analysis for the presented method is given.

1. Introduction

Many physical phenomena and engineering problems are governed by mathematical models involving differential and integral equations. For example: applications related to geophysical fluid dynamics problems, fluid dynamics, potential theory, astronomy, biology, economics, electrostatics, studies of edge effect in elastic shells, modeling oceanic and atmospheric circulation, chemical reactors theory, convection diffusion processes, and optimal control; among many other areas of applied mathematics and engineering [1–3]. For many years, nonlinear Volterra–Fredholm integro-differential equations have drawn the attention of many researchers and practitioners who devised various techniques for their numerical solutions; among them the works in [4–13,2,3,14–16].

Interests in accurately approximating the solutions of Volterra–Fredholm integro-differential equations have been the focus of attention of many scientists. There are numerous special purpose techniques to adequately deal with such problems; for example the Taylor expansion approach [4–7], the triangular functions [8], the reproducing kernel Hilbert space [9], the Legendre–Galerkin method [10], the second kind Chebyshev wavelet [11], the Adomian decomposition method [12], the maximum principle [13], the sequential approach [14], the hybrid function method [17], the Legendre matrix method [18], the Legendre wavelets method [19], the Tau method [20], the compact finite difference method [21], the Sinc–Galerkin method [22], the Haar function method [23], the Cas wavelet method [24], the differential transform method [25], the sine–cosine wavelet methods [26], the Bessel matrix method [27], the Shannon wavelets approximation [28], the Bernstein polynomial method [29], the Bernstein collocation procedure [30], the product integration and Lagrangian interpolation methods [31], and iterative method [32]. In addition, scientists consider several numerical techniques to solve system of integro-differential equations such as the variational iteration method [15] and the differential transform method [16].

In the recent years, the approximate solution of boundary value problems using the artificial neural network (ANN) has been the focus of attention of many scientists such as [33–36]. Their approaches based on artificial intelligence using neural

* Corresponding author.
E-mail addresses: bkashkari@kau.edu.sa (B.S.H. Kashkaria), m.syam@uaeu.ac.ae (M.I. Syam).

http://dx.doi.org/10.1016/j.cam.2016.07.027
0377-0427/

networks optimized global and local search techniques. However, they are relatively less exploited in this domain in spite of their efficiency.

In this paper, numerical treatment for a class of non-linear Volterra integro-differential problems is presented using the evolutionary computational intelligence. We investigate the numerical solution of a class of high-order nonlinear Volterra–Fredholm integro-differential equations of the form

$$\sum_{j=1}^{r} p_j(x)y^{(j)}(x) = f(x) + \lambda_1 \int_a^x K_1(x,t)g_1(t,y(t))dt + \lambda_2 \int_a^b K_2(x,t)g_2(t,y(t))dt, \quad a < x < b \tag{1.1}$$

subject to the mixed conditions

$$\sum_{j=0}^{r-1} \left(a_{ij}y^{(j)}(a) + b_{ij}y^{(j)}(b) + c_{ij}y^{(j)}(c)\right) = \mu_{ij} \tag{1.2}$$

for $i = 0, 1, \dots, r-1$, $a \le x \le b$, where $f(x)$, $K_1(x,t)$, $K_2(x,t)$, $g_1(t, y(t))$, $g_2(t, y(t))$, and $p_j(x)$, $j = 1, \dots, r$, are smooth functions on $a \le x, t \le b$, i.e., they have derivatives as much as the discussion required and $p_r(x) > 0$ for all $x \in [a, b]$. Also, $\lambda_1, \lambda_2, \mu_{ij}$ for $i, j = 0, \dots, r-1$ are constants and r is a nonnegative integer. Problem (1.1)–(1.2) was studied by Darania and Ivaz [4] using a Taylor expansion approach. The same approach is used by Yalcinbas and Sezer [5,6] when $g_1(t, y(t)) = g_2(t, y(t)) = y(t)$ and Maleknejad and Mahmoudi [7] when $g_1(t, y(t)) = y^p(t)$ and $g_2(t, y(t)) = y(t)$.

In Section 2, the mathematical modeling of the problem is presented. In Section 3, we present the idea of the evolutionary computational intelligence method while in Section 4, analytical results are presented. In Section 5, simulation and numerical results for a number of examples are presented and discussed. Finally, some conclusion and future work are drawn.

2. Mathematical modeling

In this section, we present the description of the designed model along with the fitness function formulation. Following the approach of Raja [36], the approximate solution of problem (1.1)–(1.2) is written as

$$y_n(x) = \sum_{i=1}^{n} \alpha_i f(w_i x + \beta_i) \tag{2.1}$$

where α_i, w_i, and β_i are the weights, n is the number of neurons, and f is the log-sigmoid transfer function which is given by

$$f(x) = \frac{1}{1+e^{-x}}.$$

Simple calculations imply that

$$\begin{aligned} y_n(x) &= \sum_{i=1}^{n} \alpha_i \left(\frac{1}{1+e^{-(w_i x+\beta_i)}}\right), \\ y_n'(x) &= \sum_{i=1}^{n} \alpha_i w_i \left(\frac{e^{-(w_i x+\beta_i)}}{\left(1+e^{-(w_i x+\beta_i)}\right)^2}\right), \\ &\ \vdots \\ y_n^{(r)}(x) &= \sum_{i=1}^{n} \alpha_i \frac{d^r}{dx^r}\left(\frac{1}{1+e^{-(w_i x+\beta_i)}}\right). \end{aligned}$$

Substitute the approximation of the derivatives in Eq. (1.1). Let

$$x_i = a + ih, \quad i = 0, 1, \dots, m$$

where $h = \frac{b-a}{m}$. Then, $\{x_0, x_1, \dots, x_m\}$ is a uniform partition of $[a, b]$. The error ϵ_1 is given by

$$\begin{aligned} \epsilon_1 = \frac{1}{m+1}\sum_{i=0}^{m}\Bigg(&\sum_{j=1}^{r} p_j(x)y_n^{(j)}(x_i) - f(x_i) - \lambda_1 \int_a^{x_i} K_1(x_i,t)g_1(t,y_n(t))dt \\ &- \lambda_2 \int_a^b K_2(x_i,t)g_2(t,y_n(t))dt\Bigg)^2. \end{aligned} \tag{2.2}$$