Solving Volterra's Population Model Using New Second Derivative Multistep Methods

[1]K. Parand *1,2 A.A. Aghaei , A.Jahandideh, J. Tadayon

[1]Department of Computer Science, Shahid Beheshti University, Tehran, Iran

2Department of Cognitive Modeling, Institute for Cognitive and Brain Sciences, Shahid Beheshti University, G.C. Tehran, Iran

3Department of Statistics and Actuarial Science, University of Waterloo, Waterloo, Canada

Abstract: In this study new second derivative multistep methods (denoted SDMM) are used to solve Volterra's model for population growth of a species within a closed system. This model is a nonlinear integro-differential where the integral term represents the effect of toxin. This model is first converted to a nonlinear ordinary differential equation and then the new SDMM, then researchers solve this equation (ODE).The accuracy of method is tested in terms of RES error and compare the obtained results with some well-known results. The numerical results obtained show that the proposed method produces a convergent solution.

Key words: multistep and multi-derivative methods, volterra's population model, integro-differential equation, stiff systems of ODEs

INTRODUCTION

Many problems arising in science and engineering are set in unbounded domain. Spectral methods have been successfully applied in the approximation of ordinary differential equations (ODEs) defined in unbounded domains in recent years

Volterra's model for the population growth of a species within a closed system is given in [8,9,10] as

$$\frac{dp}{d\tilde{t}} = ap - bp^2 - cp\int_0^{\tilde{t}} p(x)dx, \qquad p(0) = p_0, \tag{1}$$

Where $a > 0$ is the birth rate coefficient, $b > 0$ is the crowding coefficient and $c > 0$ is the toxicity coefficient. The coefficient c indicates the essential behavior of the population evolution before its level falls to zero in the long term. p_0 is the initial population and $p = p(\tilde{t})$ denotes the population at time $\tilde{t}$.

This model is a first-order integro-ordinary differential equation where the term $cp\int_0^{\tilde{t}} p(x)dx$ represents the effect of toxin accumulation on the species. We apply scale time and population by introducing the nondimensional variables

$$t = \frac{\tilde{t}c}{b}, u = \frac{pb}{a}$$

to obtain the nondimensional problem:

$$\kappa\frac{du}{dt} = u - u^2 - u\int_0^t u(x)dx, \quad u(0) = u_0 \tag{2}$$

Where u(t) is the scaled population of identical individuals at time t and $\kappa = c/(ab)$ is a prescribed non-dimensional parameter. The only equilibrium solution of eq.2 is the trivial solution $u(t) = 0$ and the following analytical solution[10] shows that $u(t) > 0$ for all t if $u_0 > 0$.

$$u(t) = u_0 \exp\left(\frac{1}{\kappa}\int_0^t \left[1 - u(\tau) - \int_0^\tau u(x)dx\right] d\tau\right)$$

It is shown in[9] that for the case $\kappa \ll 1$, where populations are weakly sensitive to toxins, a rapid rise occurs along the logistic curve that will reach a peak and then is followed by a slow exponential decay. And, for large κ , where populations are strongly sensitive to toxins, the solutions are proportional to $\mathrm{sech}^2(t)$.

In[5] Adomian decomposition method and Sinc- Galerkin method compared for the solution of some mathematical population growth models. This showed that Adomian decomposition method is more efficient and easy to use for the solution of Volterra's population model. In[11], the series solution method and the decomposition method are implemented independently to eq.2 and to a related nonlinear ordinary differential Equation. Furthermore, the Pade approximations are used in the analysis to capture the essential behavior of the populations of identical individuals, also the approximation of u_{max} and exact value of u_{max} for different κ were compared.

The solution of eq.1 has been of considerable concern. Although a closed form solution has been achieved in[8,9], it was formally shown that the closed form solution cannot lead to any insight into the behavior of the population evolution[8]. In the literature several numerical solutions for Volterra's population model have been reported. In[8], the successive approximations method was suggested for the solution of eq.2, but was not implemented. In this case the solution u(t) has a smaller amplitude compared to the amplitude of u(t) for the case $\kappa \ll 1$. In[10], several numerical algorithms namely the Euler method, the modified Euler method, the classical fourth-order Runge-Kutta method and Runge-Kutta-Fehlberg method for the solution of eq.2 are obtained. Moreover, a phase-plane analysis is implemented. In[10], the numerical results are correlated to give insight on the problem and its solution without using perturbation techniques. However, the performance of the traditional numerical techniques is well known in that it provides grid points only, and in addition, it requires a large amounts of calculations. The authors of[6,7] applied spectral method to solve Volterra's population on a semi-infinite interval. This approach is based on a Rational Tau method. They obtained the operational matrices of derivative and product of rational Chebyshev and Legendre functions and then they applied these matrices together with the Tau method to reduce the solution of this problem to the solution of system of algebraic Equations.

On the other hand, in recent years, numerous works have been focusing on the development of more advanced and efficient methods for initial value problems especially for stiff systems. For example, as Enright[2] used second derivative of solution in his algorithm, Cash[1] and Ismail[4] introduced second derivative multistep methods that have good stability properties. These methods are A-stable of high orders. One of these efficient methods that have good stability and accuracy properties is a new class of second derivative multistep methods that is introduced by *et al*.[3]. The main superiority of this new class of methods lead us to apply this new class of methods to solve Volterra's population model after converting it to a system of ODEs.

This study is arranged as follows: in the first section we describe new second derivative multistep methods. In the second section Volterra's population model is considered. This equation is first converted to an equivalent nonlinear ordinary differential equation and then our method can be applied to solve this new equation. In the next section the proposed method is applied to several numerical examples and a comparison is made with existing methods that were reported in the literature to solve similar problems. The numerical results and advantages of the method are discussed in the final section.

MATERIALS AND METHODS

New second derivative multi step methods: Let us consider the stiff initial value problem

$$y'(t) = f(t, y(t)), y(t_0)=y_0 \tag{3}$$

on the finite interval

$I = [t_0, t_N]$ where $y : [t_0, t_N] \to IR^m$

and

$f : [t_0, t_N] \times IR^m \to IR^m$ is continuous.

One of the main directions of search for higher order A-stable multi step methods is the use of higher derivatives of the solutions. By applying the second derivative of solution in algorithm of multistep methods, a new class of methods are introduced. These methods are known as second derivative multi step methods (SDMM).

The general SDMM can be written in the form:

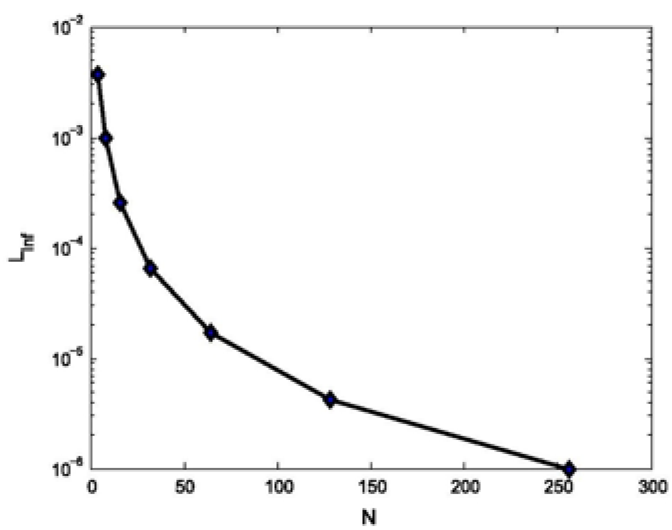
$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h \sum_{j=0}^{k} \beta_j f_{n+j} + h^2 \sum_{j=0}^{k} \gamma_j g_{n+j} \tag{4}$$

Where $\alpha_j, \beta_j, \gamma_j$ are parameters to be determined and $g_{n+j} = f^{(1)}_{n+j}$. Taylor expansion shows that the method equation 4 is of order p if and only

$$\sum_{j=0}^{k} \alpha_j j^q = q \sum_{j=0}^{k} \beta_j j^{q-1} + q(q+1) \sum_{j=0}^{k} \gamma_j j^{q-2} \tag{5}$$

Table 1
Comparison of errors with the Haar wavelet method [46] for Example 1.

2M	Haar wavelet method [46]	Present method
4	3.3E−3	3.7E−3
8	2.7E−3	1.0E−3
16	1.1E−3	2.6E−4
32	3.7E−4	6.6E−5
64	1.1E−4	1.7E−5
128	3.1E−5	4.2E−6



Comparison of exact and approxi mate solu tions for Example

$$+\sum_{i=2}^{2M}\sum_{j=2}^{2M}\frac{1}{\rho 1\rho 2}\left(\sum_{p=\alpha_1}^{\beta_1}\sum_{q=\alpha_2}^{\beta_2}K(x_p,t_q,u_q)-\sum_{p=\alpha_1}^{\beta_1}\sum_{q=\beta_2+1}^{\gamma_2}K(x_p,t_q,u_q)\right.$$
$$\left.-\sum_{p=\beta_1+1}^{\gamma_1}\sum_{q=\alpha_2}^{\beta_2}K(x_p,t_q,u_q)+\sum_{p=\beta_1+1}^{\gamma_1}\sum_{q=\beta_2+1}^{\gamma_2}K(x_p,t_q,u_q)\right)\mathrm{haar}_i(x_r)\mathrm{ihaar}_j(x_r),\quad r=1,2,\ldots,2M.$$

Proceeding as in the previous case, we can obtain the solution of the Volterra integral equation at the collocation points.

Numerical experiments

Example 2. Consider the following nonlinear Fredholm integral equation:

$$u(x)=\sin(\pi x)+\frac{1}{5}\int_0^1\cos(\pi x)\sin(\pi t)(u(t))^3\mathrm{d}t.$$

The exact solution of this problem is u(x) = 2 - x2 • In Table 1 we have compared the maximum absolute errors of the pro- posed method with those from the Haar wavelet method . The ta ble shows that the performa nce of our method is better tha n tha t of the Haar wavelet method . In Fig. 1we have shown the absolute errors for this problem . It is evident from the figure that the maximum absolute error decreases with the increase in number of collocation points. The added advantage of the new method in comparison to the Haar wavelet method is that it does not involve numerical integration. In our case we approximate the integrand with the Haar wavelet basis and perform exact integration of Haar functions.

Example 2. Consider the following nonlinear Fred holm integral equation

$$u(x)=\sin(\pi x)+\frac{1}{5}\int_0^1\cos(\pi x)\sin(\pi t)(u(t))^3\mathrm{d}t.$$

The exact solution of this problem is

$$u(x)=\sin(\pi x)+\frac{20-\sqrt{391}}{3}\cos(\pi x).$$

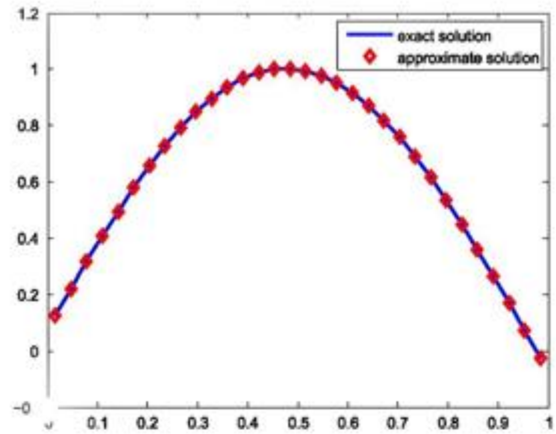


Fig. 2. Comparison of exact and approximate solutions for Example 2.

Table 2
Comparison of errors with the triangular function method [13] for Example 2.

2M	Triangular factorization method [13]	Present method
4	3.9E−2	2.8E−16
8	9.9E−3	2.9E−16
16	2.5E−3	2.8E−16
32	7.6E−4	3.3E−16

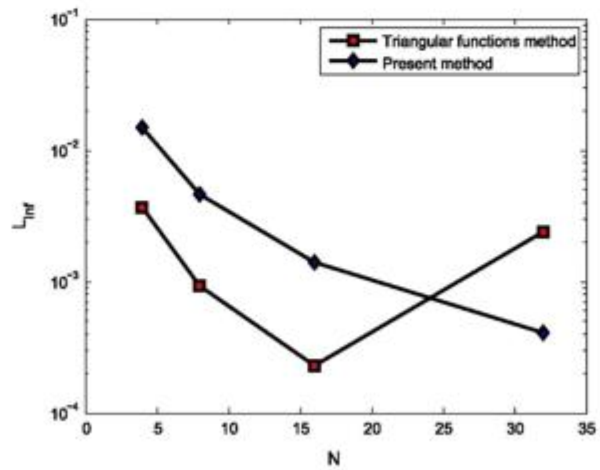


Fig. 3. Comparison of exact and approximate solutions for Example 3.

Comparison of exact and approximate solutions for Example

In Fig. 2, we have shown a comparison of the approximate solution with the exact solution. Table 2 shows the comparison of errors of the present method with those of the triangular function method . It is evident from the table that our method performed far better than the triangular function method for this problem

Example 3. Consider the following nonlinear Volterra integral equation.

$$u(x) = \frac{3}{2} - \frac{1}{2}\exp(-2x) - \int_0^x (u(t)^2 + u(t))dt.$$

The exact solution of this problem is exp(-x). I n Fig. 3, we have compared the maximum absolute errors of the present method with those of the triangular function method .The figure shows that better accuracy is obtained with the present method with the increase of number of collocation points. This shows the stability of our method while the errors in the triangular function method show an oscillatory behavior.

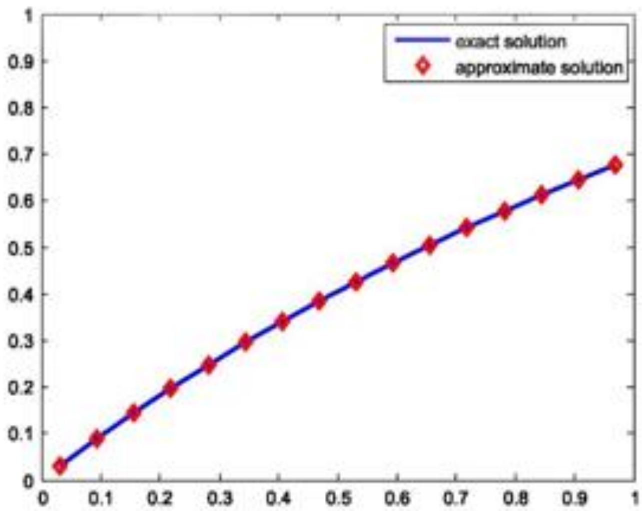


Fig. 4. Comparison of exact and approximate solutions for Example 4.

Example 4. Consider the following nonli near Volterra integral equation

$$u(x) = f(x) + \int_0^x xt^2(u(t))^2 dt,$$

where

$$f(x) = \left(1 + \frac{11}{9}x + \frac{2}{3}x^2 - \frac{1}{3}x^3 + \frac{2}{9}x^4\right)\ln(x+1) - \frac{1}{3}(x+x^4)(\ln(x+1))^2 - \frac{11}{9}x^2 + \frac{5}{18}x^3 - \frac{2}{27}x^4.$$

The exact solution of this problem is u(x) = ln(x + 1).In Fig. 4, we have shown the comparison of the approximate solution with the exact solution.

. Conclusion

Two new generic algorithm are proposed for the numerical solution of nonlinear Fred holm integral equations of the second kind and nonlinear Volterra integral equations of the second kind. A two-dimensional Haar wavelet basis is used for this purpose. The algorithms are established theoretically alongside numerical validations. The new algorithms do not need any linear system solution for evaluation of the wavelet coefficients and are more efficient than conventional Haar wavelet based methods.

Different types of integral equation can be solved numerically by the same method more accurately than previously.

Deep neural network for systemof ordinary differential equations: Vectorized algorithm and simulation

This paper is aimed at applying deep artificial neural networks for solving system of ordinary differential equations. We developed a vectorized algorithm and implemented using python code. We conducted different experiments for selecting better neural architecture. For the learning of the neural network, we utilized the adaptive moment minimization method.

Finally, we compare the method with one of the traditional numerical methods-Runge–Kutta order four.

We have shown that, the artificial neural network could provide better accuracy for smaller numbers of grid points.

Deep neural network (DNN) has obtained great attention for solving engineering problems. System of ordinary differential equations (ODEs) that can model various physical phenomena could utilize the advantages of using the method. Though there are well established traditional numerical methods for solving systems of ODEs, they have their own advantages and disadvantages in-terms of accuracy, stability, convergence, computation time etc. One of the well known method is the fourth order Runge–Kutta method (RK4). It is among the finite difference methods well suited for non-stiff problems.

Artificial neural network (ANN) is an alternative method known to the scientific community since 1940s. The beginning of ANN is often attributed to the research article by McCulloch and Pitts (1943). It was less popular due to the capacity of computational machines. The recent development and progresses in the area is attributed to the exponential improvement in the computing capacity of machines both in storing data and processing speed (Basheer & Hajmeer, 2000).

‘‘An artificial neural network is an information-processing system that has certain performance characteristics in common with biological neural networks’’ (Yadav, Yadav, Kumar, et al., 2015).

Related works

Deep neural networks

The DNN method has contributed a lot to the progress in artificial intelligence specifically in computer vision, image processing, pattern recognition and Cybersecurity (Dixit & Silakari, 2021; Dong, Wang, & Abbas, 2021; Minaee et al., 2021). The performance is due to features are learned rather than hand-crafted, the deep layers are able to capture more variances (Bruna & Dec, 2018). Some of the challenging issues related to DNN are, stability, robustness, provability and adversarial perturbation which are discussed in Haber and Ruthotto (2017), Malladi and Sharapov (2018), Szegedy et al. (2013), Zheng and Hong (2018) and Zheng, Song, Leung, and Goodfellow (2016).

The optimization problems arising form learning the DNN also need special consideration which are presented in Nouiehed and Razaviyayn (2018) and Yun, Sra, and Jadbabaie (2018).

Moreover, the search and selection of an optimal neural architecture is difficult task (Elsken, Metzen, Hutter, et al., 2019). Some widely implemented deep learning architectures — autoencoder, convolutional network, deep belief network and restricted Boltzmann machine were presented in Liu et al. (2017). A broader survey of advance in convolutional neural network can be found in Gu et al. (2018).

Works related to solving differential equations

Nowadays, researchers are applying ANN for solving differential equations. Some of the advantages of using ANN over the traditional numerical methods are: the solutions obtained by ANN are differentiable, and closed analytic form, the method could handle complex differential equations and helps to overcome the repetition of iteration (Chakraverty & Mall, 2017).

Lee and Kang (1990) implemented neural algorithm for solving differential equations. They have used the method for minimization purpose where development of highly parallel algorithms for solving the difference equations required.

Meade and Fernandez (1994) implemented a feedforward neural networks to approximate the solution of linear ODE. They have used the hard limit activation function to construct direct and non-iterative feedforward neural network. The author implemented the method on three layers, input layer, a hidden layer and output layer. Simple first and second order ordinary differential equation were considered for testing the method. Lagaris, Likas, and Fotiadis (1998) used ANN for solving ordinary and partial differential equations. For solving initial and boundary value problems, they used trail solution satisfying the given conditions. Then, network were trained to satisfy the differential equations. The results were compared with well established numerical method—finite element. The authors obtained accurate and differentiable solution in a closed analytic form.

Partial differential equation with initial and boundary condition were solved using neural network (Aarts & Van Der Veer, 2001). The architecture of the network were, multiple input units, single output unit and single hidden layer feedforward with a linear output layer

with no bias. Evolutionary algorithms were implemented for the cost minimization. The authors tested the method on problems from physics and geological process.

For solving ODE using ANN, unsupervised kernel mean square algorithm were used in Sadoghi Yazdi, Pakdaman, and Modaghegh (2011). Trial solution similar to the authors in Lagaris et al. (1998), were implemented to obtain accurate results. Nascimento, Fricke, and Viana (2020) presented the direct implementation of integration of ODE through recurrent neural networks. Berg and Nyström (2018) implemented deep feedforward ANN to approximate solution of partial differential equations in complex geometries.

They solved problems that could not be addressed or difficult by the traditional method. They did comparison between shallow versus deep networks. More recent development and applications of ANN in partial differential equations can be found in Berg and Nyström (2019), Rackauckas et al. (2020), Raissi, Perdikaris, and Karniadakis (2019) and Wang, Huan, and Garikipati (2019).

The application of ANN were also extended to the computation of integral equations. Asady, Hakimzadegan, and Nazarlue (2014), introduced an efficient application of ANN for approximating solution of linear two-dimensional Fredholm integral equation of the second kind. They have found remarkable accuracy and proposed extension to the case of more general integral equations. For the implementation of ANN, clear and reproducible algorithm with implementation needs great attention. An efficient neural network architecture has to be investigated corresponding to systems of ODE. We need to look at the effects of numbers of hidden layers in the architect as well as the numbers of neurons in the layers on accuracy, speed and performance of the model in general. We need to investigate and propose the best selection of activation function.

Addressing the issue of minimization method for the cost function is also crucial.

In this paper, we present a vectorized algorithm for solving systems of ordinary differential equation using DNN. We implement the algorithm in python and perform experimental simulations to look at the effects of different neural architecture on the performance of the model. Moreover, we observe the advantage of using the ANN over the traditional methods. Specifically, we consider the fourth order Runge–Kutta finite difference method.

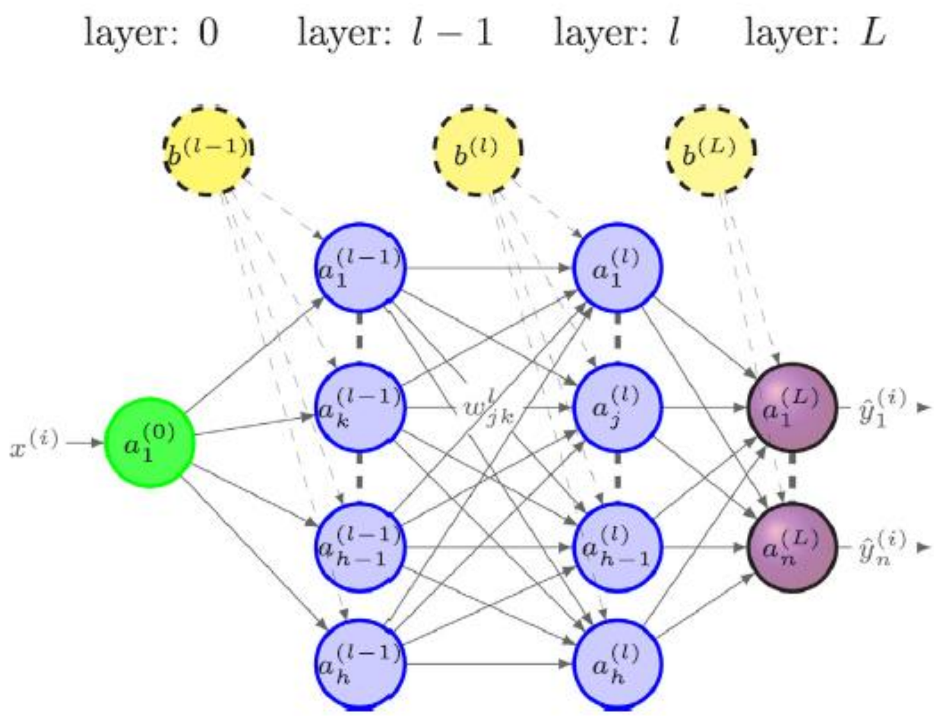


Fig. 1. The schematic diagram of deep ANN.

Systems of ordinary differential equation

The general form of a system of n ODEs is given by,

$$
\begin{aligned}
\frac{dy_1}{dt} &= f_1(t, y_1, y_2, \ldots, y_n) \\
\frac{dy_2}{dt} &= f_2(t, y_1, y_2, \ldots, y_n) \\
\vdots \quad & \quad \vdots \\
\frac{dy_n}{dt} &= f_n(t, y_1, y_2, \ldots, y_n),
\end{aligned}
\tag{1}
$$

defined on $a < t < b$ with given initial values, $y_1(a) = a_1, \ldots, y_n(a) = a_n$. The initial value problem (1) can be written in compact way as follows;

$$
\frac{d\mathbf{y}}{dt} = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(0) = \mathbf{y}_0, \tag{2}
$$

where $\mathbf{y} = [y_1, y_2, \ldots, y_n]^T$ is the unknown having dimension of $n \times 1$, and

$$
\mathbf{f}(t, \mathbf{y}) = \begin{bmatrix} f_1(t, y_1, y_2, \ldots, y_n) \\ f_2(t, y_1, y_2, \ldots, y_n) \\ \vdots \\ f_n(t, y_1, y_2, \ldots, y_n) \end{bmatrix}
$$

is given vector valued function having dimension of $n \times 1$. The uniqueness and existence of the solutions to the initial value problem is well established theory. To obtain more understanding on existence and uniqueness of solution to the initial value problem (2), one may refer Coddington and Levinson (1955).

Implementation and comparison

In this section we implement the algorithm for solving a known nonlinear systems of differential equations. First, we perform simulation for selecting appropriate number of layers and neurons in the layer. Then, we compare with the analytical solution and with a numerical solution obtained using the traditional methods. For this purpose, we consider the following problem found in Lagaris et al. (1998),

$$\frac{dy_1}{dt} = \cos(t) + y_1^2 + y_2 - (1 + t^2 + \sin^2(t)),$$
$$\frac{dy_2}{dt} = 2t - (1 + t^2)\sin(t) + y_1 y_2,$$

with $t \in [0, 1]$ and $y_1(0) = 0$ and $y_2(0) = 1$. The analytic solutions are $y_1 = \sin(t)$ and $y_2 = 1 + t^2$.

minus may not have significant effect). For this experiment, $m = 11$, uniform grid points were sampled from the given interval. The solutions using ANN and the corresponding analytical solutions are indicated in Fig. 4. The numerical quantities are indicated in Table 1. Table 2 indicates the error due to the neural network method.

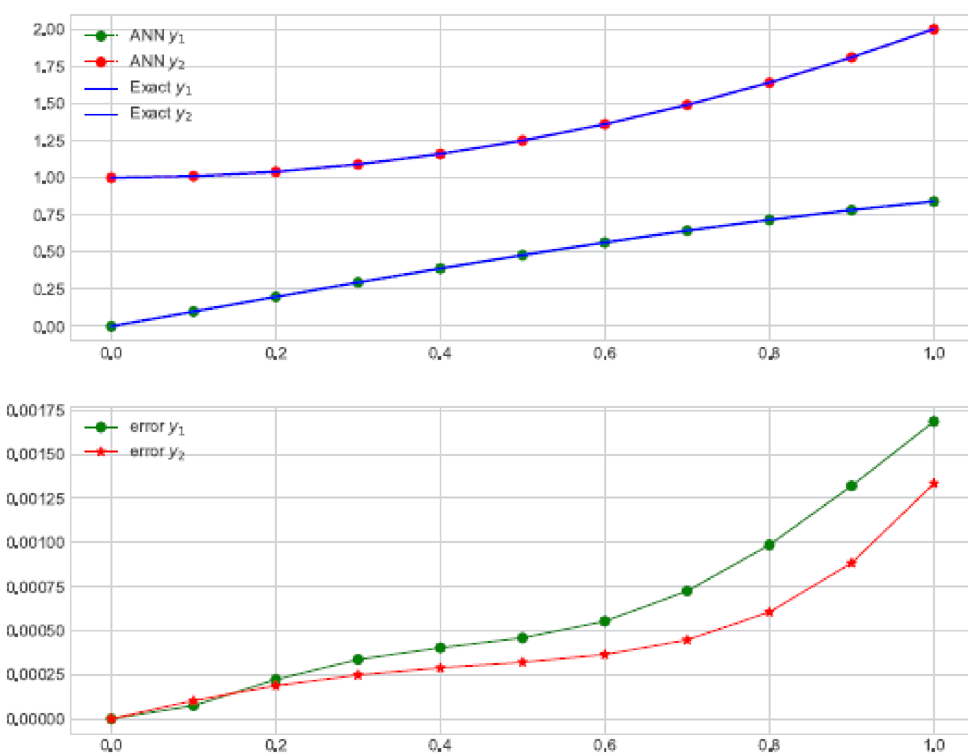


Fig. 4. Comparing the ANN solution of (8), with the exact solution and error plot.

Table 1
ANN and analytical solutions.

t	y_1 ANN	y_1 Analytic	y_2 ANN	y_2 Analytic
0.0	0.000000	0.000000	1.000000	1.00
0.1	0.099759	0.099833	1.009897	1.01
0.2	0.198447	0.198669	1.039812	1.04
0.3	0.295184	0.295520	1.089752	1.09
0.4	0.389015	0.389418	1.159711	1.16
0.5	0.478967	0.479426	1.249679	1.25
0.6	0.564089	0.564642	1.359636	1.36
0.7	0.643493	0.644218	1.489553	1.49
0.8	0.716370	0.717356	1.639394	1.64
0.9	0.782005	0.783327	1.809116	1.81
1.0	0.839784	0.841471	1.998666	2.00

Table 2
ANN Error.

t	error y_1	error y_2
0.0	0.000000	0.000000
0.1	0.000075	0.000103
0.2	0.000223	0.000188
0.3	0.000336	0.000248
0.4	0.000403	0.000289
0.5	0.000459	0.000321
0.6	0.000553	0.000364
0.7	0.000725	0.000447
0.8	0.000987	0.000606
0.9	0.001321	0.000884
1.0	0.001687	0.001334

Conclusions and outlook

In this paper, we presented a vectorized algorithm for solving systems of ODE using DNN. We conducted different experiment using python code and simulated the result using graphs. We have obtained some insight on the nature of the architecture for the model. We have seen that for some specific problems we can obtain a required accuracy even for a single neuron in the hidden layer. More neuron size provides more accuracy, but more iteration for learning the parameters.

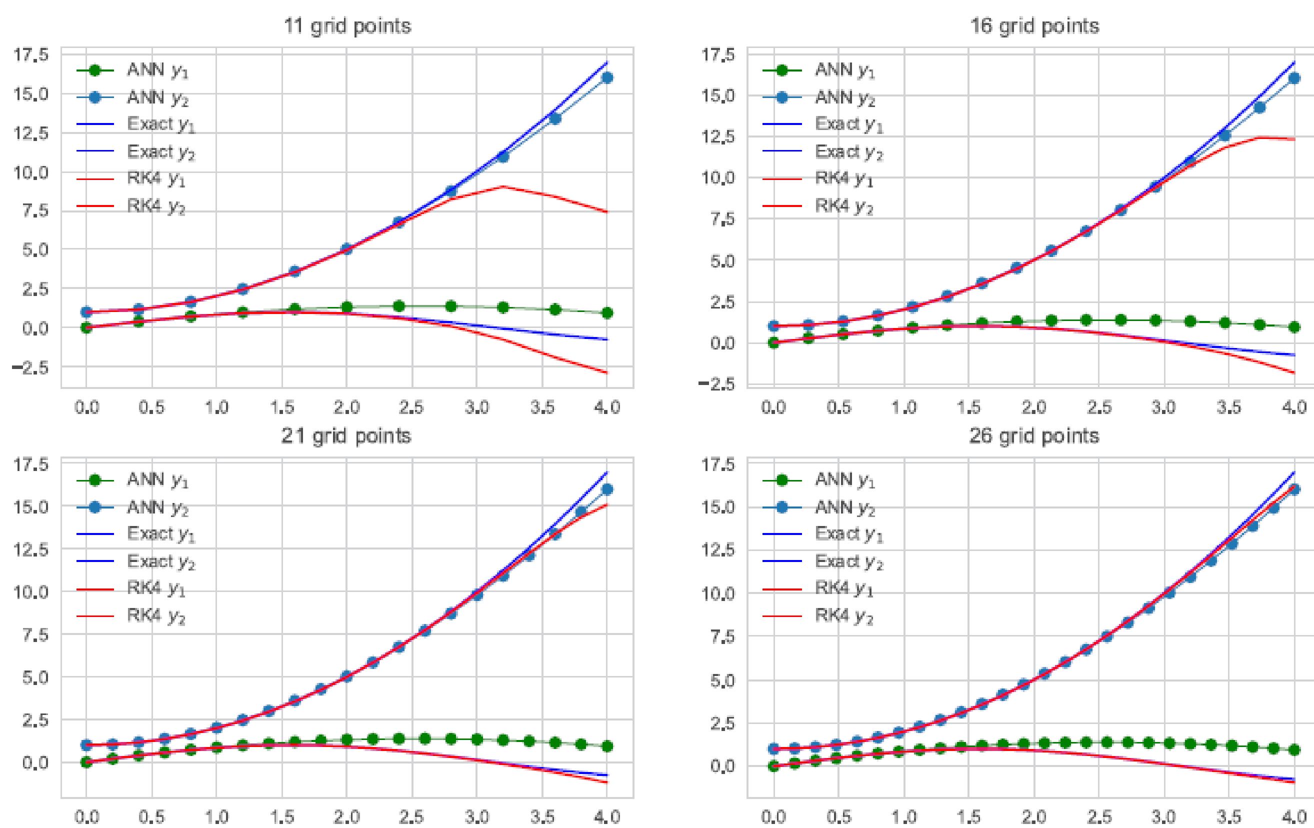


Fig. 5. Comparing the ANN solution of (8), with the exact and RK4 solutions for different grid points.

Numerical Solution of Space-time Fractional two-dimensional Telegraph Equation by Shifted LEgendre Operational Matrices

Fractional differential equations (FDEs) have attracted in the recent years a considerable interest due to their frequent appearance in various fields and their more accurate models of systems under consideration provided by fractional derivatives. For example, fractional derivatives have been used successfully to model frequency dependent damping behavior of many viscoelastic materials. They are also used in modeling of many chemical processed, mathematical biology and many other problems in engineering. The history and a comprehensive treatment of FDEs are provided by Podlubny and a review of some applications of FDEs are given by Mainardi.

The fractional telegraph equation has recently been considered by many authors. Cascaval et al. discussed the time-fractional telegraph equations, dealing with wellposedness and presenting a study involving asymptotic by using the Riemann-Liouville approach. Orsingher and Beghin discussed the time-fractional telegraph equation and telegraph processes with Brownian time, showing that some processes are governed by time-fractional telegraph equations. Chen et al.

also discussed and derived the solution of the time-fractional telegraph equation with three kinds of nonhomogeneous boundary conditions, by the method of separating variables. Orsingher and Zhao considered the space-fractional telegraph equations, obtaining the Fourier transform of its fundamental solution and presenting a symmetric process with discontinuous trajectories, whose transition function satisfies the space-fractional telegraph equation. Momani discussed analytic and approximate solutions of the space- and time-fractional telegraph differential equations by means of the so-called Adomian decomposition method. Camargo et al.

discussed the so-called general space-time fractional telegraph equations by the methods of differential and integral calculus, discussing the solution by means of the Laplace and Fourier transforms in variables t and x, respectively. In this paper, we consider the following space-time fractional two-dimensional telegraph

equation:

$$\frac{\partial^{2\alpha} u(t,x,y)}{\partial t^{2\alpha}} + p\frac{\partial^{\alpha} u(t,x,y)}{\partial t^{\alpha}} + q^2 u(t,x,y) = \frac{\partial^{\beta} u(t,x,y)}{\partial x^{\beta}} + \frac{\partial^{\beta} u(t,x,y)}{\partial y^{\beta}} + f(t,x,y),$$

with initial conditions

$$u(0,x,y)=f_1(x,y), \quad u_t(0,x,y)=f_2(x,y),$$

where p, q are constants and also $1<\beta\le 2,\ 0.5<\alpha\le 1$ and $x, y, t \in [0,1]$.

Material and Method

The Legendre polynomials defined on [−1, 1] are given by the following recurrence relation

$$L_{i+1}(z)=\frac{2i+1}{i+1}zL_i(z)-\frac{i}{i+1}L_{i-1}(z),\ i=1,2,\ldots,$$

where $L_0(z)=0$, $L_1(z)=1$. The transformation $x=\frac{z+1}{2}$ transforms the interval [−1, 1] to [0, 1] and the shifted Legendre polynomials are given by

$$P_i(x)=\sum_{k=0}^{i}(-1)^{i+k}\frac{(i+k)!}{(i-k)!}\frac{x^k}{(k!)^2},\ i=0,1,2,3,\ldots,$$

where $p_i(0)=(-1)^i$, $p_i(1)=1$.

In this paper, we approximate three-variable functions using the shifted Legendre polynomials. Then, we introduce the operational matrices of the Caputo fractional derivative in one-dimensional and two-dimensional cases and also the operational matrix of the Riemann- Liouville fractional integral using the shifted Legendre polynomials. By applying these concepts on the two-dimensional space-time fractional telegraph equation, the problem will be reduced to solve a system of algebraic equations that can be easily solved.

Application of Legendre operational matrix to solution of two dimensional non-linear Volterra integro-differential equation

This article applies the operational matrix to find the numerical solution of two-dimensional nonlinear Volterra integrodifferential equation (2DNVIDE).

Form this prospect, two-dimensional shifted Legendre functions (2DSLFs) have been presented for integration, product, and differentiation. This method converts 2DNVIDE to an algebraic system of equations, so the numerical solution of 2DNVIDE is computable. The effectiveness and accuracy of the method were examined with some examples as well. The results and comparison with other methods have shown remarkable performance.

The theory and applications of partial integral and integro-differential

equations are used in many branches of scientific research from engineering, mechanics, and physics to economics, etc [1–7]. Recently, some numerical techniques and approaches have been used to evaluate the approximate solution of the nonlinear phenomena, whereas using the operational matrix could be the essential part of these numerical solutions [11,12,19]. In particular, Legendre polynomials as powerful

tools have been employed to convert some nonlinear equations [13, 27].

There have been various numerical solutions for PDEs that benefit from the operational matrix to approximate the exact solutions in many fields of science; for example, in theory of anomalous relaxation processes in the vicinity of singular, IDE has been solved successfully by using operational matrix [14].

Some numerical methods, such as the homotopy perturbation method [24] and variational iteration method [26] have been represented to obtain the approximate solution of the mixed Volterra-Fredholm. The TFs methods have been used to approximate the numerical solution of Fredholm and Volterra integral equations [15]. Maleknejad in [18] and Khajehnasiri [17] have applied a TFs operational matrix to approximate the solution of nonlinear kind of Volterra-Fredholm integral equations and 2D nonlinear Volterra-Fredholm integro differential equations, respectively. The two-dimensional Block-Pulse functions (2D-BPFs) have been applied by Maleknejad and Mahdiani to find the solution of nonlinear mixed Volterra-Fredholm integral equations [16]. Imran Aziz has extended the Haar wavelet method to evaluate the numerical solution of 2D nonlinear integral equations [20], a class of 2D nonlinear Volterra integral equations in [25] have solved by Legendre polynomials, in [28] a 2DTFs have applied to find the nonlinear class of mixed Volterra- Fredholm integral equations. Aghazadeh in

[10] has ameliorated the Block-pulse operational matrix to evaluate the approximate solution of the nonlinear 2D Volterra integro-differential equation.

Although the 2DNVIDEs have exciting applications in Physics, Mechanics, and applied sciences, there have been a few simple numerical methods for solving these equations with high accuracy. For this reason, in this paper, we formulate a Legendre operational matrix for 2D nonlinear Volterra integro-differential equations with given supplementary conditions. The most important part of our concept was extending this polynomial to convert the nonlinear kernels of FDVIDEs, which is unsolvable in a real evaluation. The two-dimensional nonlinear Volterra integro-differential equation in the general form of can be written as:

$$\begin{aligned}\frac{\partial^n u(x,t)}{\partial x^n} &+ \frac{\partial^m u(x,t)}{\partial t^m} + \frac{\partial^{n+m} u(x,t)}{\partial x^n \partial t^m} + u(x,t) = f(x,t) \\ &+ \int_0^t \int_0^x G(x,t,y,z) R(y,z,U(y,z))) dy dz, \quad (x,t) \in [0,l] \times [0,T],\end{aligned}$$

with given supplementary initial conditions, where $u(x, t)$ is an unknown function in (и = [0, l] × [0, T]), the function $R(y, z, U(y, z))$ is given continuous in и × ($-\infty$,+∞), nonlinear in U, and the functions $f(x, t)$ and $G(x, t, y, z, u)$ are given smooth functions. Here we assume that R satisfy the following conditions:

$$| R(y,z,U(y,z)) - R(y,z,U^{'}(y,z)) | \leq \lambda | U(y,z) - U^{'}(y,z) |,$$

In the next section, we will define some basic definitions and properties of 2D shifted Legendre functions. In section three we will introduce the operational matrix for integration, product properties, as well as the operational matrix of differentiation. After which and in section 4, we are solving and obtaining the solution of two-dimensional nonlinear Volterra integro-differential equation by using 2D shifted Legendre functions. In section 5, we estimate the norm of error for the approximation of two variables smooth function on a specific domain и. Finally, we apply this the proposed method for some examples of 2DNVIDEs.

2D shifted Legendre functions (Basic definitions and properties)

where λ is positive constants, and one can prove the uniqueness and existence of the solution to Eq. (1.1) in the same In the progress of this section, we define and represent some basic definition and properties of the two-dimensional shifted Legendre functions, which are used further in the following section.

2.1. Definition and approximation of the function. The 2D shifted Legendre functions on И are defined as

$$\Psi_{m,n}(x,t) = L_m(\frac{2}{l}x - 1)L_n(\frac{2}{T}t - 1), \quad m, n = 0, 1, 2, ...,$$

where *Lm* and *Ln* are f order *m* and *n* and they are the well-known Legendre functions, which are defined on the interval [−1, 1] and they can be obtained with the following formula

$$\begin{aligned} L_0(x) &= 1, \\ L_1(x) &= x, \\ L_{m+1}(x) &= \frac{2m+1}{m+1}xL_m(x) - \frac{m}{m+1}L_{m-1}(x), \quad m = 1, 2, 3, ..., -1 \leq x \leq 1. \end{aligned}$$

All of the 2D shifted Legendre functions pairs are orthogonal such as:

$$\int_0^T \int_0^l \Psi_{i,j}(x,t)\Psi_{m,n}(x,t)dxdt = \begin{cases} \frac{lT}{(2m+1)(2n+1)}, & i = m \quad and \quad j = n, \\ 0, & otherwise. \end{cases}$$

Let *X* = *L*2(И), definition of the inner product in this space is as

$$\langle u(x,t), w(x,t)\rangle = \int_0^T \int_0^l u(x,t)w(x,t)dxdt,$$

where the norm is defined as:

$$\|u(x,t)\|_2 = \langle u(x,t), u(x,t)\rangle^{\frac{1}{2}} = \left(\int_0^T \int_0^l | u(x,t) |^2 dxdt\right)^{\frac{1}{2}}.$$

Suppose that

$$\Psi_{00}(x,t), \Psi_{01}(x,t), \cdots, \Psi_{0N}(x,t), \cdots, \Psi_{M0}(x,t), \Psi_{M1}(x,t), \cdots, \Psi_{MN}(x,t) \subset X \tag{2.5}$$

are the components of the 2D shifted Legendre functions and

$X_{M,N} = span\{\Psi_{00}(x,t), \Psi_{01}(x,t), \cdots, \Psi_{0N}(x,t), \cdots, \Psi_{M0}(x,t), \Psi_{M1}(x,t), \cdots, \Psi_{MN}(x,t)\}$

and $u(x,t)$ represent an arbitrary function in X. And $X_{M,N}$ be a finite dimensional vector space, so u is unique and it has a best approximation $u_{M,N} \in X_{M,N}$ [8], such that

$$\forall w \in X_{M,N}, \quad \|u - u_{M,N}\| \leq \|u - w\|_2. \tag{2.6}$$

In addition, since $u_{M,N} \in X_{M,N}$, unique coefficients $u_{00}, u_{01}, \cdots, u_{MN}$ are exist as follows:

$$u(x,t) \simeq u_{M,N}(x,t) = \sum_{i=0}^{M}\sum_{j=0}^{N} u_{ij}\Psi_{ij}(x,t) = U^T\Psi(x,t) = \Psi^T(x,t)U,$$

A new class of operational matrices method for solving fractional neutral pantograph differential equations

This paper uses new fractional integration operational matrices to solve a class of fractional neutral pantograph delay differential equations. A fractional-order function space is constructed where the exact solution lies in, and a set of orthogonal bases are given. Using them, we reduce the fractional delay differential equation to algebraic equations and get the approximate solution. Finally, we give the Legendre operational matrix of fractional integration to solve the equation as an example and show the efficiency of the method.

Fractional calculus is a generalization of calculus to an arbitrary order. In recent years, it is extensively applied to various fields such as viscous elastic mechanics, power fractal networks, electronic circuits [1–3].

Due to the memory and non-local characters of fractional derivative, many scholars use fractional differential equations to simulate complex phenomenon in order to make it closer to the real problem. On the other hand, the operational matrices of fractional calculus, as linear transformations, have been widely concerned.

Some operational matrices are obtained by approximating the integral of orthogonal polynomials. Some operational matrices are obtained indirectly by means of other operationalmatrices.

With the help of these matrices and orthogonal polynomials, we can reduce a fractional differential or integral equation to algebraic equations, and get the approximate solution.Much research in this field has emerged, such as Legendre operational matrices, Chebyshev operational matrices, block pulse operational matrices. Fractional delay differential equations arise in many applications,such as automatic control, long transmission lines, economy and biology .

The fractional pantograph equation, as a kind of fractional delay differential equations, plays an important role in explaining various phenomena .Therefore, it has attracted a great deal of attentions.

The authors studied the existence of solutions of nonlinear fractional pantograph equations when the order of the derivative is smaller than 1. Yang and Huang studied spectral-collocation methods for fractional pantograph delay-integro-differential equations whereas Rahimkhani, Ordokhani and Babolian gave the numerical solution of fractional pantograph differential equations by using the generalized fractional-order Bernoulli wavelet. However, only few papers are devoted to the approximate solution of

fractional neutral pantograph differential equations.

In this paper, we consider the fractional neutral pantograph differential equation

$$D_C^{\alpha}y(x) = a(x)y(px) + b(x)D_C^{\gamma}y(px) + d(x)y(x) + g(x),$$

subject to the initial condition

$$y(0) = 0,$$

where $0 < p < 1$, $0 < \gamma \leq \alpha \leq 1$, $x \in [0, 1]$. $y(x)$ is an unknown function; $a, b, d, g \in C[0, 1]$ are the known functions, the fractional derivatives are in the sense of Caputo. For simplicity of the proof, we set $y(0) = 0$, otherwise we let $v(x) = y(x) - y(0)$ and solve the fractional differential equation with the unknown function $v(x)$ and $v(0) = 0$. This paper is devoted to obtaining a class of operational matrices based on different type of fractional orthogonal polynomials and solving the fractional neutral pantograph delay differential equations by the obtained operationalmatrices. As never before, we construct suitable fractional orthogonal polynomials and get the better operational matrices based on the order of the fractional differential or integral equation. Thus we can find approximate solutions with high accuracy. We organize this paper as follows: In Sect. we give some basic definitions and useful lemmas. In Sect., we introduce operational matrices of fractional integration based on fractional orthogonal basis functions. In Sect., we give the numerical algorithms. In Sect., a fractional Legendre operational matrix is given. In Sect., two numerical examples are given to illustrate the applicability and accuracy of the proposed method.

Preliminaries

In this section, some preliminary results about the fractional integral and the Caputo fractional differential operator are recalled [11]. Then we give the appropriate space where we find the approximate solution of fractional differential equations. Throughout the paper, we assume $\alpha > 0$.

Definition 2.1 The Riemann–Liouville (R–L) fractional integral operator $J\alpha$ is given by

$$J_0^\alpha u(x) = \frac{1}{\Gamma(\alpha)} \int_0^x (x-s)^{\alpha-1} u(s)\,\mathrm{d}s,$$

where $\Gamma(\alpha) = \int_0^\infty x^{\alpha-1} e^{-x}\,\mathrm{d}x$.

Lemma 2.1 *For any* $u(x) \in L^2[0,1]$, $J_0^\alpha u(x) \in L^2[0,1]$ *holds.*

Proof

$$\begin{aligned}
&\int_0^1 \left(J_0^\alpha u(x)\right)^2 \mathrm{d}x \\
&\quad = \frac{1}{\Gamma^2(\alpha)} \int_0^1 \left(\int_0^x (x-t)^{\alpha-1} u(t)\,\mathrm{d}t\right)^2 \mathrm{d}x
\end{aligned}$$

References

1. Cash, J.R., 1981, Second Derivative Extended Backward Differentiation Formulas for the Numerical Integration of Stiff Systems, SIAM J. Numerical Anal., 18: 21-36.

2. Enright, W.H., 1974, Second Derivative Multistep methods for Stiff Ordinary Differential Equations, SIAM J. Numerical Anal., 11: 321-331.

3. Hojjati, G., M.Y. Rahimi Ardabili and S.M. Hosseini, 2006. New Second Derivative Multistep Methods for Stiff Systems, Applied Math. Modeling, 30: 466-476.

4. Ismail, G. and I. Ibrahim, 1999. New Efficient Second Derivative Multistep Methods for Stiff Systems. Applied Math. Modeling, 23: 279-288.

5. Al-Khaled, K., 2005. Numerical Approximations for Population Growth Models. Applied Math. and Comput., 160: 865-873.

6. Parand, K. and M. Razzaghi, 2004. Rational Chebyshev Tau Method for Solving Volterra's Population Model. Applied Math. and Comput., 49: 893-900.

7. Parand, K. and M. Razzaghi, 2004. Rational Legendre Approximation for Solving Some Physical Problems on Semi-Infinite Intervals. Physica Scripta, 69: 353-357.

8. Scudo, F.M. and Vito Volterra, 1971. Theoretical Ecology. Theor. Popul. Bio., 2: 1-23.

9. Small, R.D., 1989. Population Growth in a Closed System. Mathematical Modeling: Classroom Notes in Applied Math., SIAM, Philadelphia, PA.

10. Tebeest, K.G., 1997. Numerical and Analytical Solutions of Volterra's Population Model. SIAM Rev. 39: 484-493.

11. Wazwaz, A.M., 1999. Analytical Approximations

and Pade Approximants for Volterra's Population

Model. Applied Math. and Comput.,

[161 L.A. Diaz, M.T. Martin , V. Vampa, Daubechies wavelet beam and plate finite elements, Finite Elem. Anal. Des. 45 (2009) 200-209. [171 Y. Liu, Y. Liu, Z. Cen, Daubech ies wavelet meshless method for 2-D elastic problems, Tsinghua Sci. Technol. 13(2008) 605-608.

[18I W. Dahmen, A Kurdila, P. Oswald, Multisca le Wavelet Methods for Partial Differential Equations, Academ ic Press, 1997.

[191 H. Hashish, S.H. Behiry, N.A. El-Shamy, Numerica l integration using wavelets, Appl. Math. Comput. 211 (2009) 480-487.

[201 Siraj-ul-lslam, I.Aziz, F. Haq, A comparative study of numerical integration based on Haar wavelets and hybrid functions, Comput. Math. Appl. 59 (6) (2010) 2026-2036.

[21I I.Aziz, Siraj-ul-lslam, W. Khan, Quadrature rules for numerical integration based on Haar wavelets and hybrid functions, Comput. Math. Appl. 61 (9)

(2011) 2770-2781.

[221 K. Maleknejad, T. Lotfi, K. Mahdiani, Numerical solution of first kind Fredholm integral equations with wavelets-Galerkin method and wavelets precondi tion, Appl. Math. Comput. 186 (2007) 794-800.

[231 H. Saeedi, M.M. Moghadam, N. Mollahasani, G.N. Chuev, A CAS wavelet method for solving nonlinear Fredholm integro-differential equations of fractional order, Commun. Nonlinear Sci. Numer. Simul. 16 (2011) 1154-1163.

[241 M. Dehghan, M. Lakestan i, Numerical solution of nonlinear system of second-order boundary value problems using cubic 8-spline scaling functions, lnt.J. Comput. Math. 85 (2008) 1455-1461.

[25I Siraj-ul- lslam, B. Sarler. l. Aziz, F. Haq , Haar wavelet collocation method for th e numerical solution of boundary layer fluid flow problems, lnt.J.Therm.

Sci. 50 (2011) 686-697.

[26I M. Mehra, N.K .-R. Kevlahan, An adaptive wavelet collocation method for the solution of partial differential equations on the sphere, j. Compu t. Phys.

227 (2008) 5610-5632.

[271 V. Comincioli, G. Naldi, T.Scapolla, A wavelet-based method for numerica l solution of nonl inear evolution equations, Appl. Numer. Math. 33 (2000) 291-297.

[281 H.jafari, S.A. Yousefi , M.A. Fi roozjaee, S. Momani, C.M. Khalique, Application of Legendre wavelets forsolving fractional differential equations, Comput.

Math. Appl. 62 (2011) 1038-1045.

[291 j.L. Wu, A wavelet operational method for solving fractional partia l differenti al equation s numerically, Appl. Math. Comput.214 (2009) 31-40.

[301 X. Zhu, G. Lei, G. Pan.On application of fast and adaptive Battle-Lemarie wavelets to modelling of mu ltiple lossy transmission lines, J. Comput. Phys.

132 (1997) 299-311.

[31I J. Biazar, H. Ebrahimi, Chebyshev wavelets approach for nonlinear systems of Volterra integral equations, Comput. Math. Appl. 63 (2012) 608-616.

[321 E. Babolian, F. Fattahzdeh, Numeri cal solution of differential equations by using Chebyshev wavelet operational matrix of integration, Appl. Math.

Comput. 188 (2007) 417-426.

[331 K. Maleknejad, T. Lotfi, Y. Rostami, Numerica l computational method in solving Fredholm integral equations of the second kind by using Coifman wavelet, Appl. Math. Comput. 186 (2007) 212-218.

[341 E. Banifatem i, M. Razzaghi, S. Yousefi, Two-dimensiona l Legendre wavelets method for the mixed Volterra-Fredholm integral equations, j. Vib. Control 13(2007)   1667-1675.

[35I X. Shang, D. Han, Numerica l solution of Fredholm integra l equations of the first kind by using linear Legendre mu lti-wavelets, Appl. Math. Comput.

191 (2007) 440-444.

[361 o.Lepik, Numer ical solution of evolution equations by the Haa r wavelet method, Appl. Math. Comput. 185 (2007) 695-704.

[371 C.F. Chen, C.H. Hsiao, Haar wavelet method for solving lumped and distributed-paramete r systems, IEE Proc., Control Theory Appl. 144 (1997) 87-94. [381 C.H.Hsiao, Haar wavelet approach to linear stiff systems, Math . Comput. Simul. 64 (2004) 561-567.

[391 C.H. Hsiao, W.J. Wang, Haar wavelet approach to nonlinear stiff systems, Math. Comput. Simul. 57 (2001) 347-353. [401 o. Lepik, Haar wavelet method for nonlinear integro-differentia l equations, Appl. Math . Comput. 176 (2006) 324-333.

[41I K. Maleknejad, F. Mirzaee, Using rationalized Haar wavelet for solving linear integral equations, Appl. Math. Comput. 160 (2005) 579-587.

[421 K. Maleknejad, R. Mollapourasl, M. Alizadeh, Numerical solution of Volterra type integral equation of the first kind with wavelet basis, Appl. Math.

Comput. 194 (2007) 400-405.

[431 Y. Ordokhani, Solution of nonlinear Volterra -Fredholm-Hammerstein integral equations via rationalized Haar fun ctions, Appl. Math. Comput. 180 (2006) 436-443.

[441 o. Lepik, Application of the Haar wavelet tra nsform to solving integral and differen tial equations, Proc. Estonian Acad. Sci.Phys. Math. 56 (1) (2007)

28-46.

[451 E. Babolian, S. Bazm, P. Lima, Numerical solution of nonlinear two-dimensional integral equations using rationalized Haar functions, Commun.

Nonlinear Sci. Numer. Simul. 16 (2011) 1164-1175.

[461 o.Lepik, E. Tamme, Solution of nonlin ear Fredholm integral equati ons via t he Haar wavelet meth od, Proc. Estonian Acad. Sci. Phys. Math. 56 (2007) 17-27.

[471 j. Gao, Y.-L. Jiang, Trigonometric Hermite wavelet approxim ation for the integral equations of second kind with weakly singular kernel, J. Comput.

Appl. Math. 2 15 (2008) 242-259.

[481 S. Yousefi, A. Banifatemi, N umerical solution of Fred holm integral equations by using CAS wavelets, Appl. Math. Comput. 183 (2006) 458-463. [491 Y. Mahmoudi, Wavelet Galerkin method for numerical solution of nonlinear integral equation, Appl. Math. Comput. 167 (2005) 1119-1129.