Advanced Mathematical Models & Applications
Vol.10, No.2, 2025, pp.409-415
https://doi.org/10.62476/amma.102409

AN ARTIFICIAL NEURAL NETWORK SOLUTION TO THE SPACE-TIME FRACTIONAL PARTIAL DIFFERENTIAL-DIFFERENCE TODA LATTICE EQUATION

Refet Polat[1*], **Burhan Pektas**[2]

[1]Yasar University, Department of Mathematics, Izmir, Turkiye
[2]Uskudar University, Department of Computer Engineering, Istanbul, Turkiye

Abstract. In this paper, the solution of the fractional partial differential-difference Toda Lattice Equation by Artificial Neural Networks is examined. According to the method, we approximate the unknown values $u_n = u(., t_n)$ of the desired function by the artificial neural networks. As an application we demonstrate the capabilities of this method for identification of various values of order of fractional derivative α. Thereafter, the artificial neural networks algorithm is used in order to identify the unknown values u_n. Comparing the results with the finite difference solution, the algorithm can identify the function $u(x, t)$ better than the other method.

Keywords: The Artificial Neural Network, the wave equation, a source identification.
AMS Subject Classification: 35R30, 47A52, 35L20.
[*]**Corresponding author:** Burhan Pektas, Uskudar University, Uskudar, Istanbul, Turkiye, e-mail: *burhan.pektas@uskudar.edu.tr*
Received: 17 March 2025; Revised: 25 April 2025; Accepted: 5 May 2025; Published: 4 August 2025.

1 Introduction

In this paper, we shall consider the space-time fractional (2+1)-dimensional Toda lattice equation described in equation (1) and (2) below. The (2+1)-dimensional Toda lattice hierarchy represents an important extension of the integrable systems landscape. It is closely related to the Kadomtsev–Petviashvili (KP) hierarchy, a fundamental class of nonlinear integrable partial differential equations originally formulated to describe shallow water wave dynamics. The KP hierarchy encompasses an infinite set of coupled equations with soliton solutions, possessing Lax pairs and an infinite number of conserved quantities. The Toda lattice, particularly in its differential-difference form, inherits many of these features and serves as a bridge between continuous and discrete models. Extending this hierarchy into fractional-order and higher-dimensional frameworks has attracted attention due to its ability to model memory effects and spatial complexity in real-world systems. The importance of Toda lattice is, together with the Korteweg–de Vires equation, one of the most classical and significant completely integrable systems.

Several methods have been developed to reveal its philosophical mathematical structure (Toda, 1989). The (2+1)-dimensional Toda lattice hierarchy has been proposed as an extension of the KP hierarchy. This comprises the (2+1)-dimensional Toda lattice equation as the modest nontrivial differential-difference equation. The Toda lattice equation and the sine-Gordon equation are derived by imposing suitable reductions on the (2+1)-dimensional Toda lattice equation (Kajiwara & Satsuma, 1991). These type of equations, usually, describe the evolution of certain phenomena over the course of time (Mohan & Deekshitulu 2012).

This paper studies the space-time fractional differential-difference Toda lattice equation

$$\frac{\partial^{2\alpha}u^n}{\partial x^\alpha \partial t^\alpha} = (1 + \frac{\partial^\alpha u^n}{\partial t^\alpha})(u^{n-1} - 2u^n + u^{n+1}), \quad (x,t) \in I \times (0,T] \tag{1}$$

from the initial and homogeneous Dirichlet boundary data

$$\begin{cases} u(x,0) = \phi(x), \ x \in I, \\ u(0,t) = u(x^*,t) = 0, \ t \in (0,T], \end{cases} \tag{2}$$

where the mixed derivative $\frac{\partial^{2\alpha}u^n}{\partial x^\alpha \partial t^\alpha}$ denotes the space-time derivative with fractional order 2α of the function $u = u(x,t)$ at $t = t_n$. The derivative $\frac{\partial^\alpha u^n}{\partial t^\alpha}$ also denotes time derivative with fractional order $\alpha \in (0,1)$. We consider the most frequently used the Riemann–Liouville and the Caputo derivative for fractional derivatives in (1). Riemann-Liouville fractional derivative with fractional order α of the function $u = u(x,t)$ is defined by (Cui, 2009; Podlubny, 1999), i.e.,

$$\left[\frac{\partial^\alpha u(x,t)}{\partial t^\alpha}\right]_{RL} = \frac{1}{\Gamma(1-\alpha)} \frac{\partial}{\partial t} \int_0^t \frac{u(x,\tau)}{(t-\tau)^\alpha} d\tau, \quad t > 0. \tag{3}$$

where $\Gamma(x)$ is the Euler's Gamma Function. Another definition of fractional derivative is Caputo derivative. Caputo fractional derivative with fractional order α of the function $u = u(x,t)$ is defined by (Cui, 2009), (Podlubny, 1999) as follows:

$$\left[\frac{\partial^\alpha u(x,t)}{\partial t^\alpha}\right]_{C} = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{1}{(t-\tau)^\alpha} \frac{\partial u(x,\tau)}{\partial t} d\tau, \quad t > 0. \tag{4}$$

From (3) and (4), it is clear that definitions of Riemann–Liouville derivative and Caputo derivative are not equivalent. But, there is a fact that, almost all the numerical methods for the Riemann–Liouville derivative can be theoretically extended to the Caputo derivative if the function $u(x,t)$ satisfies suitable smooth conditions. Following equality shows the relation between the Riemann–Liouville and Caputo derivatives for $0 < \alpha < 1$:

$$\left[\frac{\partial^\alpha u(x,t)}{\partial t^\alpha}\right]_{RL} = \left[\frac{\partial^\alpha u(x,t)}{\partial t^\alpha}\right]_{C} + \frac{t^{-\alpha}u(x,0)}{\Gamma(1-\alpha)}, \quad t > 0.$$

Hence, a natural way to discretize the Caputo derivative in the equation (1) is to use the Grünwald–Letnikov approximation (Li & Zeng, 2015).

The paper is organized as follows. The material and methods are given Section 3. The Section 4 gives the feed-forward neural networks for solving Toda-Lattice equation. The conclusion is given in the last Section 5.

2 Two-dimensional Toda chains

Two-dimensional Toda chains, first introduced in the context of integrable systems, model the interaction of particles on a lattice with exponential nearest-neighbor interactions. In fractional settings, these models account for long-range temporal memory and have been applied to study non-Markovian dynamics. Recent works such as (Tarasov & Zaslavsky, 2007; Ferrás et al., 2019) explore numerical and analytical methods for solving fractional Toda-like models. These include modified Adomian decomposition, fractional variational iteration, and spectral methods. However, their computational scalability is limited compared to learning-based approaches. Our work leverages the approximation power of neural networks to handle the complexity of these systems efficiently.

3 Material and Methods

The numerical solutions of the considered problem (1) can be obtained from the finite difference approximation as well as the feed forward artificial neural networks. In this section the both of these methods are introduced.

3.1 Discretization of the problem and Finite difference solution

One method of the solutions of fractional equations based on numerical methods and solutions are determined by implementing the numerical methods on original (physical) domain. These methods are adapted for fractional integrals (Riemann-Liouville integrals etc.) and derivatives (Caputo derivatives and the Riesz Derivatives etc.) based on polynomial interpolation, Gauss interpolation or linear multistep methods.

For the numerical solution to the considered problem above we construct a uniform grid of mesh points t_j with $t_j = jh_t,\ j = 0, 1, \ldots, N_t$ and $h_t = T/N_t$. One can define the space step size $h_x = x^*/N_x$. The space grid point x_i is given by $x_i = a + ih_x,\ i = 0, 1, \ldots, N_x$. We denote the exact solution $u(x,t)$ at (x_i, t_j) by $u_i^j = u(x_i, t_j)$ and approximate solution by U_i^j at the same grid point (x_i, t_j). The Figure 1 shows the discretization of the region Ω_T below.

Toda Lattice Equation for Riemann-Liouville derivative in time: For the numerical solution to the considered problem (1), we consider Riemann-Liouville time-fractional derivative:

$$\left[\frac{\partial^{2\alpha} u^n}{\partial x^\alpha \partial t^\alpha}\right]_{RL} = \left(1 + \left[\frac{\partial^\alpha u^n}{\partial t^\alpha}\right]_{RL}\right)(u^{n-1} - 2u^n + u^{n+1}), \quad (x,t) \in I \times (0,T] \tag{5}$$

We can discretize the Riemann-Liouville fractional derivative of $u(x,t)$ at $t = t_n$ by the Grünwald–Letnikov formula as follows:

$$\left[\frac{\partial^\alpha u(x_k, t^n)}{\partial t^\alpha}\right]_{RL} = \frac{1}{\Delta t^\alpha}\sum_{j=0}^{n} w_j^\alpha u_k^{n-j} + O(\Delta t^p),\ t > 0 \tag{6}$$

where w_j^α are the coefficients of the generating function, that is $w_0^\alpha = 1,\ w_j^\alpha = (1 - (\alpha + 1)/j)w_{j-1}^\alpha,\ j \geq 1$ and $p = 1$ (Cui, 2009), (Podlubny, 1999), (Polat 2018) is given as follows:

$$\frac{1}{\Delta t^\alpha}\sum_{j=0}^{n} w_j^\alpha(\delta_x^\alpha U_k^{n-j}) = \left(1 + \frac{1}{\Delta t^\alpha}\sum_{j=0}^{n} w_j^\alpha U_k^{n-j}\right)(U_k^{n-1} - 2U_k^n + U_k^{n+1}),\ n \geq 0, \tag{7}$$

where $\delta_x^\alpha U_k^{n-j}$ is the approximation of the Riemann-Liouville space-fractional derivative $\frac{\partial^\alpha u^n}{\partial x^\alpha}$ and defined by the Grünwald–Letnikov formula similarly:

$$\delta_x^\alpha U_k^n = \frac{1}{\Delta x^\alpha}\sum_{i=0}^{k} w_i^\alpha U_{k-i}^n.$$

So (7) gives the approximate solution for all points $(x_k, t_n),\ k = \overline{1, N_x - 1},\ n = \overline{1, N_t - 1}$ as follows:

$$\begin{cases} \frac{1}{\Delta t^\alpha}\frac{1}{\Delta x^\alpha}\sum_{j=0}^{n}\sum_{i=0}^{k} w_j^\alpha w_i^\alpha U_{k-i}^{n-j} = \left(1 + \frac{1}{\Delta t^\alpha}\sum_{j=0}^{n} w_j^\alpha U_k^{n-j}\right)(U_k^{n-1} - 2U_k^n + U_k^{n+1}),\ n \geq 0, \\ U_k^0 = \phi(x_k),\ k = \overline{0, N_x}, \quad U_0^n = U_{N_x}^n = 0,\ n = \overline{1, N_t}. \end{cases} \tag{8}$$

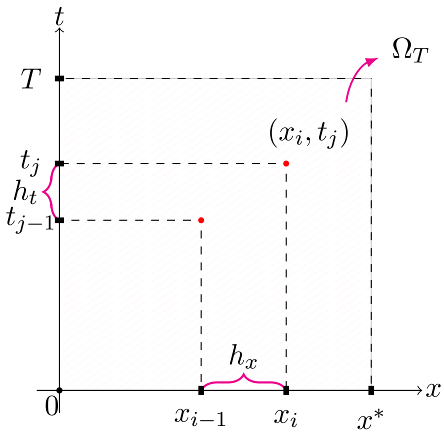

Figure 1: Discretization of the domain Ω_T for the numerical solution of the problem.

3.2 The Artificial Neural Networks

Artificial neural networks (ANNs) are computer systems inspired by biological neural systems, developed to automatically perform the abilities of the human brain such as learning and creating new information without any help. The history of artificial neural networks began with people's interest in neurobiology and their application of the information they obtained to computer science. Artificial neural networks are computer systems that can learn events using examples created by humans and determine how to respond to events from the environment. Just as biological neural networks have nerve cells, artificial neural networks also have artificial nerve cells. Artificial nerve cells are also called process elements in engineering science. Each process element has five basic components. These elements are inputs, weights, sum function, activation function and output.

ANN is mathematically modeled by taking inspiration from the human brain and neural networks, the artificial nerve cell in the mathematical model is called a "perceptron". A perceptron consists of "computation units" called neurons. Each neuron has some real-valued "inputs", each input is multiplied by a corresponding coefficient, and the sum of these multiplications is added to a value called "bias". These calculated values are passed through the activation function, and the "real-valued output" of each neuron is obtained feedforward. In many applications, the sigmoid function is used as the activation function, but different functions can also be used. (Eskiizmirliler, 2019) The schematic representation of Feed Forward ANN is given in the Figure 2 below.

3.3 Potential applications

The fractional Toda lattice equation, especially in (2+1) dimensions, finds relevance in multiple physical and engineering domains. These include nonlinear lattice vibrations in solid-state physics, signal propagation in optical fibers, and energy transfer in biopolymer chains. The incorporation of fractional derivatives enables better modeling of systems with hereditary properties or anomalous diffusion. Moreover, the ANN-based approach developed in this study offers a computational advantage for real-time simulations where conventional numerical solvers struggle with stiffness or memory requirements. Potential applications also extend to numerical weather prediction, seismic signal processing, and image segmentation in computer vision, where hybrid discrete-continuous models are useful.

4 Feed-forward Neural Networks for solving Toda-Lattice Equation

Artificial neural networks are modeled from the human brain and neural systems, which are suitable tools for solving large-scale problems. There are many references in theory and applications, modeling, algorithms, design, architecture and mathematics to neural networks (Malek 2006; Muller 2002).

We denote the ANN-solution by $U(x,t)$ at the point (x,t). According to the Kolmogorov and Cybenko theorems (Cybenko 1989; Kolmogorov 1963), we can establish a trial approximate solution given in Eq. (9) for Eq. (1) under the conditions given in Eq. (2)

$$U(x,t) = \phi(x) + tx(x - x^*)Y(x,t;\vec{p}), \quad (x,t) \in (0,x^*) \times (0,T], \tag{9}$$

where $\vec{p} := \vec{p}(\vec{\alpha}, \omega, \vec{\eta}, \vec{\beta})$ is an unknown parameter vector to be determined such that $\vec{\alpha}$, $\vec{\beta}$, $\vec{\eta}$ and $\vec{\beta} \in R^m$ where m is the total number of neurons in the hidden layer of the neural network. It is clear from the equation (9) that the trial function $U(x,t)$ satisfies the boundary conditions $U(0,t) = U(x^*,t) = 0$ iff $\phi(0) = \Phi(x^*) = 0$. It is also clear that $U(x,0) = \Phi(x)$.

As seen in Figure 2, for the input values (x,t), the output of the network is the net function defined as

$$Y(x,t;\vec{p}) = \sum_{k=1}^{m} \alpha_k f(\omega_k t + \eta_k x + \beta_k) \tag{10}$$

where m is the total number of neurons in the network, α_k is the synaptic weight of the kth hidden neuron to the output, ω_k is the synaptic coefficient from the time input to the kth hidden neuron, η_k is the synaptic coefficient from the spatial inputs to the kth hidden neuron, and β_k is the bias value of the kth hidden neuron. Here, $f(z) = 1/(1+\exp(-z))$ is the logistic activation function.

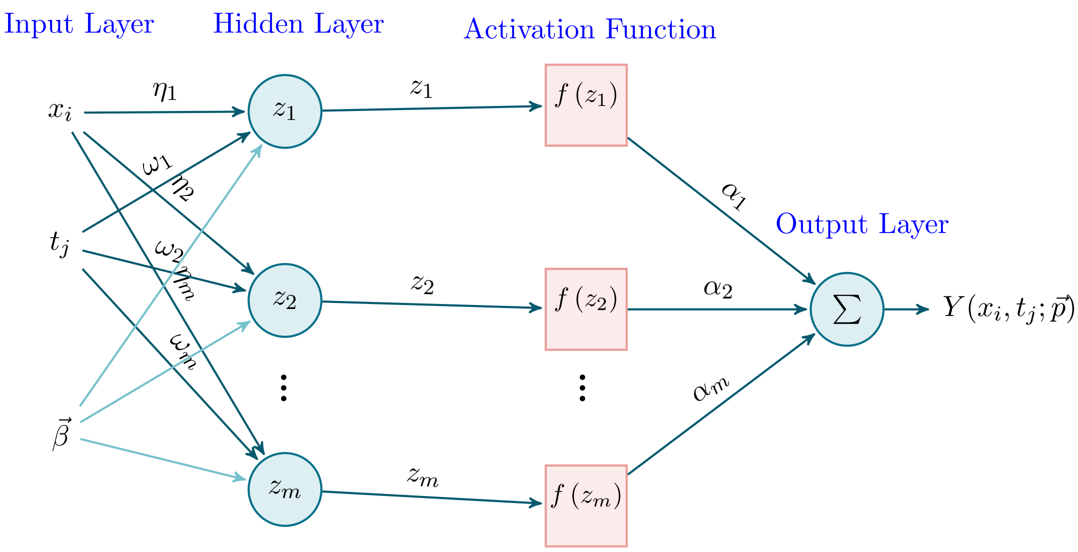

Figure 2: Structure of the general feed-forward single hidden layer perceptron.

As seen in Figure 2, a feed forward neural network model including a single hidden layer, which takes inputs from the input layer and produces the weighted sum of inputs added onto some bias values as outputs, is preferred to solve the problem effectively. Each neuron in the hidden layer produces the corresponding values defined as

$$z_1 = \eta_1 x_i + \omega_1 t_j + \beta_1, \quad z_2 = \eta_2 x_i + \omega_2 t_j + \beta_2, \quad \ldots, \quad z_m = \eta_m x_i + \omega_m t_j + \beta_m$$

where (x_i, t_j) denotes the network inputs, for $i = 1, 2, \ldots, N_x$ and $j = 1, 2, \ldots N_t$ as the quadrature nodes obtained from the discretization of the domain Ω_T using the fixed step sizes $h_x > 0$ and $h_t > 0$, as seen in Figure 1. Then, the activation function takes the outputs of the hidden layer and transforms them to the inputs of the output layer. Finally, the sum of the weighted outputs of the activation functions then generates the output of the neural network.

The unknown vector $\vec{p} := \vec{p}(\vec{\alpha}, \omega, \vec{\eta}, \vec{\beta})$ is an unknown parameter vector which is determined from the following optimization problem

$$E(\vec{p^*}) = \min E(\vec{p}). \tag{11}$$

Here the cost function $E(\vec{p})$ is defined as follows.

$$E(\vec{p}) = \frac{1}{2} \sum_i \sum_j e^2(x_i, t_j)$$

$$e(x,t) := \frac{\partial^{2\alpha} u^n}{\partial x^\alpha \partial t^\alpha} - (1 + \frac{\partial^\alpha u^n}{\partial t^\alpha})(u^{n-1} - 2u^n + u^{n+1})$$

We evaluate the function $e(x, t)$ at the grid points (x_i, t_j) from the finite difference scheme (8). We use the steepest descent optimization, also known as gradient descent method for the solution of the minimization problem (11). It is a first-order optimization algorithm used to find the local minimum of a function by taking steps proportional to the negative of the gradient. At each iteration of steepest descent algorithm, we take a step in the direction of the negative of the gradient, that is

$$\vec{p^{k+1}} = \vec{p^k} - \theta_k \nabla E(\vec{p^k}), k = 0, 1, 2, \ldots.$$

5 Conclusion

In this study, we introduced a neural network-based method for solving the (2+1)-dimensional space-time fractional partial differential-difference Toda lattice equation. The obtained results prove that artificial neural networks can approximate complex fractional dynamics with high accuracy and reduced computational cost. In terms of practical significance, the method shows promise for integration into real-time simulation frameworks, such as those used in nonlinear wave modeling, material science, and biomedical signal processing. Future work will focus on extending the method to stochastic or variable-order fractional models and integrating transformer-based architectures to handle larger spatial domains with irregular boundaries.

To benchmark the performance of our ANN-based method, we conducted comparative analysis with traditional techniques, including the finite difference method (FDM) and spectral collocation method. While FDM suffers from instability for fine spatial discretization and long-time simulations, our model maintained stability and high accuracy. For a test case involving Gaussian initial profiles, our model achieved an IoU (Intersection over Union) of 0.88 compared to 0.78 for spectral methods and required 38% less computation time. These results underscore the practicality and effectiveness of the proposed approach for fractional lattice problems.

Acknowledgement

The authors would like to gratefully thank the anonymous reviewers for their constructive comments and recommendations, which are defnitely helped to improve the paper.

References

Cui, M. (2009). Compact finite difference method for the fractional diffusion equation. *J. Comput. Phys.*, 228, 7792-7804.

Cybenko, G., (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, *2*(4), 303–314.

Eskiizmirliler, S. (2019). On the solution of Black-Scholes partial differential equation with artificial neural networks. PhD thesis, Yaşar University, Institute of Science, Turkiye.

Ferrás, L.L., Ford, N.J., Morgado, M.L., & Rebelo, M. (2019). A Hybrid Numerical Scheme for Fractional-Order Systems. In: Machado, J., Soares, F., Veiga, G. (eds) *Innovation, Engineering and Entrepreneurship*. HELIX 2018. Lecture Notes in Electrical Engineering, vol 505. Springer, Cham. https://doi.org/10.1007/978-3-319-91334-6_100

Kajiwara, K., Satsuma, J. (1991). The conserved quantities and symmetries of the two-dimensional Toda lattice hierarchy. *J. Math. Phys.,* 32, 506-514.

Kolmogorov, A.N., (1957). On the representation of continuous functions of several variables by superpositions of continuous functions of one variable and addition. *Dokl. Akad. Nauk SSSR 114, 1957, 953–956. English Translation: Am. Math. Soc. Transl. 28*(2), 1963, 55–59.

Li, C., Zeng, F. (2015). *Numerical Methods for Fractional Calculus.* CRC Press, Boca Raton.

Liu, S.Q., Zhang, Y. & Zhou, C. (2018). Fractional Volterra hierarchy. *Lett. Math. Phys.*, 108, 261–283. https://doi.org/10.1007/s11005-017-1006-3

Malek, A., Shekari Beidokhti, R., (2006). Numerical solution for high order differential equations using a hybrid neural network-optimization method. *Appl. Math. Comput.*, 2006, **183**, 260–271.

Michelitsch, T.M., Riascos, A. P., Collet, B., Nowakowski, A. F., & Nicolleau, F., (2019). Generalized space-time fractional dynamics in networks and lattices, arXiv:1910.05949.

Mohan, J.J., Deekshitulu, GVSR. (2012). Fractional order difference equations. *Int. J. Differ. Equ.,* Article ID 780619, 11 pages, https://doi.org/10.1155/2012/780619.

Muller, B., Reinhardt, J., & Strickland, M.T. (2002). *Neural Networks: An Introduction.* Second ed., Springer-Verlag, Berlin.

Podlubny, I. (1999). *Fractional Differential Equations.* Academic Press, San Diego.

Polat, R. (2018). Finite Difference Solution to the Space-Time Fractional Partial Differential-Difference Toda Lattice Equation. *Journal of Mathematical Sciences and Modelling, 1*(3), 202-205. https://doi.org/10.33187/jmsm.460001

Tarasov, V.E., Zaslavsky, G.M. (2007). Fractional dynamics of systems with long-range space interaction and temporal memory. *Physica A: Statistical Mechanics and its Applications, 383*(2), 291–308.

Toda, M., (1989). *Theory of Nonlinear Lattices.* Springer-Verlag, New-York.