Iranian Journal of Numerical Analysis and Optimization
Vol. 14, No. 4, 2024, pp 1016–1036
https://doi.org/10.22067/ijnao.2024.86494.1380
https://ijnao.um.ac.ir/

Research Article

Designing the sinc neural networks to solve the fractional optimal control problem

R. Heydari Dastjerdi*, and G. Ahmadi

Abstract

Sinc numerical methods are essential approaches for solving nonlinear problems. In this work, based on this method, the sinc neural networks (SNNs) are designed and applied to solve the fractional optimal control problem (FOCP) in the sense of the Riemann–Liouville (RL) derivative. To solve the FOCP, we first approximate the RL derivative using Grunwald–Letnikov operators. Then, according to Pontryagin's minimum principle for FOCP and using an error function, we construct an unconstrained minimization problem. We approximate the solution of the ordinary differential equation obtained from the Hamiltonian condition using the SNN. Simulation results show the efficiencies of the proposed approach.

*Corresponding author

Received 20 January 2024; revised 22 Aprail 2024; accepted 13 May 2024

Rasoul Heydari Dastjerdi
Department of Mathematics, Payame Noor University, Tehran, Iran. e-mail: rasoul59heydari@pnu.ac.ir

Ghasem Ahmadi
Department of Mathematics, Payame Noor University, Tehran, Iran. e-mail: g.ahmadi@pnu.ac.ir

How to cite this article
Heydari Dastjerdi, R. and Ahmadi, G., Designing the sinc neural networks to solve the fractional optimal control problem. *Iran. J. Numer. Anal. Optim.*, 2024; 14(4): 1016-1036. https://doi.org/10.22067/ijnao.2024.86494.1380

AMS subject classifications (2020): Primary 49M25; Secondary 68T07, 26A33.

Keywords: Sinc numerical method; Neural network; Sinc neural network; Pontryagin's minimum principle; Fractional optimal control problem.

1 Introduction

In the last decades, based on the sinc approximation, there have been developed a variety of numerical methods, which are now called the sinc numerical methods [30, 31]. Since 1974, Stenger [29] has studied the sinc approximation methods for the numerical problems. These methods are used as useful tools to solve linear and nonlinear problems arising from scientific and engineering applications. In general, the error in the sinc numerical method for the single exponential is $O(\exp(-c\sqrt{n}))$ with some positive c and n is the number of nodes used in the method [30]. In 2002, the sinc function was formed by double exponential transformation by Sugihara. He [33, 32] discovered that the error of the new method with a positive value of c is equal to $O(\exp(-\frac{cN}{\ln N}))$.

By expanding the normal integer calculus to noninteger calculus, fractional calculus is created. Today, fractional calculus is used in many branches of science and engineering, which shows its importance and application [4, 5, 14, 17, 22, 24, 28, 34, 37, 38]. An optimal control problem that includes at least one fractional derivative term in the performance index or differential equation dominating the system's dynamics is called the fraction optimal control problem. The fractional optimal control problem (FOCP) can be introduced concerning different definitions of fractional derivatives, where Riemann–Liouville (RL), Caputo, and Grunwald–Letnikov (GL) fractional derivatives are the most essential types of fractional derivatives. Because of the importance of this category of problems, research has been done to solve the FOCP. In [6], a solution of multidimensional FOCPs with inequality constraint by multiwavelets is presented. SIR and SEIR epidemic models are considered by a formulation for optimal control problems for a class of fuzzy fractional differential systems [9]. Nemati, Lima, and Torres [21] used modified hat functions for solving the FOCPs. The authors in [18] gave an extended modal series method and linear programming strategy to solve the FOCPs. In [35], for solving FOCPs with vector components, the authors used a Bernoulli polynomial method. It is hard to find the exact solution for the Hamiltonian system of equations because there exist both right and left in the Hamiltonian system. In [2, 3], the authors, to find the optimal solution of the FOCPs with the RL and Caputo fractional derivative sense, solved the Hamiltonian system of equations, which provides the necessary optimization conditions. To transform the Caputo-type FOCP into a system of algebraic equations, the authors of [7] have employed the GL approximation.

In this work, we consider the FOCP as follows:

$$\begin{aligned} &\min J\Big(x(t), u(t)\Big) = \textstyle\int_{t_0}^{t_f} F\Big(t, x(t), u(t)\Big) dt, \\ &\text{s.t. } {}_{t_0}D_{t_f}^{\alpha} x(t) = G\Big(t, x(t), u(t)\Big), \\ &\qquad x(t_0) = x_0, \end{aligned} \tag{1}$$

where $x(t) \in \mathbb{R}^p$ and $u(t) \in \mathbb{R}^q$ are the state and control variables, respectively. We assume that the integrand F, for all its arguments, has continuous first and second partial derivatives, and G is Lipschitz continuous on a set $\Omega \subset \mathbb{R}^p$. In addition, $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_p]^T, i = 1, 2, \dots, p$ and $n = [\alpha_i] + 1$, where $[\alpha_i]$ is the integer part of α_i.

The left and right RL fractional derivatives are, respectively, defined in the following form:

$$\begin{aligned} {}_aD_t^{\alpha} f(t) &= \frac{1}{\boldsymbol{\Gamma(n-\alpha)}} \left(\frac{d}{dt}\right)^n \int_a^t (t-z)^{n-\alpha-1} f(z) dz, \\ {}_tD_b^{\alpha} f(t) &= \frac{1}{\boldsymbol{\Gamma(n-\alpha)}} \left(-\frac{d}{dt}\right)^n \int_t^b (z-t)^{n-\alpha-1} f(z) dz. \end{aligned}$$

In the last decade, artificial neural networks (ANNs) have been applied to solving different problems, such as the FOCPs. The results show that the ANNs are accurate and efficient for many problems, and this method is comparable with other methods obtained by mathematical algorithms. The authors provided a method to solve the continuous-time direct adaptive optimal control for partially unknown nonlinear systems in [36]. Sabouri, Effati, and Pakdaman [25] used a neural network approach for solving a class of the FOCPs. Ghasemi and Nazemi [12] have designed a neural network to solve the FOCPs by using Mittag-Leffler. The multilayer perceptron (MLP) with a back propagation learning algorithm is employed for neonatal disease diagnosis [8]. Lagaris and Likas [16] solved the ordinary differential equations and partial differential equations for boundary and initial value problems by ANNs. The fuzzy neural networks algorithm is used for detecting cardiovascular diseases [26]. Recently, Ghasemi, Nazemi, and Hosseinpour [13] investigated the nonlinear FOCPs using ANNs. There are many references in theory and applications of neural networks, such as mathematical programming [19, 20] and optimal control problems [10].

In this work, after introducing the sinc numerical method, we design the sinc neural networks (SNNs). To solve the FOCP using this network, we first approximate the RL derivative used in the FOCP by the GL operator. Then, using Pontryagin's minimum principle (PMP) and the optimality conditions in the Hamiltonian function, we construct the unrestricted optimality problem. Finally, the adjustable parameters in the trial solution related to this problem are determined by the SNN. The main reason for using the SNN in solving FOCPs can be found in the simplicity of the network. Also, SNN has no bias and increases the efficiency of the network due to the linearity of the unknowns of the problem with respect to the output.

The article is arranged as follows: Section 2 introduces the sinc numerical method. In Section 3, the SNN is presented. In Section 4, the proposed technique is utilized for solving FOCPs. Numerical examples are presented in Section 5, and Section 6 contains concluding remarks.

2 Sinc numerical method

In recent decades, sinc numerical methods have been used to solve various problems due to their exponential convergence rate. In [30], sinc function is fully introduced. In this section, we briefly state some of its definitions and features.

The sinc function is defined for each $x \in (-\infty, \infty)$ as follows:

$$Sinc(x) = \begin{cases} \dfrac{\sin \pi x}{\pi x}, & x \neq 0, \\ 1, & x = 0. \end{cases}$$

For $h > 0$, and any integer k, the translated sinc functions with evenly spaced nodes are defined as

$$S(k,h)(x) = Sinc\left(\frac{x}{h} - k\right).$$

By these functions, a set of interpolation functions can be formed in the following form:

$$S(k,h)(jh) = \delta_{j,k} = \begin{cases} 1, & j = k, \\ 0, & j \neq k. \end{cases} \tag{2}$$

Assume that f is a function defined on real line. Then for $h > 0$, the cardinal function corresponding to f is introduced by

$$C(f,h)(x) = \sum_{k=-\infty}^{\infty} f(kh) S(k,h)(x), \tag{3}$$

while the series in (3) converges [30]. From the relations (2) and (3), it can be concluded that the function f is interpolated by the cardinal function in points $\{nh\}_{n=-\infty}^{+\infty}$. Suppose that in the complex w -plane, D_d shows the infinite strip region of width $2d$ as

$$D_d = \{w = t + is : |s| < d\}.$$

We use a conformal map ϕ as $\phi(\Gamma) = R$ for problems on a subinterval $\Gamma \subseteq R$, such that ϕ has the inverse ψ and is a conformal map of the simply-connected domain D, where $(0,1) \subseteq D$, onto D_d, then on a subinterval $\Gamma = (0,1) = \psi(R)$ with $\phi(0) = -\infty$, and $\phi(1) = +\infty$. The following interpolation can be defined as

$$f(x) \cong \sum_{k=-N}^{N} f(x_k) S_k(x),$$

where $x_k = \psi(kh)$ is defined as the sinc grid points and

$$S_k(x) = S(k,h)o\phi(x) = Sinc\left(\frac{\phi(x)}{h} - k\right)$$

is the translated sinc basic function. As a result of these discussions, double exponential and single exponential can be introduced, respectively, as follows:

$$z = \psi_1(w) = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{\pi}{2}\sinh(w)\right),$$
$$z = \psi_2(w) = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{w}{2}\right),$$

where the interval $(-\infty, +\infty)$ is mapped onto $(0, 1)$.

3 Sinc neural network (SNN)

In this part, to solve FOCP, we introduce the structure of the SNN. There are few works in the field of neural networks with the sinc activation function. SNN with a single input and a single output has been utilized for the approximation of functions with one variable by Elwasif and Fausett [11] in 1996. Suppose that $T = [t_1, t_2, \ldots, t_n]^T$, and $O = [o_1, o_2, \ldots, o_q]^T$ are the input and output vectors of the SNN, respectively. Then, the SNN can be structured as follows:

$$O = \sum_{i=1}^{m} W_i S_i(T), \tag{4}$$

where $W = [W_1, W_2, \ldots, W_m]$ is a $q \times n$ matrix of weights between the nodes of $S_i(T)$ and outputs, and $S_i(T), 1 \leq i \leq m$, is the translated sinc basic function with input vector T. The structure of SNN can be observed in Figure 1.

The structure of SNN has been motivated from the sinc numerical method as a fascinating approach in solving nonlinear problems numerically. Sinc function is a famous and beneficial function in science and engineering. It is a smooth function with positive and negative amounts. It oscillates, and its output approaches zero when its input tends to infinity.

Unlike the MLP neural network, the structure of SNN is simple with a few number of trainable parameters. In fact, it contains only one layer of

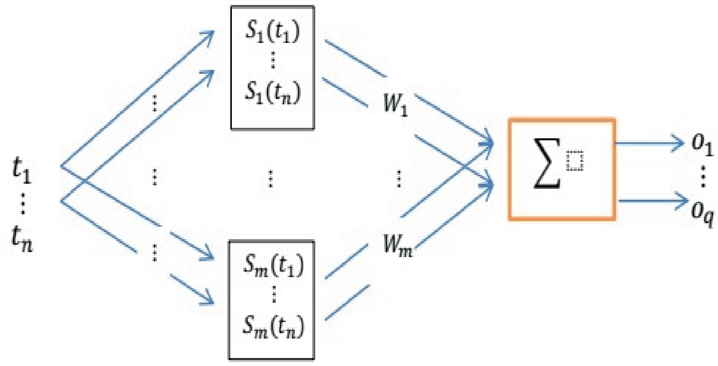

Figure 1: The structure of SNN.

trainable parameters. In SNN, the activation functions are the sinc basis functions, and therefore, the accuracy of modeling results increases when the number of sinc basis functions (nodes $S_i(T)$) increases in its structure.

To train the SNN, we can use various existing methods such as gradient methods, methods based on Lyapunov's stability theory, and so on. Since in these neural networks we only have adjustable parameters in one layer, the output of the neural network is linear with respect to the parameters, and this makes it easier to train it, and it becomes possible to use a variety of learning algorithms. In this article, based on the error functions obtained from the existing mathematical relations, an optimization problem is created, and by solving it, we get the optimal parameters of the neural network.

4 Applying SNN to solve FOCP

Before we resolve the problem, we approximate the RL fractional derivative with the GL operators. Assume $y \in c^l([a, b]), l \in N$. Then the l-order derivative of y at $t \in (a, b]$ can be defined as

$$ {}_aD_t^l y(t) = \lim_{h \to 0} \frac{1}{h^l} \sum_{k=0}^{l} (-1)^k \binom{l}{k} y(t - kh), \tag{5} $$

where $\binom{l}{k} = \frac{l!}{k!(l-k)!}$ are the binomial coefficients. By replacing the real number α instead of l, binomial coefficients can be developed using the gamma function as

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt, \quad z > 0,$$

where $\Gamma(1) = 1$ and $\Gamma(z+1) = z\Gamma(z)$, for any $z > 0$. Now, the binomial coefficients are defined for the real number α as $\binom{\alpha}{k} = \frac{\Gamma(\alpha+1)}{\Gamma(k+1)\Gamma(\alpha-k+1)}$. So, (5) can be extended to the fractional order $\alpha > 0$ by means of

$$ {}_aD_t^\alpha y(t) = \lim_{N\to\infty} \frac{1}{h^\alpha} \sum_{k=0}^{N} w_k^{(\alpha)} y(t-kh), \tag{6}$$

where $h = \frac{t-a}{N}$ and $t \in (a, b]$. Besides, we have $w_k^{(\alpha)} = (-1)^k \binom{\alpha}{k}$, which can be evaluated by using the recurrence formula as follows:

$$w_0^{(\alpha)} = 1, \quad w_k^{(\alpha)} = \big(1 - \frac{\alpha+1}{k}\big) w_{k-1}^{(\alpha)}.$$

Note that for $h > 0$, relation (6) is left GL fractional derivative, and for $h < 0$, it is called the right GL fractional derivative.

Now, to solve problem (1), we define the Hamiltonian function as

$$H\big(x(t), u(t), p(t), t\big) = F\big(x(t), u(t), t\big) + p(t).G\big(x(t), u(t), t\big),$$

where $p(t) \in \mathbb{R}^q$ is the co-state vector. Assume that $x^*(t), u^*(t)$, and $p^*(t)$ are the optimal state, control, and co-state functions, respectively. Then, to minimize the objective function (1), a necessary condition for $u^*(t)$ is as follows:

$$H\big(x^*(t), u^*(t), p^*(t), t\big) \le H\big(x^*(t), u(t), p^*(t), t\big), \tag{7}$$

where $t \in [t_0, t_f]$, for all impossible controls. Equation (7) is called the PMP, where an optimal control must minimize the Hamiltonian function [15]. According to the PMP, if $x(t), u(t)$, and $p(t)$ are the optimal solutions in (7), then they must satisfy the following conditions:

$$\begin{cases} \dfrac{\partial H(x,u,p,t)}{\partial x} = {}_tD_{t_f}^\alpha p(t), \\ \dfrac{\partial H(x,u,t,p)}{\partial p} = {}_{t_0}D_t^\alpha x(t), \\ \dfrac{\partial H(x,u,p,t)}{\partial u} = 0. \end{cases} \tag{8}$$

We try to solve the system of equations (8) using the SNN. It can define the main trial solutions using the SNN (4), so that the initial or boundary conditions are satisfied, so they are constructed in the following form:

$$\begin{cases} x_T = x_0 + (t - t_0)o_x, \\ p_T = (t - t_f)o_p, \\ u_T = o_u, \end{cases} \tag{9}$$

where

$$o_x = \sum_{i=1}^{m} W_i^x S_i^x(T), \quad o_p = \sum_{i=1}^{m} W_i^p S_i^p(T), \quad o_u = \sum_{i=1}^{m} W_i^u S_i^u(T),$$

are state, co-state, and control neural networks, respectively. Note that since $x(t_f)$ is free, then $p(t_f) = 0$. By placing the trial solutions (9) in relation (8), we have

$$\begin{cases} \dfrac{\partial H_T}{\partial x_T} = {}_tD_{t_f}^{\alpha} p_T(t), \\ \dfrac{\partial H_T}{\partial p_T} = {}_{t_0}D_t^{\alpha} x_T(t), \\ \dfrac{\partial H_T}{\partial u_T} = 0, \end{cases} \tag{10}$$

where $H_T = H\big(x_T(t), u_T(t), p_T(t), t\big)$. To solve the system of equations (10) as an unconstrained minimization problem, we put them on the $m+1$ points of the interval $[t_0, t_f]$ as $t_k = t_0 + \frac{t_f - t_0}{m}k, k = 0, 1, \ldots, m$. According to the relation (5), the left and right RL fractional derivatives can be calculated as follows:

$$\begin{aligned} {}_{t_0}D_{t_k}^{\alpha} x_T &\simeq \frac{1}{h^\alpha} \sum_{j=0}^{k} w_j^{(\alpha)} x_T(t_{k-j}), \quad k = 1, 2, \ldots, m, \\ {}_{t_k}D_{t_f}^{\alpha} p_T &\simeq \frac{1}{h^\alpha} \sum_{j=0}^{m-k} w_j^{(\alpha)} p_T(t_{k+j}), \quad k = 0, 1, \ldots, m-1. \end{aligned}$$

Now, we construct the optimization problem as

$$\min E(\eta) = \frac{1}{2} \sum_{k=0}^{m} \{E_1(t_k, \eta) + E_2(t_k, \eta) + E_3(t_k, \eta)\}, \tag{11}$$

where $\eta = (w_x, w_p, w_u)$ and

$$\begin{cases} E_1(t_k,\eta) = \left[\dfrac{\partial H_T}{\partial x_T} - \frac{1}{h^\alpha}\sum_{j=0}^{m-k} w_j^{(\alpha)} p_T(t_{k+j})\right]^2, & k = 0,1,\dots,m-1, \\ E_2(t_k,\eta) = \left[\dfrac{\partial H_T}{\partial p_T} - \frac{1}{h^\alpha}\sum_{j=0}^{k} w_j^{(\alpha)} x_T(t_{k-j})\right]^2, & k = 1,2,\dots,m, \\ E_3(t_k,\eta) = \left[\dfrac{\partial H_T}{\partial u_T}\right]^2, & k = 1,2,\dots,m. \end{cases}$$

Based on [13, Lemma 1], if η satisfies the system of equations (11), then η is an optimal solution of (1). So, one can verify that the minimization problem (11) is equivalent to the following problem:

$$\min_{\eta} E(\eta) = \frac{1}{2}||\phi(\eta)||^2. \tag{12}$$

Optimization algorithms such as steepest descent, Newton, quasi-Newton, and conjugate gradient can be used to solve the problem (12). The main advantage of the mentioned method is that it is not very complicated, and the accuracy of the approximation solution can be increased using more training points in the interval $[t_0, t_f]$.

Remark 1. In the present work, on the basis of sinc numerical methods and using the sinc basis functions, we propose the SNNs for the optimal control of fractional linear and nonlinear dynamic systems. Then, the results have been compared with MLP. In [13], the MLP is used for the optimal control of fractional dynamic systems. However, aside from the utilized neural structure, the main approach to solving FOCP is similar to [13].

In this work, we introduce the SNN as a new neural network with a simple structure for solving the FOCPs. Due to the beneficial properties of sinc function, as described in Section 3, the SNN is powerful in solving the problems. The simulation results show the capabilities of this neural network in solving FOCPs.

Remark 2. Generally, utilizing the neural networks to solve the fractional optimal control follows the presented approach in [13], and therefore, we can spread the presented theory for the convergence and stability of the method in that article to the present work.

5 Numerical examples

In order to show the efficiency and application of the proposed technique, three problems are solved in this section. In these numerical calculations, the fractional derivatives are approximated using the GL operators. In all the examples, five nodes are applied in the SNN to solve FOCP. To show the efficiency of the proposed method, we have compared this method with MLP and shown the error graph of each of these methods in the figures.

In this work, all numerical computations have been coded in MATLAB R2017b with 16GB RAM. In the proposed approach, to adjust the parameters of neural structures, we are faced with some unconstrained nonlinear optimization problems. These optimization problems have been solved using the “fminunc” function, and the optimization algorithm has been selected as “quasi-newton”.

Example 1. As the first example, consider the FOCP as follows:

$$\begin{aligned}
&\min\ J = \tfrac{1}{2}\textstyle\int_0^1 \big(3x^2(t) + u^2(t)\big)dt,\\
&{}_0D_t^{\alpha} x(t) = -x(t) + u(t),\\
&x(0) = 0, \quad x(1) = 2.
\end{aligned}$$

The exact solution of this problem [13] in the case of $\alpha = 1$ is equal to

$$\begin{cases} x(t) = \frac{2}{\sinh 2}\sinh(2t),\\ u(t) = \frac{2}{\sinh 2}\big(2\cosh(2t) + \sinh(2t)\big). \end{cases}$$

The Hamiltonian function of this example is as follows:

$$H\big(x,u,p,t\big) = \frac{1}{2}\big(3x^2(t) + u^2(t)\big) + p(t)\big(-x(t) + u(t)\big).$$

According to relation (10), we have

$$\begin{cases} {}_tD_1^{\alpha} p(t) = 3x(t) - p(t),\\ {}_0D_t^{\alpha} x(t) = -x(t) + u(t),\\ u(t) + p(t) = 0,\\ x(0) = 0, \quad x(1) = 2. \end{cases}$$

Due to the initial conditions, the trial solutions are selected as follows:

$$\begin{cases} x_T = 2t + t(t-1)o_x, \\ p_T = o_p, \\ u_T = o_u. \end{cases}$$

Figure 2 shows the exact and approximate diagram of the state and control function. Also, the error graph of the proposed method and the MLP are compared.

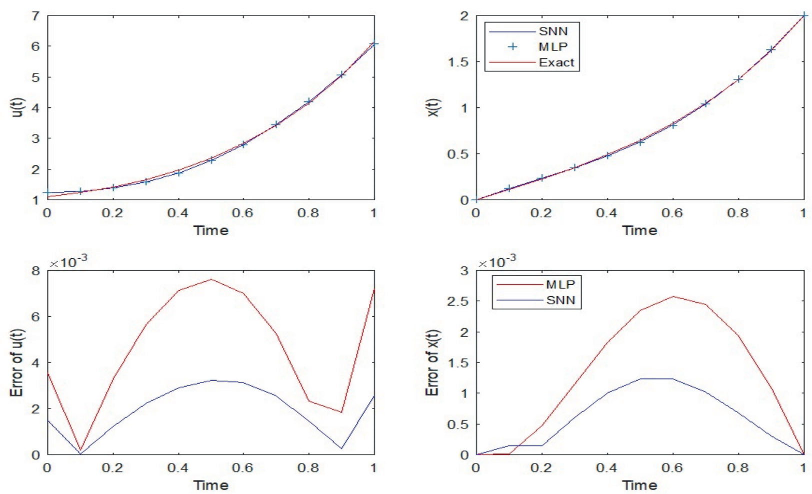

Figure 2: Exact and approximated state and control functions and their errors for Example 1.

In Table 1, we have expressed the error related to approximated state and control functions in the number of points and the process time in the SNN is 2.4 seconds, while it is 2.51 seconds in the MLP.

Example 2. Consider the following FOCP [23]:

$$\begin{aligned} \min\ J &= \tfrac{1}{2}\int_0^1 \left(x^2(t) + u^2(t)\right)dt, \\ {}_0D_t^{\alpha}x(t) &= -x(t) + u(t), \\ x(0) &= 1. \end{aligned}$$

The exact solution of this example with $\alpha = 1$ is

Table 1: Parameter values for Example 1

t	Error of $x(t)$ (SNN)	Error of $x(t)$ (MLP)	Error of $u(t)$ (SNN)	Error of $u(t)$ (MLP)
0.0	0	0	0.0023	0.0043
0.1	10^{-5}	10^{-5}	0.0004	0.0002
0.2	0.0001	0.0004	0.0012	0.0032
0.3	0.0004	0.0010	0.0025	0.0058
0.4	0.0006	0.0016	0.0034	0.0075
0.5	0.0009	0.0022	0.0039	0.0081
0.6	0.0011	0.0025	0.0037	0.0075
0.7	0.0011	0.0024	0.0029	0.0056
0.8	0.0010	0.0021	0.0012	0.0023
0.9	0.0006	0.0012	0.0014	0.0024
1.0	0	0	0.0050	0.0087

$$\begin{cases} x(t) &= \cosh(\sqrt{2}t) + \beta \sinh(\sqrt{2}t), \\ u(t) &= (1+\sqrt{2}\beta)\cosh(\sqrt{2}t) + (\sqrt{2}+\beta)\sinh(\sqrt{2}t), \end{cases}$$

where $\beta = -\frac{\cosh(\sqrt{2})+(\sqrt{2}\sinh(\sqrt{2})}{(\sqrt{2}\cosh(\sqrt{2})+\sinh(\sqrt{2})} \approx -0.98$.

Since $x(1)$ is free, we conclude $p(1) = 0$. So, due to relation (10), we have

$$\begin{cases} {}_tD_1^\alpha p(t) = x(t) - p(t), \\ {}_0D_t^\alpha x(t) = -x(t) + u(t), \\ u(t) + p(t) = 0, \\ x(0) = 1, \quad p(1) = 0. \end{cases}$$

From the initial conditions, the trial solution is written as

$$\begin{cases} x_T = (1+t)o_x, \\ p_T = (t-1)o_p, \\ u_T = o_u. \end{cases}$$

Figure 3 shows the exact and approximate diagram of the state and control function, as well as the error diagram of the proposed method and MLP.

In Table 2, we have expressed the error related to approximated state and control functions in the number of points and the process time in the SNN is 3.14 seconds, while it is 3.3 seconds in the MLP.

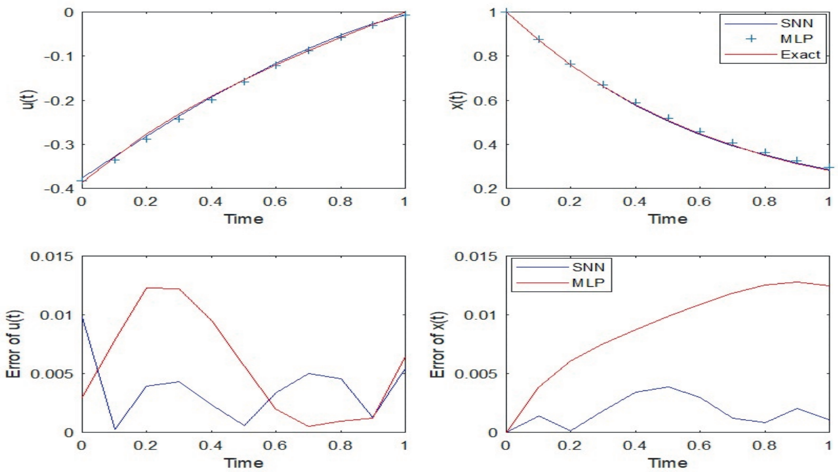

Figure 3: Exact and approximated state and control functions and their errors for Example 2.

Table 2: Parameter values for Example 2

t	Error of $x(t)$ (SNN)	Error of $x(t)$ (MLP)	Error of $u(t)$ (SNN)	Error of $u(t)$ (MLP)
0.0	0	0	0.0098	0.0030
0.1	0.0014	0.0038	0.0002	0.0078
0.2	0.0002	0.0061	0.0039	0.0123
0.3	0.0019	0.0075	0.0043	0.0122
0.4	0.0034	0.0087	0.0024	0.0095
0.5	0.0038	0.0098	0.0006	0.0056
0.6	0.0030	0.0109	0.0034	0.0020
0.7	0.0012	0.0119	0.0050	0.0005
0.8	0.0008	0.0125	0.0046	0.0009
0.9	0.0020	0.0128	0.0013	0.0012
1.0	0.0011	0.0125	0.0055	0.0065

Example 3. In this example, we consider the FOCP as [1]

$$
\begin{aligned}
&\min J = \frac{1}{2}\int_0^1 \left(x^2(t) + u^2(t)\right)dt,\\
&{}_0D_t^{\alpha} x(t) = tx(t) + u(t),\\
&x(0) = 1.
\end{aligned}
$$

According to (10), we have

$$\begin{cases} {}_tD_1^{\alpha}p(t) = x(t) + tp(t), \\ {}_0D_t^{\alpha}x(t) = tx(t) + u(t), \\ u(t) + p(t) = 0, \\ x(0) = 1, \quad p(1) = 0. \end{cases}$$

It is clear that $p(1) = 0$, because $x(1)$ is free. Also, we can select the trial solutions as

$$\begin{cases} x_T = 1 + to_x, \\ p_T = (t-1)o_p, \\ u_T = o_u. \end{cases}$$

This example does not have an exact solution; therefore, only approximate solutions for the control and state functions are shown in Figure 4. Also, approximate the state and control functions with different values of α are shown. The operation time is 3.5 seconds.

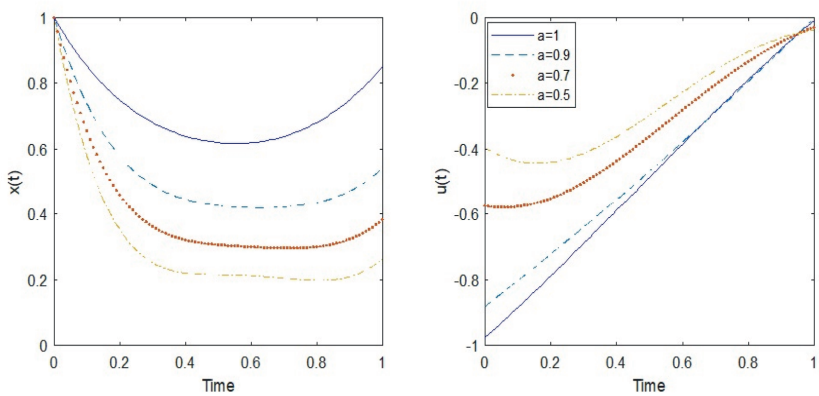

Figure 4: Approximated of the state and control functions by assuming different values of α for Example 3.

Example 4. Consider the nonlinear FOCP as follows [27]:

$$\begin{aligned} &\min\, J(x,u) = \tfrac{1}{2}\int_0^1 \big(0.625x^2(t) + 0.5x(t)u(t) + 0.5u^2(t)\big)dt, \\ &\quad {}_0D_t^{\alpha}x(t) = 0.5x(t) + u(t), \\ &\quad x(0) = 1. \end{aligned}$$

The exact solution of this problem in the case of $\alpha = 1$ is equal to

$$\begin{cases} x(t) = 0.99999 - 0.761547t + 0.499522t^2 - 0.124986t^3 + 0.0379117t^4 \\ \qquad\qquad -0.00284627t^5, \\ u(t) = \frac{-\left(\tanh(1-t)+0.5\right)\cosh(1-t)}{\cosh(1)}. \end{cases}$$

Figure 5 shows the exact and approximate diagram of the state and control function. In Table 3, we have expressed the error related to approximated state and control functions in the number of points and the process time in the SNN is 2.71 seconds, while it is 2.98 seconds in the MLP.

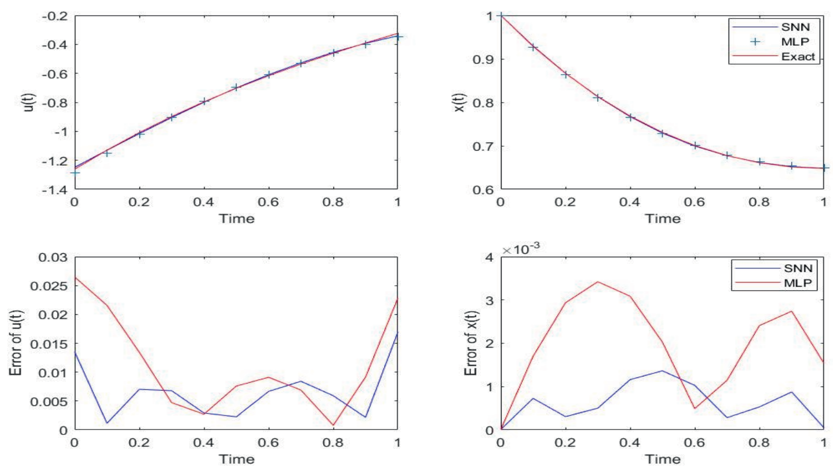

Figure 5: Exact and approximated state and control functions and their errors for Example 4.

6 Conclusion

In this work, based on the sinc numerical method, we presented a novel structure for the SNN and used this network to solve the FOCPs. Compared to MLP, this network has a simpler structure and fewer variables. The results obtained from the simulations showed that the method used is more efficient than the MLP. In general, ANNs have lower computational complexity. We can increase the accuracy of the trial solution by increasing the number of training points or different optimization algorithms. One of the important

Table 3: Parameter values for Example 4

t	Error of $x(t)$ (SNN)	Error of $x(t)$ (MLP)	Error of $u(t)$ (SNN)	Error of $u(t)$ (MLP)
0.0	0	0	0.0135	0.0265
0.1	0.0007	0.0017	0.0011	0.0216
0.2	0.0003	0.0029	0.0070	0.0134
0.3	0.0005	0.0034	0.0068	0.0047
0.4	0.0012	0.0031	0.0029	0.0027
0.5	0.0014	0.0020	0.0023	0.0076
0.6	0.0010	0.0005	0.0067	0.0091
0.7	0.0003	0.0011	0.0084	0.0069
0.8	0.0005	0.0024	0.0059	0.0008
0.9	0.0009	0.0027	0.0022	0.0092
1.0	0	0.0015	0.0169	0.0228

advantages of ANNs is that we can train the neural network at some points, but the final approximation solution can be calculated at each arbitrary point in the training interval. As a future work, the proposed method can be utilized to solve delay FOCPs.

Acknowledgements

Authors are grateful to the anonymous referees and editor for their constructive comments.

References

[1] Agrawal, O.M.P. *A general formulation and solution scheme for fractional optimal control problem*, Nonlinear Dyn. 38 (2004), 323–337.

[2] Agrawal, O.M.P. *A formulation and numerical scheme for fractional optimal control problems*, J. Vib. Control. 14 (2008), 1291–1299.

[3] Agrawal, O.M.P. and Baleanu, D. *A Hamiltonian formulation and a direct numerical scheme for fractional optimal control problems*, J. Vib. Control. 13 (2007), 1269–1281.

[4] Ahmad, H., Khan, M., Ahmad, I., Omri, M. and Alotaibi, M. *A meshless method for numerical solutions of linear and nonlinear time-fractional Black–Scholes models*, AIMS Math. 8 (2023), 19677–19698.

[5] Alquran, M., Sulaiman, T., Yusuf, A., Alshomrani, A. and Baleanu, D. *Nonautonomous lump-periodic and analytical solutions to the (3+1)-dimensional generalized Kadomtsev–Petviashvili equation*, Nonlinear Dyn. 111 (2023), 11429–11436.

[6] Ashpazzadeh, E., Lakestani, M. and Yildirim, A. *Biorthogonal multiwavelets on the interval for solving multidimensional fractional optimal control problems with inequality constraint*, Optim. Control Appl. Math. 41(5) (2020), 1477–1494.

[7] Biswas, R.K. and Sen, S. *Fractional optimal control problems with specified final time*, J. Comput. Nonlinear Dyn. 6(2) (2010), 021009.

[8] Chowdhury, D.R., Chatterjee, M. and Samanta, R.K. *An artificial neural network model for neonatal disease diagnosis*, International Journal of Artificial Intelligence and Expert Systems (IJAE), 2(3) (2011), 96–106.

[9] Dong, N.P., Long, H. and Khastan, A. *Optimal control of a fractional order model for granular SEIR epidemic with uncertainty*, Commun. Nonlinear Sci. Numer. Simul. 88 (2020), 105312.

[10] Effati, S. and Pakdaman, M. *Optimal control problem via neural networks*, Neural Comput. Appl. 23 (2013), 2093–2100.

[11] Elwasif, W. and Fausett, L.V. *Function approximation using a sinc neural network*, In: Proceedings of the SPIE, Volume 2760 (1996), 690–701.

[12] Ghasemi, S. and Nazemi, A.R. *A neural network method based on Mittag-Leffler function for solving a class of fractional optimal control problems*, AUT J. Model. Simul. 50(2) (2018), 211–218.

[13] Ghasemi, S., Nazemi, A. and Hosseinpour, S. *Nonlinear fractional optimal control problems with neural network and dynamic optimization schemes*, Nonlinear Dyn. 89 (2017), 2669–2682.

[14] Hashemi, M., Mirzazadeh, M. and Ahmad, H. *A reduction technique to solve the (2+1)-dimensional KdV equations with time local fractional derivatives*, Opt. Quantum Electron, 55(8) (2023), 721.

[15] Kirk, D.E. *Optimal control theory: An introduction*, Donver Publication, Inc. Mineola, New York, 2004.

[16] Lagaris, I.E. and Likas, A. *Hamilton–Jacobi theory over time scales and applications to linear-quadratic problems*, IEEE Trans Neural Netw. 9(5) (2012), 987–1000.

[17] Latif, S., Sabir, Z., Raja, M., Altamirano, G., Núñez, R., Gago, D., Sadat, R. and Ali, M. *IoT technology enabled stochastic computing paradigm for numerical simulation of heterogeneous mosquito model*, Multimed Tools Appl. 82 (2023), 18851–18866.

[18] Mohammadzadeh, E., Pariz, N., Hosseini sani, S.K. and Jajarmi, A. *Optimal Control for a Class of Nonlinear Fractional-Order Systems Using an Extended Modal Series Method and Linear Programming Strategy*, J. Control 10(1) (2016), 51–64.

[19] Nazemi, A. and Effati, S. *An application of a merit function for solving convex programming problems*, Comput. Ind. Eng. 66(2) (2013), 212–221.

[20] Nazemi, A. and Omidi, F. *A capable neural network model for solving the maximum flow problem*, J. Comput. Appl. Math. 236(14) (2012), 3498–3513.

[21] Nemati, S., Lima, P.M. and Torres, D.F.M. *A numerical approach for solving fractional optimal control problems using modified hat functions*, Commun. Nonlinear Sci. Numer. Simul. 78 (2019), 104849.

[22] Qayyum, M., Ahmad, E., Tauseef Saeed, S., Ahmad, H. and Askar, S. *Homotopy perturbation method-based soliton solutions of the time-fractional (2+1)-dimensionalWu–Zhang system describing long dispersive gravity water waves in the ocean*, Front Phys. 11 (2023), 1178154.

[23] Saberi Nik, H., Effati, S. and Yildirim, A. *Solution of linear optimal control systems by differential transform method*, Neural Comput. Appl. 23 (2013), 1311–1317.

[24] Sabir, Z. and Guirao, J. *A soft computing scaled conjugate gradient procedure for the fractional order Majnun and Layla romantic story*, Mathematics 11 (2023), 835.

[25] Sabouri, J., Effati, S. and Pakdaman, M. *A neural network approach for solving a class of fractional optimal control problems*,Neural Process. Lett. 45 (2017), 59–74.

[26] Shi, J., Sekar, B.D., Dong, M.C. and Hu, X.Y. *Extract knowledge from site-sampled data sets and fused hierarchical neural networks for detecting cardiovascular diseases*, 2012 International Conference on Biomedical Engineering and Biotechnology, (2012), 275–279.

[27] Singh, H., Pandey, R.K. and Kumar, D. *A reliable numerical approach for nonlinear fractional optimal control problems*, Int. J. Nonlinear Sci. Numer. Simul. 22(5) (2021), 495–507.

[28] Souayeh, B. and Sabir, Z. *Designing Hyperbolic Tangent Sigmoid Function for Solving theWilliamson Nanofluid Model*, Fractal Fract. 7 (2023), 350.

[29] Stenger, F. *A "Sinc-Galerkin" method of solution of boundary value problems*, J. Math. Comp. 33(145) (1979), 85–109.

[30] Stenger, F. *Sinc-related methods, Numerical Methods Based on Sinc and Analytic Functions*, Springer, New York, 1993.

[31] Stenger, F. *Summary of Sinc numerical methods*, J. Comput. Appl. Math. 121 (2000), 379–420.

[32] Sugihara, M. *Optimality of the double exponential formula–functional analysis approach–*, Numer. Math. (Heidelb.) 75 (1997), 379–395.

[33] Sugihara, M. *Near optimality of the sinc approximation*, Math. Comput. 72 (2003), 767–786.

[34] Sulaiman, T.A., Yusuf, A., Alshomrani, A.S. and Baleanu, D. *Wave solutions to the more general (2+1)-dimensional Boussinesq equation arising in ocean engineering*, Int. J. Mod. Phys. B 37 (22) (2023), 2350214.

[35] Taherpour, V., Nazari, M. and Nemati, A. *A new numerical Bernoulli polynomial method for solving fractional optimal control problems with vector components*, Comput. Method. Differ. Equ. 9(2) (2021), 446–466.

[36] Vrabie, D. and Lewis, F. *Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems*, Neural Netw. 22(3) (2009), 237–246.

[37] Zarin, R., Khan, M., Khan, A. and Yusuf, A. *Deterministic and fractional analysis of a newly developed dengue epidemic model*, Waves Random Complex Media, (2023), 1–34.

[38] Zil-E-Huma, Butt, A., Raza, N., Ahmad, H., Ozsahin, D. and Tchier, F. *Different solitary wave solutions and bilinear form for modified mixed-KDV equation*, Optik, 287 (2023), 171031.