RESEARCH ARTICLE

# Approximate solution to fractional differential equations using fractional sine-cosine series approach and nature inspired metaheuristic algorithms

Ratika Rastogi[a], Om Prakash Misra[a] , Rajshree Mishra[b]

[a] School of Mathematics and Allied Sciences, Jiwaji University, Gwalior, Madhya Pradesh, India; [b] Department Of Mathematics, Shrimant Madhavrao Scindia Government Model Science College, Gwalior, Madhya Pradesh, India

**ARTICLE HISTORY**

Compiled February 7, 2025

**ABSTRACT**

In this paper, a numerical method is proposed to solve the fractional differential equations (FDEs) using a fractional sine-cosine series (FSCS) as approximate function and then the metaheuristic algorithms such as Differential Evolution (DE) and Particle Swarm Optimization (PSO) are applied to get the optimal solution. In this method, an optimization problem is formulated using FDEs and associated initial conditions, where the implicit form of the fractional differential equation is considered as the objective function and the initial conditions are taken to be the constraints of the optimization problem. The effectiveness and performance of our proposed techniques, Differential Evolution - Fractional sine-cosine series (DE-FSCS) and Particle Swarm Optimization - Fractional sine-cosine series (PSO-FSCS), are compared with other existing numerical methods using mean square error (MSE) criterion. It is shown that the solutions obtained by the proposed DE-FSCS and PSO-FSCS methods are more efficient and reliable as compared to the Variational iteration method (VIM), Grey Wolf Optimization - Variational iteration method (GWO-VIM) and one other numerical iterative scheme by calculating the MSE.

CONTACT: Ratika Rastogi. Email: ratikajugwalior@gmail.com
CONTACT: O.P. Misra. Email: misraop09@gmail.com
CONTACT: Rajshree Mishra. Email: rajshreemishraa@gmail.com

**KEYWORDS**

Fractional sine-cosine series, Metaheuristic Algorithms, Differential Evolution, Particle Swarm Optimization, Fractional Differential Equations.

## 1. Introduction

Fractional calculus is currently one of the most useful areas of mathematics for simulating systems associated to real-world problems. It places a strong emphasis on the investigation of arbitrary ordered integrals and derivatives, which are useful in current mathematical research [1]. The theory of fractional calculus has been successfully applied to several real-world issues during the past forty years due to its wide applicability. Because of inclusion of memory effect, the fractional derivative has gained its importance while solving problems occurring in engineering and biological sciences. Hence, the theory of fractional integrals and derivatives has recently been extensively applied in numerous engineering and biological situations. Since fractional ordered derivatives have more applications, they provide superior modelling results than integer ordered derivatives [2]. Therefore, fractional-order derivative systems can be used to efficiently study and analyse the qualitative behaviour of many dynamical systems.

Several researchers such as Reich [3], Mateescu [4], Mastorakis [5], Babaei [6], Sadollah et al. [7] and Rastogi et al. [8] have used metaheuristic algorithms to solve a variety of ordinary differential equations. Also, the fractional-order differential equations have been solved using both analytical and numerical techniques by several researchers [9–18].

In this paper, the fractional differential equations (FDE) of the following type has been solved using Fractional sine-cosine series and metaheuristic algorithms:

$$D^{\alpha}y(t) = f(t, y(t)) \ , \quad n-1 < \alpha \leq n \tag{1}$$

where $f$ is a function of $t$ and $y(t)$, and $D^{\alpha}$ stands for the Caputo fractional derivative of order $\alpha$. The Caputo FDE have a value of $\alpha$ between $[0, 1]$ which is present in many biological and physical models [19–22]. The mathematician Leibniz stated about this kind of problem in his letter to L'Hospital in 1695.

It is noted here that the classical optimization methods might have defects for solving real-world problems due to the local optimum, considerable time and challenging implementation, while the metaheuristic approaches are run efficiently and can gain reasonable and satisfactory solutions concerning execution time and precision.

Differential evolution is a population-based optimization algorithm that efficiently explores solution spaces. Known for its simplicity and effectiveness, differential evolution has been successfully applied in various domains, such as engineering design, data mining, and parameter estimation, making it a popular choice for solving optimization problems [23–25].

Particle Swarm Optimization (PSO) is a population-based optimization algorithm inspired by the collective behavior of bird flocking or fish schooling. It has been widely used to solve optimization problems in various domains due to its simplicity and efficiency [26–28].

The structure of our paper is as follows. The basic concepts are discussed in Section 2. The terminologies related to the proposed algorithm is provided in Section 3. In Section 4, the experimental results are given. Section 5 provides the conclusion.

## 2. Basic Concepts

We shall define fractional derivatives and fractional integration in this section.

### *2.1. Riemann-Liouville fractional integer [29]*

The Riemann-Liouville fractional integral operator of order $\alpha$ of a function $f(x) \in C_\mu, \mu \geq -1$ is $J^\alpha f(x) = \frac{1}{\Gamma(\alpha)} \int_0^x (x-t)^{\alpha-1} f(t) \mathrm{dt}, x > 0, J^0 f(x) = f(x)$.

For $f(x) \in C_\mu, \mu \geq -1, \alpha, \beta \geq 0, \gamma \geq -1$, properties of the operator $J^\alpha$

$$J^\alpha J^\beta f(x) = J^\beta J^\alpha f(x), \quad J^\alpha x^\gamma = \frac{\Gamma(\gamma+1)}{\Gamma(\alpha+\gamma+1)} x^{\alpha+\gamma}$$

### *2.2. Derivative caputo fractional [29]*

According to Caputo, the fractional derivative of $f(x)$ is stated as follows:

$$D_x^\alpha f(x) = J^{n-\alpha} D^n f(x) = \frac{1}{\Gamma(n-\alpha)} \int_0^x (x-t)^{n-\alpha-1} f^{(n)}(t) \mathrm{dt}$$

$f(x) \in C_\mu^n, \mu \geq -1, \alpha, \beta \geq 0, \gamma \geq -1, n-1 < \alpha \leq n, n \in N$, properties of operator $D_x^\alpha$

$$D_x^\alpha D_x^\beta f(x) = D_x^{\alpha+\beta} f(x) = D_x^\beta D_x^\alpha f(x),$$

$$D_x^\alpha x^\gamma = \frac{\Gamma(1+\gamma)}{\Gamma(1+\gamma-\alpha)} x^{\gamma-\alpha}, \quad x > 0$$

### *2.3. Fractional Sine-Cosine Series [30]*

#### *2.3.1. Mittag-Leffler Function*

The theory of integer-order differential equations heavily relies on the exponential function $e^z$. Its one-parameter generalization, the function which is now denoted by [31]

$$E_\alpha(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + 1)} \tag{2}$$

was introduced by G.M. Mittag-Leffler [32,33].

Agarwal [34] did, in fact, introduce the two parameter function of Mittag-Leffler type, which is crucial to the fractional calculus and is now famously known as the Mittag-Leffler function. Humbert and Agarwal [35] found a variety of relationships for this function by applying the Laplace transform method.

A two-parameter function of the Mittag-Leffler type is defined by the series expansion [31]

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + \beta)} \qquad (\alpha > 0, \beta > 0) \tag{3}$$

In their research, Plotnikov [36] and Tseytlin [37] utilised two functions, $S_{c_\alpha}(z)$ and $C_{s_\alpha}(z)$, which they refer to as the fractional sine and cosine. These functions are just the particular cases of Mittag-Leffler function in two parameters:

$$S_{c_\alpha}(z) = \sum_{n=0}^{\infty} \frac{(-1)^n z^{(2-\alpha)n+1}}{\Gamma((2-\alpha)n+2)} = zE_{2-\alpha,2}(-z^{2-\alpha}) \tag{4}$$

$$C_{s_\alpha}(z) = \sum_{n=0}^{\infty} \frac{(-1)^n z^{(2-\alpha)n}}{\Gamma((2-\alpha)n+1)} = E_{2-\alpha,1}(-z^{2-\alpha}) \tag{5}$$

The properties of the fractional sine and cosine follows from the properties of the Mittag-Leffler function '(3)'.

## 3. Proposed methodology for solving FDEs

The objective of this work is to illustrate a novel approach to approximately solve a variety of FDEs by using the DE or PSO technique. A variety of fundamental concepts from various disciplines of mathematics, including variational calculus, series expansion, and evolutionary optimization algorithms, are incorporated into the formulation of the suggested method. To implement an efficient problem-solving technique, each of these components would be briefly evaluated before being appropriately merged.

### *3.1. Fractional differential equations*

The main goal of the strategy is to solve a variety of fractional differential equations by using the DE or PSO method with fractional sine-cosine series having unknown coefficients. Instead of providing discrete numerical values at different places in the solution interval, this method gives the solution function.

The general equation for a fractional-order initial value problem can be represented as:

$$D^{\alpha} y(t) = f(t, y(t)), \tag{6}$$

with initial conditions

$$y^k(0) = p_k, \quad n-1 < \alpha \leq n \tag{7}$$

where k = 0,1,...,n-1, $D^\alpha$ represents the fractional derivative of order $\alpha$, t is the independent variable, y(t) is the unknown function, f(t,y(t)) is a given function and $p_k$ is the value of $k^{th}$ derivative of y at t=0. The equation describes a fractional differential equation with initial conditions, where the fractional derivative captures the non-integer order behavior of the system. Solving this type of problem involves finding the function y(t) that satisfies the equation and the given initial conditions.

For formulation purpose, the partial sum of the Fractional sine and cosine series is proposed to be used as the approximation function

$$Y(t) = Y_{2N}(t) = \sum_{n=0}^{N-1} \frac{(-1)^n a_n t^{(2-\alpha)n}}{\Gamma((2-\alpha)n+1)} + \sum_{n=0}^{N-1} \frac{(-1)^n b_n t^{(2-\alpha)n+1}}{\Gamma((2-\alpha)n+2)} \tag{8}$$

where $a_n$ and $b_n$; $n = 0, 1, 2, \ldots, N-1$ are unknown constants that needs to be calculated. This function together with its caputo derivatives will be used to estimate the solution of '(6)'. The total number of terms of fractional sine and cosine used in approximation is denoted by 2N.

It is observed that if we could use the available evolutionary algorithms to handle a higher number of unknown variables, we may obtain greater accuracy. Theoretically, these algorithms can solve problems with any number of variables, but in fact, when dealing with vast numbers of variables, they are unable to discover the global optimum because they are trapped in local optima. Large search spaces also require more time and computational power for the properly examination.

### *3.2. Weighted-residual functional as convergence criterion*

We need a justifying criterion while working with the explicit form of the fractional differential equations to determine whether the method has achieved the desired accuracy in the approximate solution or not. We need a numerical evaluation of errors in order to have a criterion on the validity of the approximate solution. We can rely on the outcomes of the algorithm if this assessing factor is within a reasonable range. It is also necessary to provide a suitable criterion as an objective function to implement on the DE or PSO algorithm. We observe that the idea of weighted-residual functional in the variational calculus would be suitable for all these factors. The weighted residual functional is an integral that is optimized in order to evaluate the solution of the problem [38]. The FDE must be satisfied by the approximate solution $Y_{2N}$ in the form of its residual-integral, which is given as

$$WRF = \int_D |W(t)| \cdot |R(t)| \, dt \tag{9}$$

where W(t) is referred to as the weight function and R(t) is the residual [39], which is obtained in the implicit form of the differential equation '(6)'

$$g(t, y(t), y^\alpha(t)) = f(t, y(t)) - D^\alpha y(t) = 0 \tag{10}$$

by replacing $y(t)$ and $y^\alpha(t)$ with the approximate function Y(t) and its fractional derivatives.

$$R(t) \equiv g\left(t, Y(t), Y^\alpha(t)\right) \tag{11}$$

WRF will be utilised as the objective function, which is being solved numerically using the Trapezoidal or Simpson rule. The weight function W(t) is an arbitrary function used in classical weighted residual methods [39]. But in the proposed method, W(t) is considered to be 1.

In the solution process, one of the following changes might be required to obtain the desired accuracy in the execution of algorithm :

- Increase the number of terms of Fractional sine-cosine series to prevent the evolutionary process from being stuck in local optima.
- Breaking the problem solution interval into smaller subdomains and approximating the solution part by part.

### *3.3. Formulation of initial conditions*

When solving differential equation problems, we must find a solution that satisfies the equation itself and simultaneously fulfill the initial conditions of the equation. Since in the current formulation, differential equations are solved using the evolutionary optimization process, therefore, a suitable approach is needed to take into consideration the $IVs$ as constraints of the optimization problem. The homogeneous conditions are handled in their original implicit form as

$$y\left(t_0\right) = 0 \Rightarrow h_1\left(t_0\right) = \mid y\left(t_0\right) \mid \approx \mid Y\left(t_0\right) \mid \tag{12a}$$

$$y'\left(t_0\right) = 0 \Rightarrow h_2\left(t_0\right) = \mid y'\left(t_0\right) \mid \approx \mid Y'\left(t_0\right) \mid \tag{12b}$$

$\vdots$

$$y^{(n-1)}\left(t_0\right) = 0 \Rightarrow h_{n-1}\left(t_0\right) = \mid y^{(n-1)}\left(t_0\right) \mid \approx \mid Y^{(n-1)}\left(t_0\right) \mid \tag{12c}$$

The constraints of the optimization problem are represented above by $h_1$, $h_2$, $\ldots$, $h_{n-1}$. Then, all the $h_i$'s are included in the framework of a single penalty function explained in the next section.

| | | | | | | |
|---|---|---|---|---|---|---|
| $a_0$ | $b_0$ | $a_1$ | $b_1$ | ... | $a_{N-1}$ | $b_{N-1}$ |

**Figure 1.** The arrangement of the variables in the DE or PSO particles

### *3.4. Penalty Function and Fitness Function*

The DE and PSO algorithms are well known techniques for handling the unconstrained optimization problems. Therefore, one of the known strategies for handling constraints must be used in case of constrained problems. As a result, the constraints covered in the previous section are implemented using the penalty function approach. Penalizing the solutions that fail to satisfy the given criteria is managed by the penalty function. Consequently, we can calculate the fitness function for the problem by adding the value of the penalty function PFV to the weighted-residual integral WRF, i.e.,

$$\mathrm{FFV} = \mathrm{WRF} + \mathrm{PFV} \tag{13}$$

where the penalty scheme given by Rajeev and Krishnamoorthy [40] is used to calculate the value of the penalty function PFV.

$$\mathrm{PFV} = \mathrm{WRF} \cdot \sum_{\mathrm{m=1}}^{\mathrm{n_{IVs}}} \mathrm{K_m h_m} \tag{14}$$

where $n_{IVs}$ is the number of initial conditions, and $h_m$ is the normalised violation of the $\mathrm{m^{th}}$ constraint, which is obtained using equations '(12)' depending on the degree of violation for that current condition. $K_m$ is a penalty multiplier that is selected based on the importance given on satisfying each criterion. The pressure on satisfying this criterion will be great if the constant $K_m$ is selected to be a large number, hence the DE and PSO algorithms attempt to see this constraint rather than the fulfilment of the differential equation itself. On the other hand, if $K_m$ is chosen to have a low value, its corresponding criterion will only be weakly satisfied. As a result, choosing proper values for these coefficients and changing them as the evolutionary process progresses has its own significance, which is beyond the scope of this work. However, the coefficients $K_m$ are all assumed to remain constant throughout all examples considered here in order to keep things simple.

## 4. Illustrative examples

We have talked about the versatility of an algorithm for calculating approximate solution of FDEs up to this point. Here, we will use this approach to look at a number of IVPs in order to assess its applicability and precision. Examples have been chosen from some of the most well-known references [9,17,22,41] in this domain. The effectiveness of the algorithm is further demonstrated by a graphic comparison of the computed

**Table 1.** Parameters of DE algorithm for DE-FSCS method

| Parameters | Ex. 1 | Ex. 2 | Ex. 3 |
|---|---|---|---|
| $X_{min}$ | -1 | -0.8 | -9 |
| $X_{max}$ | 1 | 0.8 | 9 |
| $maxFE$ [a] | 100000 | 100000 | 100000 |
| M [b] | 70 | 70 | 16 |
| CR | 0.9 | 0.9 | 0.5 |
| F | 0.65 | 0.65 | 0.75 |
| $It_{max}$ [c] | 500 | 20 | 2000 |
| $K_1$ [d] | 0.9 | $10^{40}$ | 100000 |

[a] Maximum Function Evaluations

[b] Population Size

[c] Maximum Iterations

[d] Penalty Multipliers

approximations with the exact ones.

In order to analyze and compare the numerical results thoroughly, we mainly consider Mean Square Error (MSE) between exact solutions and approximate solutions as the most crucial evaluation criterion in this work, which is calculated by using the following expression:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} \left\| Y_N(x_i) - Y_{\text{exact}}(x_i) \right\|^2 \tag{15}$$

The mean and variance values, which demonstrate the average precision and stability of these comparing methods, constitute the MSE values.

20 separate runs were performed for each test problem in order to complete the optimization task. The proposed method was performed on an Intel(R) Core(TM) i3 processor running at 1.70 GHz with 4 GB of RAM using the MATLAB programming software (MATLAB 2021).

The chosen values for parameters of DE algorithm is mentioned in Table 1.

The chosen values for parameters of PSO algorithm is mentioned in Table 2.

**Table 2.** Parameters of PSO algorithm for PSO-FSCS method

| Parameters | Ex. 1 | Ex. 2 | Ex. 3 |
|---|---|---|---|
| $X_{min}$ | -1 | -1 | -4.2 |
| $X_{max}$ | 1 | 1 | 4.2 |
| $maxFE$ [a] | 100000 | 100000 | 100000 |
| M [b] | 70 | 14 | 16 |
| $c_1$ | 0.5 | 0.5 | 1 |
| $c_2$ | 1.5 | 1.5 | 1 |
| $\omega_{min}$ | 0.4 | 0.4 | 0.3 |
| $\omega_{max}$ | 0.9 | 0.9 | 0.8 |
| $It_{max}$ [c] | 100 | 100 | 200 |
| $K_1$ [d] | 0.9 | 1000 | 100000 |

[a] Maximum Function Evaluations

[b] Population Size

[c] Maximum Iterations

[d] Penalty Multipliers

### *4.1. Example 1 [22]*

Consider the following fractional differential equation

$$D^{\alpha}y(t) + y(t) = t^2 + \frac{2t^{1.5}}{\Gamma(2.5)},\ \ 0 \leq t \leq 1,\ \ 0 < \alpha \leq 1 \tag{16}$$

$$y(0) = 0 \tag{17}$$

The exact solution for '(16)' and '(17)' is

$$y_{\text{exact}}\ (t) = t^2\ when\ \alpha = 0.5 \tag{18}$$

The best approximate solution to this problem using Differential Evolution method and Fractional sine-cosine series (DE-FSCS) is obtained for N = 7. Thus, we get the approximate solution as

$$\begin{aligned} y(t) \approx\ & 1.08 \times 10^{-3} - 0.1751\ t + 0.7523\ t^{1.5} + 0.3009\ t^{2.5} + 0.1667\ t^3 \\ & -2.061 \times 10^{-2}\ t^4 - 1.910 \times 10^{-2}\ t^{4.5} - 3.474 \times 10^{-3}\ t^{5.5} \\ & -1.389 \times 10^{-3}\ t^6 - 1.984 \times 10^{-4}\ t^7 - 7.125 \times 10^{-5}\ t^{7.5} \\ & -8.381 \times 10^{-6}\ t^{8.5} - 2.752 \times 10^{-6}\ t^9 - 2.756 \times 10^{-7}\ t^{10} \end{aligned}$$

The best approximate solution to this problem using Particle Swarm Optimization method and Fractional sine-cosine series (PSO-FSCS) is obtained for N = 7. Thus, we get the approximate solution as

$$\begin{aligned} y(t) \approx\ & 6.92 \times 10^{-4} - 0.1748\, t + 0.7523\, t^{1.5} + 0.3009\, t^{2.5} + 0.1667\, t^{3} \\ & -2.076 \times 10^{-2}\, t^{4} - 1.91 \times 10^{-2}\, t^{4.5} - 3.474 \times 10^{-3}\, t^{5.5} \\ & -1.389 \times 10^{-3}\, t^{6} - 1.984 \times 10^{-4}\, t^{7} - 7.125 \times 10^{-5}\, t^{7.5} \\ & -2.086 \times 10^{-6}\, t^{8.5} - 2.358 \times 10^{-6}\, t^{9} - 1.315 \times 10^{-7}\, t^{10} \end{aligned}$$

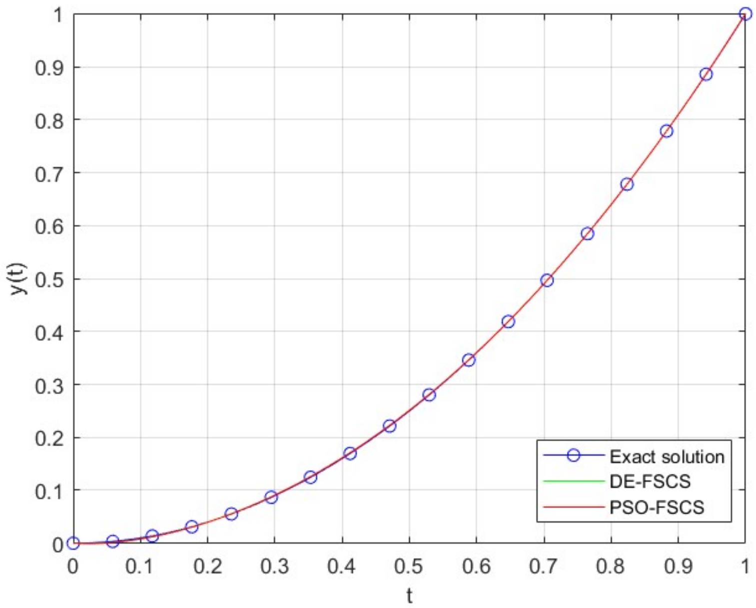


**Figure 2.** Comparison of the exact and approximate solutions of Example 1 using DE-FSCS and PSO-FSCS methods

The graphical comparison between the exact and approximate solutions of Example 1 using the DE-FSCS and PSO-FSCS methods is depicted in Fig. 2. It is clear from the graph that both the DE-FSCS and PSO-FSCS methods detected the exact solution with adequate precision.

**Table 3.** Comparison table for MSE of the solution of Example 1 obtained by the proposed DE-FSCS and PSO-FSCS methods with that of other techniques [17]

| Technique | (MSE) |
|---|---|
| DE-FSCS | $9.1299 \times 10^{-7}$ |
| PSO-FSCS | $8.5185 \times 10^{-7}$ |
| VIM [17] | 2.6283 |
| GWO-VIM [17] | $2.600 \times 10^{-3}$ |

### *4.2. Example 2 [41]*

Consider the nonlinear Riccati differential equation

$$D^{\alpha}y(t) + y^2(t) = 1,\ 0 \le t \le 1,\ 0 < \alpha \le 1 \tag{19}$$

$$y(0) = 0 \tag{20}$$

The exact solution for '(19)' and '(20)' is

$$y_{\text{exact}}(t) = \frac{e^{2t}-1}{e^{2t}+1}\ when\ \alpha = 0.75 \tag{21}$$

The best approximate solution to this problem using Differential Evolution method and Fractional sine-cosine series (DE-FSCS) is obtained for N = 7. Thus, we get the approximate solution as

$$y(t) \approx\ 0.8000\ t + 0.4174\ t^{1.25} - 0.3138\ t^{2.25} + 0.2407\ t^{2.5}$$
$$+3.1380 \times 10^{-2}\ t^{3.5} + 3.9190 \times 10^{-2}\ t^{3.75} - 3.4855 \times 10^{-3}\ t^{4.75}$$
$$+6.2467 \times 10^{-3}\ t^{5} - 5.1681 \times 10^{-4}\ t^{6} - 4.9550 \times 10^{-4}\ t^{6.25} - 6.7062 \times 10^{-6}\ t^{8.5}$$

The best approximate solution to this problem using Particle Swarm Optimization method and Fractional sine-cosine series (PSO-FSCS) is obtained for N = 7. Thus, we get the approximate solution as

$$y(t) \approx -1.167 \times 10^{-8} + t + 0.2801\, t^{1.25} - 0.3922\, t^{2.25} - 0.3009\, t^{2.5}$$
$$+8.597 \times 10^{-2}\, t^{3.5} + 6.029 \times 10^{-2}\, t^{3.75} + 1.269 \times 10^{-2}\, t^{4.75}$$
$$+5.726 \times 10^{-3}\, t^{5} - 1.389 \times 10^{-3}\, t^{6} + 8.655 \times 10^{-4}\, t^{6.25}$$
$$-1.194 \times 10^{-4}\, t^{7.25} + 3.367 \times 10^{-5}\, t^{7.5} - 8.383 \times 10^{-6}\, t^{8.5}$$

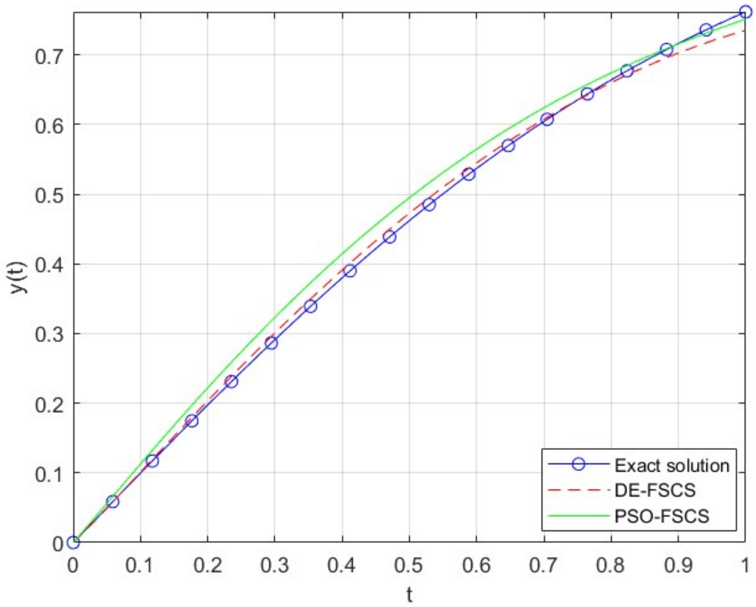


**Figure 3.** Comparison of the exact and approximate solutions of Example 2 using DE-FSCS and PSO-FSCS methods

The graphical comparison between the exact and approximate solutions of Example 2 using the DE-FSCS and PSO-FSCS methods is depicted in Fig. 3. It is seen from the graph that both the DE-FSCS and PSO-FSCS methods detected the exact solution with adequate precision, but the approximate solution obtained by DE-FSCS method in this case is better than that obtained by PSO-FSCS method as far as accuracy is concerned.

**Table 4.** Comparison table for MSE of the solution of Example 2 obtained by the proposed DE-FSCS and PSO-FSCS methods with that of other techniques [17]

| Technique | (MSE) |
|---|---|
| DE-FSCS | $9.3219 \times 10^{-5}$ |
| PSO-FSCS | $5.3229 \times 10^{-4}$ |
| VIM [17] | $4.3000 \times 10^{-3}$ |
| GWO-VIM [17] | $1.3000 \times 10^{-3}$ |

### *4.3. Example 3 [9]*

Consider the example of nonlinear fractional differential equation which is used to solve an initial value problem describing the process of cooling of a semi-infinite body by radiation

$$D^{\alpha}y(t) - \gamma(x_0 - y(t))^4 = 1,\ 0 \le t \le 1,\ 0 < \alpha \le 1 \tag{22}$$

$$y(0) = 0 \tag{23}$$

The exact solution for '(22)' and '(23)' is

$$y_{\text{exact}}(t) = x_0 - (\frac{x_0^3\sqrt{\pi}}{6\sqrt{t}+\sqrt{\pi}})^{1/3}\ when\ \alpha = 0.5 \tag{24}$$

For $\gamma = 1$ and $x_0 = 1$, the best approximate solution to this problem using Differential Evolution method and Fractional sine-cosine series (DE-FSCS) is obtained for N = 8. Thus, we get the approximate solution as

$$\begin{aligned} y(t) \approx\ & 4.307 \times 10^{-7} + 2.6122\ t - 3.9767\ t^{1.5} + 2.5804\ t^{2.5} - 0.5755\ t^3 \\ & -0.1525\ t^4 - 0.1719\ t^{4.5} - 2.545 \times 10^{-2}\ t^{5.5} + 1.25 \times 10^{-2}\ t^6 \\ & -1.786 \times 10^{-3}\ t^7 + 3.584 \times 10^{-4}\ t^{7.5} - 5.228 \times 10^{-5}\ t^{8.5} \\ & -7.548 \times 10^{-6}\ t^9 - 2.48 \times 10^{-6}\ t^{10} - 7.842 \times 10^{-8}\ t^{10.5} + 6.577 \times 10^{-8}\ t^{11.5} \end{aligned}$$

For $\gamma = 1$ and $x_0 = 1$, the best approximate solution to this problem using Particle Swarm Optimization method and Fractional sine-cosine series (PSO-FSCS) is obtained

**Table 5.** Comparison table for MSE of the solution of Example 3 obtained by the proposed DE-FSCS and PSO-FSCS methods with that of other numerical technique [30]

| Technique | (MSE) |
|---|---|
| DE-FSCS | $4.7275 \times 10^{-3}$ |
| PSO-FSCS | $5.0048 \times 10^{-3}$ |
| Numerical method mentioned by [30] for h = 0.05 | $1.0827 \times 10^{-2}$ |

for N = 8. Thus, we get the approximate solution as

$$y(t) \approx -2.066 \times 10^{-12} + 2.002\, t - 2.3669\, t^{1.5} + 0.3903\, t^{2.5} + 0.2769\, t^{3}$$
$$+8.88 \times 10^{-2}\, t^{4} - 6.078 \times 10^{-2}\, t^{4.5} + 3.07 \times 10^{-3}\, t^{5.5} - 1.212 \times 10^{-5}\, t^{6}$$
$$-8.282 \times 10^{-4}\, t^{7} - 4.137 \times 10^{-6}\, t^{7.5} + 1.734 \times 10^{-5}\, t^{8.5} + 1.613 \times 10^{-6}\, t^{9}$$
$$-7.244 \times 10^{-7}\, t^{10} - 3.599 \times 10^{-8}\, t^{10.5} - 2.315 \times 10^{-8}\, t^{11.5}$$

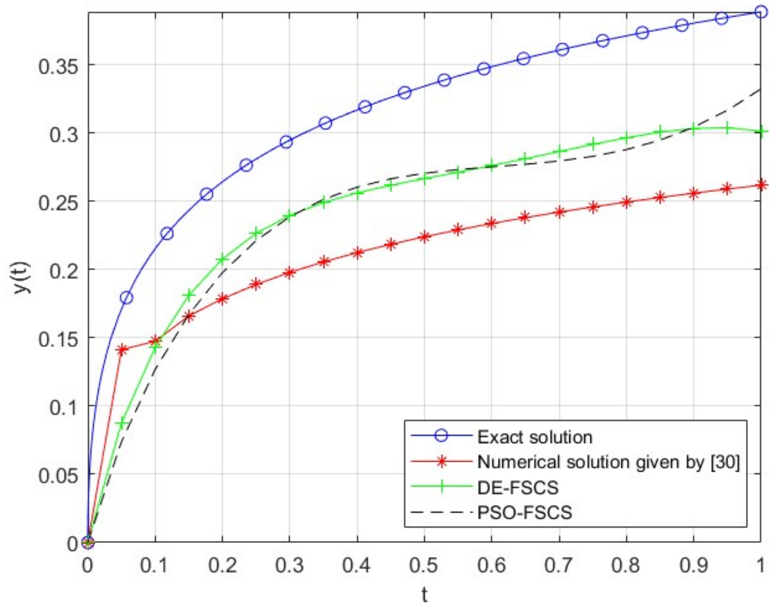


**Figure 4.** Comparison of the exact and approximate solutions of Example 3 using DE-FSCS and PSO-FSCS methods

The graphical comparison between the exact and approximate solutions of Example 3 using the DE-FSCS and PSO-FSCS methods is depicted in Fig. 4. It is seen that both the DE-FSCS and PSO-FSCS methods outperformed the other method in terms of accuracy. Also, the approximate solution obtained by DE-FSCS method in this case is better than that obtained by PSO-FSCS method as far as accuracy is concerned.

**Table 6.** Mean Function Evaluations (Mean FE) for the solutions to all the three examples obtained by the proposed DE-FSCS and PSO-FSCS methods

| Mean FE | Ex. 1 | Ex. 2 | Ex. 3 |
|---|---|---|---|
| $DE-FSCS$ | 2849 | 273 | 999.2 |
| $PSO-FSCS$ | 3433.5 | 538.3 | 1230.4 |

## 5. Conclusion

In this paper, fractional differential equations are solved numerically to get their approximate solutions. For this, a fractional sine-cosine series with unknown coefficients is taken as base approximation function and then these coefficients are evaluated using metaheuristic optimization techniques such as Differential Evolution (DE) and Particle Swarm Optimization (PSO) in order to achieve desired precision. Three problems are considered as examples to illustrate our proposed method. The MSE of the approximate solutions for Ex. 1 derived from both DE-FSCS and PSO-FSCS methods w.r.t. exact solutions are superior to that of VIM and GWO-VIM methods which were developed by Entesar and Qasim [17]. (See Fig. 2 and Table 3). Further, it may be noted that our PSO-FSCS method for this example is even better than that of our DE-FSCS method.

It is further noted that the MSE of the approximate solutions for Ex. 2 obtained by both DE-FSCS and PSO-FSCS methods w.r.t. exact solutions are better than the MSE of the solutions derived from VIM and GWO-VIM methods which were suggested by Entesar and Qasim [17]. (See Fig. 3 and Table 4). Here, it may be noted that the DE-FSCS method for this example is still better than the PSO-FSCS method.

The MSE for the approximate solutions for Ex. 3 obtained by both DE-FSCS and PSO-FSCS methods w.r.t. exact solutions are better than that of the MSE of the solution obtained by the numerical method as suggested by Podlubny [30]. (See Fig. 4 and Table 5). Further, it is noted that the DE-FSCS method in this case is even better than the PSO-FSCS method.

It is seen from Table 6 that the mean function evaluations (Mean FE) for DE-FSCS method is less than that of PSO-FSCS method in case of all the three examples 1, 2 and 3 considered in this paper. It can be concluded here that the DE-FSCS method is more efficient than PSO-FSCS method as far as mean FE is concerned.

After observing the outcomes of the proposed method in this paper, we may conclude that the performance of both the DE-FSCS and PSO-FSCS methods are much better than the methods suggested earlier by Entesar and Qasim [17] and Podlubny [30].

**Disclosure Statement** The authors report there are no competing interests to declare.

## References


[1] Miller KS, Ross B. An Introduction to The Fractional Calculus and Fractional Differential Equations, John-Wily and Sons. Inc. New York. 1993.

[2] Zhou P, Kuang F. A novel control method for integer orders chaos systems via fractional-order derivative. Discret Dyn Nat Soc. 2011;2011:1–8.

[3] Reich C. Simulation of imprecise ordinary differential equations using evolutionary algorithms. In: Proceedings of the ACM Symposium on Applied Computing. 2000. p. 428–32.

[4] Mateescu GD. On the application of genetic algorithms to differential equations. Rom J Econ Forecast. 2006;7(2):5–9.

[5] Mastorakis EN. Unstable ordinary differential equations: solution via genetic algorithms and the method of Nelder-Mead. Proceedings of the 6th WSEAS Int. Conf. on Systems Theory & Scientific Computation, 119, Elounda, Greece. 2007. p. 297-354.

[6] Babaei M. A general approach to approximate solutions of nonlinear differential equations using particle swarm optimization. Appl Soft Comput J. 2013;13(7):3354–65.

[7] Sadollah A, Eskandar H, Yoo DG, Kim JH. Approximate solving of nonlinear ordinary differential equations using least square weight function and metaheuristic algorithms. Eng Appl Artif Intell. 2015;40:117–32.

[8] Rastogi R, Misra OP, Mishra R. A Chebyshev polynomial approach to approximate solution of differential equations using differential evolution. Eng Appl Artif Intell. 2023;126:107197.

[9] Demirci E, Ozalp N. A method for solving differential equations of fractional order. J Comput Appl Math. 2012;236(11):2754–62.

[10] Ali KK, Abd El Salam MA, Mohamed EMH, Samet B, Kumar S, Osman MS. Numerical solution for generalized nonlinear fractional integro-differential equations with linear

functional arguments using Chebyshev series. Adv Differ Equations. 2020;2020(1):1–23.

[11] Bukhari AH, Raja MAZ, Sulaiman M, Islam S, Shoaib M, Kumam P. Fractional neuro-sequential ARFIMA-LSTM for financial market forecasting. IEEE Access. 2020;8:71326–38.

[12] Waseem W, Sulaiman M, Alhindi A, Alhakami H. A Soft Computing Approach Based on Fractional Order DPSO Algorithm Designed to Solve the Corneal Model for Eye Surgery. IEEE Access. 2020;8:61576–92.

[13] Kumar S, Kumar R, Osman MS, Samet B. A wavelet based numerical scheme for fractional order SEIR epidemic of measles by using Genocchi polynomials. Numer Methods Partial Differ Equ. 2021;37(2):1250–68.

[14] Cuahutenango-Barro B, Taneco-Hernández MA, Lv YP, Gómez-Aguilar JF, Osman MS, Jahanshahi H, et al. Analytical solutions of fractional wave equation with memory effect using the fractional derivative with exponential kernel. Results Phys. 2021;25:104148.

[15] Djennadi S, Shawagfeh N, Inc M, Osman MS, Gómez-Aguilar JF, Abu Arqub O. The Tikhonov regularization method for the inverse source problem of time fractional heat equation in the view of ABC-fractional technique. Phys Scr. 2021;96(9):094006.

[16] Ahmad A, Sulaiman M, Kumam P. Solutions of fractional order differential equations modeling temperature distribution in convective straight fins design. Adv Differ Equations. 2021;2021(1):1–38.

[17] Entesar A, Qasim OS. Solve fractional differential equations via a hybrid method between variational iteration method and gray wolf optimization algorithm. Asian-European J Math. 2021;14(8):2150144.

[18] Ebrahimzadeh A, Khanduzi R, A Beik SP, Baleanu D. Research on a collocation approach and three metaheuristic techniques based on MVO, MFO, and WOA for optimal control of fractional differential equation. Journal Vib Control. 2023;29(3–4):661–74.

[19] Ahmed E, El-Sayed AMA, El-Saka HAA. Equilibrium points, stability and numerical solutions of fractional-order predator-prey and rabies models. J Math Anal Appl. 2007;325(1):542–53.

[20] Al-Talib ZS, Al-Azzawi SF. Projective and hybrid projective synchronization of 4-D hyperchaotic system via nonlinear controller strategy. Telkomnika (Telecommunication Comput Electron Control). 2020;18(2):1012–20.

[21] Glöckle WG, Nonnenmacher TF. A fractional calculus approach to self-similar protein dynamics. Biophys J. 1995;68(1):46–53.

[22] Kilbas A, Srivastava H, Trujillo J. Theory and applications of fractional differential equations. J Electrochem Soc. 2006;129(204).

[23] Sharma H, Bansal JC, Arya K V. Fitness based Differential Evolution. Memetic Comput. 2012;4(4):303–16.

[24] Sharma H, Bansal JC, Arya K V. Self balanced differential evolution. J Comput Sci. 2014;5(2):312–23.

[25] Praczyk T. Using Hill Climb Modular Assembler Encoding and Differential Evolution to evolve modular neuro-controllers of an autonomous underwater vehicle acting as a Magnetic Anomaly Detector. Appl Soft Comput. 2022;127:109347.

[26] Clerc M, Kennedy J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Trans Evol Comput. 2002;6(1):58–73.

[27] Bansal JC, Deep K. A modified binary particle swarm optimization for Knapsack problems. Appl Math Comput. 2012;218(22):11042–61.

[28] Papadimitrakis M, Kapnopoulos A, Tsavartzidis S, Alexandridis A. A cooperative PSO algorithm for Volt-VAR optimization in smart distribution grids. Electr Power Syst Res. 2022;212:108618.

[29] El-Ajou A, Odibat Z, Momani S, Alawneh A. Construction of analytical solutions to fractional differential equations using homotopy analysis method. IAENG Int J Appl Math. 2010;40(2):1–9.

[30] Podlubny I. Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications. In Mathematics in science and engineering. 1998;198.

[31] Erdélyi A. Higher transcendental functions, McGraw - Hill. New York. 1955;3.

[32] Mittag-Leffler GM. Sur la nouvelle fonction E$\alpha$ (x), CR Acad. Sci. Paris, 1903;137(2):554-58.

[33] Mittag-Leffler Gm. Sopra la funzione E$\alpha$ (x), Rend. Accad. Lincei, 1904;13:3-5.

[34] Agarwal RP. A propos d'une note de M. Pierre Humbert, CR Acad. Sci. Paris, 1953;236(21):2031-32.

[35] Humbert P, Agarwal RP. Sur la fonction de Mittag-Leffler et quelques-unes de ses généralisations, Bull. Sci. Math. 1953;77(2):180-85.

[36] Plotnikov YI. Steady-state vibrations of plane and axesymmetric stamps on a viscoelastic foundation, Ph.D. thesis, Moscow (in Russian). 1979.

[37] Tseytlin AI. Applied Methods of Solution of Boundary Value Problems in Civil Engineer-

ing Stroyizdat, Moscow (in Russian). 1984.

[38] Bathe KJ. Finite Element Procedures. Prentice Hall, New Jersey. 1996.

[39] Reddy JN. Introduction to the finite element method, McGraw-Hill Education. 2019.

[40] Rajeev S, Krishnamoorthy CS. Discrete Optimization of Structures Using Genetic Algorithms. J Struct Eng. 1992;118(5):1233–50.

[41] Pradhan M, Roy PK, Pal T. Grey wolf optimization applied to economic load dispatch problems. Int J Electr Power Energy Syst. 2016;83:325–34.