Bol. Soc. Paran. Mat.
©SPM – E-ISSN-2175-1188
SPM: www.spm.uem.br/bspm

(3s.) **v. 2026 (44) 18** : 1–10.
ISSN-0037-8712
doi:10.5269/bspm.82696

Hybrid Physics-Informed Neural Networks Methodology for Solving Nonlinear Space-Fractional Reaction-Diffusion Models

Imane Hariri, Atika Radid and Karim Rhofir

ABSTRACT: Studying a Physics-Informed Neural Networks (PINNs) methodology to solve one-dimensional nonlinear fractional reaction-diffusion equations in space is the goal of our paper. PINNs are developed as global-in-time solvers by embedding the governing fractional partial differential equations (FPDE), initial and boundary conditions directly into the loss function. The fractional diffusion operator is incorporated into the PINN framework through a discrete spectral representation, while temporal derivatives are obtained via automatic differentiation. We formulate a framework and validate the efficiency and accuracy by solving three reaction-diffusion problems. We employed a high-order exponential time-differencing Runge-Kutta method of fourth order (ETDRK4) combined with a spectral discretization of the fractional Laplacian to compute accurate reference solutions. The obtained results demonstrate that the proposed PINN framework accurately captures the solution dynamics showing high agreement with exact and numerical solutions.

Keywords: Physics-informed neural networks, deep learning, reaction-diffusion models, ETDRK4, spectral methods, fractional laplacian, fractional calculus.

Contents

1. Introduction

Recently, fractional calculus has become increasingly popular in the scientific community because of its extensive applications in science and engineering [1]. The behavior of fluid flow in porous substances, theory of controlling dynamic systems, processing of signals, anomalous diffusion transport and viscoelasticity are all phenomenas described using fractional differential equations (FDEs). Obtaining analytical solutions for fractional order differential equations is often challenging, necessitating the development of reliable and efficient techniques to address FDEs.

Over the years, there have been advancements in numerical, analytical and algebraic methods for solving such equations. These methods include finite difference method (FDM) [2,3], finite element method (FEM) [4,5], Laplace transforms [6], variational iteration methods (VIM) [7] and Adomian decomposition methods (ADM) [8,9], spectral methods [10,11,12], Wavelet methods [13,14].

In recent years, machine learning methods specially physics-informed neural networks (PINNs) [15] have been proven to be a powerful tool in solving partial differential problems [16,17,18] including fractional differential equations. It has been noticed that in many scenarios, traditional numerical methods can be computationally expensive and difficult to implement for complex problems, especially in the case of high-dimensional systems. PINNs are therefore a suitable option for solving nonlinear fractional PDEs.

2020 *Mathematics Subject Classification:* 35R11, 68T07, 35K57.
Submitted March 14, 2026. Published June 19, 2026.

Moreover, PINNs are capable of dealing with large amounts of data. For example the data coming from different sensors that measure the boundary and initial conditions of a FDEs. Many researchers have used PINNs to find the solutions of different types of FDEs. For example, in [19] the authors have introduced fractional physics-informed neural networks (fPINNs) for solving spatio-temporal fractional advection-diffusion equations where they integrates deep neural networks (DNNs) with a fractional Laplacian interpolation approximation and defined the fractional Laplacian according to the directional represtentation. In [20], the authors employed adaptive fractional PINNs for solving three-dimensional forward and inverse anomalous heat conduction problems in graded materials. In [21] the authors have presented a framework based on fractional PINNs to solve time-fractional PDEs such as Cahn-Hilliard and Allen-Cahn equations using the L1 scheme as an approximation of the Caputo derivative while proposing three improved optimization strategies. The authors in [22] have proposed a physical model-driven neural network (PMNN) framework to solve time-fractional differential equations where they discretize the Caputo derivative using the L1 approximation. In [23] the authors introduces a neural networks approach using Legendre polynomials to address space and time fractional diffusion equations, while in [24] the authors introduced an innovative Hermite neural network solver by integrating the fractional Richards model with a fractional PINN and applied it to unsaturated flow problems.

The goal of this paper is to propose and analyze a physics-informed neural network framework for space-fractional reaction-diffusion equations that incorporates a discrete spectral representation of the fractional Laplacian and integrates the physical constraints directly into the loss function of the model. Therefore the rest of the paper is organized as follows: In section 2 we will describe the proposed methodology. In section 3, we will test the efficiency of the framework in solving three nonlinear fractional differential problems. Finally, section 4 includes the conclusions.

2. Description of the Method

2.1. Deep Neural Networks

Deep neural networks (DNNs) are parametric functions composed of multiple hidden layers where each layer consist of linear transformations followed by nonlinear activation functions. In particular, fully connected feed-forward neural networks (FCNNs) are neural networks (NNs) in which each neuron in a layer is connected to all neurons in the subsequent layer. It has been demonstrated that DNNs are universal function approximators [25] which make them well suited for approximating smooth solutions of partial differential equations. Let $(x,t) \in \mathbb{R}^2$ denote the input variables. The neural network approximation of the solution $\Phi(x,t)$ is given by

$$\Phi_\theta(x,t) = \mathcal{N}_\theta(x,t), \tag{2.1}$$

where $\mathcal{N}_\theta$ denotes the neural network with trainable parameters θ (weights and biases). Hyperbolic tangent activation functions are used in the hidden layers to ensure smoothness and stable automatic differentiation.

2.2. Physics-Informed Neural Networks

Physics-Informed Neural Networks (PINNs) are NNs that incorporate the governing physical laws directly into the training process by embedding the differential equation into the loss function. Instead of relying on labeled data, PINNs enforce the PDE, initial conditions and boundary conditions in a soft-constrained manner. We will focus in this paper on solving one-dimensional nonlinear space-fractional reaction-diffusion equations of the form:

$$\frac{\partial \Phi(x,t)}{\partial t} = -k_\alpha(-\Delta)^{\alpha/2}\Phi(x,t) + R(x,t,\Phi(x,t)), \quad (x,t) \in (0,L) \times (0,T], \tag{2.2}$$

subject to the initial condition

$$\Phi(x,0) = \Phi_0(x), \quad x \in (0,L), \tag{2.3}$$

and homogeneous Dirichlet boundary conditions. Here, $k_\alpha > 0$ denotes the diffusion coefficient, $\alpha \in (1,2]$ is the fractional order, $(-\Delta)^{\alpha/2}$ is the fractional Laplacian operator, and $R(x,t,u)$ represents a nonlinear reaction term. In this case, the PINN output $\Phi_\theta(x,t)$ must satisfy the PDE residual given by:

$$\mathcal{R}(x,t;\theta) = \frac{\partial \Phi_\theta}{\partial t}(x,t) + k_\alpha(-\Delta)^{\alpha/2}\Phi_\theta(x,t) - R(x,t,\Phi_\theta(x,t)). \tag{2.4}$$

We define the PINN total loss function as:

$$\mathcal{L}_{total}(\theta) = \omega_{res}\mathcal{L}_{res}(\theta) + \omega_{ic}\mathcal{L}_{ic}(\theta) + \omega_{bc}\mathcal{L}_{bc}(\theta), \tag{2.5}$$

where each loss component is given by:

$$\mathcal{L}_{res}(\theta) = \frac{1}{N_{res}} \sum_{i=1}^{N_{res}} \left|\mathcal{R}(x_i,t_i;\theta)\right|^2, \tag{2.6}$$

$$\mathcal{L}_{bc}(\theta) = \frac{1}{N_b} \sum_{j=1}^{N_b} \left(|\Phi_\theta(0,t_j)|^2 + |\Phi_\theta(L,t_j)|^2\right), \tag{2.7}$$

$$\mathcal{L}_{ic}(\theta) = \frac{1}{N_i} \sum_{k=1}^{N_i} \left|\Phi_\theta(x_k,0) - \Phi_0(x_k)\right|^2. \tag{2.8}$$

Here, N_{res}, N_b and N_i denote the number of collocation, boundary and initial points, respectively, and ω_{res}, ω_{bc} and ω_{ic} are penalty weights. The time derivative $\partial\Phi_\theta/\partial t$ is computed exactly using automatic differentiation, while the space-fractional Laplacian is approximated numerically, as described below.

2.3. Discretization of the Fractional Laplacian Operator

To handle the nonlocal fractional Laplacian operator, we discretize the spatial domain using N interior grid points and approximate $(-\Delta)^{\alpha/2}$ via a spectral method based on the sine basis, which naturally satisfies homogeneous Dirichlet boundary conditions. Let $\{\phi_k(x) = \sin(k\pi x/L)\}_{k=1}^N$ denote the eigenfunctions of the Laplacian operator. The corresponding eigenvalues of the fractional Laplacian are

$$\lambda_k = \frac{2}{h^2}\left(1 - \cos\left(\frac{k\pi}{N+1}\right)\right), \tag{2.9}$$

with $h = L/(N+1)$. Using this basis, the fractional Laplacian operator can be approximated in matrix form as

$$(-\Delta)^{\alpha/2}\mathbf{\Phi} \approx \mathbf{A}\mathbf{\Phi}, \tag{2.10}$$

where $\mathbf{A} = \mathbf{S}\,\mathrm{diag}(\lambda_k)\,\mathbf{S}^{-1}$ and $\mathbf{S} \in \mathbb{R}^{N\times N}$ is the discrete sine transform (DST) matrix given by:

$$S_{ik} = \sin\left(\frac{ik\pi}{N+1}\right). \tag{2.11}$$

This approach allows efficient and accurate evaluation of the fractional operator inside the PINN framework. Algorithm 1 outlines the main steps of the proposed PINN methodology.

Algorithm 1 PINN for space-fractional reaction-diffusion FDEs

```
Define the neural network architecture and activation functions
Generate collocation points, boundary points and initial points
Construct the fractional Laplacian matrix A
Define the optimizer and the learning rate
for each training iteration do
    Evaluate Φ_θ(x, t)
    Compute ∂Φ_θ/∂t using automatic differentiation
    Evaluate (−Δ)^{α/2} Φ_θ ≈ A Φ_θ
    Compute residual, boundary condition and initial condition losses
    Update the network parameters θ using the optimizer
end for
Return neural network prediction Φ_θ(x, t)
```

3. Numerical Application

3.1. Exponential Time Differencing Runge-Kutta Method (ETDRK4)

To validate the accuracy of the PINN solution, a reference numerical solution is computed using the fourth-order Exponential Time Differencing Runge-Kutta (ETDRK4) method. After spatial discretization of the Laplacian operator using the sine spectral method, the PDE (2.2) can be written in semi-discrete form as

$$\frac{d\mathbf{\Phi}}{dt} = \mathbf{L}\mathbf{\Phi} + \mathbf{R}(\mathbf{\Phi}), \tag{3.1}$$

where $\mathbf{L} = -k_\alpha \operatorname{diag}(\lambda_k)$ represents the linear fractional diffusion operator in spectral space and $\mathbf{R}(\mathbf{\Phi})$ corresponds to the nonlinear reaction term. The ETDRK4 scheme integrates the linear part exactly and treats the nonlinear term explicitly using a fourth-order Runge-Kutta formulation. Let Δt denote the time step. The update formula reads

$$\mathbf{\Phi}^{n+1} = e^{\mathbf{L}\Delta t}\mathbf{\Phi}^n + \Delta t \sum_{i=1}^{4} b_i \mathbf{R}_i, \tag{3.2}$$

where the coefficients b_i depend on exponential φ-functions and $\mathbf{N}_i$ denote evaluations of the nonlinear term at intermediate stages. This method is particularly well suited for stiff fractional diffusion operators, providing high accuracy and stability.

3.2. Test Problems

In this section, we will investigate the performance of the proposed framework by solving three one-dimensional nonlinear space-fractional reaction-diffusion problems. For validation purposes, the obtained results will be compared with reference solution computed using the ETDRK4 method as well as analytical ones.

Problem 1 *We consider the following reaction-diffusion equation:*

$$\frac{\partial \Phi}{\partial t} = -k_\alpha(-\Delta)^{\frac{\alpha}{2}}\Phi + f(x,t,\Phi), \quad (x,t) \in [0,1]^2, \alpha \in]1,2], \tag{3.3}$$

where the source function is:

$$f(x,t,\Phi) = \frac{k_\alpha}{4} t^\alpha \left(3[1+(2\pi)^\alpha]\sin(2\pi x) - [1+(6\pi)^\alpha]\sin(6\pi x)\right) + \alpha t^{\alpha-1}\sin^3(2\pi x) - k_\alpha \Phi,$$

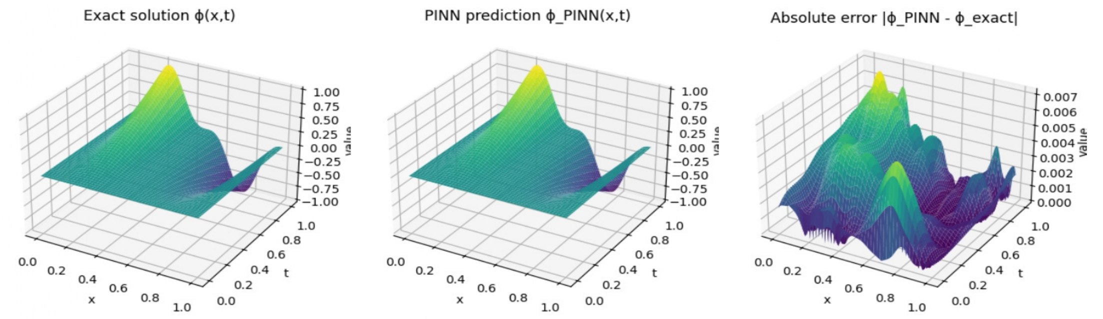
with $k_\alpha > 0$.
The initial and boundary conditions are given by:

$$\begin{cases} \Phi(x,0) &= 0, \\ \Phi(0,t) &= \Phi(1,t) = 0, \qquad t > 0. \end{cases}$$

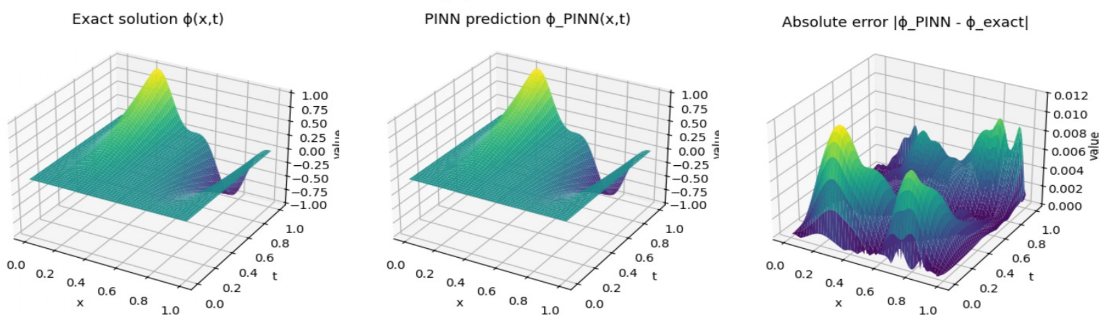
The exact solution of this equation is given by:

$$\Phi(x,t) = t^\alpha \sin^3(2\pi x).$$

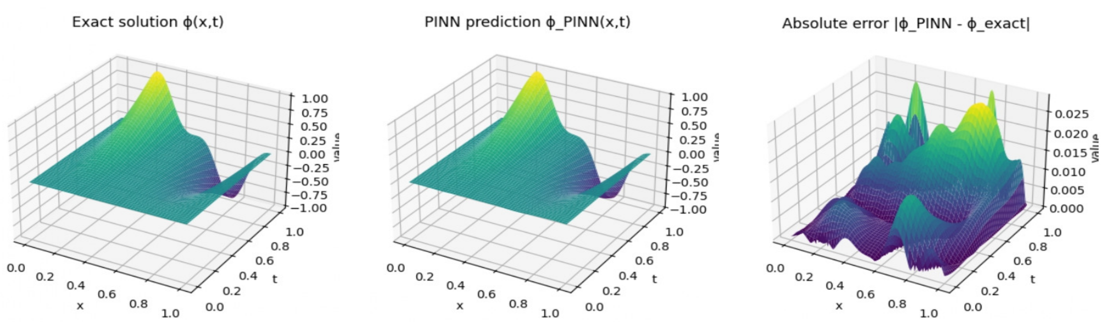
We choose for this example $k_\alpha = 1$. We considered a FCNN with three hidden layers, each with 64 neurons and tanh *activation functions. The number of training (collocation) points, initial condition points and boundary points are $N_{res} = 6400$, $N_i = 128$ and $N_b = 100$ respectively. The training was performed using the following weights $w_{res} = 10^{-2}$ and $w_{ic} = w_{bc} = 10$. We trained our model for 50k iterations using Adam optimizer. The training took about 5 minutes. The exact and predicted solutions along with the absolute error for different values of α are presented in Figure 1. It can be seen from the figure that our network performed well. In Table 1, we calculated the absolute error at some points of the domain for $\alpha = 1.6$. It can be observed from the tables that the proposed framework produced accurate solutions.*



(a) For $\alpha = 1.4$.



(b) For $\alpha = 1.6$.



(c) For $\alpha = 1.8$.

Figure 1: Exact solution, PINN prediction and absolute error for three values of α.

x	t	Absolute error
0.0	0.0	5.8×10^{-5}
	0.5	1.7×10^{-4}
	1.0	4.5×10^{-4}
0.5	0.0	7.0×10^{-5}
	0.5	7.6×10^{-4}
	1.0	1.8×10^{-3}
1.0	0.0	4.5×10^{-4}
	0.5	7.8×10^{-4}
	1.0	1.2×10^{-3}

Table 1: Absolute error between the exact solution and PINN prediction at different points for $\alpha = 1.6$

Problem 2 *We consider the following one dimensional Gray-Scott system:*

$$\begin{cases} \dfrac{\partial \Phi}{\partial t} &= -k_\Phi(-\Delta)^{\frac{\alpha}{2}}\Phi - \Phi\Psi^2 + F(1-\Phi), \\ \dfrac{\partial \Psi}{\partial t} &= -k_\Psi(-\Delta)^{\frac{\alpha}{2}}\Psi - (F+\mu)\Psi, \qquad (x,t) \in [0,1]^2, \end{cases} \tag{3.4}$$

where $k_\Phi = 10^{-3}$, $k_\Psi = 10^{-5}$, $F = 0.024$ *and* $\mu = 0.06$. *The initial and boundary conditions are given by:*

$$\begin{cases} \Phi(x,0) &= 1 - \dfrac{1}{2}\sin^{100}(\pi x), \\ \Psi(x,0) &= \dfrac{1}{4}\sin^{100}(\pi x), \\ \Phi(0,t) &= \Phi(1,t) = 0, \\ \Psi(0,t) &= \Psi(1,t) = 0. \end{cases}$$

For the numerical reference solution, we discretized the spatial domain using $N = 500$ *interior grid points, while the temporal integration was carried out with a time step* $\Delta t = 10^{-3}$. *For the PINN model, we employed a FCNN with four hidden layers of 128 neurons each and* tanh *activation functions. The network takes as input the space-time coordinates* (x,t) *and outputs the two solutions* (Φ, Ψ). *We considered* 10^4 *collocation points for enforcing the governing equations,* 500 *points for the initial condition and* 200 *points at the boundaries. The weights were set at* $w_{res} = w_{ic} = w_{bc} = 1$ *and we trained our network for 50k iterations using Adam optimizer which took about 27 minutes. We plotted the reference solutions, the predicted solutions and their absolute errors for* $\alpha = 1.5$. *As shown in Figure 2, the PINN shows very good agreement with the numerical solution while the error remains low across the domain.*

Problem 3 *For the third test problem, we consider the following nonlinear space fractional reaction-diffusion equation:*

$$\frac{\partial \Phi}{\partial t} = -k_\alpha(-\Delta)^{\frac{\alpha}{2}}\Phi - \frac{\Phi}{\Phi+1}, \quad (x,t) \in [0,1]^2, \alpha \in]1,2], \tag{3.5}$$

subject to the following initial condition:

$$\Phi(x,0) = \sin(\pi x), \tag{3.6}$$

and homogeneous Dirichlet boundary condition. The network architecture, the number of training points and the collocation strategy are the same as in the Gray-Scott problem. To improve training stability, different weights were assigned to the loss components. We set the weights to $w_{res} = 1$, $w_{bc} = 10^{-3}$ *and* $w_{ic} = 100$, *thereby strongly enforcing the initial condition while relaxing the boundary constraints. This choice was motivated by the observed sensitivity of the solution to the initial condition. The total training time for this problem was approximately 16 minutes for the 50k iteration epochs. Figure 3 compares the ETDRK4 numerical reference solution and the PINN prediction, together with the point-wise absolute error. In Table 2, we report the numerical values of the PINN prediction and the numerical reference solutions at different space-time points. As show from the results, the PINN solution matches accurately the reference numerical solution over the space-time domain.*

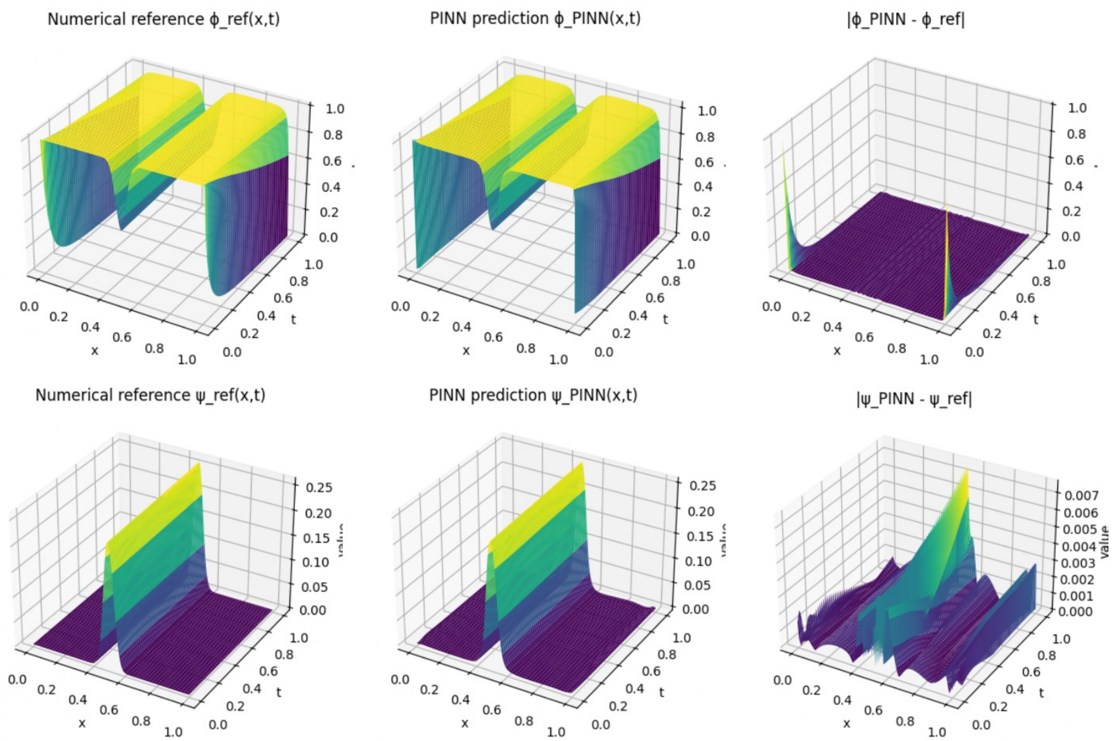


Figure 2: Numerical solutions, PINN predictions and absolute error for $\alpha = 1.5$.

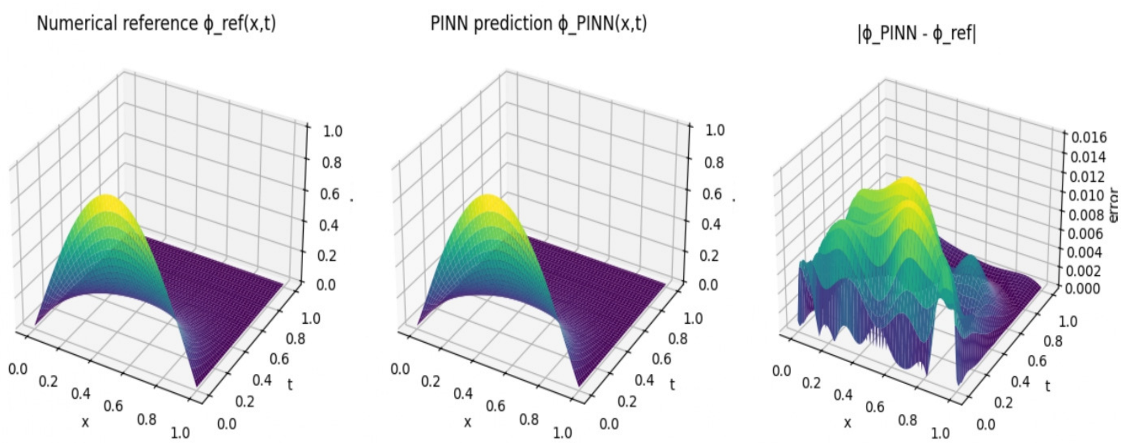


Figure 3: Numerical solution, PINN prediction and absolute error for $\alpha = 1.5$ and $k_\alpha = 1$.

t	x	$\Phi_\theta(x,t)$	$\Phi_{num}(x,t)$
0.0	0.0	0.00087672	0.00000000
	0.1	0.30458024	0.30901644
	0.3	0.80684280	0.80901365
	0.5	0.99665654	0.99999508
	0.7	0.80913305	0.80901365
	0.9	0.29815087	0.30901644
	1.0	0.00099599	0.00000000
0.3	0.0	0.00001991	0.00000000
	0.1	0.04294714	0.04823641
	0.3	0.11415466	0.12659001
	0.5	0.14158553	0.15666046
	0.7	0.11473043	0.12659001
	0.9	0.04593173	0.04823641
	1.0	0.00004060	0.00000000
0.5	0.0	0.00002019	0.00000000
	0.1	0.01147469	0.01358487
	0.3	0.03024084	0.03559510
	0.5	0.04138006	0.04401664
	0.7	0.03043015	0.03559510
	0.9	0.00837822	0.01358487
	1.0	0.00005302	0.00000000
0.9	0.0	0.00001285	0.00000000
	0.1	0.00371524	0.00105464
	0.3	0.00444307	0.00276128
	0.5	0.00376712	0.00341325
	0.7	0.00253999	0.00276128
	0.9	0.00152363	0.00105464
	1.0	0.00001562	0.00000000

Table 2: Numerical values of the PINN prediction and the numerical solution at different points for $\alpha = 1.5$

4. Conclusions

In this paper a hybrid numerical and physics-informed learning framework has been presented to solve some nonlinear space-fractional reaction-diffusion equations. The fractional diffusion operator was incorporated into the PINN framework through a discrete spectral representation consistent with the numerical solver, while the time derivatives were calculated using the neural network automatic differentiation. To evaluate the performance of the framework we computed high-accuracy reference solutions using a spectral discretization of the fractional Laplacian coupled with a fourth-order exponential time-differencing Runge-Kutta scheme (ETDRK4). These numerical solutions were employed exclusively for the validation of proposed model. The numerical experiments demonstrate the effectiveness of the proposed approach and its capability of accurately capturing the complex spatio-temporal dynamics of the solutions. This validates our framework and shows that it can be applied as candidate to solve other fractional partial differential equations. Future work will focus on extending the framework to multi-dimensional problems and comparing the results with some methods mentioned in the introduction.

Acknowledgments

The National Center for Scientific and Technical Research (CNRST) of Morocco has supported this research as part of the PhD-ASsociate Scholarship (PASS) program.

References

1. T.M. Atanackovic, S. Pilipovic, B. Stankovic, D. Zorica, *Fractional Calculus with Applications in Mechanics: From the Cell to the Ecosystem*, Wiley-ISTE, (2014).

2. Z. Sun and G. Gao, *Fractional differential equations: finite difference methods.* De Gruyter, (2020).

3. M. M. Meerschaert and C. Tadjeran, *Finite difference approximations for fractional advection-dispersion flow equations*, J. Comput. Appl. Math., vol. 172, no. 1, pp. 65-77, (2004).

4. W. Deng, *Finite element method for the space and time fractional Fokker-Planck equation*, SIAM J. Numer. Anal., vol. 47, no. 1, pp. 204-226, (2009).

5. Y. Jiang and J. Ma, *High-order finite element methods for time-fractional partial differential equations*, J. Comput. Appl. Math., vol. 235, no. 11, pp. 3285-3290, (2011).

6. S. Salahshour, T. Allahviranloo, and S. Abbasbandy, *Solving fuzzy fractional differential equations by fuzzy laplace transforms*, Commun. Nonlinear Sci. Numer. Simul., vol. 17, no. 3, pp. 1372-1381, (2012).

7. Z. M. Odibat and S. Momani, *Application of variational iteration method to nonlinear differential equations of fractional order*, Int. J. Nonlinear Sci. Numer. Simul., vol. 7, no. 1, pp. 27-34, (2006).

8. S. S. Ray and R. K. Bera, *An approximate solution of a nonlinear fractional differential equation by Adomian decomposition method*, Appl. Math. Comput., vol. 167, no. 1, pp. 561-571, (2005).

9. S. Momani and Z. Odibat, *Analytical solution of a time-fractional Navier-Stokes equation by Adomian decomposition method*, Appl. Math. Comput., vol. 177, no. 2, pp. 488-494, (2006).

10. F. Zeng, F. Liu, C. Li, K. Burrage, I. Turner, and V. Anh, *A Crank-Nicolson ADI spectral method for a two-dimensional Riesz space fractional nonlinear reaction-diffusion equation*, SIAM J. Numer. Anal., vol. 52, no. 6, pp. 2599-2622, (2014).

11. A. Ahmadian, S. Salahshour, D. Baleanu, H. Amirkhani, and R. Yunus, *Tau method for the numerical solution of a fuzzy fractional kinetic model and its application to the oil palm frond as a promising source of xylose*, J. Comput. Phys., vol. 294, pp. 562-584, (2015).

12. A. H. Bhrawy, M. M. Tharwat, and A. Yildirim, *A new formula for fractional integrals of Chebyshev polynomials: Application for solving multi-term fractional differential equations*, Appl. Math. Model., vol. 37, no. 6, pp. 4245-4252, (2013).

13. F. Mohammadi and C. Cattani, *A generalized fractional-order legendre wavelet Tau method for solving fractional differential equations*, J. Comput. Appl. Math., vol. 339, pp. 306-316, (2018).

14. S. Kumar, A. Ahmadian, R. Kumar, D. Kumar, J. Singh, D. Baleanu, and M. Salimi, *An efficient numerical method for fractional SIR epidemic model of infectious disease by using Bernstein wavelets*, Mathematics, vol. 8, no. 4, p. 558, (2020).

15. M. Raissi, P. Perdikaris and G. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics, 378:686–707, (2019).

16. I. Hariri, A. Radid, K. Rhofir, *Physics-informed neural networks methodology for the resolution of some Turing's type models.* Advanced Mathematical Models & Applications, 10(1), 61-80. (2025)

17. I. Hariri, A. Radid, K. Rhofir, *Physics-informed neural networks for the reaction-diffusion Brusselator model*, Mathematical Modeling and Computing, Vol. 11, No. 2, pp. 448-454, (2024).

18. I. Hariri, A. Radid, K. Rhofir, *Embedding physical laws into Deep Neural Networks for solving generalized Burgers-Huxley equation*, Mathematical Modeling and Computing, Vol. 11, No. 2, pp. 505-511, (2024).

19. G. Pang, L. Lu , G. Karniadakis, *fPINNs: Fractional Physics-Informed Neural Networks*, SIAM Journal on Scientific Computing. 41. A2603-A2626, (2019).

20. X. Ma, L. Qiu, B. Zhang, G. Wu, F. Wang, *Adaptive fractional physics-informed neural networks for solving forward and inverse problems of anomalous heat conduction in functionally graded materials*, International Journal of Heat and Mass Transfer 236 (2025).

21. X. Kang, Y. Li, Y. Li, J. Hu, K. Zheng, *Solving the Fractional Allen-Cahn Equation and the Fractional Cahn-Hilliard Equation with the Fractional Physics-Informed Neural Networks.* Fractal Fract, 9, 773 (2025).

22. M. Zhiying, H. Jie, Z. Wenhao, P. Yaxin, L. Ying, *PMNN: Physical model-driven neural network for solving time-fractional differential equations*, Chaos, Solitons & Fractals, Volume 177, (2023).

23. H. Qu, Z. She, X. Liu, *Neural network method for solving fractional diffusion equations*, Applied Mathematics and Computation, (2021).

24. X. Wang, X. Wang, H. Xu, H. Qi, *Numerical analysis and comparison of fractional physics-informed neural networks in unsaturated flow process*, Commun Nonlinear Sci Numer Simulat 147 (2025).

25. K. Hornik, M. Stinchcombe, H. White, *Multilayer feedforward networks are universal approximators*, Neural Networks 2, 359-366 (1989).

Imane Hariri,
LAM2A, Department of Mathematics and Computer Science,
Faculty of Sciences Aïn Chock, Hassan II University of Casablanca,
Morocco.
E-mail address: imane.hariri2000@gmail.com

and

Atika Radid,
LAM2A, Department of Mathematics and Computer Science,
Faculty of Sciences Aïn Chock, Hassan II University of Casablanca,
Morocco.
E-mail address: atikaradid@gmail.com

and

Karim Rhofir,
LaSTI, National School of Applied Sciences Khouribga (ENSAK),
Sultan Moulay Slimane University, Beni Mellal,
Morocco.
E-mail address: krhofir@gmail.com