



Research article

Second-order time integrators with the Fourier spectral method in application to multidimensional space-fractional FitzHugh-Nagumo model

Harish Bhatt*

Department of Mathematics, Utah Valley University, UT 84058, USA

* **Correspondence:** Email: harishbht24@gmail.com.

Abstract: This paper investigated the propagation and interaction behavior of the fractional-in-space multidimensional FitzHugh-Nagumo model using second-order time integrators in combination with the Fourier spectral method. The study focused on analyzing the accuracy, efficiency and stability of these time integrators by comparing numerical results. The experimental findings highlight the ease of implementation and suitability of the methods for long-time simulations. Furthermore, the method's capability to capture the influence of the fractional operator on the equation's dynamics was examined.

Keywords: fourier spectral method; second-order time integrators; space-fractional reaction-diffusion systems; excitable systems; FitzHugh-Nagumo model

1. Introduction

In the field of excitable media, FitzHugh [1] introduced the FitzHugh model as a mathematical model to understand the generation and propagation of electrical signals in neuronal dynamics. Initially, the FitzHugh model was formulated as a nonlinear ordinary differential equation. Later, Nagumo in [2] extended this model to create the FitzHugh-Nagumo (FHN) model. The FitzHugh-Nagumo model is a well known two components classical reaction-diffusion model that describes the dynamics of excitable systems such as neurons and cardiac tissues [3]. However, when heterogeneity and complex connectivity properties characterize the medium in which the transport phenomenon is observed on a very large number of scales, it cannot be adequately described using the classical FHN model. Hence, the use of space-fractional FHN (SFFHN) model becomes necessary. The SFFHN model is obtained by replacing the standard Laplacian operator in the FHN model by a fractional one. In this

study, we consider the following SFFHN model.

$$\begin{aligned}\frac{\partial u(X, t)}{\partial t} &= -D_u(-\Delta)^{\frac{\alpha}{2}}u(X, t) + F_1(u, v), \quad X \in \Omega \subset \mathbb{R}^d, \quad t > 0, \\ \frac{\partial v(X, t)}{\partial t} &= F_2(u, v), \quad X \in \Omega \subset \mathbb{R}^d, \quad t > 0,\end{aligned}$$

with the periodic, homogeneous Neumann, or Dirichlet boundary conditions along with appropriate initial conditions

$$u(X, 0) = u_0(X), \quad v(X, 0) = v_0(X), \quad X \in \Omega, \quad (1.1)$$

where u is a normalized transmembrane potential, v is a dimensionless time-dependent recovery variable, D_u is the diffusion coefficient, d is the dimension of the problem, $F_1(u, v) = u(u - \mu)(1 - u) - v$, $F_2(u, v) = \epsilon(\beta u - \gamma v - \delta)$ are nonlinear functions that satisfy the local Lipschitz condition, $\epsilon, \mu, \beta, \gamma$, and δ are constants that describe the model behavior and $(-\Delta)^{\frac{\alpha}{2}}$, $1 < \alpha \leq 2$ is the fractional Laplacian. As this model often exhibits complex and nonlinear behavior, analytical solutions to this model are often not available or they are very challenging to obtain. This has led to the development of a variety of efficient and accurate numerical methods to simulate the propagation of electrical signals, with varying degrees of accuracy and computational efficiency. In recent years, there has been an increasing interest in the development of various-order numerical methods for the simulation of partial differential equation models such as the SFFHN model. To name a few of them, Liu et al. [4] developed an implicit operator splitting method with a shifted Grünwald-Letnikov approximation to solve the two-dimensional (2D) SFFHN model with zero Dirichlet boundary conditions. Bueno-Orovio et al. [5] proposed a backward Euler method along with a Fourier spectral method as accurate and efficient numerical stencils for the SFFHN model, with the advantage of fully diagonal discretization that can scale to any spatial dimension. Bu et al. in [6] introduced a Crank–Nicolson (CN) alternating direction implicit Galerkin finite element method for a 2D fractional FitzHugh–Nagumo monodomain model with homogeneous Dirichlet boundary conditions. In another study, Li and Song [7] introduced a second-order operator splitting spectral element method for solving the space fractional reaction-diffusion (SFRD) equations including the SFFHN. Liu and Zhang [8] developed a semi-alternating direction Galerkin finite element method to solve a 2D fractional FHN monodomain model on an irregular domain. Li et al. [9] used a fourth-order Runge-Kutta method along with a Fourier spectral method to solve the FitzHugh-Nagumo model with Riesz fractional-derivatives. Zhang et al. in [10] introduced a numerical scheme based on the CN method in time and a Legendre spectral method in space for the SFFHN model. In [11, 12], authors proposed a second-order stabilized semi-implicit method and a second-order operator splitting Fourier spectral method respectively to solve SFRD equations, including the 2D SFFHN model. Wang et al. [13] proposed a ghost structure finite difference method for a fractional FHN monodomain model on a moving irregular domain. Che et al. in [14] introduced the Fourier transform for spatial discretization and the Runge-Kutta method for time discretization for the time dependent 2D SFFHN model. Authors in [15] proposed a fast high-order method for the multidimensional SFRD equation with general boundary conditions, including the 2D SFFHN model.

Second-order numerical methods in particular have shown promise in providing increased accuracy and efficiency over first-order methods. However, the application of these methods to multidimensional (especially 3D) SFFHN models with general boundary conditions (periodic, Dirichlet and Neumann) remains relatively unexplored, despite the potential benefits that they could provide. Therefore, the

primary objective of this research project is to investigate the effectiveness of second-order time integrators in combination with the Fourier spectral method for solving the multidimensional SFFHN model (1.1), with given initial and boundary conditions. We aim to explore the accuracy, stability, and computational efficiency of second-order numerical methods in capturing the dynamics of the SFFHN model for varying α , comparing the results based on two testing parameters: Accuracy and CPU time consumption.

The paper is organized as follows: Section 2 describes the spatial discretization approach using the Fourier spectral method for different boundary conditions. Section 3 introduces four second-order time integrators for solving the SFFHN model. The linear stability analysis of the proposed methods is presented in Section 4. Section 5 provides numerical results and discussions to evaluate the accuracy and computational efficiency of the methods. Finally, Section 6 concludes the work and outlines potential avenues for future research.

2. Fourier spectral method

There are various definitions available for the fractional Laplacian operator $(-\Delta)^{\frac{\alpha}{2}}$, and for more detailed information, readers are encouraged to refer to the references [16, 17]. The selection of an appropriate numerical approximation method depends on the specific definition being employed. In this study, we define the fractional Laplacian $(-\Delta)^{\alpha/2}$ operator in terms of the spectral decomposition of the Laplacian $(-\Delta)$ operator.

Suppose the d dimensional Laplacian $(-\Delta)$ has a complete set of orthonormal eigenfunctions $\phi_{\eta_1, \dots, \eta_d}$ corresponding to eigenvalues $\lambda_{\eta_1, \dots, \eta_d}$ on the bounded region $\Omega = [a, b]^d$ with periodic, homogeneous Dirichlet or homogeneous Neumann boundary conditions, that is $(-\Delta)\phi_{\eta_1, \dots, \eta_d} = \lambda_{\eta_1, \dots, \eta_d}\phi_{\eta_1, \dots, \eta_d}$, then $(-\Delta)^{\alpha/2}$ is defined by

$$(-\Delta)^{\alpha/2}u = \sum_{\eta_1=0}^{\infty} \cdots \sum_{\eta_d=0}^{\infty} \hat{u}_{\eta_1, \dots, \eta_d} \lambda_{\eta_1, \dots, \eta_d}^{\frac{\alpha}{2}} \phi_{\eta_1, \dots, \eta_d},$$

in which u can be expressed as follows

$$u = \sum_{\eta_1=0}^{\infty} \cdots \sum_{\eta_d=0}^{\infty} \hat{u}_{\eta_1, \dots, \eta_d} \phi_{\eta_1, \dots, \eta_d}, \quad \hat{u}_{\eta_1, \dots, \eta_d} = \langle u, \phi_{\eta_1, \dots, \eta_d} \rangle, \quad \sum_{\eta_1=0}^{\infty} \cdots \sum_{\eta_d=0}^{\infty} (|\hat{u}_{\eta_1, \dots, \eta_d}| |\lambda_{\eta_1, \dots, \eta_d}|^{\alpha}) < \infty,$$

where $\langle u, \phi_{\eta_1} \rangle = \int_{\Omega} u \phi_{\eta_1} dx$ for $d = 1$ and eigenvalues $\lambda_{\eta_1, \dots, \eta_d}$ and eigenvectors $\phi_{\eta_1, \dots, \eta_d}$ will depend on the provided boundary conditions:

Case 1: For periodic boundary conditions:

$$\lambda_{\eta_1, \dots, \eta_d} = \sum_{j=1}^d \left(\frac{2\pi\eta_j}{L} \right)^2, \quad \forall \eta_1, \dots, \eta_d,$$

$$\phi_{\eta_1, \dots, \eta_d} = \left(\sqrt{\frac{1}{L}} \right)^d \prod_{j=1}^d e^{(\sqrt{-1} \frac{2\eta_j \pi (x-a)}{L})}, \quad \forall \eta_1, \dots, \eta_d.$$

Case 2: For homogeneous Dirichlet boundary conditions:

$$\lambda_{\eta_1, \dots, \eta_d} = \sum_{j=1}^d \left(\frac{(\eta_j + 1)\pi}{L} \right)^2, \quad \forall \eta_1, \dots, \eta_d,$$

$$\phi_{\eta_1, \dots, \eta_d} = \left(\sqrt{\frac{2}{L}} \right)^d \prod_{j=1}^d \sin \left(\frac{(\eta_j + 1)\pi(X - a)}{L} \right), \quad \forall \eta_1, \dots, \eta_d.$$

Case 3 For homogeneous Neumann boundary conditions:

$$\lambda_{\eta_1, \dots, \eta_d} = \sum_{j=1}^d \left(\frac{(\eta_j)\pi}{L} \right)^2, \quad \forall \eta_1, \dots, \eta_d,$$

$$\phi_{\eta_1, \dots, \eta_d} = \left(\sqrt{\frac{2}{L}} \right)^d \prod_{j=1}^d \cos \left(\frac{(\eta_j)\pi(X - a)}{L} \right), \quad \forall \eta_1, \dots, \eta_d,$$

where $L = b - a$. If we consider a finite number of the orthonormal trigonometric eigenfunctions $\{\phi_i\}$ (equal to the number of discretization points N), then $(-\Delta)^{\frac{\alpha}{2}} u$ is approximated by a truncated series:

$$(-\Delta)^{\alpha/2} u \approx \sum_{\eta_1=0}^{N-1} \cdots \sum_{\eta_d=0}^{N-1} \hat{u}_{\eta_1, \dots, \eta_d} \lambda_{\eta_1, \dots, \eta_d}^{\frac{\alpha}{2}} \phi_{\eta_1, \dots, \eta_d}. \quad (2.1)$$

Efficient computation of the coefficients $\hat{u}_{\eta_1, \dots, \eta_d}$ in Eq (2.1) and the inverse reconstruction of u in physical space can be accomplished using fast and robust algorithms such as direct and inverse Discrete Sine Transforms for the homogeneous Dirichlet boundary, direct and inverse Discrete Cosine Transforms for the homogeneous Neumann boundary and direct and inverse Discrete Fourier Transforms for the periodic boundary data, as detailed in [18].

To illustrate the idea of the Fourier spectral method applied to the fractional Laplacian, let's consider the following equation on one-dimensional (1D) space for simplicity:

$$\frac{\partial u(x, t)}{\partial t} = -D(-\Delta)^{\frac{\alpha}{2}} u(x, t) + f(x, t, u). \quad (2.2)$$

The following is the representation of Eq (2.2) in Fourier space by using the definition of the fractional Laplacian and applying the Fourier transform:

$$\frac{d\mathcal{F}(\mathbf{U}(t))}{dt} + A\mathcal{F}(\mathbf{U}) = \mathcal{F}(\mathbf{F}(\mathbf{U})), \quad (2.3)$$

where $A = D(\lambda_{\eta_1})^{\frac{\alpha}{2}}$, $\mathbf{U} := [u_1(t), u_2(t) \cdots, u_N(t)]^\top$, $\mathbf{F}(\mathbf{U}) = [f(x_1, t, u), f(x_2, t, u), \cdots, f(x_N, t, u)]^\top$.

In the actual calculation, the choice of the mesh will depend on the specified boundary data, and we generate the grids in MATLAB as follows. For simplicity, we consider a 1D space. The 2D/three-dimensional (3D) case is defined analogously.

For periodic boundary conditions,

$$x = \frac{L}{N} \left[-\frac{N}{2} + 1 : \frac{N}{2} \right],$$

$$\lambda_{\eta_1} = \left(\frac{2\pi}{L} \left[0 : \frac{N}{2} - \frac{N}{2} + 1 : -1 \right] \right)^2.$$

For homogeneous Neumann boundary conditions,

$$x_i = a + (i - 1)h + \frac{h}{2}, \quad h = \frac{L}{N}, \quad i = 1, 2, \dots, N,$$

$$\lambda_{\eta_1} = \left(\frac{\pi}{L} [0 : N - 1] \right)^2.$$

For homogeneous Dirichlet boundary conditions,

$$x_i = a + (i)h, \quad h = \frac{L}{N + 1}, \quad i = 1, 2, \dots, N,$$

$$\lambda_{\eta_1} = \left(\frac{\pi}{L} [1 : N] \right)^2.$$

3. Second-order time integrators

In this section, we briefly discuss four second-order time integration methods for solving the SFFHN model. To describe the methods, let us consider the following system of nonlinear ordinary differential equations obtained by approximating the fractional Laplacian in (2.2) with the Fourier spectral technique discussed above:

$$\hat{\mathbf{U}}_t = -A\hat{\mathbf{U}} + \hat{\mathcal{F}}(\mathbf{U}), \quad (3.1)$$

where $\hat{\mathbf{U}} = \mathcal{F}(\mathbf{U})$ and $\hat{\mathcal{F}} = \mathcal{F}(\mathbf{F}(\mathbf{U}))$. Let $k = t_{n+1} - t_n$ be the time step size, then using a variation of constant formula, we come up with the following formula:

$$\hat{\mathbf{U}}(t_{n+1}) = e^{-kA}\hat{\mathbf{U}}(t_n) + e^{-kA} \int_0^k e^{A\tau} \hat{\mathcal{F}}(\mathbf{U}(t_n + \tau), t_n + \tau) d\tau, \quad (3.2)$$

The Eq (3.2) serves as the foundation for various time integrators, including the exponential time differential (ETD) and the integrating factor (IF), which vary based on how we approximate the matrix exponential function and nonlinear function.

3.1. Extrapolated exponential time differencing backward Euler (EETD-BE) method

To get the first order ETD method, the simplest approximation to the integral is to impose that $\hat{\mathcal{F}}$ is constant for $t \in [t_n, t_{n+1}]$ i.e., $\hat{\mathcal{F}} \approx \hat{\mathcal{F}}_n$. Denoting the semi-discrete approximation to $\hat{\mathbf{U}}(t_n)$ by $\hat{\mathbf{U}}_n$ yields:

$$\hat{\mathbf{U}}_{n+1} = e^{-kA}\hat{\mathbf{U}}_n - A^{-1}(e^{-kA} - 1)\hat{\mathcal{F}}_n. \quad (3.3)$$

However, the effectiveness of this semi-discrete method hinges on discretizing the matrix exponential. By incorporating the (0, 1)-Padé approximation, the scheme becomes more practical and computationally efficient. This approximation facilitates a direct discretization of the matrix exponential and eliminates the need for explicit computation of the inverse matrix A .

Definition 3.1. The $(r + s)^{\text{th}}$ order rational Padé approximation to e^{-z} is defined as:

$$R_{r,s}(z) = \frac{P_{r,s}(z)}{Q_{r,s}(z)},$$

where

$$P_{r,s}(z) = \sum_{j=0}^r \frac{(s+r-j)!r!}{(s+r)!j!(r-j)!} (-z)^j \text{ and } Q_{r,s}(z) = \sum_{j=0}^s \frac{(s+r-j)!}{(s+r)!j!(s-j)!} (z)^j.$$

Implementing (0, 1)-Padé approximation $R_{0,1}(kA) = (1+kA)^{-1}$ into Eq (3.3), yields the ETD backward Euler method (ETD-BE):

$$\hat{U}_{n+1} = (1+kA)^{-1} [\hat{U}_n + k\hat{\mathcal{F}}_n]. \quad (3.4)$$

The ETD-BE method is only first order accurate in time. So, in order to enhance the temporal accuracy to second-order, we implement a local extrapolation approach proposed by Lawson and Morris [19]:

Considering Eq (3.4) written over a double time step $2k$ and denoting the result with $\hat{U}^{(2)}$, we have:

$$\hat{U}^{(2)} = (1+2kA)^{-1} [\hat{U}_n + 2k\hat{\mathcal{F}}_n]. \quad (3.5)$$

Alternatively, if Eq (3.4) is computed over two single time steps and if we denote the result with $\hat{U}^{(1)}$ we have:

$$\hat{U}^{(1)} = (1+kA)^{-1} [\hat{U}_{n+1} + k\hat{\mathcal{F}}_{n+1}]. \quad (3.6)$$

As a result, Equations (3.5) and (3.6) represent two distinct ETD-BE schemes for computing the solution at double time steps. Neither (3.5) nor (3.6) achieves an accuracy of $O(k^2)$. Nonetheless, by expanding $\hat{U}^{(1)}$ and $\hat{U}^{(2)}$ to $O(k^2)$ and combining these expansions through the operation of subtracting $\hat{U}^{(2)}$ from $2\hat{U}^{(1)}$, the local extrapolation of ETD-BE is obtained [20]:

$$\hat{U}_{n+2} = 2\hat{U}^{(1)} - \hat{U}^{(2)}. \quad (3.7)$$

For the efficient implementation of EETD-BE, in the following Algorithm the fast Fourier transform (FFT) computations of the EETD-BE method are provided for the case of 3D with Neumann boundary conditions:

Algorithm 1 EETD-BE Method for the case of 3D with Neumann boundary conditions

- 1: Given $L = b - a, N, h = \frac{L}{N}, \alpha, D_u, w_n$, and $k = \frac{T}{M}$.
 - 2: Compute $\lambda = \left((0 : N - 1) \frac{\pi}{L}\right)^2 + \left((0 : N - 1) \frac{\pi}{L}\right)^2 + \left((0 : N - 1) \frac{\pi}{L}\right)^2$
 - 3: Compute $x = y = z = a + (0 : N - 1)h + \frac{h}{2}$
 - 4: Set $A = kD_u \lambda^{\frac{\alpha}{2}}$
 - 5: Precompute the following arrays:
 - 6: $P = (1 + A)^{-1}$
 - 7: $Q = (1 + 2A)^{-1}$
 - 8: **for** $n = 1, \dots, \frac{M}{2}$ **do**
 - 9: $t_n = nk$.
 - 10: $\hat{w}_n = \text{dctm}(w_n)$ ▷ *dctm* is MATLAB built-in function
 - 11: Compute $\mathcal{F}(w_n, t_n)$ as $\mathcal{F}(w_n)$
 - 12: $w^* = \text{idctm}(P(\hat{w}_n + k \text{dctm}(\mathcal{F}(w_n))))$ ▷ *idctm* is MATLAB built-in function
 - 13: Compute $\mathcal{F}(w^*, t_n + k)$ as $\mathcal{F}(w^*)$
 - 14: $w^{(1)} = \text{idctm}(P(\text{dctm}(w^*) + k \text{dctm}(\mathcal{F}(w^*))))$
 - 15: $w^{(2)} = \text{idctm}(Q(\hat{w}_n + 2k \text{dctm}(\mathcal{F}(w_n))))$
 - 16: $w_{n+2} = 2w^{(1)} - w^{(2)}$
 - 17: $w_n = w_{n+2}$
 - 18: **end for**
-

3.2. Exponential time differencing Crank-Nicolson (ETD-CN) method

Using ETD-BE as a prediction of $\hat{\mathbf{U}}_{n+1}$ and approximating the nonlinear term $\hat{\mathcal{F}}$ [21] in (3.2) with:

$$\hat{\mathcal{F}} \approx \hat{\mathcal{F}}(\mathbf{U}_n, t_n) + (t - t_n) \frac{\hat{\mathcal{F}}(\hat{\mathbf{a}}_n, t_n + k) - \hat{\mathcal{F}}(\mathbf{U}_n, t_n)}{k}, \quad t \in [t_n, t_{n+1}],$$

where $\hat{\mathbf{a}}_n = e^{-kA} \hat{\mathbf{U}}_n - A^{-1} (e^{-kA} - 1) \hat{\mathcal{F}}_n$ yields

$$\begin{aligned} \hat{\mathbf{U}}_{n+1} &= e^{-kA} \hat{\mathbf{U}}(t_n) + ke^{-kA} \int_0^1 e^{kA\tau} \left(\hat{\mathcal{F}}(\mathbf{U}_n, t_n) + (t - t_n) \frac{\hat{\mathcal{F}}(\hat{\mathbf{a}}_n, t_n + k) - \hat{\mathcal{F}}(\mathbf{U}_n, t_n)}{k} \right) d\tau, \\ &= \hat{\mathbf{a}}_n + \frac{A^{-2}}{k} (e^{-kA} - 1 + kA) (\hat{\mathcal{F}}(\hat{\mathbf{a}}_n, t_n + k) - \hat{\mathcal{F}}(\mathbf{U}_n, t_n)). \end{aligned}$$

Replacing e^{-kA} in (3.8) with (1, 1)-Padé approximation $R_{1,1}(kA) = (2 - kA)(2 + kA)^{-1}$ results the ETD-CN method:

$$\hat{\mathbf{U}}_{n+1} = \mathbf{a}_n + k(2 + kA)^{-1} [\hat{\mathcal{F}}(\hat{\mathbf{a}}_n, t_n + k) - \hat{\mathcal{F}}(\mathbf{U}_n, t_n)], \quad (3.8)$$

where

$$\mathbf{a}_n = [4(2 + kA)^{-1} - 1] \hat{\mathbf{U}}_n + 2k(2 + kA)^{-1} \hat{\mathcal{F}}(\mathbf{U}_n, t_n).$$

For the efficient implementation of ETD-CN, the FFT computations of the method are provided for the case of 3D with periodic boundary conditions:

Algorithm 2 ETD-CN Method for the case of 3D with periodic boundary conditions

- 1: Given $L = b - a, N, h = \frac{L}{N}, \alpha, D_u, w_n$, and $k = \frac{T}{M}$.
 - 2: Compute $\lambda = ([0 : \frac{N}{2} - \frac{N}{2} + 1 : -1] \frac{2\pi}{L})^2 + ([0 : \frac{N}{2} - \frac{N}{2} + 1 : -1] \frac{2\pi}{L})^2 + ([0 : \frac{N}{2} - \frac{N}{2} + 1 : -1] \frac{2\pi}{L})^2$
 - 3: Compute $x = y = z = \frac{L}{N} [-\frac{N}{2} + 1 : \frac{N}{2}]$
 - 4: Set $A = kD_u \lambda^{\frac{\alpha}{2}}$.
 - 5: Precompute the following arrays:
 - 6: $P = (2 + A)^{-1}$.
 - 7: **for** $n = 1, \dots, M$ **do**
 - 8: $t_n = nk$.
 - 9: $\hat{w}_n = \text{fft}(w_n)$. ▷ *fft* is MATLAB built-in function
 - 10: Compute $\mathcal{F}(w_n, t_n)$ as $\mathcal{F}(w_n)$.
 - 11: $a_n = \text{ifft}(P(4\hat{w}_n + 2k\text{fft}(\mathcal{F}(w_n)))) - \hat{w}_n$ ▷ *ifft* is MATLAB built-in function
 - 12: Compute $\mathcal{F}(a_n, t_n + k)$ as $\mathcal{F}(a_n)$.
 - 13: $w_{n+1} = \text{ifft}(\text{fft}(a_n) + kP[\mathcal{F}(a_n) - \mathcal{F}(w_n)])$
 - 14: **end for**
-

3.3. ETD-Padé (0, 2)

Replacing e^{-kA} in (3.8) with (0, 2)-Padé approximation $R_{0,2}(kA) = 2(2 + 2kA + (kA)^2)^{-1}$ yields the ETD-Padé (0, 2) method:

$$\hat{\mathbf{U}}_{n+1} = \mathbf{a}_n + k(2 + 2kA + (kA)^2)^{-1} (1 + kA) [\hat{\mathcal{F}}(\hat{\mathbf{a}}_n, t_{n+1}) - \hat{\mathcal{F}}(\mathbf{U}_n, t_n)], \quad (3.9)$$

where

$$\mathbf{a}_n = R_{0,2}(kA)\hat{\mathbf{U}}_n + k(2 + 2kA + (kA)^2)^{-1}(2 + kA)\hat{\mathcal{F}}(\mathbf{U}_n, t_n).$$

For the efficient implementation of ETD-Padé (0, 2), the FFT computations of the method are provided for the case of 3D with Dirichlet boundary conditions:

Algorithm 3 ETD-Padé (0, 2) method for the case of 3D with Dirichlet boundary conditions

- 1: Given $L = b - a, N, h = \frac{L}{N+1}, \alpha, D_u, w_n,$ and $k = \frac{T}{M}$.
 - 2: Compute $\lambda = \left((1 : N)\frac{\pi}{L}\right)^2 + \left((1 : N)\frac{\pi}{L}\right)^2 + \left((1 : N)\frac{\pi}{L}\right)^2$
 - 3: Compute $x = y = z = a + (1 : N)h$
 - 4: Set $A = kD_u\lambda^{\frac{\alpha}{2}}$.
 - 5: Precompute the following arrays:
 - 6: $R_{02} = (2 + 2kA + (kA)^2)^{-1}$.
 - 7: $P = 2R_{02}$.
 - 8: $Q = k(1 + A)R_{02}$.
 - 9: $R = k(2 + A)R_{02}$.
 - 10: **for** $n = 1, \dots, M$ **do**
 - 11: $t_n = nk$.
 - 12: $\hat{w}_n = \text{dstn}(w_n)$. ▷ *dstn* is MATLAB built-in function
 - 13: Compute $\mathcal{F}(w_n, t_n)$ as $\mathcal{F}(w_n)$.
 - 14: $a_n = \text{idstn}(P\hat{w}_n + R\text{fft}(\mathcal{F}(w_n)))$ ▷ *idstn* is MATLAB built-in function
 - 15: Compute $\mathcal{F}(a_n, t_n + k)$ as $\mathcal{F}(a_n)$.
 - 16: $w_{n+1} = \text{ifftn}(\text{fft}(\mathbf{a}_n) + kP[\mathcal{F}(\mathbf{a}_n) - \mathcal{F}(w_n)])$
 - 17: **end for**
-

3.4. IIF2 method

To construct the second-order implicit integration factor (IIF2) method [22], we approximate $e^{kA\tau}\hat{\mathcal{F}}(\mathbf{U}(t_n+k\tau), t_n+k\tau)$ using second-order Lagrange interpolation polynomial involving points $t_{n+1}, t_n,$ and t_{n-1} :

$$e^{A\tau}\hat{\mathcal{F}}(\mathbf{U}(t_n + \tau), t_n + \tau) \approx \frac{1}{k} \left[\hat{\mathcal{F}}(\mathbf{U}_n)(k - \tau) + e^{kA}\hat{\mathcal{F}}(\mathbf{U}_{n+1})\tau \right], \quad 0 \leq \tau \leq k.$$

With the above approximation, the direct evaluation of the integral in (3.2) leads to the IIF2 method given as below:

$$\hat{\mathbf{U}}_{n+1} = e^{-kA} \left(\hat{\mathbf{U}}_n + \frac{k}{2}\hat{\mathcal{F}}(\mathbf{U}_n) \right) + \frac{k}{2}\hat{\mathcal{F}}(\mathbf{U}_{n+1}). \quad (3.10)$$

For the efficient implementation of the IIF2 method, the FFT computations of the method are provided for the case of 3D with Dirichlet boundary conditions:

Algorithm 4 IIF2 method for the case of 3D with Dirichlet boundary conditions

```

1: Given  $L = b - a, N, h = \frac{L}{N+1}, \alpha, D_u, w_n, t_0, t_{\text{end}}, \text{Tol}$ , and  $k = \frac{T}{M}$ .
2: Compute  $\lambda = \left((1 : N)\frac{\pi}{L}\right)^2 + \left((1 : N)\frac{\pi}{L}\right)^2 + \left((1 : N)\frac{\pi}{L}\right)^2$ 
3: Compute  $x = y = z = a + (1 : N)h$ 
4: Precompute  $P = \text{expm}(-kD_u\lambda^{\frac{\alpha}{2}})$  ▶ expm is MATLAB built-in function
5: while istop== 0 do
6:   if  $t_0 + k > t_{\text{end}}$ . then
7:      $k = t_{\text{end}} - t_0$ 
8:      $P = \text{expm}(-kD_u\lambda)$ 
9:     istop= 1
10:  end if
11:   $\hat{w}_n = \text{dstn}(w_n)$ . ▶ dstn is MATLAB built-in function
12:  Compute  $\mathcal{F}(w_n, t_n)$  as  $\mathcal{F}(w_n)$ .
13:   $a_n = P\left(\hat{w}_n + \frac{k}{2}\mathcal{F}(w_n)\right)$ 
14:  Temp =  $\text{idstn}\left(a_n + \frac{k}{2}\text{fftn}(\mathcal{F}(w_n))\right)$  ▶ idstn is MATLAB built-in function
15:  Compute res = norm(Temp(:) -  $w_n$ (:))
16:  while max(res) > Tol do
17:     $w_n = \text{Temp}$ 
18:    Temp =  $\text{idstn}\left(a_n + \frac{k}{2}\text{fftn}(\mathcal{F}(w_n))\right)$ 
19:    Compute res = norm(Temp(:) -  $w_n$ (:))
20:  end while
21:   $w_n = \text{Temp}$ 
22:   $t_0 = t_0 + k$ 
23: end while

```

4. Stability regions

We consider the following nonlinear autonomous ordinary differential equation (ODE):

$$v_t = -cv + F(v), \quad c \geq 0. \quad (4.1)$$

In order to get the stability regions of an EETD-BE, we linearize Eq (4.1) about a fixed point u_0 such that $-cu_0 + F(u_0) = 0$ to obtain:

$$u_t = -cu + \lambda u, \quad (4.2)$$

where u is perturbation of u_0 and $\lambda = F'(u_0)$. In order to keep a fixed point u_0 stable we need to have $\text{Re}(\lambda - c) < 0$ (note that the fixed points of the ETD schemes are the same as those of the ODE (4.1), in contrast to the IF methods, which do not preserve the fixed points for the ODE that they discretize) [23].

The boundaries of the stability regions (a family of curves for different values of ck , where k is the time step) based on Eq (4.2) are presented below for EETD-BE, ETD-CN and ETD-Padé (0, 2).

To obtain the stability regions, we applied EETD-BE, ETD-CN and ETD-Padé (0, 2) to (4.2), which leads to the following amplification factors:

4.1. Amplification factor of EETD-BE

$$r(x, y) = c_0 + c_1x + c_2x^2,$$

where

$$c_0 = \frac{y^2 + 2y - 1}{(y - 1)^2(2y - 1)}, \quad c_1 = \frac{2y^2 + 4y - 2}{(y - 1)^2(2y - 1)}, \quad c_2 = \frac{2}{(y - 1)^2}.$$

4.2. Amplification factor of ETD-CN

$$r(x, y) = c_0 + c_1x + c_2x^2,$$

where

$$c_0 = \frac{2 + y}{2 - y}, \quad c_1 = \frac{4}{(2 - y)^2}, \quad c_2 = \frac{2}{(2 - y)^2}.$$

4.3. Amplification factor of ETD-Padé (0, 2)

$$r(x, y) = c_0 + c_1x + c_2x^2,$$

where

$$c_0 = \frac{2}{2 + 2y + y^2}, \quad c_1 = \frac{y^2 + 4y + 4}{(2 + 2y + y^2)^2}, \quad c_2 = \frac{y^2 + 3y + 2}{(2 + 2y + y^2)^2}.$$

In general, the parameters c and λ may both be complex-valued. The stability region is four dimensional (4D) and, therefore, difficult to represent [23]. However, if both c and λ are pure imaginary [24] or real [23] or if λ is complex and c is fixed and real [25], then the stability region is 2D. The boundaries of the stability region are obtained by substituting $r = e^{i\theta}$, $\theta \in [0, 2\pi]$ into Eq (4.2) and solving for x . In order to plot a 2D stability region in the complex x -plane, where the solution is bounded as $n \rightarrow \infty$, we start our analysis by assuming c to be fixed and real and λ as complex.

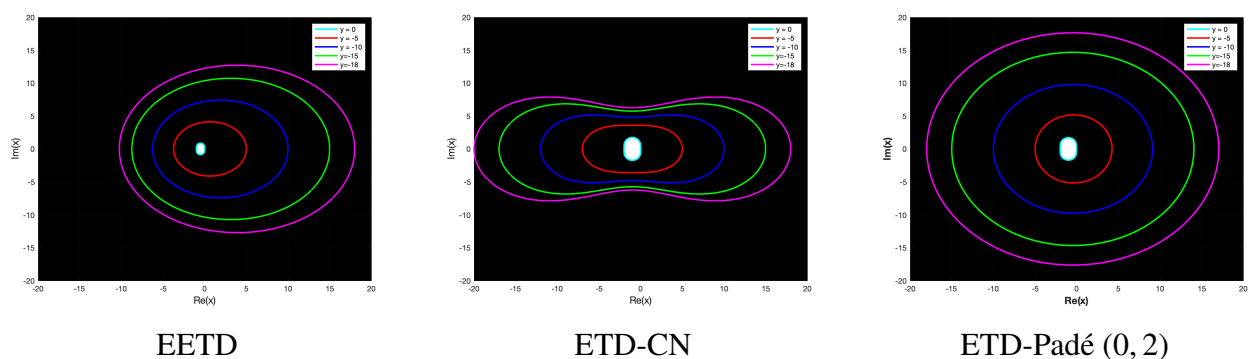


Figure 1. Log-log plots of time rates of convergence of the proposed methods for varying α in the case of homogeneous Neumann boundary.

Figure 1 illustrates the stability regions of the methods for different values of $y = 0, -3, -5, -10, -15,$ and -20 in the complex x -plane. According to Beylkin et al. [25], it is crucial

for a method to have stability regions that grow as ck becomes larger to be useful. From the stability regions shown in Figure 1, we observed that as $y \rightarrow -\infty$, the methods maintain their shape and exhibit larger stability regions. This characteristic enables the methods to utilize larger time-step sizes as $|c| \rightarrow \infty$ while solving the problem. This observation indicates the stability of the methods EETD-BE, ETD-CN and ETD-Padé(0, 2). Comparing the stability regions in Figure 1, we can note that ETD-CN exhibits slightly better stability than EETD and ETD-Padé(0, 2), as its stability region encompasses larger values along the real axis.

4.4. Amplification factor of IIF2

For the linear stability analysis and amplification factor of the IIF2 method, readers are referred to the work of [22]. In their study, the authors stated that the IIF2 method is unconditionally stable and exhibits excellent stability properties when used to solve stiff reaction systems.

5. Numerical experiments and discussions

In this section, five test problems are considered to demonstrate the effectiveness of the methods based on the two key testing parameters: Accuracy and CPU time consumption for the integrating SFFHN model with different fractional powers α , different initial conditions along with periodic, homogeneous Dirichlet, or Neumann boundary conditions. The accuracy of the method is measured in terms of maximum error norm $\|\cdot\|_\infty$ and the empirical rates of convergence (in time direction) of the methods is computed utilizing the following formula:

$$\text{rate} = \frac{\log_{10}(E_k/E_{\frac{k}{2}})}{\log_{10}(2)},$$

where $E_k = \|u_k - u_{2k}\|_\infty$ and $E_{\frac{k}{2}} = \|u_{\frac{k}{2}} - u_k\|_\infty$ are maximum norm errors at k and $\frac{k}{2}$. Throughout this study, we fixed the parameters $D_u = 10^{-4}$, $\epsilon = 0.01$, $\mu = 0.1$, $\beta = 0.1$, $\gamma = 1$, and $\delta = 0$, which are known to generate stable patterns in the system. All the computations are conducted on the MATLAB 2020b platform based on a MacBook Pro with 2.6 GHz 6-Core Intel Core i7 CPU and 16 GB RAM.

Example 1 (2D SFFHN model with Neumann Boundary): In this example, we consider the 2D SFFHN model with homogeneous Neumann boundary conditions, along with the following initial conditions on domain $[0, 2.5]^2$:

$$u(x, y, 0) = \begin{cases} 1, & \text{if } 0 \leq x, y \leq 1.25 \\ 0, & \text{otherwise} \end{cases} \quad (5.1)$$

$$v(x, y, 0) = \begin{cases} 0.1, & \text{if } 0 \leq x \leq 2.5, \quad 1.25 \leq y \leq 2.5 \\ 0, & \text{otherwise.} \end{cases}$$

To test the order of accuracy of the methods mentioned in Section 3 for varying α along with Neumann boundary conditions, we simulated the solution profile of the component u up to the final time $t = 10$. In the experiment we set the spectral resolution $N = 256$ large enough to ensure that the

errors primarily arise from the time integration process rather than spatial discretization and repeatedly halving the temporal step size in each simulation, starting with the initial value $k = 0.1$. The results of our experiments are illustrated in Figure 2 for various values of $\alpha = 2, 1.7$ and 1.5 . Figure 2 confirms second-order convergence for the ETD-CN, ETD-Padé (0, 2), IIF2, and EETD methods. These second-order methods effectively integrate the system for varying α values.

In Figure 3, we extended the comparison of the methods by analyzing their accuracy in relation to the CPU time. The results depicted in Figure 3 highlight a noticeable variation in CPU time among the methods, even at a comparable level of accuracy. Consequently, it can be concluded that the computational time required by each method is not similar, indicating the presence of significant differences in efficiency. The performance of the ETD-CN method closely resembles that of the ETD-Padé (0, 2), as evident from the identical errors that scale with the expected second-order as shown in Figure 2. However, it should be noted that the IIF2 method is the most computationally expensive due to the involvement of a nonlinear solver, while the EETD method is the least demanding in terms of computational resources.

Considering both accuracy and computational cost when selecting a second-order method for time stepping for Example 1, it becomes evident that the ETD-CN method stands out as the most efficient choice in this regard.

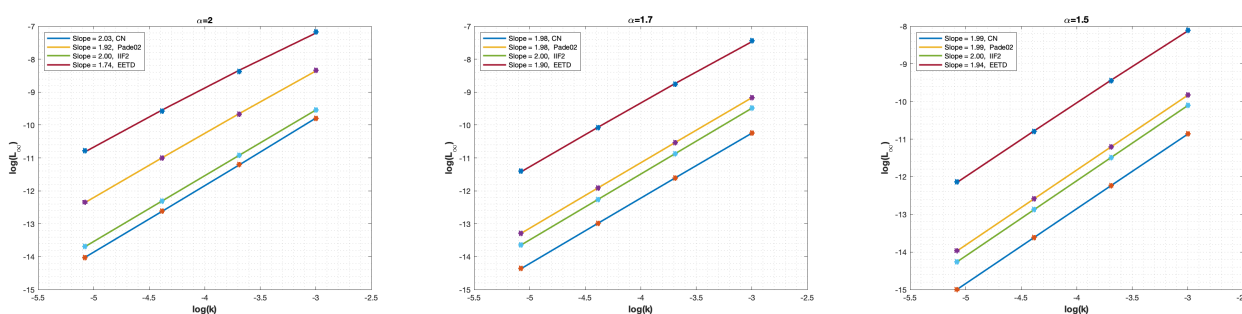


Figure 2. Log-log plots of time rates of convergence of the proposed methods for varying α in the case of homogeneous Neumann boundary.

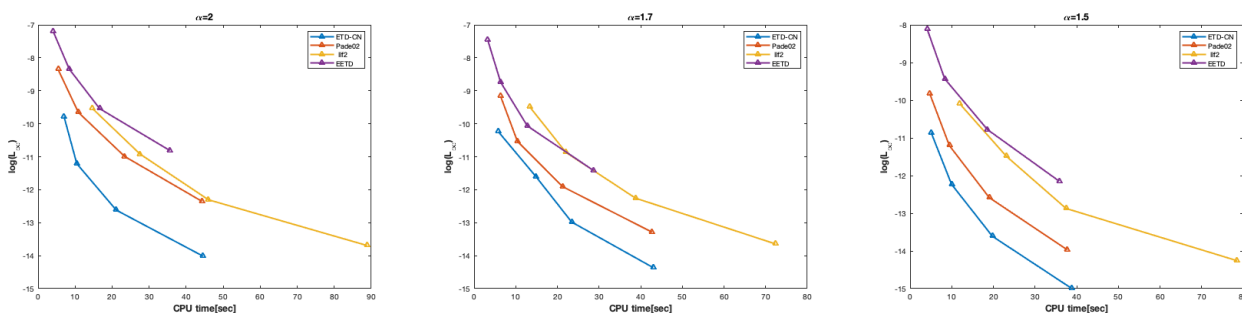


Figure 3. Comparison of the computational efficiency of the proposed methods for varying α in the case of homogeneous Neumann boundary.

In this experiment, we utilized the ETD-CN method to investigate the effect of the fractional order in the SFFHN model. Figure 4 displays the stable rotation solutions of the component u at $t = 2000$ with

$k = 1$ and $N = 256$ for four different fractional orders $\alpha = 2, 1.8, 1.7$ and 1.5 . As we observed in Figure 4, decreasing the fractional power has a notable effect on the width of the excitation wavefront and the wavelength of the system. A smaller value of α leads to a significantly reduced width of the wavefront and a shorter wavelength, allowing the domain to accommodate a larger number of wavefronts. These visualizations align with the findings presented by Bueno-Orovio et al. in [5]. In addition, these results highlight the application of fractional diffusion as a modeling tool to characterize intermediate dynamic states that cannot be solely described by pure diffusion mechanisms.

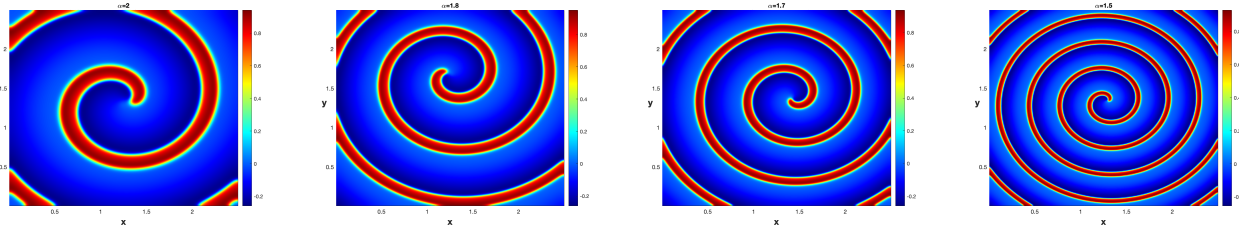


Figure 4. Propagation speed of the spiral wave in FitzHugh-Nagumo model for varying $\alpha = 2, 1.8, 1.7, 1.5$ at $t = 2000$.

Example 2 (2D SFFHN model with Periodic Boundary): In this example, we consider the 2D SFFHN model with periodic boundary conditions, along with following initial conditions on domain $[-2.5, 2.5]^2$ to analyze the effectiveness and accuracy of the methods for periodic boundary conditions and varying α values.

$$u(x, y, 0) = \begin{cases} u(\frac{3N}{8} : \frac{5N}{8}, \frac{5N}{8} : \frac{6N}{8}) = 1 \\ u(\frac{3N}{8} : \frac{5N}{8}, \frac{2N}{8} : \frac{3N}{8}) = 1 \\ 0, \text{ elsewhere} \end{cases} \quad (5.2)$$

$$v(x, y, 0) = \begin{cases} v(\frac{5N}{8} : N, \frac{N}{4} : N) = 0.1 \\ v(\frac{N}{8} : \frac{3N}{8}, \frac{N}{4} : N) = 0.1 \\ 0, \text{ elsewhere.} \end{cases}$$

In this experiment, we conducted a comparative analysis of the methods to assess their accuracy, computational cost, and effectiveness for solving SFFHN in the case of periodic boundary conditions and varying fractional order α . We simulated the solution profile of the component u up to the final time $t = 10$ by setting the fixed spectral resolution $N = 512$, which was chosen to be large enough to ensure that the errors primarily originated from the time integration process rather than spatial discretization.

In each simulation, we systematically reduced the temporal step size by halving it, starting with an initial value of $k = 0.1$. This step allowed us to investigate the impact of different temporal resolutions on the accuracy of the results and assess the convergence behavior of the numerical method. In Figures 5 and 6, we plotted the maximum error as a function of temporal step and the CPU time, respectively. From these figures, it is evident that all the methods achieved expected second-order convergence in the temporal direction for varying α . Additionally, there is a noticeable variation in the CPU time among the methods, even when they achieve a comparable level of accuracy.

When considering both accuracy and computational cost in the selection of a second-order method for time stepping in Example 2, the ETD-CN method emerges as the most efficient choice, making it a

favorable option when balancing accuracy and computational cost.

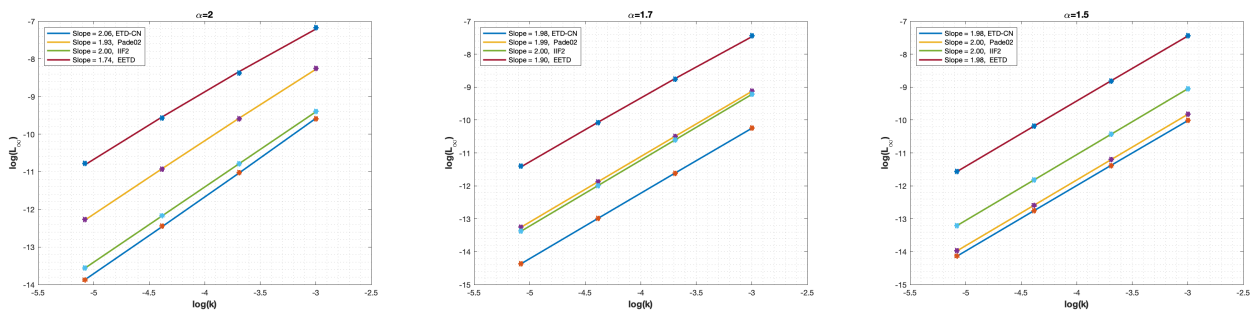


Figure 5. Log-log plots of time rates of convergence of proposed methods for varying α in the case of periodic boundary conditions.

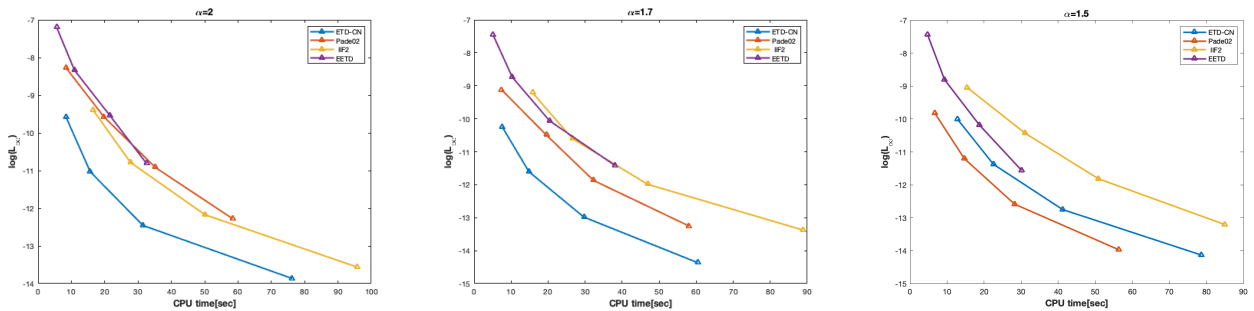


Figure 6. Comparison of the computational efficiency of proposed methods for varying α in the case of periodic boundary conditions.

After ensuring that the proposed methods provide second-order accuracy for Example 2, in this sets of experiments we simulated the spatio-temporal evolution profile of the component v and studied the effects of the super-diffusion on the SFFHN model. All the plots shown in Figure 7 are the aerial views of the numerical solutions v and are obtained via the ETD-CN method with $N = 512, k = 1$ and $\alpha = 2, 1.8, 1.7, 1.5$ for various times. Since the evolution profile of the component u is similar to the v , we will not discuss it in this paper. From Figure 7, we observe that as the fractional order decreases, both the width of the excitation wavefront and the wavelength of the system become significantly smaller. Consequently, the domain is capable of accommodating a greater number of wavefronts for lower orders. In the case of super-diffusion, which is characterized by the presence of a fractional Laplacian operator, the wave propagation exhibits long-tailed characteristics. This means that the influence of the wave extends over a larger spatial range compared to traditional diffusion. As a result, for wavefronts with similar widths, the distance between consecutive wavelengths becomes larger in the case of super-diffusion. These observations are in accordance with previous investigations conducted by researchers in [9].

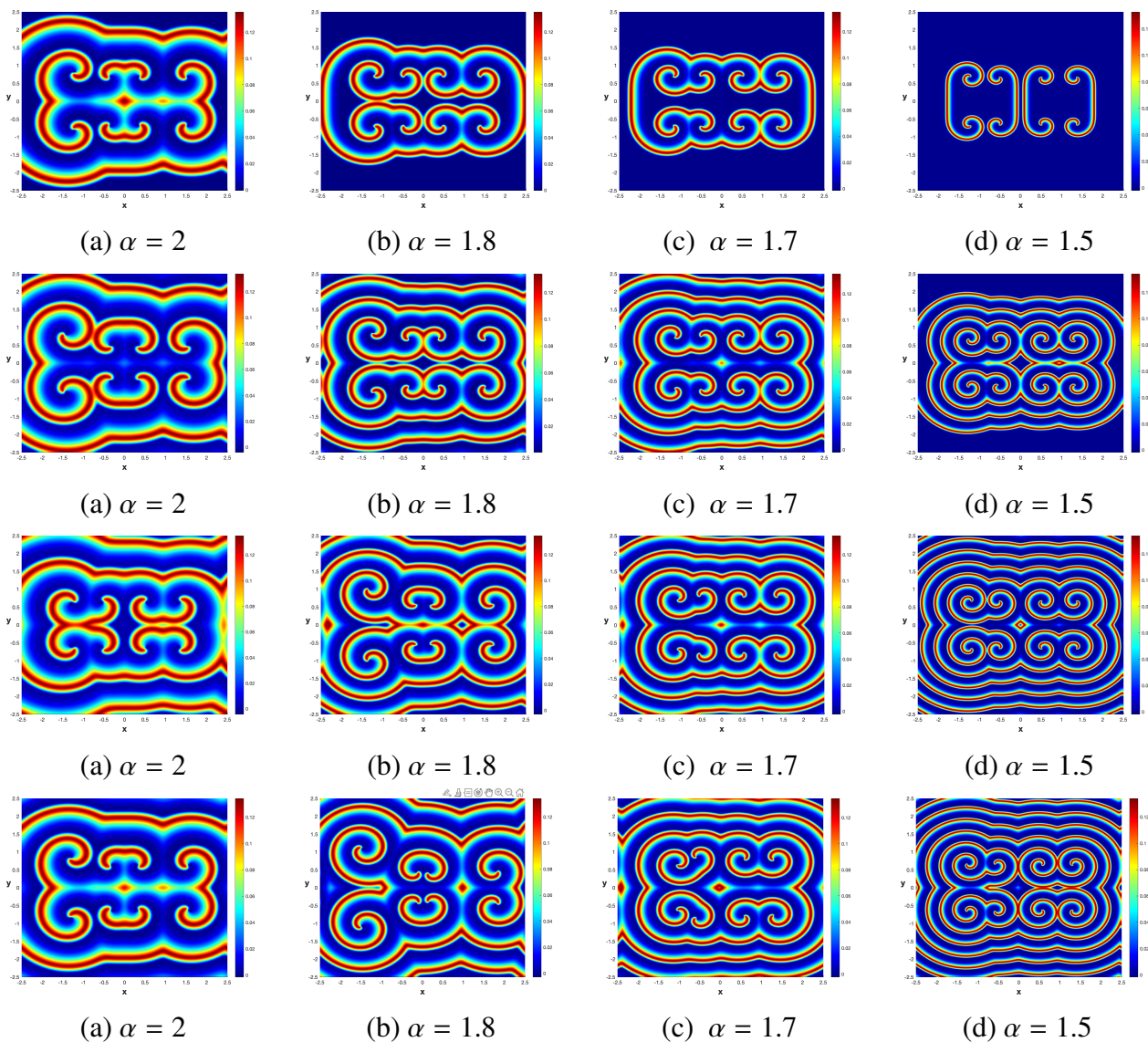


Figure 7. Spatio-temporal wave propagation profile of the component v for varying $t = 400, 1000, 2000$ and 3000 (top-bottom).

Example 3 (2D SFFHN model with Dirichlet Boundary): In this example, we consider the 2D SFFHN model with homogeneous Dirichlet boundary conditions, along with the following initial conditions on domain $[-1.25, 1.25]^2$ to compare accuracy, efficiency and effectiveness of the methods for homogeneous Dirichlet boundary conditions and varying α .

$$u(x, y, 0) = \begin{cases} 1, & \text{if } x, y < 0 \\ 0, & \text{otherwise} \end{cases} \quad (5.3)$$

$$v(x, y, 0) = \begin{cases} 0.1, & \text{if } -1.25 \leq x \leq 1.25, y > 0 \\ 0, & \text{otherwise.} \end{cases}$$

In order to analyze the empirical rates of convergence, computational efficiency, and effectiveness

of the methods for the case of homogenous Dirichlet boundary conditions and super diffusion, we simulated the solution profile of the component u with the methods up to time $t = 10$. We fixed $N = 256$ and ran the simulations for various values of temporal step size starting with $k = 0.1$ and illustrated the maximum errors as a function of temporal step and CPU time in Figures 8 and 9, respectively. From these figures, it is evident that all the methods were able to capture second-order accuracy in the time direction for varying fractional orders α . The convergence rates were consistent with the expected behavior. Moreover, there was noticeable variation in the CPU time among the methods, similar to what was observed in Examples 1 and 2, even when achieving a comparable level of accuracy.

Considering the case of Dirichlet boundary conditions with the specified initial conditions, again the ETD-CN method emerged as the most efficient choice, taking into account both accuracy and computational efficiency. The results depicted in Figures 8 and 9 further support the superiority of the ETD-CN method in this particular scenario.

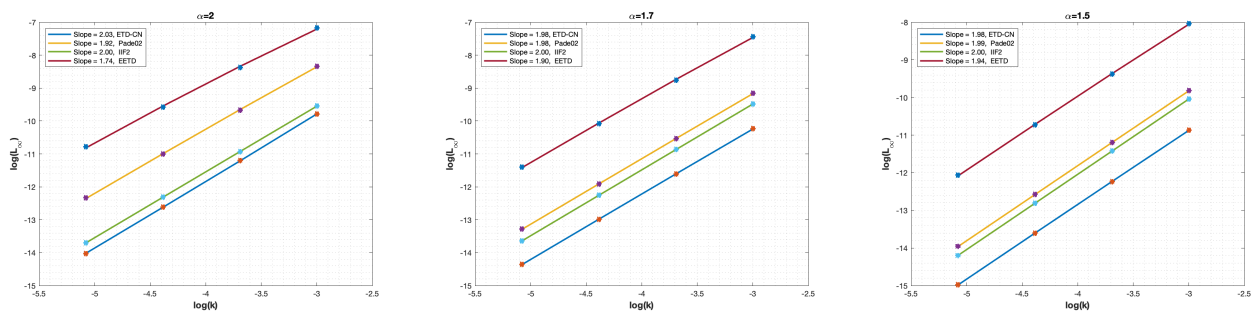


Figure 8. Log-log plots of time rates of convergence of proposed methods for varying α in the case of homogeneous Dirichlet boundary.

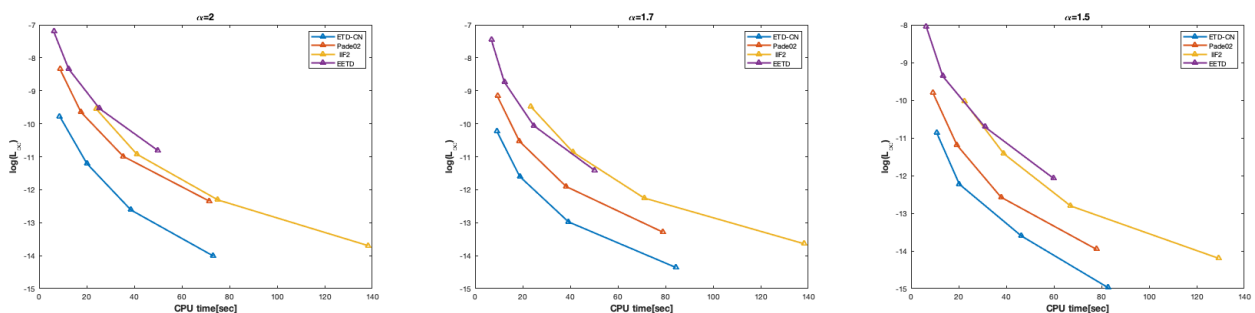


Figure 9. Comparison of the computational efficiency of proposed methods for varying α in the case of homogeneous Dirichlet boundary.

Example 4 (3D SFFHN model with periodic boundary): In this example, we consider the 3D SFFHN model with periodic boundary conditions, along with the following initial conditions on do-

main $[-2.5, 2.5]^3$:

$$u(x, y, z, 0) = \begin{cases} u\left(\frac{3N}{8} : \frac{5N}{8}, \frac{5N}{8} : \frac{6N}{8}, \frac{5N}{8} : \frac{6N}{8}, 0\right) = 1 \\ u\left(\frac{3N}{8} : \frac{5N}{8}, \frac{2N}{8} : \frac{3N}{8}, \frac{2N}{8} : \frac{3N}{8}, 0\right) = 1 \\ 0, \text{ elsewhere} \end{cases} \quad (5.4)$$

$$v(x, y, z, 0) = \begin{cases} v\left(\frac{5N}{8} : N, \frac{N}{4} : N, \frac{N}{4} : N\right) = 0.1 \\ v\left(\frac{N}{8} : \frac{3N}{8}, \frac{N}{4} : N, \frac{N}{4} : N\right) = 0.1 \\ 0, \text{ elsewhere.} \end{cases}$$

To further demonstrate the effectiveness and applicability of the methods, we applied them to investigate pattern formation in the 3D SFFHN model. In our analysis, we aimed to compare the accuracy and efficiency of the methods by conducting simulations on Example 4 until time $t = 5$. We kept the spectral resolution fixed at $N = 128$ and systematically reduced the temporal step size by halving it in each simulation, starting with an initial value of $k = 0.0625$.

The results regarding the order of convergence and computational time are depicted in Figures 10 and 11, respectively. From these figures, it is evident that all the methods achieved second-order convergence in the time direction for varying α values, as expected. Similar to the 2D case, the ETD-CN method stood out as the most efficient choice, considering both accuracy and computational efficiency. It demonstrated excellent performance in solving the 3D SFFHN model with periodic boundary conditions and varying fractional orders. These findings further emphasize the superiority of the ETD-CN method in capturing the dynamics and patterns of the 3D SFFHN model accurately while maintaining computational efficiency.

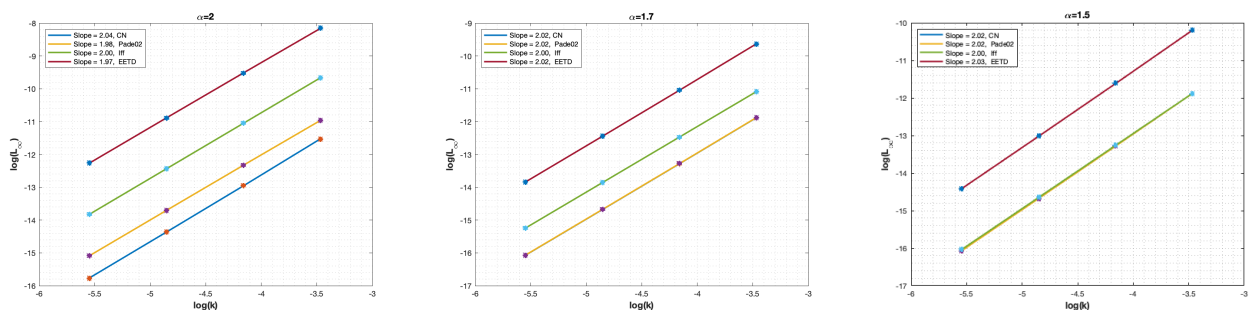


Figure 10. Empirical convergence analysis of the proposed methods in temporal direction for varying α in the case of periodic boundary conditions.

After confirming the accuracy and efficiency of the ETD-CN method for solving the 3D SFFHN model, we proceeded with a series of experiments to simulate the spatio-temporal evolution profiles of the component u and v for varying $\alpha = 2, 1.9, 1.8, 1.7$ to investigate the effects of super-diffusion on the SFFHN model. In these simulations, we used $N = 128, k = 1$ and ran the simulations until different times $t = 400, 500, 1000$, and 2000 . The complex patterns obtained from these simulations are illustrated in Figures 12 and 13. Notably, some of the patterns observed in these figures differ from those obtained in previous numerical work [9]. These unique patterns highlight the ability of the methods to capture novel and distinct dynamical behavior in the context of the SFFHN model. The results presented in Figures 12 and 13 provide valuable insights into the intricate spatio-temporal

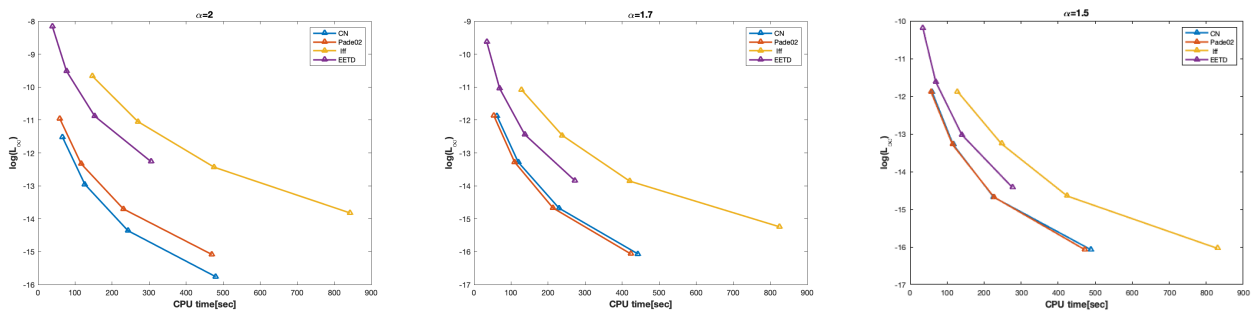


Figure 11. Comparison of the computational efficiency of proposed methods for varying α in the case of periodic boundary conditions.

dynamics and pattern formation in the 3D SFFHN model under the influence of super-diffusion.

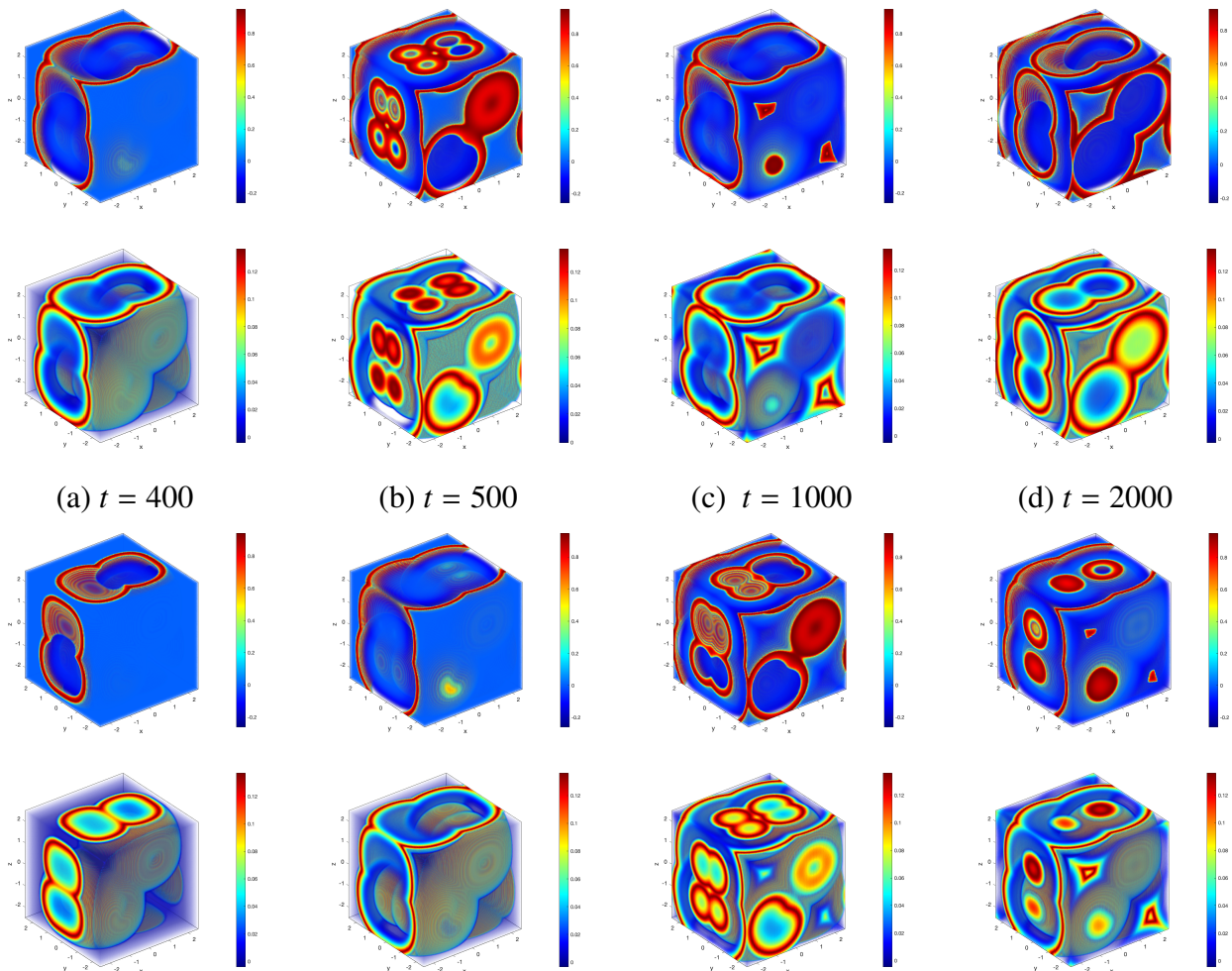


Figure 12. Complex patterns of the component u (odd rows) and v (even rows) for varying $\alpha = 2, 1.9$ (top-bottom) at various times $t = 400, 500, 1000$ and 2000 (left-right).

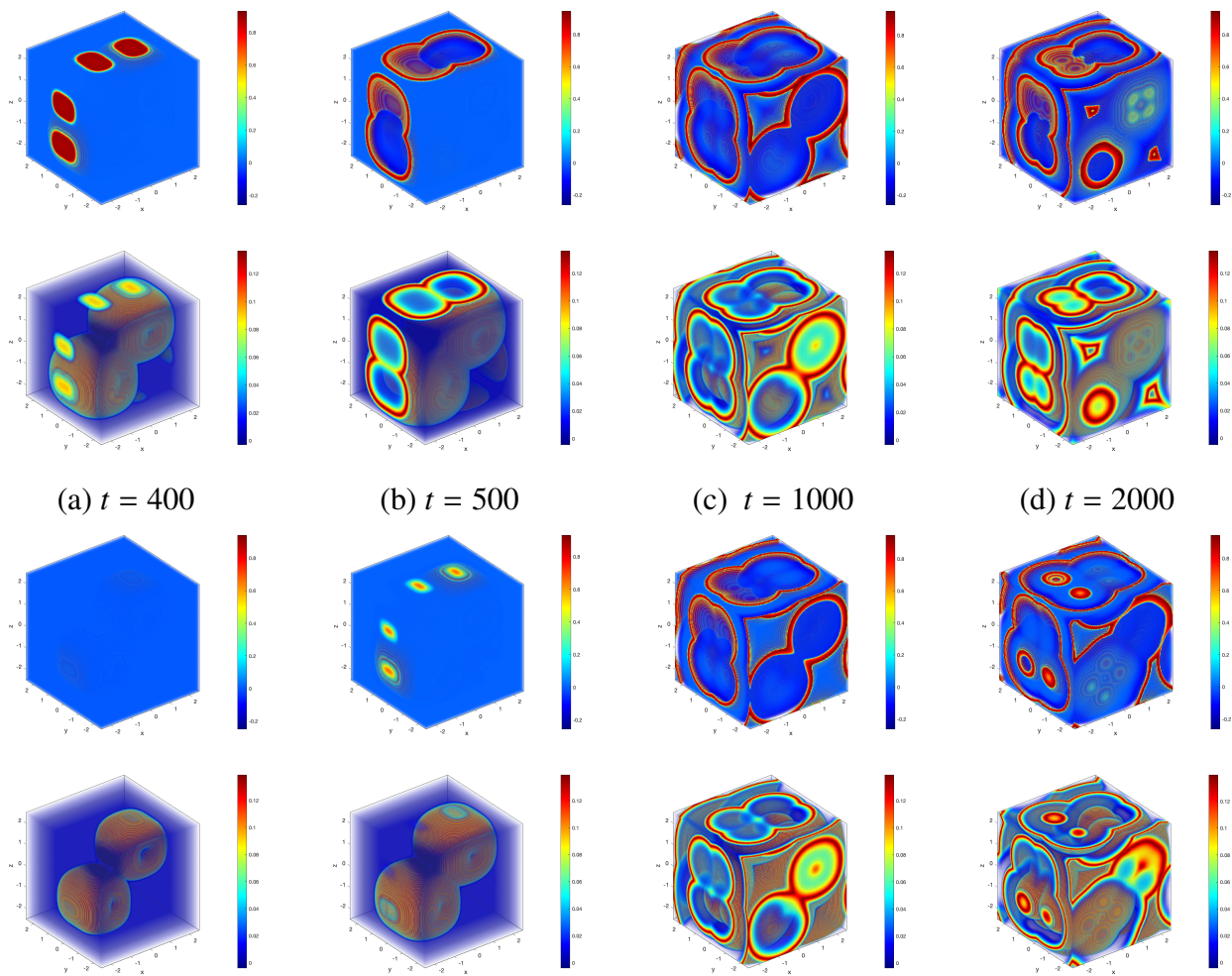


Figure 13. Complex patterns of the component u (odd rows) and v (even rows) for varying $\alpha = 1.8, 1.7$ (top-bottom) at various times $t = 400, 500, 1000$ and 2000 (left-right).

Example 5 (3D SFFHN model with Dirichlet boundary): In this example, we consider the 3D SFFHN model with homogeneous Dirichlet boundary conditions, along with the following initial conditions on domain $[-1.25, 1.25]^3$.

$$\begin{aligned}
 u(x, y, z, 0) &= \begin{cases} 1, & \text{if } x, y, z < 0 \\ 0, & \text{otherwise} \end{cases} \\
 v(x, y, z, 0) &= \begin{cases} 0.1, & \text{if } -1.25 \leq x \leq 1.25, y > 0, -1.25 \leq z \leq 1.25 \\ 0, & \text{otherwise.} \end{cases}
 \end{aligned} \tag{5.5}$$

In this experiment we conducted a series of simulations using the ETD- $\text{Padé}(0, 2)$ method with a fixed spectral resolution of $N = 128$, temporal step size of $k = 1$ and varying fractional orders $\alpha = 2, 1.8, 1.7, 1.5$. The simulations were performed until time $t = 2000$ to investigate the effect of the fractional operator on the wave propagation process in the 3D SFFHN model. Figure 14 provides visual representation of the system's evolution, highlighting the generation of a spiral wave from

the initial data. The process begins with the emergence of a traveling pulse, which subsequently starts rotating in a clockwise direction. Notably, as the fractional power decreases, the wave propagation extends toward the edge of the region. This phenomenon is accompanied by a noticeable reduction in both the width of the excitation wavefront and the wavelength of the system. The observed generation of spiral waves and the subsequent propagation dynamics shed light on the complex relationship between the fractional order and the spatio-temporal patterns in the system.

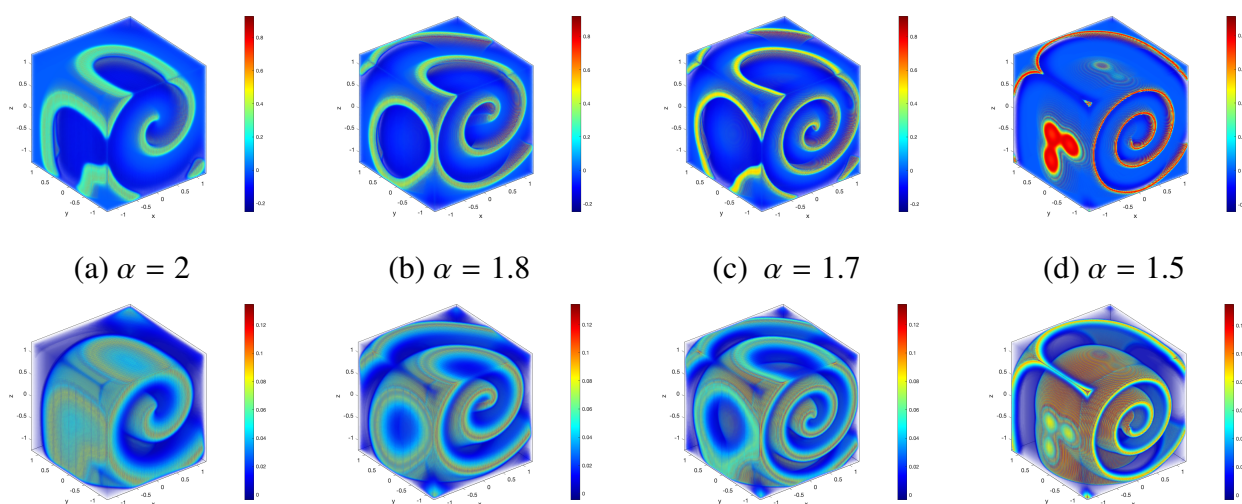


Figure 14. Spatio-temporal wave propagation scenario of the component u (first row) and v (second row) for varying $\alpha = 2, 1.8, 1.7, 1.5$ (left-right) at time $t = 2000$.

6. Conclusions

In this manuscript, we have introduced four second-order time integrators along with the Fourier spectral method to investigate the influence of the fractional operator α in the 2D and 3D SFFHN model with periodic, Dirichlet, or Neumann boundary conditions. The effectiveness of these methods was compared based on two key parameters: Accuracy and computational efficiency. The experimental results have demonstrated that all the considered methods accurately captured the long-time spatio-temporal solution profiles of the system for varying α , although with noticeable variations in CPU time. Considering both accuracy and computational cost when selecting a second-order method for time stepping the SFFHN model suggests that certain methods are more preferable than others, although all the methods yielded satisfactory results. Notably, the ETD-CN method emerged as the most efficient choice in this regard.

However, it is crucial to note that these conclusions are specific to the studies conducted on the SFFHN model with the specified initial and boundary conditions. The performance of the methods may vary depending on the specific case, such as the choice of initial conditions or other problem scenarios. It is important to emphasize that these results are not universally applicable and may differ for different sets of initial conditions, boundary conditions, and problem scenarios.

The methods introduced in this paper have the potential to be extended for solving fractional non-linear reaction-diffusion models with variable coefficients or models with a fractional Laplacian of complex order [26] posed in higher dimensions, which will be the focus of our future research project.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The author is grateful to the scholarly activities committee of Utah Valley University for the summer research award. In addition, the author is grateful to the editor and anonymous referees for their valuable comments and suggestions, which helped us to improve the quality of this manuscript.

Conflict of interest

There is no conflict of interest to declare.

References

1. R. FitzHugh, Impulses and physiological states in theoretical models of nerve membrane, *Biophys. J.*, **1** (1961), 445–466.
2. J. Nagumo, S. Arimoto, S. Yoshizawa, An active pulse transmission line simulating nerve axon, *Proc. IRE*, **50** (1962), 2061–2070. <https://doi.org/10.1109/jrproc.1962.288235>
3. J. Keener, J. Sneyd, *Mathematical Physiology I: Cellular Physiology*, 2nd edition, Springer, 2009.
4. F. Liu, I. Turner, V. Anh, Q. Yang, K. Burrage, A numerical method for the fractional FitzHugh–Nagumo monodomain model, *Anziam J.*, **54** (2013), C608–C629. <https://doi.org/10.21914/anziamj.v54i0.6372>
5. A. Bueno-Orovio, D. Kay, K. Burrage, Fourier spectral methods for fractional in-space reaction-diffusion equations, *BIT Numer. Math.*, **54** (2014), 937–954. <https://doi.org/10.1007/s10543-014-0484-2>
6. W. Bu, Y. Tang, Y. Wu, J. Yang, Crank–Nicolson ADI Galerkin finite element method for two-dimensional fractional FitzHugh–Nagumo monodomain model, *Appl. Math. Comput.*, **257** (2015), 355–364. <https://doi.org/10.1016/j.amc.2014.09.034>
7. Q. Li, F. Song, Splitting spectral element method for fractional reaction-diffusion equations, *J. Algorithms Comput. Technol.*, **14** (2020), 1–10. <https://doi.org/10.1177/1748302620966705>
8. F. W. Liu, P. H. Zhuang, I. Turner, V. Anh, K. Burrage, A semi-alternating direction method for a 2-D fractional FitzHugh–Nagumo monodomain model on an approximate irregular domain, *J. Comput. Phys.*, **293** (2015), 252–263. <https://doi.org/10.1016/j.jcp.2014.06.001>
9. X. Li, C. Han, Y. Wang, Novel patterns in fractional-in-space nonlinear coupled FitzHugh–Nagumo models with Riesz fractional derivative, *Fractal Fract.*, **6** (2022), 1–13. <https://doi.org/10.3390/fractalfract6030136>
10. J. Zhang, S. Lin, Z. Liu, F. Lin, An efficient numerical approach to solve the space fractional FitzHugh–Nagumo model, *Adv. Differ. Equ.*, **350** (2019). <https://doi.org/10.1186/s13662-019-2270-6>

11. H. G. Lee, A second-order operator splitting Fourier spectral method for fractional-in-space reaction–diffusion equations, *J. Comput. Appl. Math.*, **333** (2018), 395–403. <https://doi.org/10.1016/j.cam.2017.09.007>
12. H. Zhang, X. Jiang, F. Zeng, G. E. Karniadakis, A stabilized semi-implicit Fourier spectral method for nonlinear space-fractional reaction-diffusion equations, *J. Comput. Phys.*, **405** (2020), 1–17. <https://doi.org/10.1016/j.jcp.2019.109141>
13. Y. Wang, L. Cai, X. Feng, X. Luo, H. Gao, A ghost structure finite difference method for a fractional FitzHugh-Nagumo monodomain model on moving irregular domain, *J. Comput. Phys.*, **428** (2021), 110081. <https://doi.org/10.1016/j.jcp.2020.110081>
14. H. Che, W. Y. Lan, L. Z. Yuan, Novel patterns in a class of fractional reaction–diffusion models with the Riesz fractional derivative, *Math. Comput. Simul.*, **202** (2022), 149–163. <https://doi.org/10.1016/j.matcom.2022.05.037>
15. M. Almushaira, H. Bhatt, A. M. Al-rassas, Fast high-order method for multi-dimensional space-fractional reaction–diffusion equations with general boundary conditions, *Math. Comput. Simul.*, **182** (2021), 235–258. <https://doi.org/10.1016/j.matcom.2020.11.001>
16. S. Duo, H. Wang, Y. Zhang, A comparative study on nonlocal diffusion operators related to the fractional Laplacian, *Discrete Contin. Dyn. Syst.-Ser. B*, **24** (2019), 231–256. <https://doi.org/10.3934/dcdsb.2018110>
17. C. Pozrikidis, *The Fractional Laplacian*, CRC Press, Boca Raton, 2016.
18. C. V. Loan, *Computational Frameworks for the Fast Fourier Transform*, Frontiers in Applied Mathematics, SIAM, Philadelphia, PA, 1992.
19. J. D. Lawson, J. L. Morris, The extrapolation of first order methods for parabolic partial differential equations, *SIAM J. Numer. Anal.*, **15** (1978), 1212–1224. <https://doi.org/10.1137/0715082>
20. H. P. Bhatt, A. Q. M. Khaliq, The locally extrapolated exponential time differencing LOD scheme for multidimensional reaction-diffusion systems, *J. Comput. Appl. Math.*, **285** (2015), 256–278. <https://doi.org/10.1016/j.cam.2015.02.017>
21. B. Kleefeld, A. Q. M. Khaliq, B. A. Wade, An ETD Crank-Nicolson method for reaction-diffusion systems, *Numer. Methods Partial Differ. Equations*, **28** (2012), 1309–1335. <https://doi.org/10.1002/num.20682>
22. Q. Nie, Y. T. Zhang, R. Zhao, Efficient semi-implicit schemes for stiff systems, *J. Comput. Phys.*, **214** (2006), 521–537. <https://doi.org/10.1016/j.jcp.2005.09.030>
23. S. M. Cox, P. C. Matthews, Exponential time differencing for stiff systems, *J. Comput. Phys.*, **176** (2002), 430–455. <https://doi.org/10.1006/jcph.2002.6995>
24. B. Fornberg, T. A. Driscoll, A fast spectral for nonlinear wave equations with linear dispersion, *J. Comput. Phys.*, **155** (1999), 456–467. <https://doi.org/10.1006/jcph.1999.6351>
25. G. Beylkin, J. M. Keiser, L. Vozovoi, A new class of time discretization schemes for the solution of nonlinear PDEs, *J. Comput. Phys.*, **147** (1998), 362–387. <https://doi.org/10.1006/jcph.1998.6093>

-
26. A. Bueno-Orovio, D. Kay, K. Burrage, Complex-order fractional diffusion in reaction-diffusion systems, *Commun. Nonlinear Sci. Numer. Simul.*, **119** (2023), 107–120. <https://doi.org/10.1016/j.cnsns.2023.107120>



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)