Advanced Numerical Treatment of Newell Whitehead Segel Equation Using the Method of Lines with Third-Order Finite Difference Approximations

Muhammad Ameer Hamza[1], Muhammad Yaseen[1,*], Salma Trabelsi[2] and Meraa Arab[2,*]

[1] Department of Mathematics, University of Sargodha, Sargodha, Pakistan

[2] Department of Mathematics and Statistics, College of Science, King Faisal University, Al Ahsa, Saudi Arabia

INFORMATION

Keywords:
Newell-Whitehead-Segel equation
method of lines
higher order difference schemes
stability
convergence

DOI: 10.23967/j.rimni.2026.10.78494

SCIPEDIA Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería

Revista Internacional Métodos numéricos para cálculo y diseño en ingeniería RIMNI

In cooperation with CIMNE

Advanced Numerical Treatment of Newell Whitehead Segel Equation Using the Method of Lines with Third-Order Finite Difference Approximations

Muhammad Ameer Hamza[1], Muhammad Yaseen[1,*], Salma Trabelsi[2] and Meraa Arab[2,*]

[1]Department of Mathematics, University of Sargodha, Sargodha, Pakistan

[2]Department of Mathematics and Statistics, College of Science, King Faisal University, Al Ahsa, Saudi Arabia

ABSTRACT

The Newell-Whitehead-Segel equation (NWSE) is a foundational nonlinear model for understanding pattern development and bifurcation in a variety of physical and engineering systems, such as Rayleigh-Bénard convection, material microstructure evolution, and nanostructure self-assembly. This study proposes a strong high-order numerical technique for solving the NWSE that combines the Method of Lines with third-order finite difference approximations for spatial derivatives. The spatial discretization transforms the governing partial differential equation into a system of ordinary differential equations, which are then integrated in time using the standard fourth-order Runge-Kutta technique. A thorough stability and convergence analysis is carried out to determine the theoretical validity of the proposed method. Numerous numerical studies show that the approach is highly accurate, stable, and computationally advantageous across a number of examples of testing. This work makes a novel contribution by constructing third-order one-sided finite-difference stencils at the boundaries, which ensure high-order accuracy while successfully implementing Dirichlet boundary conditions and avoiding precision loss near domain boundaries. The suggested numerical framework is a reliable and effective tool for describing challenging pattern-forming systems, as well as precisely parametric studies for design and control applications in engineering and scientific studies.

OPEN ACCESS

Received: 01/01/2026

Accepted: 27/02/2026

Published: 29/05/2026

DOI
10.23967/j.rimni.2026.10.78494

Keywords:
Newell-Whitehead-Segel equation
method of lines
higher order difference schemes
stability
convergence

1 Introduction

Nonlinear partial differential equations are fundamental in modeling complex phenomena across numerous scientific and engineering disciplines including fluid dynamics, chemical engineering, material sciences and biomechanics. Among these, NWSE holds particular importance. Atta et al. [1] developed a spectral collocation method to solve the time-fractional Newell–Whitehead–Segel equation, showing that the approach yields accurate and efficient numerical solutions for fractional reaction–diffusion models. This connection makes it directly relevant to engineering problems in heat transfer, convective cooling and geophysical fluid dynamics. Beyond this, the NWSE serves as a canonical model for the emergence of stripped patterns in two-dimensional systems, a phenomenon critical to understanding material wrinkling, optical grating fabrication and organization of biological tissues.

*Correspondence: Muhammad Yaseen, Meraa Arab (yaseen.yaqoob@uos.edu.pk, marab@kfu.edu.sa).

In its general form, the NWSE is expressed as:

$$\frac{\partial \theta}{\partial t} = \gamma \frac{\partial^2 \theta}{\partial^2 s} + \alpha\theta - \beta\theta^q, \qquad s \in [a, b], \qquad t > 0, \tag{1}$$

subject to the initial condition

$$\theta(s, 0) = w(s) \tag{2}$$

and Dirichlet boundary conditions

$$\begin{cases} \theta(a, t) = f_1(t), \\ \theta(b, t) = f_2(t), \end{cases} \qquad t > 0, \tag{3}$$

here a, b, $w(s)$, $f_1(t)$ and $f_2(t)$ are given, and γ, α, β are real parameters with $\gamma > 0$ and q is positive integer. The function $\theta(s, t)$ can represent, for instance, a temperature distribution in a thin long rod or the flow velocity in a narrow pipe. The nonlinear source term $\alpha\theta - \beta\theta^q$ governs the growth and saturation of the amplitude. Notably, with $\alpha = \beta = \gamma = 1$ and $q = 3$, Eq. (1) reduces to the well known Allen-Cahn equation.

Despite the existence of numerous analytical and numerical techniques for solving nonlinear parabolic PDEs like the NWSE, there remains a need for computationally efficient and highly accurate numerical schemes that are suitable for engineering simulations. Such schemes must be rigorously analyzed for stability and convergence to ensure reliable predictions. Many existing semi-analytical approaches can have limitations in handling complex boundary conditions or achieving high precision over extended domains-scenarios commonplace in engineering design. A robust, high order numerical method that balances accuracy with computational efficiency is therefore essential for performing detailed simulations, sensitivity analysis and optimization studies in engineering systems modeled by the NWSE from microfluidic device design to large-scale thermal convection analysis.

In recent years, numerous methods and techniques have been developed to solve non-linear, parabolic partial differential equations. Several methods have been proposed to solve Newell-Whitehead-Segel-type equations. The differential transform approach was used by Aasaraai [2] to obtain the analytical solution of the NWSE. The bounded solution of the NWS equation is computed by Macias-Diaz and Ruiz-Ramirez [3] using a non-standard symmetry-preserving approach. The NWSE was solved by Hilal et al. [4] using the exponential finite difference method. Mahgoub and Sedeeg [5] solved the NWSE using the Elzaki Adomian decomposition method. The homotopy perturbation method was employed by Saravanan and Magesh [6] to obtain an accurate solution to the NWSE. He's variational iteration method was used by Prakash and Kumar [7] to solve the NWSE. The variational iteration method was utilized by Soori et al. [8] to determine the analytical solution to the NWSE. The cubic B-spline collocation approach was utilized by Zahra et al. [9] to obtain a numerical solution for the NWSE. Ezzati and Shakibi [10] solved the NWSE using multi-quadric quasi-interpolation and Adomian's decomposition method. Pue-On [11] applied the Laplace-Adomian decomposition method to solve the NWSE. Kheiri et al. [12] obtained analytic solutions of the NWSE by the homotopy analysis method and the homotopy Pade method. Patade and Bhalekar [13] applied a new iterative method to derive an approximate analytical solution to the NWSE, contributing to more efficient solutions in mathematical modeling contexts. Akinlabi and Edeki [14] introduced the perturbation iteration transform method for solving NWSE, enhancing computational efficiency for complex nonlinear equations. Jassim [15] employed the homotopy perturbation algorithm with Laplace transforms for the NWSE, enabling simplified and accurate solutions. Hashemi and Mirzazadeh [16] derived exact analytical solutions of the nonlinear stochastic

Newell–Whitehead–Segel equation using a reduction technique, revealing the impact of stochastic effects on the model's dynamics and pattern formation. Nadeem et al. [17] presented a numerical solution of the nonlinear fractional Newell–Whitehead–Segel equation using the natural transform method, demonstrating its effectiveness in handling fractional reaction–diffusion models. Bektaş and Anaç [18] developed a hybrid method based on the Shehu transform and q-homotopy analysis to solve the fractional Newell–Whitehead–Segel equation. Their approach efficiently captures nonlinear fractional dynamics and yields accurate approximate solutions with improved performance compared to existing methods. Mahgoub [19] utilized the homotopy perturbation method for solving the NWSE providing an alternative analytical approach for nonlinear equations. Nourazar et al. [20] explored exact solutions of the NWSE using the homotopy perturbation method advancing methods for obtaining precise solutions. Areshi et al. [21] developed an analytical framework based on a novel transform method to solve the fractional-order Newell–Whitehead–Segel equation, demonstrating its effectiveness in analyzing nonlinear fractional reaction–diffusion models. Hariharan [22] developed a Legendre wavelet-based approximation method for the NWS and Allen–Cahn equations contributing an efficient numerical method for solving differential equations. Inan et al. [23] examined analytical and numerical solutions for mathematical biology models, including the NWS and Allen–Cahn equations, broadening the application of these equations to biological systems. Yaseen and Batool [24] proposed a third-order finite difference scheme within the numerical method of lines to efficiently solve the classical heat equation. Their numerical results demonstrate improved accuracy and computational efficiency compared with the schemes for heat conduction problems. The finite difference method (FDM) has shown outstanding flexibility in solving complex physical models, especially those that include fractional-order dynamics and transport processes. Recent studies have demonstrated its usefulness in a variety of applications. For example, FDM-based methods have been used successfully to describe mass transport processes regulated by time-varying space-fractional Riesz derivatives, resulting in accurate and stable numerical solutions for anomalous diffusion systems [25]. Furthermore, finite difference approximations were combined with spectral approaches to analyze fractional heat transfer models in composite cylindrical structures, and Chebyshev collocation methods were employed to improve computational accuracy and efficiency. These studies demonstrate FDM's versatility and robustness in handling complex physical problems, highlighting its importance in the numerical research of nonlinear reaction-diffusion equations such as the model used in this study [26].

This attempt intends to formulate and validate a high-order numerical scheme for the NWSE that is both theoretically sound and useful to engineering studies. This paper presents specialized third-order one-sided stencils at domain boundaries that retain global accuracy without fog areas or sacrificing stability. This work provides a robust and accurate computational tool with thoroughly evaluated boundary treatments for investigating pattern selection, stability thresholds, and temporal dynamics in real-world systems.

The primary objective of this work is to develop and validate a high-order numerical scheme for the NWSE that is theoretically sound and readily applicable to engineering investigations. The distinct approach used here involves construction of specialized third-order ond-sided stencils at domain boundaries which maintain global third-order accuracy without requiring ghost points or sacrificing stability. By providing a stable and accurate computational tool with rigorously analyzed boundary treatments, this work facilitates the study of pattern selection, stability thresholds and transient dynamics in real-world systems.

The structure of the paper proceeds as follows: In Section 2, we describe the computational method using the method of lines and third-order approximations to solve the NWS equation.

Sections 3 and 4 focus on the stability and convergence analysis of this method to ensure it is reliable for numerical simulations. In Section 5, we compare our results with those from other numerical methods found in the literature. Finally, the conclusion summarizes our findings and highlights the effectiveness of the proposed method.

2 The Derivation of the Scheme

Let $h = \frac{b-a}{N+1}$ and $\Delta t = \frac{T_f}{N_t}$, where N is a natural number, T_f is the final time N_t is the number of time steps. The spatial domain $a \leq s \leq b$ is divided into $N+2$ equivalent sub-intervals $[s_j, s_{j+1}]$, $j = 0, 1, 2, ..., N+1$ such that $a = s_0 < s_1 < ... < s_{N-1} < s_N < s_{N+1} = b$ with a uniform step size h. Thus the interval, $[a, b]$ is partitioned into $N+1$ sub intervals each of width h. This discretization, together with the temporal step Δt generates a rectangular mesh with grid points (s_m, t_n), where s_m and t_n denote the spatial and temporal grid points, respectively. The set of grid points covers the computation region $R = \{(s, t) : a \leq s \leq b, t > 0\}$ subject to the boundary conditions at $s = a$ and $s = b$ and the initial condition at $t = 0$. Assuming that the spatial derivative in Eq. (1) is approximated using a five-point difference formula, a third-order accuracy can be achieved at a typical grid point (s_m, t_n). Accordingly following [24], we obtain

$$\frac{\partial^2\theta(s,t)}{\partial s^2} = \frac{1}{h^2}\{\mu_1\theta(s-h,t) + \mu_2\theta(s,t) + \mu_3\theta(s+h,t) + \mu_4\theta(s+2h,t) + \mu_5\theta(s+3h,t)\}. \tag{4}$$

As observed, the above approximation is not symmetric; therefore, a five-point stencil is employed. By expanding $\theta(s-h,t)$, $\theta(s+h,t)$, $\theta(s+2h,t)$ and $\theta(s+3h,t)$ in (4) using Taylor series expansions about the point (s,t), we obtain

$$\begin{aligned} h^2\frac{\partial^2\theta(s,t)}{\partial s^2} &= (\mu_1+\mu_2+\mu_3+\mu_4+\mu_5)\theta(s,t) \\ &+ (-\mu_1+\mu_3+2\mu_4+3\mu_5)h\frac{\partial\theta(s,t)}{\partial s} \\ &+ \frac{1}{2!}(\mu_1+\mu_3+4\mu_4+9\mu_5)h^2\frac{\partial^2\theta(s,t)}{\partial s^2} \\ &+ \frac{1}{3!}(-\mu_1+\mu_3+8\mu_4+27\mu_5)h^3\frac{\partial^3\theta(s,t)}{\partial s^3} \\ &+ \frac{1}{4!}(\mu_1+\mu_3+16\mu_4+81\mu_5)h^4\frac{\partial^4\theta(s,t)}{\partial s^4} \\ &+ \frac{1}{5!}(-\mu_1+\mu_3+32\mu_4+243\mu_5)h^5\frac{\partial^5\theta(s,t)}{\partial s^5} + ... \end{aligned} \tag{5}$$

By equating the coefficients of h^k for $k = 0, 1, 2, 3, 4$ in Eq. (5), the following system of linear equations is obtained:

$$\begin{aligned}
\mu_1 + \mu_2 + \mu_3 + \mu_4 + \mu_5 &= 0,\\
-\mu_1 + \mu_3 + 2\mu_4 + 3\mu_5 &= 0,\\
\mu_1 + \mu_3 + 4\mu_4 + 9\mu_5 &= 2,\\
-\mu_1 + \mu_3 + 8\mu_4 + 27\mu_5 &= 0,\\
\mu_1 + \mu_3 + 16\mu_4 + 81\mu_5 &= 0.
\end{aligned} \tag{6}$$

Solving the system (6) yields

$$\mu_1 = \frac{11}{12},\ \mu_2 = \frac{-5}{3},\ \mu_3 = \frac{1}{2},\ \mu_4 = \frac{1}{3},\ \mu_5 = \frac{-1}{12}. \tag{7}$$

Substituting these coefficients into Eq. (4), the resulting finite difference approximation becomes

$$\begin{aligned}
\frac{\partial^2\theta(s,t)}{\partial s^2} = \frac{1}{12h^2}\{&11\theta(s-h,t) - 20\theta(s,t) + 6\theta(s+h,t) + 4\theta(s+2h,t)\\
&-\theta(s+3h,t)\} + \frac{h^3}{12}\frac{\partial^5\theta(s,t)}{\partial s^5} + O(h^4)\ as\ h \to 0.
\end{aligned} \tag{8}$$

To maintain third-order accuracy at the boundary points (s_{N-1}, t_n) and (s_N, t_n), additional finite difference formulas are required. These approximations must preserve the truncation error term $\frac{h^3}{12}\frac{\partial^5\theta(s,t)}{\partial s^5}$. Accordingly six-point stencils are employed at these locations. Retaining the factor $(12h^2)^{-1}$, as used in Eq. (8), facilitates consistency across the scheme. At the grid point (s_{m-1}, t_n), the approximation is assumed in the form

$$\begin{aligned}
12h^2\frac{\partial^2\theta(s,t)}{\partial s^2} &= \mu_1\theta(s-3h,t) + \mu_2\theta(s-2h,t) + \mu_3\theta(s-h,t)\\
&+ \mu_4\theta(s,t) + \mu_5\theta(s+h,t) + \mu_6\theta(s+2h,t) + h^5\frac{\partial^5\theta(s,t)}{\partial s^5}.
\end{aligned} \tag{9}$$

Then

$$\begin{aligned}
12h^2\frac{\partial^2\theta(s,t)}{\partial s^2} &= (\mu_1 + \mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6)\theta(s,t)\\
&+ (-3\mu_1 - 2\mu_2 - \mu_3 + \mu_5 + 2\mu_6)h\frac{\partial\theta(s,t)}{\partial s}\\
&+ \frac{1}{2!}(9\mu_1 + 4\mu_2 + \mu_4 + 4\mu_6)h^2\frac{\partial^2\theta(s,t)}{\partial s^2}\\
&+ \frac{1}{3!}(-27\mu_1 - 8\mu_2 - \mu_3 + \mu_5 + 8\mu_6)h^3\frac{\partial^3\theta(s,t)}{\partial s^3}\\
&+ \frac{1}{4!}(81\mu_1 + 16\mu_2 + \mu_3 + \mu_5 + 16\mu_6)h^4\frac{\partial^4\theta(s,t)}{\partial s^4}\\
&+ \frac{1}{5!}(-243\mu_1 - 32\mu_2 - \mu_3 + \mu_5 + 32\mu_6)h^5\frac{\partial^5\theta(s,t)}{\partial s^5} + \ldots
\end{aligned} \tag{10}$$

Equating h^k for $k = 0, 1, 2, 3, 4, 5$ in (10) gives

$$\begin{aligned}
&\mu_1 + \mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6 = 0,\\
&-3\mu_1 - 2\mu_2 - \mu_3 + \mu_5 + 2\mu_6 = 0,\\
&9\mu_1 + 4\mu_2 + \mu_3 + \mu_5 + 4\mu_6 = 24,\\
&-27\mu_1 - 8\mu_2 - \mu_3 + \mu_5 + 8\mu_6 = 0,\\
&81\mu_1 + 16\mu_2 + \mu_3 + \mu_5 + 16\mu_6 = 0,\\
&-243\mu_1 - 32\mu_2 - \mu_3 + \mu_5 + 32\mu_6 = 120.
\end{aligned} \tag{11}$$

The system of linear Eqs. (11) has the solution:

$$\mu_1 = 1,\ \mu_2 = -6,\ \mu_3 = 26,\ \mu_4 = -40,\ \mu_5 = 21,\ \mu_6 = -2. \tag{12}$$

Hence, the third-order accurate approximation at (s_{m-1}, t_n) is given by

$$\begin{aligned}
\frac{\partial^2\theta(s,t)}{\partial s^2} = \frac{1}{12h^2}\{&\theta(s-3h,t) - 6\theta(s-2h,t) + 26\theta(s-h,t) - 40\theta(s,t)\\
&+ 21\theta(s+h,t) - 2\theta(s+2h,t)\} + \frac{h^3}{12}\frac{\partial^5\theta(s,t)}{\partial s^5} + O(h^4)\ as\ h \to 0.
\end{aligned} \tag{13}$$

Similarly for the boundary point (s_m, t_n), a backward six-point stencil is constructed as:

$$\begin{aligned}
12h^2\frac{\partial^2\theta(s,t)}{\partial s^2} = &\mu_1\theta(s-4h,t) + \mu_2\theta(s-3h,t) + \mu_3\theta(s-2h,t)\\
&+ \mu_4\theta(s-h,t) + \mu_5\theta(s,t) + \mu_6\theta(x+h,t) + h^5\frac{\partial^5\theta(s,t)}{\partial s^5}.
\end{aligned} \tag{14}$$

Then, we have

$$\begin{aligned}
12h^2\frac{\partial^2\theta(s,t)}{\partial s^2} = &(\mu_1 + \mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6)\theta(s,t)\\
&+ (-4\mu_1 - 3\mu_2 - 2\mu_3 - \mu_4 + \mu_6)h\frac{\partial\theta(s,t)}{\partial s}\\
&+ \frac{1}{2!}(16\mu_1 + 9\mu_2 + 4\mu_3 + \mu_4 + \mu_6)h^2\frac{\partial^2\theta(s,t)}{\partial s^2}\\
&+ \frac{1}{3!}(-64\mu_1 - 27\mu_2 - 8\mu_3 - \mu_4 + \mu_6)h^3\frac{\partial^3\theta(s,t)}{\partial s^3}\\
&+ \frac{1}{4!}(256\mu_1 + 81\mu_2 + 16\mu_3 + \mu_4 + \mu_6)h^4\frac{\partial^4\theta(s,t)}{\partial s^4}\\
&+ \frac{1}{5!}(-1024\mu_1 - 243\mu_2 - 32\mu_3 - \mu_4 + \mu_6)h^5\frac{\partial^5\theta(s,t)}{\partial s^5} + \ldots
\end{aligned} \tag{15}$$

Equating h^k for $k = 0, 1, 2, 3, 4$ in (15) gives

$$\begin{aligned}
&\mu_1 + \mu_2 + \mu_3 + \mu_4 + \mu_5 + \mu_6 = 0,\\
&-4\mu_1 - 3\mu_2 - 2\mu_3 - \mu_4 + \mu_6 = 0,\\
&16\mu_1 + 9\mu_2 + 4\mu_3 + \mu_4 + \mu_6 = 24,\\
&-64\mu_1 - 27\mu_2 - 8\mu_3 - \mu_4 + \mu_6 = 0,\\
&256\mu_1 + 81\mu_2 + 16\mu_3 + \mu_4 + \mu_6 = 0,\\
&-1024\mu_1 - 243\mu_2 - 32\mu_3 - \mu_4 + \mu_6 = -120.
\end{aligned} \tag{16}$$

The system of linear Eqs. (16) has solution:

$$\mu_1 = 2,\ \mu_2 = -11,\ \mu_3 = 24,\ \mu_4 = -14,\ \mu_5 = -10,\ \mu_6 = 9. \tag{17}$$

Therefore, for the grid point (s_m, t_n), the intended approximation is as follows:

$$\begin{aligned}
\frac{\partial^2\theta(s,t)}{\partial s^2} = \frac{1}{12h^2}\{&2\theta(s-4h,t) - 11\theta(s-3h,t) + 24\theta(s-2h,t)\\
&- 14\theta(s-h,t) - 10\theta(s,t) + 9\theta(s+h,t)\} + \frac{h^3}{12}\frac{\partial^5\theta(s,t)}{\partial s^5} + O(h^4)\ as\ h \to 0.
\end{aligned} \tag{18}$$

The proposed approach's fundamental spatial discretization consists of the approximations (8), (13) and (18). The inside Formula (8) uses a normal five-point asymmetric stencil. However, the boundary stencils (13) and (18) are distinctive in that they use six-point one-sided formulas that maintain third-order precision at the last two grid points without extending beyond the domain. This enables the technique to handle Dirichlet conditions directly while maintaining the same level of accuracy across the spatial mesh.

Finally, substituting Eqs. (8), (13) and (18) into the governing equation yields the following system of ordinary differential equations:

$$\begin{aligned}
\frac{\partial\theta_i}{\partial t} = \gamma\frac{1}{12h^2}\{&11\theta(s_i-h,t) - 20\theta(s_i,t) + 6\theta(s_i+h,t)\\
&+ 4\theta(s_i+2h,t) - \theta(s_i+3h,t)\} + \alpha\theta - \beta\theta^q,\ 1 \le i \le m-2.
\end{aligned} \tag{19}$$

For $i = N-1$, we have

$$\begin{aligned}
\frac{\partial\theta_{N-1}}{\partial t} = \gamma\frac{1}{12h^2}\{&\theta(s_{N-1}-3h,t) - 6\theta(s_{N-1}-2h,t) + 26\theta(s_{N-1}-h,t)\\
&- 40\theta(s_{N-1},t) + 21\theta(s_{N-1}+h,t) - 2\theta(b,t) + \alpha\theta - \beta\theta^q
\end{aligned} \tag{20}$$

and for $i = N$, we obtain

$$\begin{aligned}
\frac{\partial\theta_N}{\partial t} = \gamma\frac{1}{12h^2}\{&2\theta(s_N-4h,t) - 11\theta(s_N-3h,t) + 24\theta(s_N-2h,t)\\
&- 14\theta(s_N-h,t) - 10\theta(s_N,t) + 9\theta(b,t)\} + \alpha\theta - \beta\theta^q,
\end{aligned} \tag{21}$$

where $\theta(a,t)$ and $\theta(b,t)$ are known boundary conditions. The resulting system of ODEs is solved numerically using the classical fourth-order Runge-Kutta (RK4) method.

Algorithm 1 shows the implementation of the proposed method.

Algorithm 1: Method of lines with RK4 for the NWSE

Input : Spatial domain $[a, b]$, number of spatial subintervals m, final time T_f, number of time steps N_t, parameters γ, α, β, q, initial condition $w(s)$, boundary conditions $f_1(t), f_2(t)$.
Output: Numerical solution θ_i^n at grid points s_i and time levels t_n.
Step 1: Discretize spatial domain
$h = (b-a)/N + 1$
Define spatial grid points $s_i = a + ih$ for $i = 0, 1, \ldots, N+1$
Step 2: Set time step
$\Delta t = T_f/N_t$
Step 3: Apply initial condition
for $i = 0$ **to** N **do**
$\theta_i^0 = w(s_i)$
Step 4: Time marching
for $n = 0$ **to** $N_t - 1$ **do**
4a. Form ODE system: Assemble the right-hand side vector $\mathbf{F}(\boldsymbol{\theta}^n)$, where the component F_i for each interior point i is given by:

- For $1 \le i \le N-2$: use centered scheme.
- For $i = N-1$: use near-boundary scheme.
- For $i = N$: use boundary-adjacent scheme.

Boundary values θ_0^{n+1} and θ_{N+1}^{n+1} are set directly from $f_1(t_{n+1})$ and $f_2(t_{n+1})$.
4b. Apply RK4 for time integration:

$$\mathbf{k}_1 = \mathbf{F}(\boldsymbol{\theta}^n)$$

$$\mathbf{k}_2 = \mathbf{F}\left(\boldsymbol{\theta}^n + \tfrac{\Delta t}{2}\mathbf{k}_1\right)$$

$$\mathbf{k}_3 = \mathbf{F}\left(\boldsymbol{\theta}^n + \tfrac{\Delta t}{2}\mathbf{k}_2\right)$$

$$\mathbf{k}_4 = \mathbf{F}\left(\boldsymbol{\theta}^n + \Delta t\mathbf{k}_3\right)$$

$$\boldsymbol{\theta}^{n+1} = \boldsymbol{\theta}^n + \tfrac{\Delta t}{6}\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right)$$

Step 5: Return the solution matrix $\boldsymbol{\theta}$.

3 Stability Analysis

This section presents a detailed stability analysis of the proposed scheme. To facilitate the analysis, we first linearize the NWSE equation by neglecting the nonlinear term appearing in Eq. (1), that is, $\alpha\theta - \beta\theta^q = 0$. Consequently, the linearized form of the NWSE reduces to the following equation:

$$\frac{\partial \theta}{\partial t} = \gamma \frac{\partial^2 \theta}{\partial^2 s}, \qquad t > 0. \tag{22}$$

The semi-discrete system of ordinary differential equations obtained earlier from Eqs. (8), (12) and (16) can be written compactly in matrix form as

$$\frac{\partial \boldsymbol{\theta}}{\partial t} = \mathbf{R}\boldsymbol{\theta}, \tag{23}$$

subject to the initial condition

$\boldsymbol{\theta}(s_j, 0) = \boldsymbol{w}(s_j)$,

where the system matrix $\mathbf{R}$ is defined by

$$\mathbf{R} = \frac{\gamma}{12h^2}\begin{bmatrix} 11 & -20 & 6 & 4 & -1 & 0 & 0 & \dots & 0 \\ 0 & 11 & -20 & 6 & 4 & -1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & -6 & 26 & -40 & 21 & -2 \\ 0 & \dots & 0 & 2 & -11 & 24 & -14 & -10 & 9 \end{bmatrix}$$

for $j = 0, 1, 2, ..., m+1$ and the solution vector is given by

$$\boldsymbol{\theta} = \begin{pmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_{m+1} \end{pmatrix}.$$

The temporal discretization of the system (23) is carried out using the classical fourth-order Runge-Kutta method. Accordingly, the numerical solution is advanced in time by the iterative scheme

$$\begin{cases} \mathbf{n}_1 = \mathbf{R}\boldsymbol{\theta}_k \\ \mathbf{n}_2 = \mathbf{R}(\boldsymbol{\theta}_k + \dfrac{\Delta t}{2}\mathbf{n}_1) \\ \mathbf{n}_3 = \mathbf{R}(\boldsymbol{\theta}_k + \dfrac{\Delta t}{2}\mathbf{n}_2) \\ \mathbf{n}_4 = \mathbf{R}(\boldsymbol{\theta}_k + \Delta t\mathbf{n}_3) \\ \boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \dfrac{\Delta t}{6}(\mathbf{n}_1 + 2\mathbf{n}_2 + 2\mathbf{n}_3 + \mathbf{n}_4), \end{cases} \tag{24}$$

where $\widehat{\theta}_k$ denotes the numerical solution vector at the k-th time level. To perform a von Neumann stability analysis, we assume a Fourier-mode of the form

$\boldsymbol{\theta}_k = \boldsymbol{\rho}^k e^{\iota\omega\Delta t}$,

where $\boldsymbol{\rho}^k$ represents the amplification factor and ω denotes the mode number. Substituting this assumed solution into Eq. (24), we obtain

$$\boldsymbol{\rho}^{k+1}e^{\iota\omega\Delta t} = \boldsymbol{\rho}^k e^{\iota\omega\Delta t} + \frac{\Delta t}{6}(\mathbf{n}_1 + 2\mathbf{n}_2 + 2\mathbf{n}_3 + \mathbf{n}_4). \tag{25}$$

Now we substitute the expression for $\mathbf{n}_1$, $\mathbf{n}_2$, $\mathbf{n}_3$, and $\mathbf{n}_4$ into above equation:

$$\begin{aligned} \boldsymbol{\rho}^{k+1}e^{\iota\omega\Delta t} &= \boldsymbol{\rho}^k e^{\iota\omega\Delta t} \\ &+ \frac{\Delta t}{6}\Big[\mathbf{R}\boldsymbol{\rho}^k e^{\iota\omega\Delta t} + 2\mathbf{R}\Big(\boldsymbol{\rho}^k e^{\iota\omega\Delta t} + \frac{\Delta t}{2}\mathbf{n}_1\Big) \\ &+ 2\mathbf{R}\Big(\boldsymbol{\rho}^k e^{\iota\omega\Delta t} + \frac{\Delta t}{2}\mathbf{n}_2\Big) + \mathbf{R}\big(\boldsymbol{\rho}^k e^{\iota\omega\Delta t} + \Delta t\,\mathbf{n}_3\big)\Big]. \end{aligned}$$

After algebraic simplification, the above expression reduces to

$$\boldsymbol{\rho}^{k+1}e^{\iota\omega\Delta t} = \boldsymbol{\rho}^{k}e^{\iota\omega\Delta t}(\boldsymbol{I} + \Delta t\mathbf{R} + \frac{(\Delta t\mathbf{R})^2}{2} + \frac{(\Delta t\mathbf{R})^3}{6} + \frac{(\Delta t\mathbf{R})^4}{24}). \tag{26}$$

Equivalently, Eq. (26) may be written in the compact form

$$\widehat{\boldsymbol{\theta}}_{k+1} = \widehat{\boldsymbol{\theta}}_{k}(\boldsymbol{I} + \Delta t\mathbf{R} + \frac{(\Delta t\mathbf{R})^2}{2} + \frac{(\Delta t\mathbf{R})^3}{6} + \frac{(\Delta t\mathbf{R})^4}{24}). \tag{27}$$

From Eq. (27), we define the amplification matrix $\boldsymbol{C}$ associated with RK4 time integration as

$$\boldsymbol{C} = \boldsymbol{I} + \Delta t\mathbf{R} + \frac{(\Delta t\mathbf{R})^2}{2} + \frac{(\Delta t\mathbf{R})^3}{6} + \frac{(\Delta t\mathbf{R})^4}{24}.$$

Thus the fully discrete scheme can be expressed as

$$\boldsymbol{\theta}_{k+1} = \mathbf{C}\boldsymbol{\theta}_{k}. \tag{28}$$

For stability, it is necessary that the spectral radius of the amplification matrix $\mathbf{C}$ satisfies $\rho(\mathbf{C}) \leq 1$. Since the RK4 method is an explicit time integration scheme, it is not A-stable and its stability depends critically on the time step Δt and the eigenvalues of the system matrix $\mathbf{R}$. In particular, for stiff system or when $\mathbf{R}$ possesses large negative eigenvalues, the time step Δt must be sufficiently small to ensure that all eigenvalues of $\Delta t\mathbf{R}$ remain within the stability region of the RK4 method. Despite this restriction, the RK4 scheme provides excellent stability and accuracy for a wide class of problems, provided that an appropriate time step is chosen in accordance with the spectral radius of $\mathbf{R}$. This can be validated from Table 1.

Table 1: Spectral radius $\rho(\mathbf{C})$ of the amplification matrix for different time steps Δt and diffusion coefficients γ

Δt	γ		
	1	2	0.9
10^{-2}	1.64×10^{4}	4.63×10^{5}	1.00×10^{4}
10^{-3}	5.44	26.1	3.6
10^{-4}	1.19	1.41	0.17
10^{-5}	1.02	1.03	0.02

Practical Time-Step Restriction

For the linearized diffusion problem $\theta_t = \gamma\theta_{ss}$, the spatial discretization matrix $\mathbf{R}$ has eigenvalues λ_j that depend on h and the boundary stencils. The RK4 stability region in the complex plane requires that $\Delta t\,\lambda_j$ lie within this region for all j.

For a purely real and negative spectrum (as in the centered diffusion discretization), the RK4 stability interval along the negative real axis extends to approximately

$$\Delta t \max_{j} |\lambda_j| \lesssim 2.785.$$

The eigenvalues of $\mathbf{R}$ for the third-order stencil (8), (13) and (18) can be bounded via Gershgorin's theorem or computed numerically. A conservative estimate for the maximum eigenvalue magnitude is

$$|\lambda_{\max}| \approx \frac{\gamma}{h^2}\, c,$$

where c is a constant determined by the stencil coefficients. For the interior stencil (8), $c \approx 8$. Including boundary stencils slightly increases this bound.

Therefore, a practical stability condition is

$$\Delta t \lesssim \frac{2.785}{|\lambda_{\max}|} \approx \frac{0.35\, h^2}{\gamma},$$

which mirrors the usual parabolic CFL condition for explicit methods. In all numerical experiments, we chose Δt respecting this condition, ensuring $\rho(\mathbf{C}) \leq 1$.

4 Convergence Analysis

Convergence analysis of the fourth-order Runge-Kutta ($RK4$) method for a system of $m+1$ ordinary differential equations is concerned with demonstrating that the numerical solution converges to the exact solution as the time step Δt tends to zero. In this section, we examine the local truncation error (LTE) of the RK4 method and discuss its implications for the global convergence behavior of the scheme.

Consider the system of ordinary differential equations

$$\frac{\partial \boldsymbol{\theta}_i}{\partial t} = F_i(t, \boldsymbol{\theta}), \;\; i = 0, 1, 2, ..., N, N+1, \tag{29}$$

where t is independent variable, $\boldsymbol{\theta}$ denotes the vector of dependent variables evaluated at the spatial grid points and F_i represents the right hand side function of the system. When the system (29) is discretized in time using the RK4 method, the numerical solution is advanced according to the following scheme:

$$\begin{cases} \boldsymbol{r}_1 = \boldsymbol{F}(t_n, \boldsymbol{\theta}_n) \\ \boldsymbol{r}_2 = \boldsymbol{F}(t_n + \dfrac{\Delta t}{2}, \boldsymbol{\theta}_n + \dfrac{\Delta t}{2}\boldsymbol{r}_1) \\ \boldsymbol{r}_3 = \boldsymbol{F}(t_n + \dfrac{\Delta t}{2}, \boldsymbol{\theta}_n + \dfrac{\Delta t}{2}\boldsymbol{r}_2) \\ \boldsymbol{r}_4 = \boldsymbol{F}(t_n + \Delta t, \boldsymbol{\theta}_n + \Delta t \boldsymbol{r}_3) \\ \boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}_n + \dfrac{\Delta t}{6}(\boldsymbol{r}_1 + 2\boldsymbol{r}_2 + 2\boldsymbol{r}_3 + \boldsymbol{r}_4), \end{cases} \tag{30}$$

where $\boldsymbol{\theta}_n$ denotes the numerical approximation at time $t = t_n$. The LTE of a numerical method measures the error introduced during a single time step when the exact solution is substituted into the numerical scheme. For the $RK4$ method, the LTE is given by

$$LTE = \frac{1}{720}\Delta t^5 \theta^{(5)}(\zeta), \tag{31}$$

where $\theta^{(5)}$ denotes the 5-th derivative of the exact solution with respect to time and $\zeta \in [t_n, t_n + \Delta t]$. Eq. (31) shows that the LTE of the RK4 method is proportional to Δt^5 and hence decreases rapidly as $\Delta t \to 0$. This property is a direct consequence of the fourth-order accuracy of the RK4 method.

The global error, which represents the accumulated error over all time steps up to a fixed final time, is obtained by summing the local truncation errors over approximately $O(\Delta t^5)$ time steps. As a result, the global error of the RK4 method behaves as $O(\Delta t^4)$ confirming the fourth-order convergence of the method. To make this argument precise, let us define

$$P = \max_{\zeta \in [t_n, t_n + \Delta t]} \mid \theta^{(5)}(\zeta) \mid, \tag{32}$$

where P denotes an upper bound on the magnitude of the fifth derivative of the exact solution over the time interval $[t_n, t_n + \Delta t]$. Provided that the exact solution is sufficiently smooth, the quantity P remains bounded. Using the bound, the magnitude of LTE satisfies

$$\mid LTE \mid = \frac{1}{720}\Delta t^5 \mid \theta^{(5)}(\zeta) \mid \leq \frac{1}{720}\Delta t^5 P,$$

which clearly tends to zero as $\Delta t \to 0$. The fourth order convergence of the $RK4$ method implies that reducing the time step Δt leads to a significant decrease in the numerical error, allowing high accuracy to be achieved with relatively few time steps compared to lower-order methods. However, it should be noted that for stiff problems, stability constraints may necessitate the use of smaller time steps despite the favorable convergence properties.

Overall Convergence of the Full Scheme

The overall numerical scheme proposed in this work consists of two distinct sources of error: the spatial discretization error arising from the third-order finite difference approximations (yielding a local truncation error of $O(h^3)$ and the temporal discretization error arising from the fourth-order Runge-Kutta method (yielding a local truncation error of $O(\Delta t^5)$ and a global error of Δt^4. Assuming the exact solution is sufficiently smooth, the global error of the fully discrete method is expected to be of the form $O(h^3) + O(\Delta t^4)$.

5 Results and Discussion

This section presents a variety of test problems to examine and evaluate the accuracy of the suggested method. The following errors are defined to measure the precision of the solutions as well as to quantify the errors.

Error Metrics

Let $\theta(s_j, t_n)$ denote the exact solution and θ_j^n the numerical approximation at spatial grid point $s_j = a + jh$ and time level $t_n = n\Delta t$, with $j = 0, 1, \ldots, N+1$ and $n = 0, 1, \ldots, N_t$. The following error measures are used:

- **Absolute error** at a grid point:

  $E_{\text{abs}}(s_j, t_n) = |\theta(s_j, t_n) - \theta_j^n|.$

- L_∞ **error** at time t_n:

  $L_\infty(t_n) = \max_{0 \leq j \leq m+1} E_{\text{abs}}(s_j, t_n).$

- L_2 **error** at time t_n:

  $$L_2(t_n) = \sqrt{h \sum_{j=0}^{m+1} \left(\theta(s_j, t_n) - \theta_j^n\right)^2},$$

where $h = (b-a)/(N+1)$ is the spatial step.

Example 1: *Consider Eq. (1) with* $\gamma = 1$, $\alpha = 1$, $\beta = 1$ *and* $q = 4$*:*

$$\frac{\partial \theta}{\partial t} = \frac{\partial^2 \theta}{\partial^2 s} + \theta - \theta^4, \quad 0 \le s \le 1 \quad t > 0, \tag{33}$$

with boundary conditions,

$$\begin{cases} \theta(0,t) = \left(\frac{1}{2} + \frac{1}{2} tanh\left(\frac{21}{20}t\right)\right)^{\frac{2}{3}}, \\ \theta(1,t) = \left(\frac{1}{2} + \frac{1}{2} tanh\left(-\frac{3}{2\sqrt{10}}\left(1 - \frac{7t}{\sqrt{10}}\right)\right)\right)^{\frac{2}{3}}, \end{cases} \tag{34}$$

and the initial condition,

$$\theta(s,0) = \left(1 + e^{\frac{3s}{\sqrt{10}}}\right)^{-\frac{2}{3}}. \tag{35}$$

The exact solution for this problem is given by

$$\theta(s,t) = \left(\frac{1}{2} + \frac{1}{2}\tanh\left(-\frac{3}{2\sqrt{10}}\left(s - \frac{7t}{\sqrt{10}}\right)\right)\right)^{\frac{2}{3}}.$$

The provided methodology is applied to solve the aforementioned problem. Table 2 presents a comparison of the absolute errors obtained using the present scheme with those reported in [4] and [9] for the parameters $T = 1$, $h = 0.0125$ and $\Delta t = 10^{-4}$. Table 3 reports the L_∞ and L_2 error norms at $T = 10$ with spatial step size $h = 0.05$ and compares the results with those obtained using the method proposed in [9]. Table 4 presents the spatial convergence results for Example 1. The numerical errors decrease consistently as the mesh size h is refined and the observed orders are approximately 3, confirming the third order accuracy spatial accuracy. the CPU time increase moderately with mesh refinements. The results in these tables clearly indicate that the present method produces more accurate solutions than the methods presented in [4,9]. Fig. 1 illustrates a comparison between the exact and numerical solutions at different spatial points. Fig. 2 shows the space-time surface plots of the approximate and exact solutions. In addition, the 3d error distribution for $h = 0.05$ is displayed in Fig. 3.

Table 2: Absolute errors for Example 1 with $\Delta t = 10^{-4}$ and $h = 0.0125$

s/t	Method	0.2	0.4	0.6	0.8	1
0.2	**Present**	1.06×10^{-9}	6.51×10^{-10}	8.05×10^{-11}	5.37×10^{-10}	5.72×10^{-10}
[4]	UCBS	3.80×10^{-4}	8.19×10^{-4}	1.11×10^{-3}	1.18×10^{-3}	1.08×10^{-3}
	TCBS	3.95×10^{-4}	8.36×10^{-4}	1.13×10^{-3}	1.20×10^{-3}	1.10×10^{-3}
	ECBS	9.67×10^{-4}	1.90×10^{-3}	2.45×10^{-3}	2.52×10^{-3}	2.25×10^{-3}
[9]	I-EFD	3.66×10^{-5}	3.46×10^{-5}	2.78×10^{-5}	2.08×10^{-5}	1.49×10^{-5}
	FI-EFD	3.64×10^{-5}	3.55×10^{-5}	2.98×10^{-5}	2.33×10^{-5}	1.73×10^{-5}
0.4	**Present**	1.69×10^{-9}	1.22×10^{-9}	6.72×10^{-11}	7.61×10^{-10}	9.07×10^{-10}

(Continued)

Table 2 (continued)

s/t	Method	0.2	0.4	0.6	0.8	1
[4]	UCBS	4.23×10^{-4}	1.11×10^{-3}	1.61×10^{-3}	1.78×10^{-3}	1.66×10^{-3}
	TCBS	4.45×10^{-4}	1.14×10^{-3}	1.64×10^{-3}	1.81×10^{-3}	1.69×10^{-3}
	ECBS	1.16×10^{-3}	2.64×10^{-3}	3.59×10^{-3}	3.81×10^{-3}	3.46×10^{-3}
[9]	I-EFD	3.52×10^{-5}	3.67×10^{-5}	3.04×10^{-5}	2.31×10^{-5}	1.66×10^{-5}
	FI-EFD	3.45×10^{-5}	3.76×10^{-5}	3.30×10^{-5}	2.66×10^{-5}	2.03×10^{-5}
0.6	**Present**	1.74×10^{-9}	1.43×10^{-9}	2.48×10^{-10}	7.12×10^{-10}	9.55×10^{-10}
[4]	UCBS	2.88×10^{-4}	1.01×10^{-3}	1.57×10^{-3}	1.79×10^{-3}	1.70×10^{-3}
	TCBS	3.11×10^{-4}	1.03×10^{-3}	1.60×10^{-3}	1.82×10^{-3}	1.73×10^{-3}
	ECBS	8.86×10^{-4}	2.44×10^{-3}	3.51×10^{-3}	3.84×10^{-3}	3.56×10^{-3}
[9]	I-EFD	3.65×10^{-5}	3.92×10^{-5}	3.35×10^{-5}	2.62×10^{-5}	1.94×10^{-5}
	FI-EFD	3.55×10^{-5}	3.99×10^{-5}	3.60×10^{-5}	2.98×10^{-5}	2.31×10^{-5}
0.8	**Present**	1.18×10^{-9}	1.06×10^{-9}	2.73×10^{-10}	4.40×10^{-10}	6.65×10^{-10}
[4]	UCBS	1.12×10^{-4}	6.12×10^{-4}	1.02×10^{-3}	1.20×10^{-3}	1.17×10^{-3}
	TCBS	1.27×10^{-4}	6.30×10^{-4}	1.04×10^{-3}	1.23×10^{-3}	1.19×10^{-3}
	ECBS	4.22×10^{-4}	1.51×10^{-3}	2.29×10^{-3}	2.58×10^{-3}	2.44×10^{-3}
[9]	I-EFD	4.03×10^{-5}	4.20×10^{-5}	3.72×10^{-5}	3.02×10^{-5}	2.32×10^{-5}
	FI-EFD	3.95×10^{-5}	4.23×10^{-5}	3.87×10^{-5}	3.26×10^{-5}	2.57×10^{-5}

Table 3: Error norms with $n = 19$, $\theta \in [0, 1]$, $\Delta t = 10^{-4}$ and $h = 0.05$ for Example 1

t	I-EFD		FI-EFD		Present	
	L_∞ Error	L_2 Error	L_∞ Error	L_2 Error	L_∞ Error	L_2 Error
0.01	3.13×10^{-4}	1.26×10^{-4}	3.13×10^{-4}	1.25×10^{-4}	1.13×10^{-9}	8.95×10^{-9}
0.1	4.13×10^{-4}	2.95×10^{-4}	4.10×10^{-4}	2.89×10^{-4}	9.74×10^{-10}	6.23×10^{-8}
5	9.10×10^{-6}	6.60×10^{-6}	9.20×10^{-6}	6.70×10^{-6}	5.42×10^{-13}	2.07×10^{-11}
10	2.51×10^{-10}	1.83×10^{-10}	2.53×10^{-10}	1.84×10^{-10}	2.1×10^{-14}	4.13×10^{-16}

Table 4: Spatial convergence as well as CPU time study for Example 1 with $\Delta t = 10^{-5}$, $t = 1$

h	L_∞ Error	Observed Order	CPU Time (s)
0.1	5.24×10^{-7}	—	0.653
0.05	6.45×10^{-8}	3.02	1.388
0.025	8.05×10^{-9}	3.002	2.672
0.0125	1.005×10^{-9}	3.0017	4.323

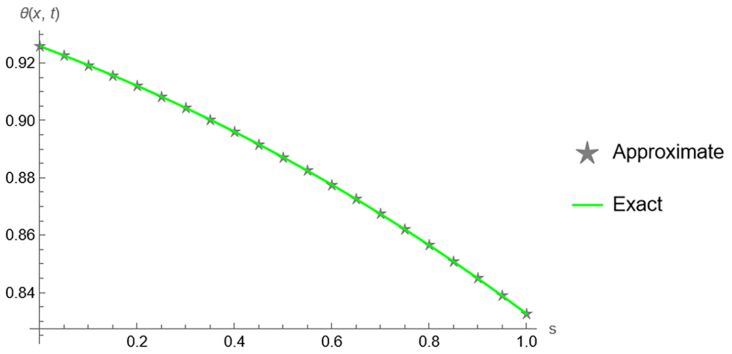

Figure 1: Accurate (solid line) and numerical (stars) solutions with a step size of $h = 0.05$ at $t = 1$ for Example 1

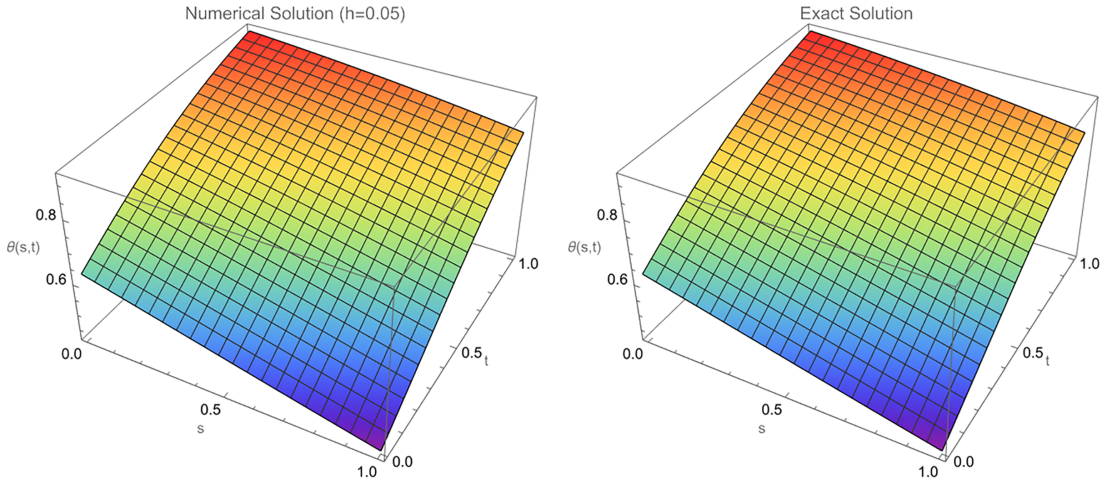

Figure 2: Approximate and exact solutions for Example 1

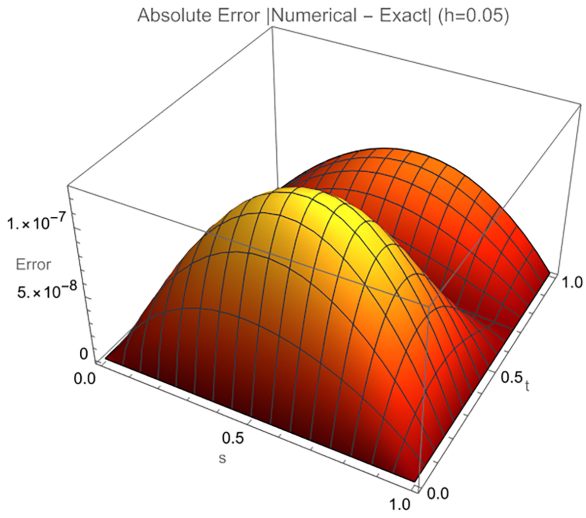

Figure 3: Error distribution with a step size of $h = 0.05$ for Example 1

Key Observation: The present scheme yields errors several orders of magnitude smaller than the comparator methods. This improvement is attributable to (i) the third-order spatial discretization, (ii) the exact treatment of Dirichlet boundaries via high-order one-sided stencils, and (iii) the fourth-order time integration. In contrast, the methods in [4,9] use lower-order spatial approximations and/or larger effective time steps, which explains the observed discrepancy in accuracy.

To verify the third-order spatial accuracy, we fix a sufficiently small time step $\Delta t = 10^{-5}$ and refine the spatial step h.

The computed order approaches 3 as h decreases, confirming the third-order spatial accuracy.

Example 2: *Consider Eq. (1) with* $\gamma = 1$, $\alpha = 3$, $\beta = 4$ *and* $q = 3$*:*

$$\frac{\partial \theta}{\partial t} = \frac{\partial^2 \theta}{\partial^2 s} + 3\theta - 4\theta^3, \quad 0 \le s \le 1 \quad t > 0, \tag{36}$$

with boundary conditions,

$$\begin{cases} \theta(0,t) = \sqrt{\dfrac{3}{4}} \dfrac{1}{1 + e^{-\frac{9}{2}t}}, \\ \theta(1,t) = \sqrt{\dfrac{3}{4}} \dfrac{e^{\sqrt{6}}}{e^{\sqrt{6}s} + e^{\frac{\sqrt{6}}{2}s}} \end{cases} \tag{37}$$

and the initial condition,

$$\theta(s,0) = \sqrt{\frac{3}{4}} \frac{e^{\sqrt{6}s}}{e^{\sqrt{6}s} + e^{\frac{\sqrt{6}-9t}{2}}}. \tag{38}$$

The exact solution for this problem is given by

$$\theta(s,t) = \sqrt{\frac{3}{4}} \frac{e^{\sqrt{6}}s}{e^{\sqrt{6}s} + e^{\frac{\sqrt{6}s}{2} - \frac{9t}{2}}}.$$

The proposed methodology is applied to solve the aforementioned problem. Table 5 presents a comparison of the absolute errors obtained using the present scheme with those reported in [4] and [9] for the parameters $T = 1$, $h = 0.0125$ and $\Delta t = 10^{-4}$. Table 6 reports the L_∞ and L_2 error norms at $T = 1$ with $h = 0.05$ and compares the results with those obtained using the method proposed in [9]. Table 7 shows the spatial convergence results and CPU time at $t = 0.8$. The errors decrease rapidly as the mesh is refined and the observed orders remain close to 3, confirming the third-order spatial accuracy. The results presented in these tables clearly indicate that the present method yields more accurate solutions at various spatial locations. Fig. 4 illustrates a comparison between the exact and numerical solutions at various spatial locations. Fig. 5 shows the space-time surface plots of the approximate and exact solutions. Furthermore, 3d error distribution for $h = 0.05$ is shown in Fig. 6.

Table 5: Absolute errors for Example 2 with $\Delta t = 10^{-4}$ and $h = 0.0125$

s/t	Method	0.2	0.4	0.6	0.8	1
0.2	**Present**	4.73×10^{-10}	2.07×10^{-9}	5.38×10^{-10}	1.85×10^{-11}	7.12×10^{-11}
[4]	UCBS	6.13×10^{-2}	7.12×10^{-2}	4.31×10^{-2}	4.66×10^{-2}	1.56×10^{-2}
	TCBS	6.13×10^{-2}	7.13×10^{-2}	6.43×10^{-2}	4.67×10^{-2}	1.56×10^{-2}
	ECBS	5.17×10^{-2}	5.40×10^{-2}	4.51×10^{-2}	2.86×10^{-2}	1.52×10^{-2}
[9]	I-EFD	5.84×10^{-5}	3.31×10^{-5}	1.51×10^{-5}	6.40×10^{-6}	9.77×10^{-4}
	FI-EFD	6.70×10^{-5}	4.24×10^{-5}	2.05×10^{-5}	9.00×10^{-6}	9.77×10^{-4}
0.4	**Present**	9.65×10^{-10}	2.73×10^{-9}	6.89×10^{-10}	3.31×10^{-11}	9.77×10^{-11}
[4]	UCBS	8.71×10^{-2}	1.01×10^{-1}	8.77×10^{-2}	6.00×10^{-2}	1.64×10^{-2}
	TCBS	8.72×10^{-2}	1.01×10^{-1}	8.79×10^{-2}	6.02×10^{-2}	1.65×10^{-2}
	ECBS	7.22×10^{-2}	7.42×10^{-2}	5.89×10^{-2}	3.32×10^{-2}	4.32×10^{-3}
[9]	I-EFD	4.24×10^{-5}	2.35×10^{-5}	1.01×10^{-5}	4.10×10^{-6}	1.09×10^{-3}
	FI-EFD	5.53×10^{-5}	3.65×10^{-5}	1.76×10^{-5}	7.70×10^{-6}	1.09×10^{-3}
0.6	**Present**	1.30×10^{-9}	2.39×10^{-9}	5.74×10^{-10}	4.24×10^{-11}	9.15×10^{-11}
[4]	UCBS	8.03×10^{-2}	9.07×10^{-2}	7.64×10^{-2}	4.90×10^{-2}	9.88×10^{-3}
	TCBS	8.04×10^{-2}	9.09×10^{-2}	7.65×10^{-2}	4.91×10^{-2}	9.97×10^{-3}
	ECBS	6.46×10^{-2}	6.46×10^{-2}	4.82×10^{-2}	2.33×10^{-2}	9.68×10^{-3}
[9]	I-EFD	3.37×10^{-5}	1.80×10^{-5}	7.4×10^{-6}	3.0×10^{-6}	9.07×10^{-4}
	FI-EFD	4.68×10^{-5}	3.02×10^{-5}	1.44×10^{-5}	6.20×10^{-6}	9.06×10^{-4}
0.8	**Present**	1.11×10^{-9}	1.40×10^{-9}	3.10×10^{-10}	3.70×10^{-11}	5.86×10^{-11}
[4]	UCBS	4.86×10^{-2}	5.32×10^{-2}	6.42×10^{-2}	2.58×10^{-2}	2.86×10^{-3}
	TCBS	4.86×10^{-2}	5.33×10^{-2}	4.32×10^{-2}	2.59×10^{-2}	2.92×10^{-3}
	ECBS	3.72×10^{-2}	3.59×10^{-2}	2.50×10^{-2}	9.64×10^{-3}	9.12×10^{-3}
[9]	I-EFD	3.23×10^{-5}	1.60×10^{-5}	6.60×10^{-6}	2.70×10^{-6}	5.80×10^{-4}
	FI-EFD	4.13×10^{-5}	2.38×10^{-5}	1.10×10^{-5}	4.70×10^{-6}	5.80×10^{-4}

Table 6: Error norms with $n = 19$, $\theta \in [0, 1]$, $\Delta t = 10^{-4}$ and $h = 0.05$ for Example 2

t	I-EFD		FI-EFD		Present	
	L_∞ Error	L_2 Error	L_∞ Error	L_2 Error	L_∞ Error	L_2 Error
0.01	6.93×10^{-4}	2.39×10^{-4}	6.91×10^{-4}	2.40×10^{-4}	5.06×10^{-8}	3.01×10^{-8}
0.1	8.40×10^{-4}	4.59×10^{-4}	8.52×10^{-4}	5.06×10^{-4}	6.29×10^{-8}	4.39×10^{-8}
5	6.76×10^{-11}	4.9×10^{-11}	6.9×10^{-11}	5.0×10^{-11}	2.0×10^{-15}	1.50×10^{-15}

Table 7: Spatial convergence as well as CPU time study for Example 1 with $\Delta t = 10^{-5}$, $t = 0.8$

h	L_∞ Error	Observed Order	CPU Time (s)
0.1	1.15×10^{-7}	—	0.469
0.05	1.42×10^{-8}	3.0177	0.766
0.025	1.76×10^{-9}	3.012	1.266
0.0125	2.18×10^{-10}	3.013	3.938

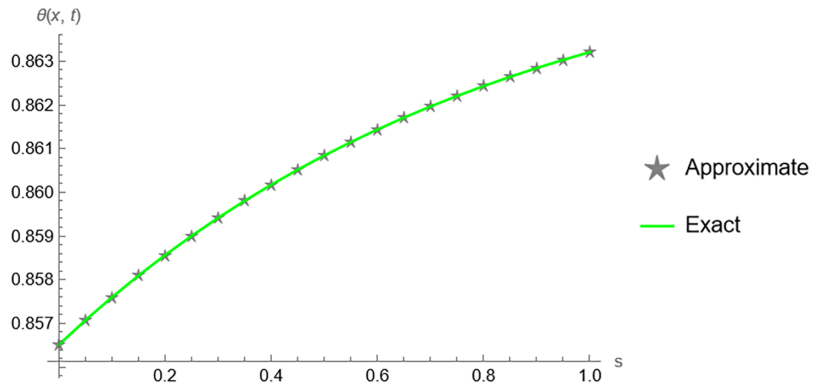

Figure 4: The accurate (solid) and numerical (stars) solutions with a step size $h = 0.05$ at time $t = 1$ for Example 2

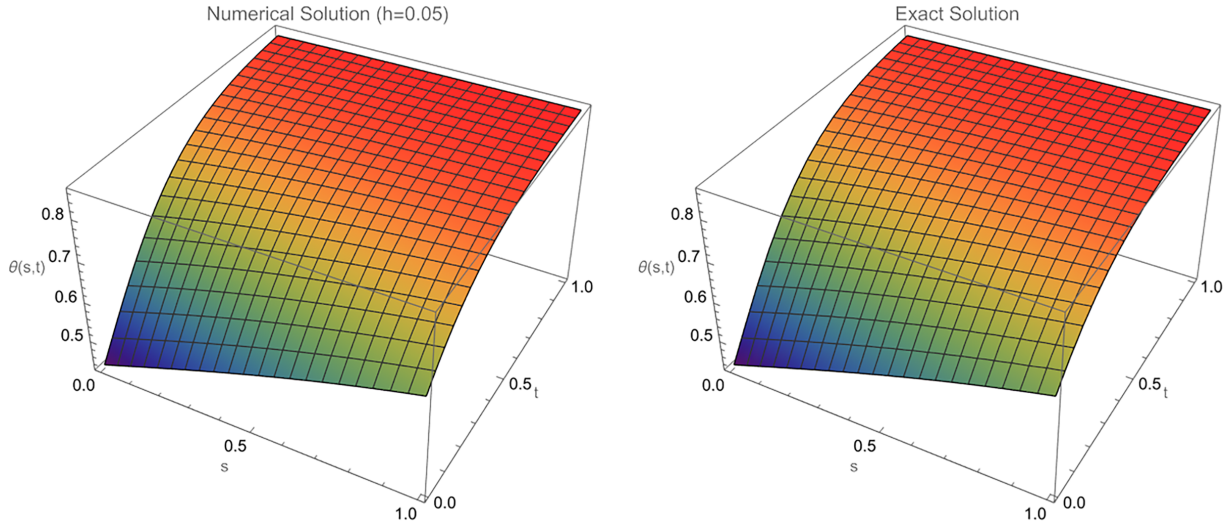

Figure 5: Approximate and exact solution for Example 2

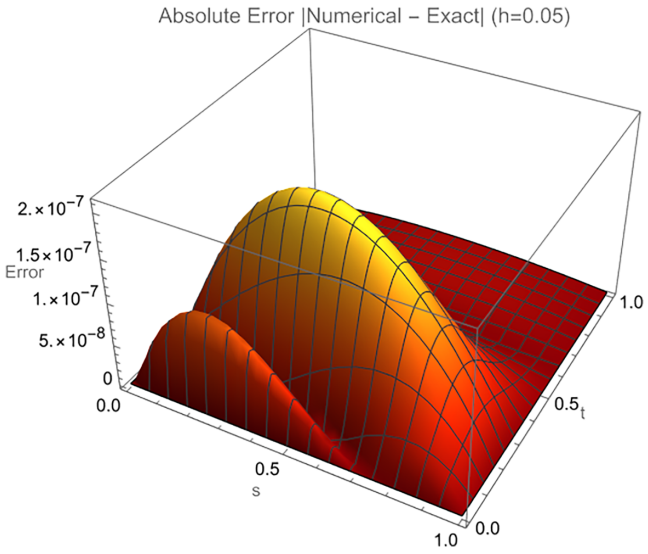

Figure 6: Error distribution with step size of $h = 0.05$ for Example 2

To verify the third-order spatial accuracy, we fix a sufficiently small time step $\Delta t = 10^{-5}$ and refine the spatial step h.

The computed order approaches 3 as h decreases, confirming the third-order spatial accuracy.

Example 3: *Consider Eq. (1) with* $\gamma = 1$, $\alpha = 1$, $\beta = 1$ *and* $q = 2$*:*

$$\frac{\partial \theta}{\partial t} = \frac{\partial^2 \theta}{\partial^2 s} + \theta - \theta^2, \quad 0 \le s \le 1 \quad t > 0, \tag{39}$$

with boundary conditions,

$$\begin{cases} \theta(0,t) = \dfrac{1}{\left(1 + e^{-\frac{5t}{6}}\right)^2}, \\ \theta(1,t) = \dfrac{1}{\left(1 + e^{\frac{1}{\sqrt{6}} - \frac{5t}{6}}\right)^2}, \end{cases} \tag{40}$$

and the initial condition,

$$\theta(s,0) = \frac{1}{\left(1 + e^{\frac{s}{\sqrt{6}}}\right)^2}. \tag{41}$$

The exact solution for this problem is given by

$$\theta(s,t) = \frac{1}{\left(1 + e^{\frac{s}{\sqrt{6}} - \frac{5t}{6}}\right)^2}.$$

The provided methodology is employed to solve the aforementioned problem. Table 8 presents a comparison of the absolute errors obtained using present scheme with those reported in [4,9], for

parameters $T = 1$, $h = 0.0125$ and $\Delta t = 10^{-4}$. Table 9 reports the L_∞ and L_2 error norms at $T = 5$ with spatial step size $h = 0.05$ and compares the result with those obtained by the method presented in [9]. Table 10 presents the spatial convergence results and CPU time at $t = 0.5$. The numerical errors decrease consistently with mesh refinement and the observed orders are close to 3, confirming third-order spatial accuracy of the scheme. The results in these tables demonstrate that the present method yields more accurate solutions than the method presented in [4,9]. Fig. 7 illustrates a comparison between the exact and numerical solutions at various spatial locations. Fig. 8 depicts the space-time surface plots of both the approximate and exact solutions. Furthermore, the 3d error distribution for $h = 0.05$ is shown in Fig. 9.

Table 8: Absolute errors for Example 3 with $\Delta t = 10^{-4}$ and $h = 0.0125$

s/t	Method	0.2	0.4	0.6	0.8	1
0.2	**Present**	1.41×10^{-10}	5.67×10^{-11}	5.63×10^{-11}	6.07×10^{-11}	5.74×10^{-11}
[4]	UCBS	2.81×10^{-4}	3.03×10^{-4}	2.48×10^{-4}	1.66×10^{-4}	7.41×10^{-5}
	TCBS	2.72×10^{-4}	3.25×10^{-4}	2.34×10^{-4}	1.51×10^{-4}	5.79×10^{-5}
	ECBS	6.91×10^{-4}	6.76×10^{-4}	5.18×10^{-4}	3.20×10^{-4}	1.09×10^{-4}
[9]	I-EFD	1.94×10^{-5}	2.25×10^{-5}	2.38×10^{-5}	2.43×10^{-5}	2.44×10^{-5}
	FI-EFD	1.86×10^{-5}	2.17×10^{-5}	2.31×10^{-5}	2.38×10^{-5}	2.40×10^{-5}
0.4	**Present**	2.20×10^{-10}	8.22×10^{-11}	8.23×10^{-11}	9.15×10^{-11}	8.91×10^{-11}
[4]	UCBS	4.27×10^{-4}	4.74×10^{-4}	3.97×10^{-4}	2.76×10^{-4}	1.39×10^{-4}
	TCBS	4.13×10^{-4}	4.55×10^{-4}	3.76×10^{-4}	2.54×10^{-4}	1.15×10^{-4}
	ECBS	1.06×10^{-3}	1.06×10^{-3}	8.33×10^{-4}	5.38×10^{-4}	2.23×10^{-4}
[9]	I-EFD	1.71×10^{-5}	2.16×10^{-5}	2.32×10^{-5}	2.40×10^{-5}	2.42×10^{-5}
	FI-EFD	1.60×10^{-5}	2.04×10^{-5}	2.21×10^{-5}	2.31×10^{-5}	2.37×10^{-5}
0.6	**Present**	2.14×10^{-10}	7.75×10^{-11}	7.96×10^{-11}	9.13×10^{-11}	9.12×10^{-11}
[4]	UCBS	4.38×10^{-4}	4.91×10^{-4}	4.18×10^{-4}	2.99×10^{-4}	1.62×10^{-4}
	TCBS	4.25×10^{-4}	4.72×10^{-4}	3.97×10^{-4}	2.77×10^{-4}	1.38×10^{-4}
	ECBS	1.09×10^{-3}	1.10×10^{-3}	8.80×10^{-4}	5.90×10^{-4}	2.74×10^{-4}
[9]	I-EFD	1.60×10^{-5}	2.09×10^{-5}	2.27×10^{-5}	2.37×10^{-5}	2.42×10^{-5}
	FI-EFD	1.51×10^{-5}	1.96×10^{-5}	2.16×10^{-5}	2.28×10^{-5}	2.35×10^{-5}
0.8	**Present**	1.30×10^{-10}	4.75×10^{-11}	5.12×10^{-11}	6.04×10^{-11}	6.17×10^{-11}
[4]	UCBS	3.05×10^{-4}	3.37×10^{-4}	2.89×10^{-4}	2.11×10^{-4}	1.20×10^{-4}
	TCBS	2.96×10^{-4}	2.91×10^{-4}	2.76×10^{-4}	1.96×10^{-4}	1.04×10^{-4}
	ECBS	7.48×10^{-4}	7.55×10^{-4}	6.12×10^{-4}	4.22×10^{-4}	2.13×10^{-4}
[9]	I-EFD	1.67×10^{-5}	2.04×10^{-5}	2.22×10^{-5}	2.34×10^{-5}	2.41×10^{-5}
	FI-EFD	1.59×10^{-5}	1.95×10^{-5}	2.14×10^{-5}	2.28×10^{-5}	2.37×10^{-5}

Table 9: Error norms with $n = 19$, $\theta \in [0, 1]$, $\Delta t = 10^{-4}$ and $h = 0.05$ for Example 3

t	I-EFD		FI-EFD		Present	
	L_∞ Error	L_2 Error	L_∞ Error	L_2 Error	L_∞ Error	L_2 Error
0.01	1.49×10^{-4}	5.38×10^{-4}	1.49×10^{-4}	5.34×10^{-5}	1.06×10^{-9}	8.50×10^{-10}
0.1	2.0×10^{-4}	1.31×10^{-4}	1.98×10^{-4}	1.25×10^{-4}	1.29×10^{-9}	8.51×10^{-10}
5	3.52×10^{-5}	2.82×10^{-5}	3.59×10^{-5}	3.05×10^{-5}	1.03×10^{-10}	7.56×10^{-11}
10	6.67×10^{-5}	4.87×10^{-5}	6.68×10^{-5}	4.88×10^{-5}	8.60×10^{-12}	6.23×10^{-12}

Table 10: Spatial convergence as well as CPU time study for Example 1 with $\Delta t = 10^{-5}$, $t = 0.5$

h	L_∞ Error	Observed Order	CPU Time (s)
0.1	4.27×10^{-8}	—	0.375
0.05	5.4×10^{-9}	2.983	0.984
0.025	6.86×10^{-10}	2.977	2.047
0.0125	7.95×10^{-11}	3.109	3.734

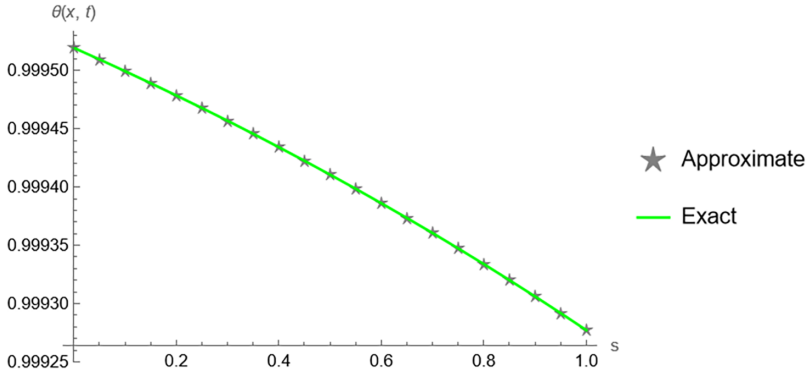

Figure 7: The accurate (solid) and numerical (stars) solutions with a step size $h = 0.05$ at time $t = 1$ for Example 3

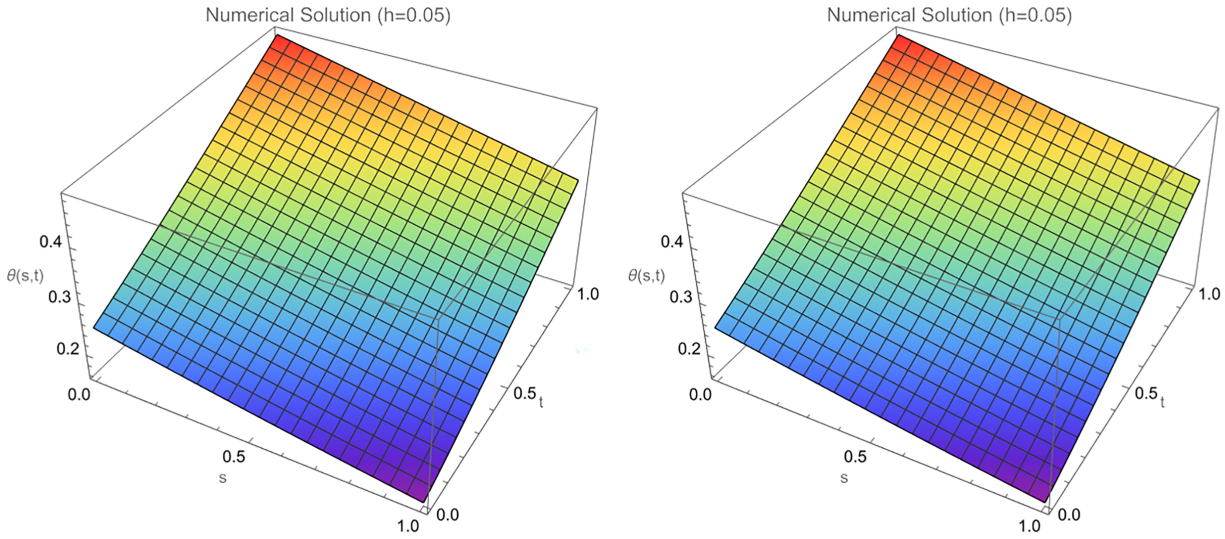

Figure 8: Approximate and exact solutions for Example 3

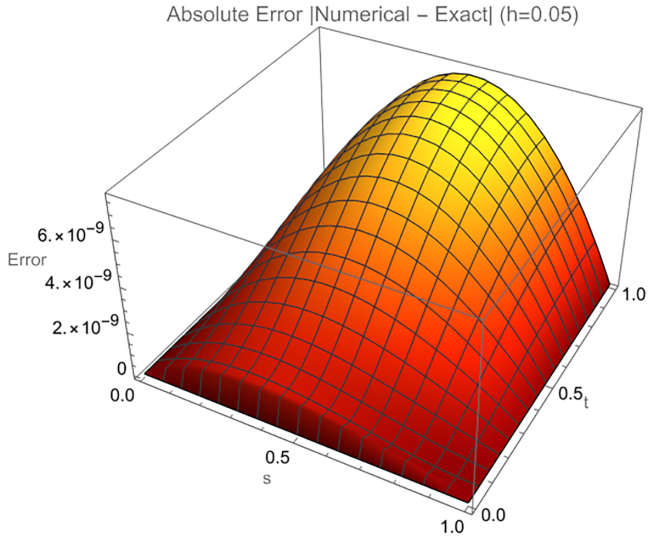

Figure 9: Error distribution with step size of $h = 0.05$ for Example 3 at time $t = 0.01$

The computed order approaches 3 as h decreases, confirming the third-order spatial accuracy.

Discussion on Computational Cost and Efficiency

It is important to point out that for explicit methods such as RK4 for parabolic problems, the time step is constrained by a stability condition of the form $\Delta t \leq C \cdot h^2$, where C is a constant that depends on the problem data (namely, γ). This diffusive stability constraint (CFL condition) means that as the spatial mesh is refined, the number of time steps must increase quadratically in order to satisfy the stability constraint. This means that the overall computational cost for reaching a final time T_f is $O(N^3)$, as the number of spatial points N, the overall computational cost due to the high-order accuracy in space and time is that much more efficient, since a much coarser mesh (smaller N) can be used to satisfy a given error tolerance, as opposed to lower-order methods.

6 Concluding Remarks

This research established and carefully studied a novel numerical technique for solving the Newell-Whitehead-Segel equation. The primary breakthrough is the design of third-order accurate one-sided finite difference stencils at the boundaries. The methodology combines the Line method with high-order spatial formulas and incorporates the resulting ODE system using the classical fourth-order Runge-Kutta method. Theoretical analysis confirms the scheme's stability and convergence, while comprehensive numerical experiments demonstrate its high accuracy and efficiency. The significance of this work extends beyond theoretical mathematics into practical engineering domains. The presented scheme provides a reliable and efficient computational tool for simulating pattern formation and bifurcation dynamics in applied contexts. The method's framework is readily adaptable to similar reaction-diffusion systems, such as the Fisher-KPP or Ginzburg-Landau equations, which model phenomena in population dynamics, corrosion and superconductivity. Future work will focus on extending the scheme to multi-dimensional domains and coupled systems, further enhancing its utility for complex engineering simulations.

Acknowledgement: We acknowledge the support of the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia.

Funding Statement: This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Grant No. KFU261039.].

Author Contributions: Conceptualization, Muhammad Yaseen and Muhammad Ameer Hamza; methodology, Salma Trabelsi and Muhammad Ameer Hamza; software, Muhammad Yaseen; validation, Muhammad Ameer Hamza and Meraa Arab; formal analysis, Salma Trabelsi and Meraa Arab; investigation, Muhammad Ameer Hamza, Muhammad Yaseen and Salma Trabelsi; writing—original draft preparation, Muhammad Ameer Hamza and Muhammad Yaseen; data curation, Muhammad Yaseen, Salma Trabelsi and Meraa Arab; visualization, Muhammad Yaseen and Meraa Arab; funding acquisition, Salma Trabelsi and Meraa Arab. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the Corresponding Author upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Atta AG, Abd-Elhameed WM, Youssri YH. Approximate collocation solution for the time-fractional Newell-Whitehead-Segel equation. J Appl Comput Mech. 2025;11(2):529–40.
2. Aasaraai A. Analytic solution for Newell-Whitehead–Segel equation by differential transform method. Middle East J Sci Res. 2011;10(2):270–3.
3. Macias-Diaz JE, Ruiz-Ramirez J. A non-standard symmetry-preserving method to compute bounded solutions of a generalized Newell-Whitehead–Segel equation. Appl Numer Math. 2011;61:630–40.
4. Hilal N, Injrou S, Karroum R. Exponential finite difference methods for solving Newell-Whitehead-Segel equation. Arab J Math. 2020;9(2):367–80.

5. Mahgoub MMA, Sedeeg AKH. On the solution of Newell-Whitehead–Segel equation. Am J Math Comput Model. 2016;1:21–4.
6. Saravanan A, Magesh N. A comparison between the reduced differential transform method and the Adomian decomposition method for the Newell-Whitehead-Segel equation. J Egypt Math Soc. 2013;21: 259–65.
7. Prakash A, Kumar M. He's variational iteration method for the solution of nonlinear Newell–Whitehead–Segel equation. J Appl Anal Comput. 2016;6:738–48.
8. Soori M, Nourazar S, Nazari-Golshan A. The variational iteration method for the Newell-Whitehead-Segel equation. Theor Phys Appl Math Sci Essay. 2016;5(1):17–26.
9. Zahra WK, Ouf WA, El-Azab MS. Cubic B-spline collocation algorithm for the numerical solution of Newell-Whitehead-Segel type equations. Electron J Math Anal Appl. 2014;2:81–100.
10. Ezzati R, Shakibi K. Using Adomians decomposition and multi quadric quasi-interpolation methods for solving Newell-Whitehead equation. Procedia Comput Sci. 2011;3:1043–8.
11. Pue-On P. Laplace Adomian decomposition method for solving Newell-Whitehead-Segel equation. Appl Math Sci. 2013;7(132):6593–600.
12. Kheiri H, Ali pour N, Dehghani R. Homotopy analysis and homotopy Padé methods for the modified Burgers-Korteweg-de Vries and Newell-Whitehead equations. Math Sci. 2011;5:33–50.
13. Patade J, Bhalekar S. Approximate analytical solution of Newell-Whitehead–Segel equation using a new iterative method. World J Model Simul. 2015;11(2):94–103.
14. Akinlabi GO, Edeki SO. Perturbation iteration transform method for the solution of Newell-Whitehead–Segel model equations. J Math Statistics. 2017;13(1):24–9.
15. Jassim HK. Homotopy perturbation algorithm using Laplace transform for Newell-Whitehead–Segel equation. Int J Adv Appl Math Mech. 2015;2(4):8–12.
16. Hashemi MS, Mirzazadeh M. Exact solutions of nonlinear stochastic Newell-Whitehead-Segel equation by a reduction technique. Eur Phys J Plus. 2023;138(11):1016.
17. Nadeem M, He JH, He CH, Sedighi HM, Shirazi A. A numerical solution of nonlinear fractional Newell-Whitehead-Segel equation using natural transform. TWMS J Pure Appl Math. 2022;13(2):168–82.
18. Bektaş U, Anaç H. A hybrid method to solve a fractional-order Newell-Whitehead–Segel equation. Bound Value Probl. 2024;2024(1):38.
19. Mahgoub MMA. Homotopy perturbation method for solving Newell-Whitehead–Segel equation. Adv Theor Appl Math. 2016;11:399–406.
20. Nourazar SS, Soori M, Nazari-Golshan A. On the exact solution of Newell-Whitehead–Segel equation using the homotopy perturbation method. Aust J Basic Appl Sci. 2011;5(8):1400–11.
21. Areshi M, Khan A, Shah R, Nonlaopon K. Analytical investigation of fractional-order Newell-Whitehead-Segel equations via a novel transform. Aims Math. 2022;7(4):6936–58.
22. Hariharan G. An efficient legendre Wavelet-based approximation method for a few Newell-Whitehead and Allen-Cahn equations. J Membr Biol. 2014;247:371–80.
23. Inan B, Osman MS, Ak T, Baleanu D. Analytical and numerical solutions of mathematical biology models: the Newell-Whitehead–Segel and Allen-Cahn equations. Math Methods Appl Sci. 2020;43:2588–600.
24. Yaseen M, Batool S. Utilizing a third order difference scheme in numerical method of lines for solving the heat equation. Phys Scr. 2024;99(4):045204.
25. Adel E, El-kalla IL, Elsaid A, Sameeh M. Modeling mass transport using time-variable space-fractional Riesz derivative. Int J Dyn Control. 2026;14(1):19.
26. Mohamed S, El-Azab M, Sameeh M. Chebyshev collocation method for fractional temporary heat transfer model in composite cylinders with hollow interiors. Damietta's J Eng Res Future Technol. 2025;1(1):9–18.