



Nayrouz Hilal · Sami Injrou · Ramez Karroum

# Exponential finite difference methods for solving Newell–Whitehead–Segel equation

Received: 5 August 2019 / Accepted: 18 February 2020 / Published online: 2 March 2020  
© The Author(s) 2020

**Abstract** This work presents two different finite difference methods to compute the numerical solutions for Newell–Whitehead–Segel partial differential equation, which are implicit exponential finite difference method and fully implicit exponential finite difference method. Implicit exponential methods lead to nonlinear systems. Newton method is used to solve the resulting systems. Stability and consistency are discussed. To illustrate the accuracy of the proposed numerical methods, some examples are delivered at the end.

**Mathematics Subject Classification** 65M06

## 1 Introduction

Nonlinear partial differential equations play an important role in modeling complicated phenomenon in physics, chemistry, biology, and mechanics. One of these equations is Newell–Whitehead–Segel (NWS) partial differential equation. It describes the appearance of the stripe pattern in two dimensions. On the other hand, this equation is applied as a mathematical model in different systems, such as Rayleigh–Benard convection, Faraday instability, and chemical reaction.

Newell–Whitehead equation has the form [15]:

$$U_t(x, t) - aU_{xx}(x, t) = bU(x, t) - cU^3(x, t); \quad (1.1)$$

Segel modified the previous equation to the following form [22]:

$$U_t(x, t) = kU_{xx}(x, t) + aU(x, t) - bU^q(x, t), \quad (1.2)$$

where  $a$ ,  $b$  and  $k$  are real numbers with  $k > 0$  and  $q$  is a positive integer,  $U(x, t)$  may express a nonlinear distribution of temperature in an infinitely thin and long rod or as the flow velocity of a fluid in an infinitely long pipe with small diameter, and  $aU - bU^q$  presents the effect of the source term.

Recently, researchers used different methods to find the analytical solutions of Problem (1.2). Aasarai [1] used differential transform method to present the analytical solution, Pue-On [19] solved (1.2) by Laplace Adomian decomposition method. Jassim [11] applied Homotopy perturbation algorithm using Laplace transform

---

N. Hilal (✉) · S. Injrou · R. Karroum  
Department of Mathematics, Tishreen University, Lattakia, Syria  
E-mail: nayrouz.hilal.11@gmail.com

S. Injrou  
E-mail: s.injrou@tishreen.edu.sy

R. Karroum  
E-mail: d.karroum.r@gmail.com



to give analytical solution; however, Mahgoub and Sedeeg [13, 14] solved it by Elzaki Adomian decomposition method. Soori et al. [23] used the variational iteration method and Prakash et al. [18] utilized He's variational iteration method to solve Problem (1.2).

For the numerical solution of Problem (1.2), Ruiz-Ramírez and Macías-Díaz [12] used non-standard symmetry-preserving method to compute bounded solution of a generalized NWS problem, Zahraa et al. [24] found numerical solutions by Cubic B-Spline methods, Patade and Bhalekar [17] used a new iterative method which was found by Jafari and Daftardar-Gejji [5] to give analytical solutions for NWS equation with initial condition, and Akinlabi and Edeki [2] used perturbation iteration transform method and gave approximate solutions for the initial value problems of NWS equation.

Different finite difference schemes have been developed for solving different differential equations, Bahadir [3] applied exponential finite difference method to KDV equation for small times, Ramos [21] used explicit finite difference methods for the equal width (EW) and regularized long-wave (RLW) equations, Inan and Bahadir [9] used Hopf–Cole transform to linearize Burgers' equation, then they applied an explicit exponential finite difference method to find the numerical solution, also they presented an implicit exponential finite difference scheme for solving generalized Burgers–Huxley equation [10], Huang and Abduwali [7] used Crank–Nicolson method to modify the numerical scheme of generalized Burgers–Huxley equation, Celikten et al. [4] presented four different explicit exponential finite difference methods to solve modified Burgers' equation, and Inan [8] applied Crank–Nicolson exponential finite difference scheme to generalized Fitzhugh–Nagumo equation.

The aim of this manuscript is to introduce two different finite difference schemes to compute the numerical solutions of Problem (1.2). The first one is implicit exponential finite difference scheme and the second one is fully implicit exponential finite difference scheme. Stability and consistency of both schemes are discussed. Moreover, the rate and order of convergence are discussed numerically. Some examples are presented to show the efficiency of these methods to solve the equation.

## 2 Numerical methods

In general, finite difference methods for solving partial differential equations depend on transforming a calculus problem into an algebra problem by discretizing the continuous domain into a discrete difference grid  $(x_i, t_j)$ , where  $x_i = i \Delta x$ ;  $0 \leq i \leq N$ ,  $t_j = j \Delta t$ ;  $j = 0, 1, 2, \dots$ ,  $\Delta x$  is the spatial mesh size and  $\Delta t$  is the time step. Then, replacing the individual exact partial derivatives by algebraic finite difference approximations. In this work, we use the difference operators:

$$\left(\frac{\partial u}{\partial t}\right)_{i,j+1} \approx \frac{u_i^{j+1} - u_i^j}{\Delta t}, \quad (2.1)$$

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_{i,j+1} \approx \frac{u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}}{(\Delta x)^2}, \quad (2.2)$$

where  $u_i^j$  is the numerical approximation of the exact solution at point  $(x_i, t_j)$ .

The numerical solution of the proposed problem is obtained by solving the resulting system of algebraic equations [6].

### 2.1 Implicit exponential finite difference scheme (I-EFD)

The implicit exponential finite difference scheme of Eq. (1.2) is given by the form:

$$u_i^{j+1} = u_i^j \exp\left(\frac{1}{u_i^j} \left[ r(u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}) + a \Delta t u_i^j - b \Delta t (u_i^j)^q \right]\right). \quad (2.3)$$



### 2.2 Fully implicit exponential finite difference scheme (FI-EFD)

The fully implicit exponential finite difference scheme of Eq. (1.2) is given by the form:

$$u_i^{j+1} = u_i^j \exp \left( \frac{1}{u_i^j} \left[ r(u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}) + a\Delta t u_i^{j+1} - b\Delta t (u_i^{j+1})^q \right] \right), \tag{2.4}$$

where  $u_i^j$  is the numerical exponential approximation of  $U(x, t)$  at the point  $(x_i, t_j)$ ,  $1 \leq i \leq N - 1$ ,  $j = 0, 1, 2, \dots$ , and  $r = \frac{k\Delta t}{(\Delta x)^2}$ .

An iterative method is applied to solve the nonlinear systems of algebraic equations (Eqs. (2.3) and (2.4)). Suppose these nonlinear systems are given by the form:

$$G(u) = 0, \tag{2.5}$$

where  $G = [g_1, g_2, \dots, g_{N-1}]^T$ ,  $u = [u_1^{j+1}, u_2^{j+1}, \dots, u_{N-1}^{j+1}]^T$ .

Newton method is used to solve the system (2.5) by the following way:

1. Set  $u^{(0)}$ , an initial estimate.
2. For  $K = 0, 1, 2, \dots$  until convergence do:

$$\text{Solve: } u^{K+1} = u^K - J(u^{(K)})^{-1} G(u^{(K)}),$$

where  $J(u^{(K)})$  is the Jacobian matrix.

The solution at every time level is considered an initial estimate to the solution at the next time level. At each time step, Newton iteration stops when  $\|G(u^{(K)})\|_\infty \leq 10^{-5}$ ; it usually needs two or three iterations.

Absolute,  $L_\infty$  and  $L_2$  norms will be used to measure the accuracy of both proposed methods which are defined, respectively, by:

$$\text{errorAbs} = |u(x_i, t_j) - U(x_i, t_j)|, \tag{2.6}$$

$$\text{error}L_\infty = \|u - U\|_\infty = \max_{0 \leq i \leq N} |u_i - U_i|, \tag{2.7}$$

$$\text{error}L_2 = \|u - U\|_2 = \left( \Delta x \sum_{i=0}^{i=N} |u_i - U_i|^2 \right)^{\frac{1}{2}}, \tag{2.8}$$

where  $U$  is the exact solution of the proposed problem and  $u$  is the numerical exponential finite difference approximation.

## 3 Errors and consistency

### 3.1 Local truncation error (LTE)

Since the schemes are exponential, the investigation will be developed by expanding the exponential term of the schemes into a Taylor series, and using the first two terms of the expansion, then substituting the coefficients  $u_i^{j+1}, u_{i+1}^{j+1}$ , and  $u_{i-1}^{j+1}$  by Taylor series expansions:

$$u_i^{j+1} = u_i^j + \frac{\Delta t}{1!} \left( \frac{\partial u}{\partial t} \right)_{i,j} + \frac{(\Delta t)^2}{2!} \left( \frac{\partial^2 u}{\partial t^2} \right)_{i,j} + \frac{(\Delta t)^3}{3!} \left( \frac{\partial^3 u}{\partial t^3} \right)_{i,j} + \frac{(\Delta t)^4}{4!} \left( \frac{\partial^4 u}{\partial t^4} \right)_{i,j} + \dots, \tag{3.1}$$

$$\begin{aligned}
 u_{i+1}^{j+1} = & u_i^j + \frac{\Delta x}{1!} \left( \frac{\partial u}{\partial x} \right)_{i,j} + \frac{\Delta t}{1!} \left( \frac{\partial u}{\partial t} \right)_{i,j} + \frac{(\Delta x)^2}{2!} \left( \frac{\partial^2 u}{\partial x^2} \right)_{i,j} + \frac{(\Delta t)^2}{2!} \left( \frac{\partial^2 u}{\partial t^2} \right)_{i,j} \\
 & + \Delta x \Delta t \left( \frac{\partial^2 u}{\partial x \partial t} \right)_{i,j} + \frac{(\Delta x)^3}{3!} \left( \frac{\partial^3 u}{\partial x^3} \right)_{i,j} + \frac{(\Delta t)^3}{3!} \left( \frac{\partial^3 u}{\partial t^3} \right)_{i,j} + \frac{(\Delta x)^2}{2!} \Delta t \left( \frac{\partial^3 u}{\partial x^2 \partial t} \right)_{i,j} \\
 & + \Delta x \frac{(\Delta t)^2}{2!} \left( \frac{\partial^3 u}{\partial x \partial t^2} \right)_{i,j} + \frac{(\Delta x)^4}{4!} \left( \frac{\partial^4 u}{\partial x^4} \right)_{i,j} + \frac{(\Delta t)^4}{4!} \left( \frac{\partial^4 u}{\partial t^4} \right)_{i,j} \\
 & + \frac{(\Delta x)^2 (\Delta t)^2}{4} \left( \frac{\partial^4 u}{\partial x^2 \partial t^2} \right)_{i,j} + \frac{(\Delta x)^3}{3!} \Delta t \left( \frac{\partial^4 u}{\partial x^3 \partial t} \right)_{i,j} + \Delta x \frac{(\Delta t)^3}{3!} \left( \frac{\partial^4 u}{\partial x \partial t^3} \right)_{i,j} + \dots, \tag{3.2}
 \end{aligned}$$

$$\begin{aligned}
 u_{i-1}^{j+1} = & u_i^j - \frac{\Delta x}{1!} \left( \frac{\partial u}{\partial x} \right)_{i,j} + \frac{\Delta t}{1!} \left( \frac{\partial u}{\partial t} \right)_{i,j} + \frac{(\Delta x)^2}{2!} \left( \frac{\partial^2 u}{\partial x^2} \right)_{i,j} + \frac{(\Delta t)^2}{2!} \left( \frac{\partial^2 u}{\partial t^2} \right)_{i,j} \\
 & - \Delta x \Delta t \left( \frac{\partial^2 u}{\partial x \partial t} \right)_{i,j} - \frac{(\Delta x)^3}{3!} \left( \frac{\partial^3 u}{\partial x^3} \right)_{i,j} + \frac{(\Delta t)^3}{3!} \left( \frac{\partial^3 u}{\partial t^3} \right)_{i,j} + \frac{(\Delta x)^2}{2!} \Delta t \left( \frac{\partial^3 u}{\partial x^2 \partial t} \right)_{i,j} \\
 & - \Delta x \frac{(\Delta t)^2}{2!} \left( \frac{\partial^3 u}{\partial x \partial t^2} \right)_{i,j} + \frac{(\Delta x)^4}{4!} \left( \frac{\partial^4 u}{\partial x^4} \right)_{i,j} + \frac{(\Delta t)^4}{4!} \left( \frac{\partial^4 u}{\partial t^4} \right)_{i,j} + \frac{(\Delta x)^2 (\Delta t)^2}{4} \left( \frac{\partial^4 u}{\partial x^2 \partial t^2} \right)_{i,j} \\
 & - \frac{(\Delta x)^3}{3!} \Delta t \left( \frac{\partial^4 u}{\partial x^3 \partial t} \right)_{i,j} - \Delta x \frac{(\Delta t)^3}{3!} \left( \frac{\partial^4 u}{\partial x \partial t^3} \right)_{i,j} + \dots \tag{3.3}
 \end{aligned}$$

3.1.1 LTE of implicit exponential finite difference scheme (I-EFD)

The Eq. (2.3) can be written as:

$$u_i^{j+1} = u_i^j + r(u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}) + a \Delta t u_i^j - b \Delta t (u_i^j)^q. \tag{3.4}$$

Substituting (3.1)–(3.3) into the resulting system (3.4), we get:

$$\left( \frac{\partial u}{\partial t} \right)_{i,j} + \mathcal{O}(\Delta t) = k \left( \frac{\partial^2 u}{\partial x^2} \right)_{i,j} + \mathcal{O}(\Delta x)^2 + a u_i^j - b (u_i^j)^q. \tag{3.5}$$

Therefore, LTE of (3.5) is:

$$LTE = \lim_{\Delta t, \Delta x \rightarrow 0} [\mathcal{O}(\Delta t) + \mathcal{O}(\Delta x)^2] = 0. \tag{3.6}$$

3.1.2 LTE of fully implicit exponential finite difference scheme (FI-EFD)

The Eq. (2.4) can be written as:

$$u_i^{j+1} = u_i^j + r(u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}) + a \Delta t u_i^{j+1} - b \Delta t (u_i^{j+1})^q. \tag{3.7}$$

As the previous discussion in 3.1.1, we find:

$$\left( \frac{\partial u}{\partial t} \right)_{i,j} + \mathcal{O}(\Delta t) = k \left( \frac{\partial^2 u}{\partial x^2} \right)_{i,j} + \mathcal{O}(\Delta x)^2 + a \left( u_i^j + \mathcal{O}(\Delta t) \right) - b \left( (u_i^j)^q + \mathcal{O}(\Delta t) \right). \tag{3.8}$$

$$LTE = \lim_{\Delta t, \Delta x \rightarrow 0} [\mathcal{O}(\Delta t) + \mathcal{O}(\Delta x)^2] = 0. \tag{3.9}$$

3.2 Consistency

Since the difference between the PDE and the finite difference scheme gives LTE, and according to (3.6) and (3.9), the proposed schemes are consistent with Eq. (1.2). And they are first order in time and second order in space.

### 4 Stability analysis

We consider Von-Neumann stability analysis to investigate stability of the linear form of NWS equation.

#### 4.1 Stability of implicit exponential finite difference scheme (I-EFD)

We suppose  $(u_i^j)^{q-1} = \alpha = constant$ , so the linear form of (3.4) can be written as:

$$u_i^{j+1} = u_i^j + r(u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}) + a\Delta t u_i^j - \alpha b \Delta t u_i^j. \tag{4.1}$$

By substituting the Fourier mode  $u_i^j = \zeta^j e^{I\beta i \Delta x}$ ;  $I^2 = -1$  into (4.1), we get:

$$\zeta^{j+1} = g\zeta^j; g = \frac{1 + a\Delta t - \alpha b \Delta t}{1 + 4r \sin^2\left(\frac{\beta \Delta x}{2}\right)}.$$

Suppose that  $\alpha = 1$ , so  $g$  has the form:

$$g = \frac{1 + a\Delta t - b \Delta t}{1 + 4r \sin^2\left(\frac{\beta \Delta x}{2}\right)}. \tag{4.2}$$

The stability condition is  $|g| \leq 1$ .

When  $a \leq b$ , the scheme is unconditionally stable.

When  $a > b$ , suppose that  $a - b = c > 0$ :

$$\implies -1 - 4r \sin^2\left(\frac{\beta \Delta x}{2}\right) \leq 1 + c\Delta t \leq 1 + 4r \sin^2\left(\frac{\beta \Delta x}{2}\right) \implies c\Delta t \leq 4r \sin^2\left(\frac{\beta \Delta x}{2}\right).$$

Therefore, the stability condition is:

$$a - b \leq \frac{4r}{\Delta t}.$$

#### 4.2 Stability of fully implicit exponential finite difference scheme (FI-EFD)

We suppose  $(u_i^{j+1})^{q-1} = \alpha = constant$ , so the linear form of (3.7) can be written as:

$$u_i^{j+1} = u_i^j + r(u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}) + a\Delta t u_i^{j+1} - \alpha b \Delta t u_i^{j+1}. \tag{4.3}$$

By substituting the Fourier mode  $u_i^j = \zeta^j e^{I\beta i \Delta x}$ ;  $I^2 = -1$  into (4.3), we get:

$$\zeta^{j+1} = g\zeta^j; g = \frac{1}{1 - a\Delta t + \alpha b \Delta t + 4r \sin^2\left(\frac{\beta \Delta x}{2}\right)}.$$

Suppose that  $\alpha = 1$ , so  $g$  has the form:

$$g = \frac{1}{1 - a\Delta t + b \Delta t + 4r \sin^2\left(\frac{\beta \Delta x}{2}\right)}. \tag{4.4}$$

The stability condition is  $|g| \leq 1$ .

When  $a \leq b$ , the scheme is unconditionally stable.

When  $a > b$ , the stability condition is  $-1 \leq \frac{1}{1 - a\Delta t + b \Delta t + 4r \sin^2\left(\frac{\beta \Delta x}{2}\right)} \leq 1$ .

If  $1 - a\Delta t + b \Delta t + 4r \sin^2\left(\frac{\beta \Delta x}{2}\right) > 0$ , then  $a - b \leq \min\left\{\frac{2+4r}{\Delta t}, \frac{4r}{\Delta t}\right\}$ , so the stability condition is:

$$a - b \leq \frac{4r}{\Delta t}.$$

If  $1 - a\Delta t + b \Delta t + 4r \sin^2\left(\frac{\beta \Delta x}{2}\right) < 0$ , then  $a - b \geq \max\left\{\frac{2+4r}{\Delta t}, \frac{4r}{\Delta t}\right\}$ , so the stability condition is,

$$a - b \geq \frac{2 + 4r}{\Delta t}.$$

**Table 1** Absolute errors of Example 5.1 of I-EFD, FI-EFD (for  $\Delta x = 0.0125, \Delta t = 10^{-4}$ ), and the methods of [24]

$x/t$	Method	0.2	0.4	0.6	0.8	1
0.2	UCBS	3.800E-04	8.190E-04	1.112E-03	1.184E-03	1.080E-03
	TCBS	3.951E-04	8.362E-04	1.130E-03	1.204E-03	1.103E-03
	ECBS	9.673E-04	1.900E-03	2.446E-03	2.519E-03	2.251E-03
	I-EFD	3.66E-5	3.46E-5	2.78E-5	2.08E-5	1.49E-5
	FI-EFD	3.64E-5	3.55E-5	2.98E-5	2.33E-5	1.73E-5
0.4	UCBS	4.230E-04	1.111E-03	1.613E-03	1.777E-03	1.658E-03
	TCBS	4.448E-04	1.137E-03	1.640E-03	1.806E-03	1.688E-03
	ECBS	1.159E-03	2.641E-03	3.587E-03	3.805E-03	3.462E-03
	I-EFD	3.52E-5	3.67E-5	3.04E-5	2.31E-5	1.66E-5
	FI-EFD	3.45E-5	3.76E-5	3.30E-5	2.66E-5	2.03E-5
0.6	UCBS	2.890E-04	1.008E-03	1.569E-03	1.789E-03	1.703E-03
	TCBS	3.111E-04	1.034E-03	1.596E-03	1.818E-03	1.734E-03
	ECBS	8.863E-04	2.444E-03	3.512E-03	3.838E-03	3.558E-03
	I-EFD	3.65E-5	3.92E-5	3.35E-5	2.62E-5	1.94E-5
	FI-EFD	3.55E-5	3.99E-5	3.60E-5	2.98E-5	2.31E-5
0.8	UCBS	1.120E-04	6.120E-04	1.024E-03	1.206E-03	1.172E-03
	TCBS	1.270E-04	6.300E-04	1.042E-03	1.225E-03	1.192E-03
	ECBS	4.224E-04	1.505E-03	2.293E-03	2.583E-03	2.441E-03
	I-EFD	4.03E-5	4.20E-5	3.72E-5	3.02E-5	2.32E-5
	FI-EFD	3.95E-5	4.23E-5	3.87E-5	3.26E-5	2.57E-5

**5 Numerical results and stability discussion**

In this section, we will discuss three different standard examples to test the both proposed methods.

*Example 5.1* [16]

$$u_t - u_{xx} - u + u^4 = 0, (x, t) \in [0, 1] \times [0, T].$$

The initial condition is:

$$u(x, 0) = \left(1 + e^{\frac{3x}{\sqrt{10}}}\right)^{-\frac{2}{3}}$$

with the boundary conditions:

$$u(0, t) = \left(\frac{1}{2} + \frac{1}{2} \tanh\left(\frac{21t}{20}\right)\right)^{\frac{2}{3}},$$

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

The exact solution is given as:  $u(x, t) = \left(\frac{1}{2} + \frac{1}{2} \tanh\left(-\frac{3}{2\sqrt{10}}\left(x - \frac{7t}{\sqrt{10}}\right)\right)\right)^{\frac{2}{3}}.$

**Stability of Example 5.1.**

- Stability of I-EFD scheme

In this problem, we have  $a = 1, b = 1$ ; so,  $g$  in (4.2) is given as  $g = \frac{1}{1+4r \sin^2(\frac{\beta\Delta x}{2})}$  which satisfies the condition  $|g| \leq 1$  and the scheme is unconditionally stable.

- Stability of FI-EFD scheme

$g$  in (4.4) is given as  $g = \frac{1}{1+4r \sin^2(\frac{\beta\Delta x}{2})}$  and the scheme is also unconditionally stable.

Table 1 shows absolute errors of I-EFD, FI-EFD methods, and the methods of [24], where  $T = 1, \Delta x = 0.0125$  and  $\Delta t = 10^{-4}$  (for finite difference methods). It also shows that the presented methods offer better results than the methods of [24]. Tables 2, 3 show  $L_\infty$  and  $L_2$  errors for  $T = 10, \Delta x = 0.05$  and two different choices of  $\Delta t$ . These tables show that the errors of I-EFD and FI-EFD methods are almost equal, and they offer high accurate solutions.

Figure 1 shows exact and numerical solutions computed by I-EFD method for  $T = 5, \Delta t = 0.001$ , and  $\Delta x = 0.05$ . Figure 2 shows numerical solution computed by I-EFD method for  $T = 1, \Delta t = 0.05$ , and  $\Delta x = 0.1$ .

*Example 5.2* [16]

$$u_t - u_{xx} - 3u + 4u^3 = 0, (x, t) \in [0, 1] \times [0, T].$$

The initial condition is:

**Table 2**  $L_\infty$  and  $L_2$  errors of Example 5.1 for  $\Delta x = 0.05$  and  $\Delta t = 10^{-3}$

Method	I-EFD		FI-EFD	
	Error $L_\infty$	Error $L_2$	Error $L_\infty$	Error $L_2$
$t$				
0.01	3.131E-4	1.260E-4	3.129E-4	1.254E-4
0.1	4.134E-4	2.954E-4	4.101E-4	2.886E-4
5	9.1E-6	6.6E-6	9.2E-6	6.7E-6
10	2.511E-10	1.830E-10	2.529E-10	1.843E-10

**Table 3**  $L_\infty$  and  $L_2$  errors of Example 5.1 for  $\Delta x = 0.05$  and  $\Delta t = 10^{-4}$

Method	I-EFD		FI-EFD	
	Error $L_\infty$	Error $L_2$	Error $L_\infty$	Error $L_2$
$t$				
0.01	3.17E-5	1.28E-5	3.17E-5	1.28E-5
0.1	4.14E-5	2.96E-5	4.11E-5	2.90E-5
5	9.1E-6	6.6E-6	9.1E-6	6.6E-6
10	2.495E-10	1.815E-10	2.497E-10	1.817E-10

$$u(x, 0) = \sqrt{\frac{3}{4}} \frac{e^{\sqrt{6}x}}{e^{\sqrt{6}x} + e^{\frac{\sqrt{6}}{2}x}},$$

with the boundary conditions:

$$u(0, t) = \sqrt{\frac{3}{4}} \frac{1}{1 + e^{-\frac{9}{2}t}},$$

$$u(1, t) = \sqrt{\frac{3}{4}} \frac{e^{\sqrt{6}}}{e^{\sqrt{6}} + e^{\frac{\sqrt{6}-9t}{2}}}.$$

The exact solution is given as:  $u(x, t) = \sqrt{\frac{3}{4}} \frac{e^{\sqrt{6}x}}{e^{\sqrt{6}x} + e^{(\frac{\sqrt{6}}{2}x - \frac{9}{2}t)}}$ .

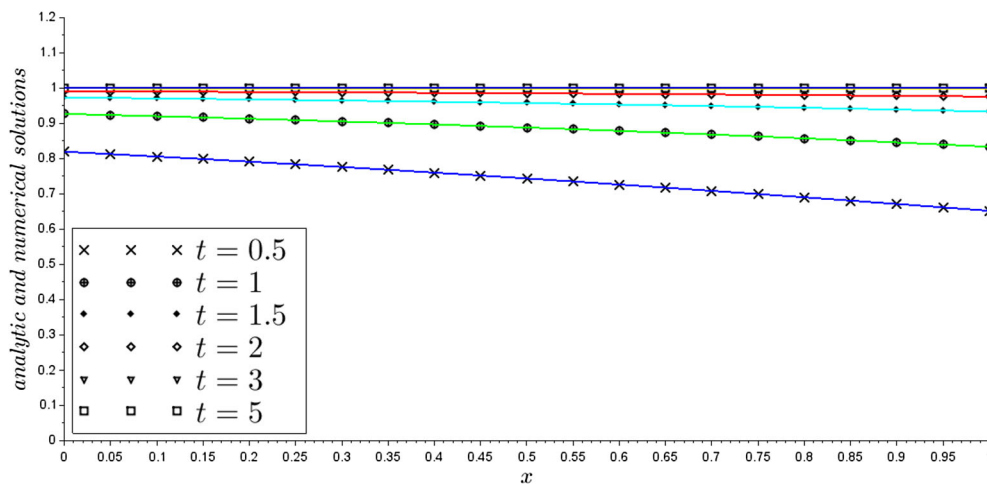
**Stability of Example 5.2.**

- Stability of I-EFD scheme

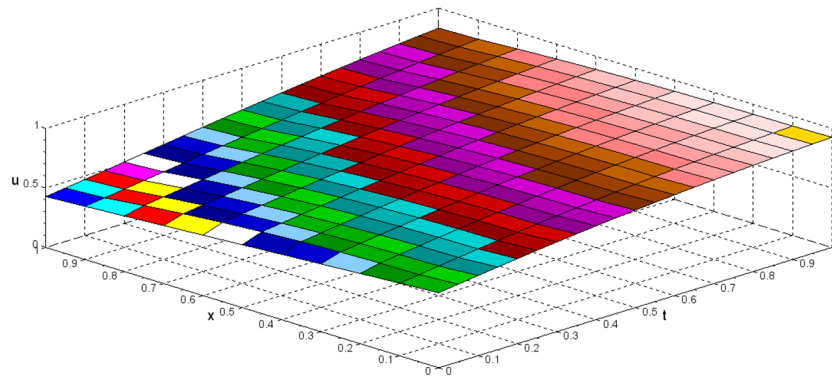
In this problem, we have  $a = 3$ ,  $b = 4$ , and so,  $g$  in (4.2) is given as  $g = \frac{1-\Delta t}{1+4r \sin^2(\frac{\beta\Delta x}{2})}$  which satisfies the condition  $|g| \leq 1$  and the scheme is unconditionally stable.

- Stability of FI-EFD scheme

$g$  in (4.4) is given as  $g = \frac{1}{1+\Delta t+4r \sin^2(\frac{\beta\Delta x}{2})}$  and the scheme is also unconditionally stable.



**Fig. 1** I-EFD solution of Example 5.1 for  $\Delta t = 0.001$  and  $\Delta x = 0.05$



**Fig. 2** I-EFD solution of Example 5.1 for  $\Delta t = 0.05$  and  $\Delta x = 0.1$

**Table 4** Absolute errors of Example 5.2 of I-EFD, FI-EFD (for  $\Delta x = 0.0125$ ,  $\Delta t = 10^{-4}$ ), and the methods of [24]

$x/t$	Method	0.2	0.4	0.6	0.8	1
0.2	UCBS	6.129E-02	7.120E-02	4.312E-02	4.661E-02	1.560E-02
	TCBS	6.134E-02	7.129E-02	6.430E-02	4.670E-02	1.563E-02
	ECBS	5.166E-02	5.402E-02	4.511E-02	2.860E-02	1.518E-02
	I-EFD	5.84E-5	3.31E-5	1.51E-5	6.4E-6	9.771E-4
	FI-EFD	6.70E-5	4.24E-5	2.05E-5	9.0E-6	9.771E-4
	0.4	UCBS	8.711E-02	1.005E-01	8.771E-02	6.001E-02
TCBS		8.718E-02	1.006E-01	8.786E-02	6.015E-02	1.650E-02
ECBS		7.224E-02	7.421E-02	5.888E-02	3.321E-02	4.323E-03
I-EFD		4.24E-5	2.35E-5	1.01E-5	4.1E-6	1.0948E-3
FI-EFD		5.53E-5	3.65E-5	1.76E-5	7.7E-6	1.0945E-3
0.6		UCBS	8.030E-02	9.072E-02	7.637E-02	4.899E-02
	TCBS	8.036E-02	9.085E-02	7.651E-02	4.911E-02	9.969E-03
	ECBS	6.461E-02	6.456E-02	4.822E-02	2.328E-02	9.679E-03
	I-EFD	3.37E-5	1.80E-5	7.4E-6	3.0E-6	9.066E-4
	FI-EFD	4.68E-5	3.02E-5	1.44E-5	6.2E-6	9.063E-4
	0.8	UCBS	4.857E-02	5.319E-02	6.420E-02	2.582E-02
TCBS		4.862E-02	5.328E-02	6.4321E-02	2.590E-02	2.915E-03
ECBS		3.721E-02	3.588E-02	2.500E-02	9.644E-03	9.118E-03
I-EFD		3.23E-5	1.60E-5	6.6E-6	2.7E-6	5.799E-4
FI-EFD		4.13E-5	2.38E-5	1.10E-5	4.7E-6	5.798E-4

**Table 5**  $L_\infty$  and  $L_2$  errors of Example 5.2 for  $\Delta x = 0.05$  and  $\Delta t = 10^{-3}$

Method	I-EFD		FI-EFD	
	Error $L_\infty$	Error $L_2$	Error $L_\infty$	Error $L_2$
$t$				
0.01	6.927E-4	2.387E-4	6.911E-4	2.404E-4
0.1	8.403E-4	4.591E-4	8.515E-4	5.057E-4
5	6.759E-11	4.898E-11	6.895E-11	4.996E-11
10	4.219E-15	3.042E-15	3.220E-15	2.229E-15

Table 4 shows absolute errors of I-EFD, FI-EFD methods, and the methods of [24], where  $T = 1$ ,  $\Delta x = 0.0125$  and  $\Delta t = 10^{-4}$  (for finite difference methods). It also shows that I-EFD and FI-EFD methods give more accurate results than the methods of [24]. Tables 5, 6 show  $L_\infty$  and  $L_2$  errors for  $T = 10$ ,  $\Delta x = 0.05$  and two different choices of  $\Delta t$ . These tables also show the high accuracy of the solutions.

Figure 3 shows exact and numerical solutions computed by I-EFD method for  $T = 5$ ,  $\Delta t = 0.001$ , and  $\Delta x = 0.05$ . Figure 4 shows numerical solution computed by I-EFD method for  $T = 1$ ,  $\Delta t = 0.05$ , and  $\Delta x = 0.1$ .

**Example 5.3** [16]

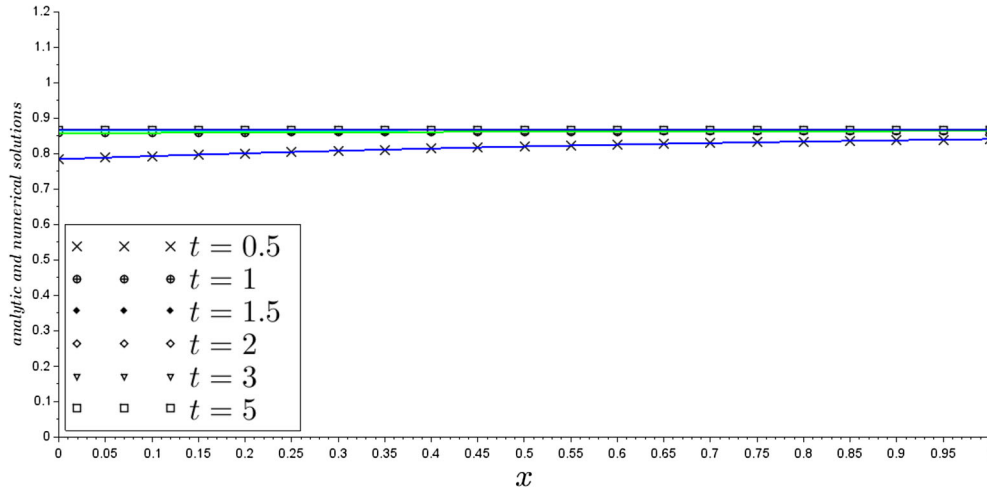
$$u_t - u_{xx} - u + u^2 = 0, (x, t) \in [0, 1] \times [0, T].$$

The initial condition is:

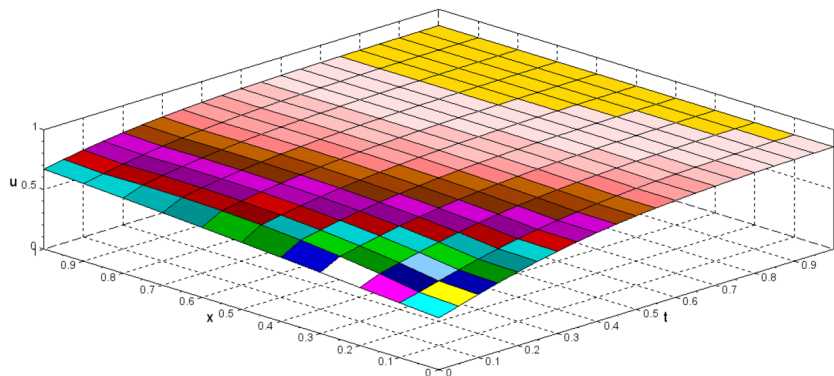


**Table 6**  $L_\infty$  and  $L_2$  errors of Example 5.2 for  $\Delta x = 0.05$  and  $\Delta t = 10^{-4}$

Method	I-EFD		FI-EFD	
	Error $L_\infty$	Error $L_2$	Error $L_\infty$	Error $L_2$
0.01	7.01E-5	2.42E-5	6.99E-5	2.43E-5
0.1	8.34E-5	4.42E-5	8.45E-5	4.88E-5
5	6.727E-11	4.869E-11	6.742E-11	4.880E-11
10	5.418E-14	4.008E-14	4.929E-14	3.692E-14



**Fig. 3** (I-EFD) solution of Example 5.2 for  $\Delta t = 0.001$  and  $\Delta x = 0.05$



**Fig. 4** I-EFD solution of Example 5.2 for  $\Delta t = 0.05$  and  $\Delta x = 0.1$

$$u(x, 0) = \frac{1}{\left(1 + e^{\frac{x}{\sqrt{6}}}\right)^2},$$

with the boundary conditions:

$$u(0, t) = \frac{1}{\left(1 + e^{-\frac{5t}{6}}\right)^2},$$

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

The exact solution is given as:  $u(x, t) = \frac{1}{\left(1 + e^{\frac{x}{\sqrt{6}} - \frac{5t}{6}}\right)^2}.$

**Stability of Example 5.3.**

**Table 7** Absolute errors of Example 5.3 of I-EFD, FI-EFD (for  $\Delta x = 0.0125, \Delta t = 10^{-4}$ ), and the methods of [24]

$x/t$	Method	0.2	0.4	0.6	0.8	1
0.2	UCBS	2.810E-04	3.030E-04	2.480E-04	1.660E-04	7.410E-05
	TCBS	2.716E-04	3.249E-04	2.341E-04	1.506E-04	5.793E-05
	ECBS	6.909E-04	6.760E-04	5.184E-04	3.196E-04	1.094E-04
	I-EFD	1.94E-5	2.25E-5	2.38E-5	2.43E-5	2.44E-5
	FI-EFD	1.86E-5	2.17E-5	2.31E-5	2.38E-5	2.40E-5
0.4	UCBS	4.270E-04	4.740E-04	3.970E-04	2.760E-04	1.390E-04
	TCBS	4.126E-04	4.551E-04	3.764E-04	2.536E-04	1.148E-04
	ECBS	1.056E-03	1.059E-03	8.328E-04	5.384E-04	2.226E-04
	I-EFD	1.71E-5	2.16E-5	2.32E-5	2.40E-5	2.42E-5
	FI-EFD	1.60E-5	2.04E-5	2.21E-5	2.31E-5	2.37E-5
0.6	UCBS	4.380E-04	4.910E-04	4.180E-04	2.990E-04	1.620E-04
	TCBS	4.247E-04	4.723E-04	3.972E-04	2.765E-04	1.381E-04
	ECBS	1.085E-03	1.099E-03	8.799E-04	5.897E-04	2.744E-04
	I-EFD	1.62E-5	2.09E-5	2.27E-5	2.37E-5	2.42E-5
	FI-EFD	1.51E-5	1.96E-5	2.16E-5	2.28E-5	2.35E-5
0.8	UCBS	3.050E-04	3.370E-04	2.890E-04	2.110E-04	1.200E-04
	TCBS	2.959E-04	2.905E-04	2.755E-04	1.962E-04	1.044E-04
	ECBS	7.482E-04	7.547E-04	6.123E-04	4.219E-04	2.127E-04
	I-EFD	1.67E-5	2.04E-5	2.22E-5	2.34E-5	2.41E-5
	FI-EFD	1.59E-5	1.95E-5	2.14E-5	2.28E-5	2.37E-5

**Table 8**  $L_\infty$  and  $L_2$  errors of Example 5.3 for  $\Delta x = 0.05$  and  $\Delta t = 10^{-3}$

Method	I-EFD		FI-EFD	
	Error $L_\infty$	Error $L_2$	Error $L_\infty$	Error $L_2$
$t$				
0.01	1.489E-4	5.38 E-5	1.485E-4	5.34E-5
0.1	1.996E-4	1.306E-4	1.978 E-4	1.252E-4
5	3.52E-5	2.82E-5	3.59E-5	3.05E-5
10	6.67E-5	4.87E-5	6.68E-5	4.88E-5

**Table 9**  $L_\infty$  and  $L_2$  errors of Example 5.3 for  $\Delta x = 0.05$  and  $\Delta t = 10^{-4}$

Method	I-EFD		FI-EFD	
	Error $L_\infty$	Error $L_2$	Error $L_\infty$	Error $L_2$
$t$				
0.01	1.51E-5	5.5E-6	1.50E-5	5.4E-6
0.1	2.00E-5	1.32E-5	1.98E-5	1.26E-5
5	4.0209E-3	2.9334E-3	4.0217E-3	2.9340E-3
10	6.62E-5	4.83E-5	6.62E-5	4.83E-5

In this problem, we have  $a = 1, b = 1$ , so as Example 5.1, both of the methods are unconditionally stable.

Table 7 shows absolute errors of I-EFD, FI-EFD methods, and the methods of [24], where  $T = 1, \Delta x = 0.0125$  and  $\Delta t = 10^{-4}$  (for finite difference methods). It also shows that the results of the presented methods are better than the results of [24]. Tables 8, 9 show  $L_\infty$  and  $L_2$  errors for  $T = 10, \Delta x = 0.05$  and two different choices of  $\Delta t$ . These tables clearly show that the numerical results of I-EFD and FI-EFD methods are in good agreement with the exact solution.

Figure 5 shows exact and numerical solutions computed by I-EFD method for  $T = 5, \Delta t = 0.001$ , and  $\Delta x = 0.05$ . Figure 6 shows numerical solution computed by I-EFD method for  $T = 1, \Delta t = 0.05$ , and  $\Delta x = 0.1$ .

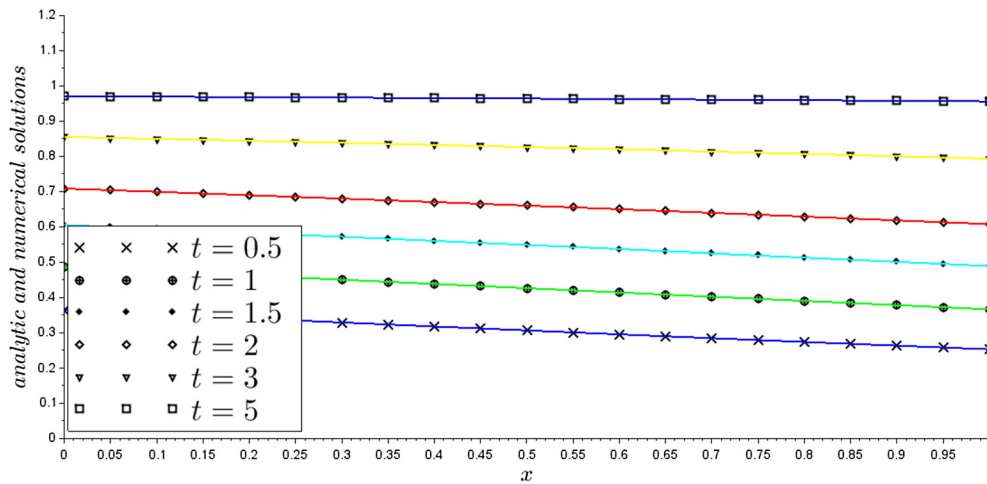


Fig. 5 I-EFD solution of Example 5.3 for  $\Delta t = 0.001$  and  $\Delta x = 0.05$

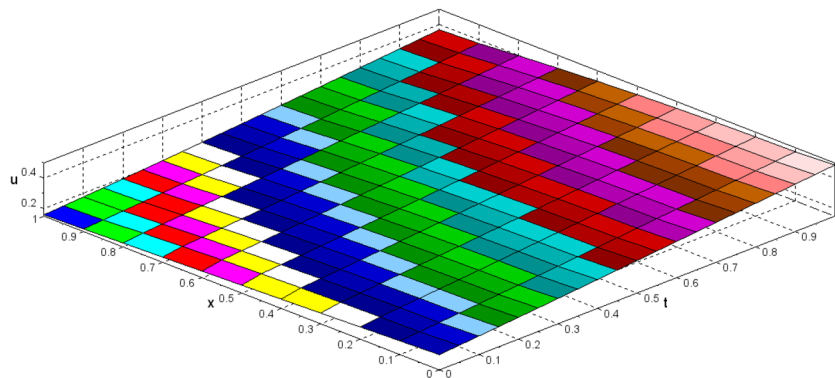


Fig. 6 I-EFD solution of Example 5.3 for  $\Delta t = 0.05$  and  $\Delta x = 0.1$

### 6 Rate and order of convergence

#### 6.1 Rate of convergence

To compute the rate of convergence, we half the grid size  $h$ , and then use the formula:

$$\text{rate} = \log_2 \left( \frac{E^h}{E^{\frac{h}{2}}} \right), \tag{6.1}$$

where  $E$  is a norm given by:

$$E = \left[ \frac{\sum_{i=0}^{i=N} |U_i - u_i|^2}{\sum_{i=0}^{i=N} |U_i|^2} \right]^{\frac{1}{2}}. \tag{6.2}$$

#### 6.2 Order of convergence

**Definition 6.1** A sequence  $\{x^{(k)}\}$  [20] generated by a numerical method is said to converge to  $\alpha$  with order  $p \geq 1$  if:

$$\exists C > 0 : \frac{|x^{(k+1)} - \alpha|}{|x^{(k)} - \alpha|^p} \leq C, \quad \forall k_0 \leq k,$$

where  $k_0 \geq 0$  is a suitable integer.

**Table 10** Rate and order of convergence for  $T = 4$ ,  $\Delta t = 10^{-4}$ , and  $\Delta x$  changes, Example 5.1

Method	I-EFD		FI-EFD	
	Rate	Order	Rate	Order
4	0.1133171	–	0.1133255	–
8	0.0507913	2.2310337	0.0507933	2.2311112
16	0.0239502	2.1207046	0.0239508	2.120735
32	0.0116674	2.0527453	0.0116678	2.0527263
64	0.0060771	1.919896	0.0060781	1.9196459

**Table 11** Rate and order of convergence for  $T = 2$ ,  $\Delta x = 0.1$ , and  $\Delta t$  changes, Example 5.1

Method	I-EFD		FI-EFD	
	Rate	Order	Rate	Order
128	1.0005681	–	1.0033521	–
256	0.9949199	1.005677	0.9972370	1.006132
512	0.9868013	1.0082272	0.9897985	1.0075152
1024	0.9723711	1.0148402	0.9774782	1.0126042
2048	0.9452501	1.0286919	0.9547440	1.0238118

To estimate  $p$ , we suppose  $|x^{(k+1)} - \alpha| = \varepsilon_{k+1}$ ,  $|x^{(k)} - \alpha| = \varepsilon_k$ , and so, the previous inequality will be  $\varepsilon_{k+1} \leq C\varepsilon_k^p$ .  
Therefore:

$$p \approx \frac{\log\left(\frac{\varepsilon_{k+1}}{\varepsilon_k}\right)}{\log\left(\frac{\varepsilon_k}{\varepsilon_{k-1}}\right)}.$$

Table 10 shows rate and order of convergence of Example 5.1 using I-EFD and FI-EFD methods for  $T = 4$ ,  $\Delta t = 10^{-4}$ , and  $\Delta x$  changes as  $N$  changes. It also shows that I-EFD and FI-EFD are second-order accurate in space, and that the methods are consistent, because the error goes to zero as  $\Delta x$  is getting smaller.

Table 11 shows rate and order of convergence of Example 5.1 using I-EFD and FI-EFD methods for  $T = 2$ ,  $\Delta x = 0.1$ , and  $\Delta t$  changes as  $N$  changes. It also shows that I-EFD and FI-EFD are first-order accurate in time, and that the methods are consistent, because the error goes to zero as  $\Delta t$  is getting smaller.

## 7 Conclusion

This manuscript introduced two different exponential finite difference methods, which are implicit exponential finite difference method and fully implicit exponential finite difference method for solving the NWS equation in (1.2). It is shown that both schemes are consistent. Furthermore, it is shown that the local truncation errors are first-order accurate in time and second-order accurate in space. Moreover, the numerical results confirmed the order of convergence in time and in space. The presented methods are computationally consistent, stable, convergent, and offer better accuracy than other methods.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.



## References

1. Aasaraai, A.: Analytic solution for Newell–Whitehead–Segel equation by differential transform method. *Middle East J. Sci. Res.* **10**(2), 270–273 (2011)
2. Akinlabi, G.O.; Edeki, S.O.: Perturbation iteration transform method for the solution of Newell–Whitehead–Segel model equations. *J. Math. Stat.* (2017)
3. Bahadir, A.R.: Exponential finite-difference method applied to Korteweg–de Vries equation for small times. *Appl. Math. Comput.* **160**, 675–682 (2005)
4. Celikten, G.; Aksan, E.N.: Explicit exponential finite difference methods for the numerical solution of modified Burgers' equation. *Eastern Anatol. J. Sci.* **3**, 45–50 (2017)
5. Daftardar-Gejji, V.; Jafari, H.: An iterative method for solving nonlinear functional equations. *J. Math. Anal. Appl.* **316**, 753–763 (2006)
6. Hoffman, J.D.: *Numerical Methods for Engineers and Scientists*, 2nd edn. CRC, Boca Raton (2001)
7. Huang, P.; Abduwali, A.: Modified local Crank–Nicolson method for generalized Burgers–Huxley equation. *Math. Rep.* **18**(68), 109–120 (2016)
8. Inan, B.: A finite difference method for solving generalized FitzHugh–Nagumo equation. AIP. 020018-1 - 020018-7 (2018)
9. Inan, B.; Bahadir, A.R.: An explicit exponential finite difference method for the Burgers' equation. *Eur. Int. J. Sci. Technol.* **2**(10), 61–72 (2013)
10. Inan, B.; Bahadir, A.R.: Numerical solutions of the generalized Burgers–Huxley equation by implicit exponential finite difference method. *JAMSI* **11**, 57–67 (2015)
11. Jassim, H.K.: Homotopy perturbation algorithm using Laplace transform for Newell–Whitehead–Segel equation. *Int. J. Adv. Appl. Math. Mech.* **2**(4), 8–12 (2015)
12. Macias-Diaz, J.E.; Ruiz-Ramirez, J.: A non-standard symmetry-preserving method to compute bounded solutions of a generalized Newell–Whitehead–Segel equation. *Appl. Numer. Math.* **61**, 630–640 (2011)
13. Mahgoub, M.M.A.: Homotopy perturbation method for solving Newell–Whitehead–Segel equation. *Adv. Theor. Appl. Math.* **11**, 399–406 (2016)
14. Mahgoub, M.M.A.; Sedeeg, A.K.H.: On the solution of Newell–Whitehead–Segel equation. *Am. J. Math. Comput. Model.* **1**, 21–24 (2016)
15. Newell, A.C.; Whitehead, J.A.: Finite bandwidth, finite amplitude convection. *J. Fluid Mech.* **38**(2), 279–303 (1969)
16. Nourazar, S.S.; Soori, M.; Nazari-Golshan, A.: On the exact solution of Newell–Whitehead–Segel equation using the homotopy perturbation method. *Aust. J. Basic Appl. Sci.* **5**(8), 1400–1411 (2011)
17. Patade, J.; Bhalekar, S.: Approximate analytical solution of Newell–Whitehead–Segel equation using a new iterative method. *World J. Model. Simul.* **11**(2), 94–103 (2015)
18. Prakash, A.; Kumar, M.: He's variational iteration method for the solution of nonlinear Newell–Whitehead–Segel equation. *J. Appl. Anal. Comput.* **6**, 738–748 (2016)
19. Pue-ON, P.: Laplace Adomian decomposition method for solving Newell–Whitehead–Segel equation. *Appl. Math. Sci.* **7**, 6593–6600 (2013)
20. Quarteroni, A.; Sacco, R.; Saleri, F.: *Text in Applied Mathematics, Numerical Mathematics*, 2nd edn. Springer, Berlin (2007)
21. Ramos, J.I.: Explicit finite difference methods for the EW and RLW equations. *Appl. Math. Comput.* **179**, 622–638 (2006)
22. Segel, L.A.: Distant side-walls cause slow amplitude modulation of cellular convection. *J. Fluid Mech.* **38**(1), 203–224 (1969)
23. Soori, M.; Nourazar, S.; Nazari-Golshan, A.: The variational iteration method for the Newell-Whitehead–Segel equation. *Theor. Phys. Appl. Math. Sci. Essay* **5**(1), 17–26 (2016)
24. Zahra, W.K.; Ouf, W.A.; El-Azab, M.S.: Cubic B-spline collocation algorithm for the numerical solution of Newell-Whitehead-Segel type equations. *Electron. J. Math. Anal. Appl.* **2**, 81–100 (2014)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

