Int. J. Nonlinear Anal. Appl. 13 (2022) 1, 1507-1518
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2021.25004.2878

Numerical solutions of nonlinear Burgers-Huxley equation through the Richtmyer type nonstandard finite difference scheme

F. Izadi[a], H. Saberi Najafi[a,*], A.H. Refahi Sheikhani

[a] *Department of Applied Mathematics, Faculty of Mathematical Sciences, Lahijan Branch, Islamic Azad University, Lahijan, Iran*

(Communicated by Madjid Eshaghi Gordji)

Abstract

The Burger-Huxley equation as a well-known nonlinear physical model is studied numerically in the present paper. In this respect, the nonstandard finite difference (NSFD) scheme in company with the Richtmyer's (3, 1, 1) implicit formula is formally adopted to accomplish this goal. Moreover, the stability, convergence, and consistency analyses of nonstandard finite difference schemes are investigated systematically. Several case studies with comparisons are provided, confirming that the current numerical scheme is capable of resulting in highly accurate approximations.

Keywords: Burger–Huxley equation; Nonstandard finite difference scheme; Richtmyer's (3, 1, 1) implicit formula; Consistency; Convergence; Stability
2010 MSC: 35J05,35J10,35K05,35L05.

1. Introduction

Most of the problems in various field as biology, chemistry, physics, and engineering are modeled by nonlinear partial differential equations. The generalized Burgers-Huxley equation is the form;

$$\frac{\partial u}{\partial t} + \alpha u^{\delta}\frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial x^2} = \beta u(1-u^{\delta})(u^{\delta}-\gamma), \qquad 0 \le x \le 1, t \ge 0, \tag{1.1}$$

*Corresponding author
Email addresses: farnooshizadi@yahoo.com (F. Izadi), hnajafi@guilan.ac.ir (H. Saberi Najafi), ah_refahi@yahoo.com (A.H. Refahi Sheikhani)

Received: August 2021 *Accepted:* October 2021

with the following initial and boundary conditions

$$
\begin{aligned}
u(x,0) &= f_1(x),\\
u(0,t) &= g_1(t),\\
u(1,t) &= g_2(t),
\end{aligned}
\tag{1.2}
$$

where α, β, δ and γ are parameters that $\beta \geq 0, \delta \succ 0, \gamma \in (0,1)$. When $\alpha \neq 0, \beta \neq 0, \delta = 1$ equation (1.2) becomes the following Burgers-Huxley equation:

$$
\frac{\partial u}{\partial t} + \alpha u \frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial x^2} = \beta u(1-u)(u-\gamma), \tag{1.3}
$$

This equation shows being a nonlinear partial differential equation is of high importance for describing the interaction between reaction mechanisms, convection effects and diffusion transports. This equation was investigated by Satsuma in 1986.

In literature, many numerical methods have been proposed for approximating solution of this equation. Adomian was applied to the generalized Burger-Huxley equation [1]. The collocation method was applied to solving this equation [2]. The variational iteration method [3], a numerical solution of the equation, based on collocation method using Radial basis function [4], the differential transform method [5], Haar Wavelet method [6], the discrete Adomian decomposition method [7]. A fourth order finite difference scheme [8] and finite difference method [9], were applied for solving this equation.

This paper is organized as follows: Section 2, introduced the computational technique to approximate solution of the model (1.3), here we prove that our NSFD scheme is consistency and convergence to the exact solution, an analysis of nonlinear stability is presented [10], [11]. Section 3, for Huxley-Burger and generalized Huxley equations this scheme was applied and the numerical results are reported and compared with the results of method used in [9], [12], [7], [5]. Tables are presented for ability of the method to solve the equation for different values of It is clearly seen that the numerical results are reasonably in good agreement with the exact solution. Finally a conclusion is given in section 4, all the numerical experiments presented in this section were computed using the MATLAB 10 on a pc with a 2.5 GHz, 64-bit processor, 4 GB memory.

2. Numerical method

2.1. Finite-difference Scheme

The main idea behind the Finite difference methods for obtaining the solution of a given partial differential equation is to approximate the derivatives appearing in the equation by the function at a selected number of points. The most usual way to generate these approximations is through the use of Taylor series. Let M and N be positive integers. In order to approximate the Equation (1.3) over the real line, we restrict our attention to a bounded spatial domain and impose appropriate boundary conditions. In order to approximate the solution of the Huxley-Berger equation problem under study over a temporal interval $[0,T]$ we set $0 = t_0 < t_1 < \cdots < t_M = T$ and $a_s = x_0 < x_1 < \cdots < x_n = b_s$ of $[0,T], [a_s, b_s], \Delta x = \dfrac{b_s - a_s}{N}$, and $\Delta t = \dfrac{T}{M}$. u_n^k, is the approximation provided by the numerical method for the exact value of $u(x_n, t_k)$ for $n = 0, \cdots, N, k = 0, 1, 2, \cdots, M$.

2.2. Nonstandard finite difference scheme

We construct a general NSFD scheme for the equation (1.3) by using the Richtmyer's (3, 1, 1) implicit formula [10], [12]. This formula is a three-point and three-level formula. In this scheme, a weighted average of finite difference approximation to the time derivative is used. The following nomenclatures are introduced to approximate the partial derivatives, u with respect to t and x at the point (x_n, t_k).

$$\frac{du}{dt}\mid_n^k = (1-\theta)\frac{u_n^{k+1}-u_n^k}{\phi(\Delta t)} + \theta\frac{u_n^k-u_n^{k-1}}{\phi(\Delta t)} + O((1+2\theta)\Delta t, (\Delta x)^2), \tag{2.1}$$

$$\frac{d^2u}{dx^2}\mid_n^k = \frac{u_{n+1}^k - 2u_n^k + u_{n-1}^k}{\psi(\Delta x)} + O(\Delta(x)^2), \tag{2.2}$$

$$\frac{du}{dx}\mid_n^k = \frac{u_n^k - u_{n-1}^k}{\phi(\Delta x)} + O(\Delta(x)), \tag{2.3}$$

Where $\phi(\Delta t) = \frac{1-e^{\frac{-\Delta t}{2}}}{\frac{1}{2}}, \phi(\Delta x) = \frac{1-e^{\frac{-\Delta x}{2}}}{\frac{1}{2}}$, and $\psi(\Delta x) = 4\sinh^2\left(\frac{\Delta x}{2}\right)$ By these conventions in hand, we will approximate solutions of Equation (1.3) in the and $[0, T]$, through the finite-difference scheme

$$\frac{du}{dt}\mid_n^k + \alpha(u_n^{k+1})\frac{du}{dx}\mid_n^k = \frac{d^2u}{dx^2}\mid_n^{k+1} + \beta u_n^{k+1} f(u_n^k), \tag{2.4}$$

where,

$$f(u_n^k) = (u_n^k - \gamma)(1-u_n^k). \tag{2.5}$$

The Nonstandard finite difference scheme (2.4) may be conveniently rewritten as

$$A_1 u_{n+1}^{k+1} + A_2 u_n^{k+1} + A_1 u_{n-1}^{k+1} = A_3 u_n^k + A_4 u_n^{k-1}, \tag{2.6}$$

with

$$A_1 = -R_2, A_2 = 2R_2 + 1 - \theta - \beta\phi(\Delta t) f(u_n^k) + \alpha R_1(u_n^k - u_n^{k-1}), A_3 = 1-2\theta, A_4 = \theta.$$

where

$$R_1 = \frac{\phi(\Delta t)}{\phi(\Delta x)}, R_2 = \frac{\phi(\Delta t)}{\psi(\Delta x)}. \tag{2.7}$$

are the Fourier numbers of the NSFD scheme (2.4), the coefficients A_1, A_2, A_3, A_4 depend on u_n^k.

2.3. Matrix representation

In this work, we will impose constraints on the form

$$u(a_s, t) = a_0(t) \quad \text{and} \quad u(b_s, t) = a_1(t) \tag{2.8}$$

Satisfied for every $t \geq 0$. Here, a_0, a_1 are non-negative, real function which is less than or equal to 1. Let M_n be the vector space of all matrices over $\mathbb{R}$ of size $(n * n)$, for each positive integer n. The numerical method (2.6) can be presented in matrix form as the following

$$Au^{k+1} = b^k, \tag{2.9}$$

for $k \in \{1, \ldots, M-1\}$, u^k is the $(N+1)$-dimensional vector $(u_0^k, u_1^k, ..., u_N^k)$, for $k \in \{0, 1, \ldots, M\}$. We let

$$b^k = Bu^k + Cu^{k-1} + d^k, \tag{2.10}$$

for every $k \in \{0, 1, \ldots, M\}$, where B and C are the diagonal matrices M_{N+1} given by:

$$B = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & A_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & A_3 & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}, C = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & A_4 & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & A_4 & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}. \tag{2.11}$$

The matrix A is a matrix of M_{N+1}, the vector d^k is an $(n+1)$-dimensional vector. The system (2.9) can be solved under the method in [14], [15].
By employing discrete Dirichlet constraints in the form of $u_0^k = a_0(t_k)$ and $u_N^k = a_1(t_k)$, for $k \in \{0, 1, \ldots, M\}$ we have the following presentations of A and d^k.

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ A_1 & A_2 & A_1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & A_1 & A_2 & A_1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & A_1 & A_2 & A_1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 \end{bmatrix}, d^k = \begin{bmatrix} a_0(t_k) \\ 0 \\ \vdots \\ \vdots \\ 0 \\ a_1(t_k) \end{bmatrix}. \tag{2.12}$$

2.4. Numerical properties

In this Section, we show the properties of stability, convergence and consistency for NSFD scheme (2.6).

2.4.1. Convergence Analysis

The local truncation error of our scheme at (x_n, t_k) is

$$\begin{aligned} \ell_n^k = {} & (1-\theta)\frac{u_n^{k+1} - u_n^k}{\varphi(\Delta t)} + \theta\frac{u_n^{k+1} - u_n^{k-1}}{\varphi(\Delta t)} + \alpha u_n^{k+1}\frac{u_n^k - u_{n-1}^k}{\varphi(\Delta x)} \\ & - \frac{u_{n+1}^{k+1} - 2u_n^{k+1} + u_{n-1}^{k+1}}{\psi(\Delta x)} - \beta u_n^{k+1} f(u_n^k). \end{aligned} \tag{2.13}$$

Considering u is the exact solution of (1.3) and using Taylor's series expansion, we have

$$\ell_n^k = (1-2\theta)\left.\frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2}\right|_n^k - \alpha\left.\frac{\Delta x}{2}\frac{\partial^2 u}{\partial x^2}\right|_n^k - \left.\frac{(\Delta x)^2}{12}\frac{\partial^4 u}{\partial x^4}\right|_n^{k+1}. \tag{2.14}$$

We assume that u_{tt}, u_{xxxx}, u_{xx} are continuous in $[0, T] \times [a_s, b_s]$, so there are constant K_1, K_2, K_3 such that

$$\left|\ell_n^k\right| \leq K_1\Delta t + K_2(\Delta x) + K_3(\Delta x)^2 \equiv E.$$

Rearranging the terms of (2.13) we have

$$A_1 u_{n+1}^{k+1} + A_2 u_n^{k+1} + A_1 u_{n-1}^{k+1} = A_3 u_n^k + A_4 u_n^{k-1} + \ell_n^k \Delta t. \tag{2.15}$$

Let $e_n^k = u_n^k - U_n^k$, U is numerical solution at (x_n, t_k). We subtract (2.5) from (2.15) and assume $1 - \theta \geq 0$.
After taking absolute of both sides of the equation (2.10), the following inequality is then obtained

$$\begin{aligned} A_1 \left|e_{n+1}^{k+1}\right| + (2R_2 + 1 - \theta)\left|e_n^{k+1}\right| + (\beta\Delta t K_4)\left|e_n^{k+1}\right| + A_1\left|e_{n-1}^{k+1}\right| \leq A_3\left|e_n^k\right| + A_4\left|e_n^{k-1}\right| \\ - \alpha R_1\left|e_n^k - e_{n-1}^k\right|\left|e_n^{k+1}\right| \\ + E\Delta t, \end{aligned} \tag{2.16}$$

Where K_4 is the maximum value of $f'(u)$, if we let

$$e^k = \max_{0 \leq n \leq N} \left|e_n^k\right|,$$

then, the above inequality becomes

$$e^{k+1} \leq (1 - 2\theta)Me^k + \theta Me^{k-1} + ME\Delta t, \tag{2.17}$$

that,

$$M = 1 - \theta - K_4\beta\Delta t.$$

Since $e^0 = e^{-1} = 0$, from equation (2.17) we have

$$\begin{aligned} e^k \leq (1 + M(1 - 2\theta) + \cdots + M^{k-1}(1 - 2\theta)^{k-1})ME\Delta t + (k - 1)(M\theta + 2(M\theta)^2 \\ + \cdots + (k - 1)(M\theta)^{k-1})ME\Delta t, \end{aligned} \tag{2.18}$$

so $e^k \to 0$ as $\Delta t, \Delta x \to 0$. Thus we have proved the following theorem.

Theorem 2.1. *If the solution of (1.3) has continuous u_{tt}, u_{xxxx}, u_{xx} in $[a_s, b_s] \times [0, T]$ then the approximation solution generated by the NSFD scheme (2.6) convergence to the exact one as $\Delta t, \Delta x \to 0$, keeping $0 \leq \theta \leq 1$.*

2.4.2. Stability analysis

For stability analysis of Huxley-Burger equation we use matrix method, we assume $u_n^k = K$ as a constant, if we write the equation (2.6) in terms of constants K, the coefficient matrix of equation (2.4) can be given by

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ A_1 & F & A_1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & A_1 & F & A_1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & A_1 & F & A_1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 \end{bmatrix}.$$

Here, $F = 2R_2 + 1 - \theta - q(1 - K)(K - \gamma)$, which is also a constant that $q = \beta\phi(\Delta t)$.
The coefficient matrix is symmetric and positive definite, then its Eigenvalues are also positive and minimize the errors Eigenvalue of coefficient matrix must be less than or equal to one, so $\lambda_1 \leq 1$.
Eigenvalue for above matrix can be represented as [13].

$$1 + 2R_2 - \theta - q(1 - K)(K - \gamma) + 2\cos\frac{\pi l}{r + 1} \leq 1, \tag{2.19}$$

where $l = 1, 2, 3, \cdots, r$. When value of

a) $\cos\frac{\pi l}{r+1} = 0$, then $\frac{R_2}{q(1-K)(K-\gamma)+\theta} \leq \frac{1}{2}$,
b) $\cos\frac{\pi l}{r+1} = 1$, then $\frac{R_2}{q(1-K)(K-\gamma)+\theta} \leq \frac{1}{2}$.

2.4.3. Consistency analysis

The local truncation error (LTE) of a numerical method is an estimate of the error introduced in a single iteration of the method, assuming that everything fed into the method was perfectly accurate. Expanding the coefficients, $u_n^{k+1}, u_{n+1}^{k+1}, u_{n-1}^{k+1}, u_n^{k-1}, u_{n-1}^k$, by Taylor series method.

$$u_n^{k+1} = u_n^k + \frac{(\Delta t)}{1!}\frac{\partial u}{\partial t} + \frac{(\Delta t)^2}{2!}\frac{\partial^2 u}{\partial t^2} + \frac{(\Delta t)^3}{3!}\frac{\partial^3 u}{\partial t^3} + \cdots + O(\Delta t^4), \tag{2.20}$$

$$\begin{aligned} u_{n+1}^{k+1} = u_n^k &+ \frac{(\Delta x)}{1!}\frac{\partial u}{\partial x} + \frac{(\Delta x)^2}{2!}\frac{\partial^2 u}{\partial x^2} + \frac{(\Delta x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \cdots + O(\Delta x^4) + \frac{(\Delta t)}{1!}\frac{\partial u}{\partial t} \\ &+ \frac{(\Delta t)^2}{2!}\frac{\partial^2 u}{\partial t^2} + \frac{(\Delta t)^3}{3!}\frac{\partial^3 u}{\partial t^3} + \cdots + O(\Delta t^4), \end{aligned} \tag{2.21}$$

$$\begin{aligned} u_{n-1}^{k+1} = u_n^k &- \frac{(\Delta x)}{1!}\frac{\partial u}{\partial x} + \frac{(\Delta x)^2}{2!}\frac{\partial^2 u}{\partial x^2} - \frac{(\Delta x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \cdots + O(\Delta x^4) + \frac{(\Delta t)}{1!}\frac{\partial u}{\partial t} \\ &+ \frac{(\Delta t)^2}{2!}\frac{\partial^2 u}{\partial t^2} + \frac{(\Delta t)^3}{3!}\frac{\partial^3 u}{\partial t^3} + \cdots + O(\Delta t^4), \end{aligned} \tag{2.22}$$

$$u_n^{k-1} = u_n^k - \frac{(\Delta t)}{1!}\frac{\partial u}{\partial t} + \frac{(\Delta t)^2}{2!}\frac{\partial^2 u}{\partial t^2} - \frac{(\Delta t)^3}{3!}\frac{\partial^3 u}{\partial t^3} + \cdots + O(\Delta t^4), \tag{2.23}$$

$$u_{n-1}^k = u_n^k - \frac{(\Delta x)}{1!}\frac{\partial u}{\partial x} + \frac{(\Delta x)^2}{2!}\frac{\partial^2 u}{\partial x^2} - \frac{(\Delta x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \cdots + O(\Delta x^4). \tag{2.24}$$

Now substituting the value of equations (2.20), (2.21), (2.22), (2.23), (2.24) in equation (2.6), we get

$$\begin{aligned} \frac{1}{\Delta t}[\frac{\partial u}{\partial t}(\Delta t) + O(\Delta t^2)] + \frac{1}{\Delta x}[\alpha\frac{\partial u}{\partial x}(\Delta x) + O(\Delta x^2)] &= \frac{1}{(\Delta x^2)}[\frac{\partial^2 u}{\partial x^2}(\Delta x^2) + O(\Delta x^4)] \\ &+ \frac{1}{\Delta t}[\beta u_n^{k+1} f(u_n^k)(\Delta t) + O(\Delta t^2)]. \end{aligned} \tag{2.25}$$

Local Truncation error for above equation can be written as,

$$\begin{aligned} LTE = \lim_{\Delta x, \Delta t \to 0} &(1-2\theta)\frac{(\Delta t^2)}{2!}\frac{\partial^2 u}{\partial t^2} + \frac{(\Delta t^3)}{3!}\frac{\partial^3 u}{\partial t^3} + \cdots - 2\frac{(\Delta x^4)}{4!}\frac{\partial^4 u}{\partial x^4} - 2\frac{(\Delta x^6)}{6!}\frac{\partial^6 u}{\partial t^6} \\ &+ \cdots + \alpha(-\frac{(\Delta x^2)}{2!}\frac{\partial^2 u}{\partial x^2} + \frac{(\Delta x^3)}{3!}\frac{\partial^3 u}{\partial x^3} - \cdots + \frac{(\Delta t)}{1!}\frac{\partial u}{\partial t} + \cdots) \\ &- \beta f(u_n^k)(\frac{(\Delta t)}{1!}\frac{\partial u}{\partial t} + \frac{(\Delta t^2)}{2!}\frac{\partial^2 u}{\partial t^2} + \cdots) = 0. \end{aligned} \tag{2.26}$$

A finite difference representation of PDE is said to be consistent if we can show that the difference between PDE and its FDE representation vanishes as meh is refined. so we can write as,

$$\lim_{mesh \to 0}(PDE - FDE) = \lim_{mesh \to 0}(LTE) = 0, \tag{2.27}$$

Since $\Delta t, \Delta x$ approaches to zero, so from equation (2.26), local truncation error becomes zero, therefore the NSFD scheme (2.6) is consistent. And solving the equation (2.25) and comparing with equation (2.6) we can say order of our proposed scheme is first in time and second order in space.

3. Examples

3.1. Example1

In this Section, two examples are provided to illustrate the validity and effectiveness of the proposed methods. The initial and boundary conditions are directly obtained from analytical solution.

Consider the following Burgers-Huxley equation in the domain $[0, 1]$

$$\frac{\partial u}{\partial t} + \alpha u \frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial x^2} = \beta u(1-u)\,(u-\gamma). \tag{3.1}$$

With the initial condition

$$u(x,0) = \left\{\frac{\gamma}{2} + \frac{\gamma}{2}\tanh\left(G_1 x\right)\right\}, \tag{3.2}$$

and the boundary conditions

$$u(0,t) = \left\{\frac{\gamma}{2} + \frac{\gamma}{2}\tanh\left(-G_1 G_2 x\right)\right\}, \tag{3.3}$$

$$u(1,t) = \left\{\frac{\gamma}{2} + \frac{\gamma}{2}\tanh\left(G_1 (1 - G_2 t\right)\right\}, \tag{3.4}$$

The exact solution is presented in [14] by

$$u(x,t) = \left\{\frac{\gamma}{2} + \frac{\gamma}{2}\tanh\left\{G_1(x - G_2 t)\right\}\right\}, \tag{3.5}$$

that

$$G_1 = \frac{-\alpha + \sqrt{\alpha^2 + 8\beta}}{8}\gamma$$

$$G_2 = \frac{\gamma\alpha}{2} - \frac{(2-\gamma)(-\alpha + \sqrt{\alpha^2 + 8\beta}}{8}. \tag{3.6}$$

By using the NSFD scheme (2.6) for solving equation (1.3), we presented the absolute error for various values for x and t at $alpha = 0.1, \beta = 0.001, \gamma = 0.0001$ in Table1, and the CPU time at these points is computed.

Table 1: Shows the absolute errors for the numerical approximations are obtained by NSFD scheme (2.6), respect to the exact solution in example 1 with $\alpha = 0.1, \beta = 0.001, \gamma = 0.0001, \theta = 0.01$ and the CPU time for each point

x	t	Absolute Error	CPU time
0.1	0.02	9.2609×10^{-15}	0.001056
	0.1	1.1053×10^{-14}	0.000463
	1	1.1054×10^{-14}	0.000555
0.5	0.02	2.7310×10^{-14}	0.001145
	0.1	3.3733×10^{-14}	0.000580
	1	3.3737×10^{-14}	0.000462
0.9	0.02	1.1040×10^{-14}	0.001135
	0.1	1.5361×10^{-13}	0.000463
	1	1.3218×10^{-14}	0.000455

Figure 1, show the graphs for the numerical results of the method for various values for t. The figure shows that the numerical and exact solutions are exactly coincident together.

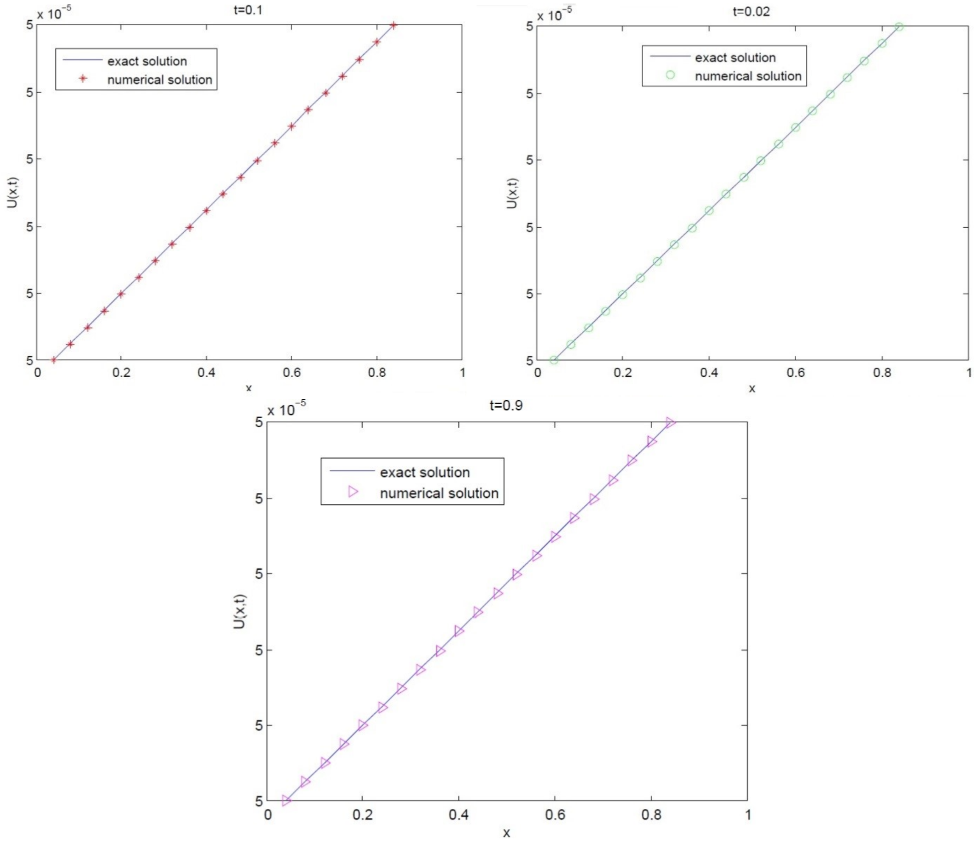

Figure 1: The graphs of the approximate and exact solutions of the partial differential Equation (1.3) in Example (1) by the model NSFD(9) for $x \in [0, 1]$ and several times. $t = 0.02, 0.1, 0.9$ the model parameter $\theta = 0.01$ and $\alpha = 1, \beta = 0.001, \gamma = 0.0001$ along with the discrete steps $\Delta t = 0.001$, $\Delta x = 0.05$.

Figure 2, illustrates the graph of the absolute error for the numerical method NSFD (9), with $\alpha = 0.1, \beta = 0.001, \gamma = 0.0001$ at different time level, for $\Delta t = 0.001$, $\Delta x = 0.05$.

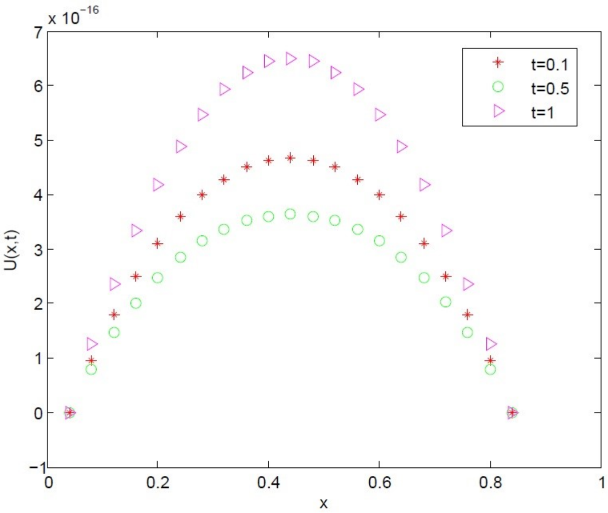

Figure 2: The absolute error of NSFD (9) model in Example 1 with $\alpha = 0.1, \beta = 0.001, \gamma = 0.0001$ at different time levels using $\Delta t = 0.0001, \Delta x = 0.05$.

Table2, shows comparison of present method with [1], [3], [5], [7], [9] for $\alpha = \beta = 1, \gamma = 0.001$

From table 1-2, it can be observed that the computed results show excellent agreement with the exact solution.

Table 2: Shows the comparison of absolute error for the Example 1 with $\alpha = 1, \beta = 1, \gamma = 0.001$.

x	t	Present method	[1]	[3]	[5]	[7]	[9]
0.1	0.05	1.0842×10^{-18}	1.8740×10^{-8}	1.8740×10^{-8}	1.8740×10^{-8}	1.8740×10^{-8}	1.5454×10^{-8}
	0.1	1.5179×10^{-18}	3.7481×10^{-8}	3.7481×10^{-8}	3.7481×10^{-8}	3.7481×10^{-8}	2.2593×10^{-8}
	1	3.2526×10^{-19}	3.7481×10^{-7}	3.7481×10^{-7}	3.7481×10^{-7}	3.7481×10^{-7}	3.3729×10^{-8}
0.5	0.05	3.8520×10^{-10}	1.8740×10^{-8}	1.8740×10^{-8}	1.8740×10^{-8}	1.8740×10^{-8}	3.4705×10^{-8}
	0.1	2.1825×10^{-10}	3.7481×10^{-8}	1.3748×10^{-8}	3.7481×10^{-8}	3.7481×10^{-8}	5.7659×10^{-8}
	1	3.8520×10^{-10}	3.7481×10^{-7}	3.7481×10^{-7}	3.7481×10^{-7}	3.7481×10^{-7}	9.3698×10^{-8}
0.9	0.05	2.5680×10^{-10}	1.8740×10^{-8}	1.8740×10^{-8}	1.8740×10^{-8}	1.8740×10^{-8}	3.4705×10^{-8}
	0.1	2.5680×10^{-10}	3.7481×10^{-8}	3.7481×10^{-8}	3.7481×10^{-8}	3.7481×10^{-8}	5.7659×10^{-8}
	1	2.2458×10^{-10}	3.7481×10^{-7}	3.7481×10^{-7}	3.7481×10^{-7}	3.7481×10^{-7}	9.3698×10^{-8}

3.2. Example2

Consider the generalized Huxley equation of the form

$$\frac{\partial u}{\partial t} - \alpha \frac{\partial^2 u}{\partial x^2} = \beta u(1 - u^\delta)\,(u - \gamma^\delta), \tag{3.7}$$

with the initial condition

$$u(x, 0) = \left\{ \frac{\gamma}{2} + \frac{\gamma}{2} \tanh\left(\sigma\gamma x\right) \right\}^{\frac{1}{\delta}}, \tag{3.8}$$

and the boundary conditions

$$u(0, t) = \left\{ \frac{\gamma}{2} + \frac{\gamma}{2} \tanh\{\sigma\gamma\{\frac{(1+\delta-\gamma)\rho}{2(1+\delta)}\}t\} \right\}^{\frac{1}{\delta}}, \tag{3.9}$$

and

$$u(1, t) = \left\{ \frac{\gamma}{2} + \frac{\gamma}{2} \tanh\{\sigma\gamma\{1 + \frac{(1+\delta-\gamma)\rho}{2(1+\delta)}\}t\} \right\}^{\frac{1}{\delta}}, \tag{3.10}$$

the exact solution of this equation was derived by [13] using nonlinear transformations and is given by

$$u(x, t) = \left\{ \frac{\gamma}{2} + \frac{\gamma}{2} \tanh\{\sigma\gamma\{x + \frac{(1+\delta-\gamma)\rho}{2(1+\delta)}\}t\} \right\}^{\frac{1}{\delta}}, \tag{3.11}$$

where

$$\sigma = \frac{\delta\rho}{4(1+\delta)},$$

and

$$\rho = \sqrt{4\beta(1+\delta)}.$$

For the numerical computation, we use the parameters $\beta = 1, \gamma = 0.001, \delta = 1, 2, 3$. Numerical results absolute errors of different values of t are given in table 3-5. The comparison of the present method with [9] is shown in these tables. From these results, it is obvious that numerical solutions are in excellent agreement with the exact solution.

Table 3: Shows the comparison of absolute error for the Example 2 with $\beta = 1, \gamma = 0.001, \delta = 1$.

x	t	Absolute error Present method	Absolute error[9]
0.1	0.05	4.9504×10^{-11}	1.03030×10^{-8}
	0.1	9.9439×10^{-11}	1.50629×10^{-8}
	1	1.4929×10^{-10}	2.24877×10^{-8}
0.5	0.05	4.9513×10^{-11}	2.36369×10^{-8}
	0.1	9.9492×10^{-11}	3.84395×10^{-8}
	1	1.4947×10^{-10}	6.24653×10^{-8}
0.9	0.05	4.9512×10^{-11}	1.03030×10^{-8}
	0.1	9.9483×10^{-11}	1.50629×10^{-8}
	1	1.4943×10^{-10}	2.24877×10^{-8}

Table 4: Shows the comparison of absolute error for the Example 2 with $\beta = 1, \gamma = 0.001, \delta = 2$.

x	t	Absolute error Present method	Absolute error[9]
0.1	0.05	7.4163×10^{-11}	1.40343×10^{-6}
	0.1	1.4909×10^{-10}	2.05158×10^{-6}
	1	2.2388×10^{-10}	3.05621×10^{-6}
0.5	0.05	7.4179×10^{-11}	3.15158×10^{-6}
	0.1	1.4917×10^{-10}	5.23566×10^{-6}
	1	2.2416×10^{-10}	8.49007×10^{-6}
0.9	0.05	7.4180×10^{-11}	1.40337×10^{-6}
	0.1	1.4917×10^{-10}	2.05162×10^{-6}
	1	2.2412×10^{-10}	3.05644×10^{-6}

Table 5: Shows the comparison of absolute error for the Example 2 with $\beta = 1, \gamma = 0.001, \delta = 3$.

x	t	Absolute error Present method	Absolute error[9]
0.1	0.05	7.3471×10^{-11}	8.79142×10^{-6}
	0.1	1.4842×10^{-10}	1.28501×10^{-5}
	1	2.2323×10^{-10}	1.90246×10^{-5}
0.5	0.05	7.3485×10^{-11}	1.97445×10^{-5}
	0.1	1.4850×10^{-10}	3.27943×10^{-5}
	1	2.2350×10^{-10}	5.28544×10^{-5}
0.9	0.05	7.3483×10^{-11}	8.79142×10^{-6}
	0.1	1.4849×10^{-10}	1.28495×10^{-5}
	1	2.2345×10^{-10}	1.90252×10^{-5}

Figure3 illustrates the graphs of the absolute errors of numerical solutions of the NSFD (9) with $\delta = 1, 2, 3, 4$, for $\Delta t = 0.0001$, $\Delta x = 0.01, \theta = 0.01$ in Example 2.

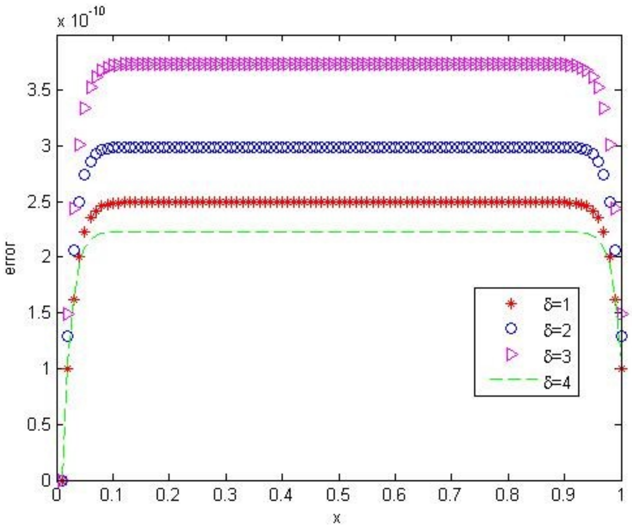

Figure 3: The graphs of absolute error of NSFD (9) for, $\delta = 1, 2, 3, 4$ with $\Delta t = 0.0001$, $\Delta x = 0.01$ in Example2.

Figure4, shows the (3D) graph of the numerical solution and exact solution of NSFD(9) for Example 2. The figure shows that the numerical and exact solutions are exactly coincident together.

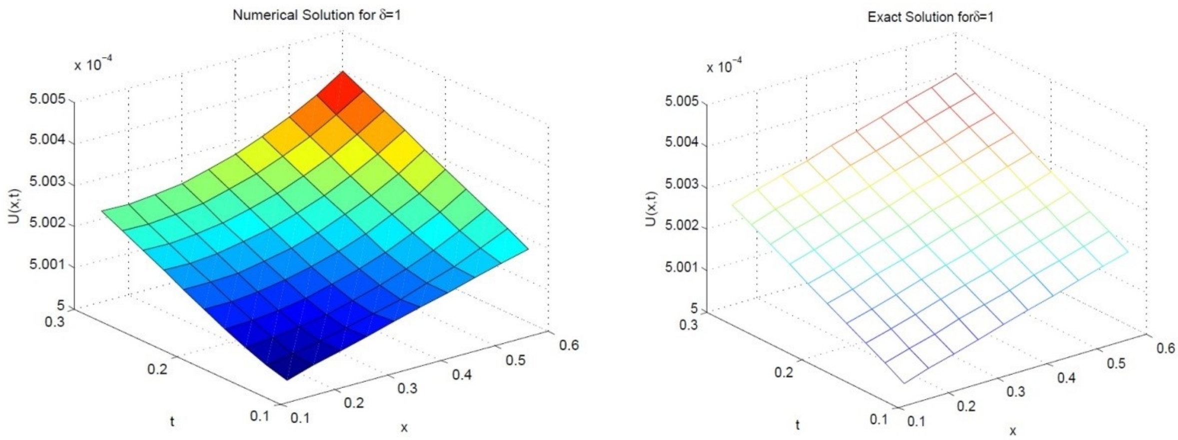

Figure 4: 3(D) graphs of the approximate and exact solutions of the partial differential Equation (3.7) in Example (2) obtained by the model NSFD(9) with the discrete steps $\Delta t = 0.0001$, $\Delta x = 0.01$.

Tables 6, 7, illustrate the absolute error between the exact and numerical results obtained by NSFD (9), for the Equation (3.1) and (3.7) with some amount of θ, these results show that the absolute error of these NSFD schemes depend on θ, when close to zero the NSFD scheme has the lowest absolute error.

Table 6: Shows the absolute error for the results obtained by NSFD(9) in Example1 with $\Delta t = 0.001, \Delta x = 0.01, \theta = 0.0001.0.1, 0.5, \alpha = 0.1, \beta = 0.001, \gamma = 0.0001$ at $t = 0.5$.

θ	0.0001	0.1	0.5
error	7.6549×10^{-15}	5.8084×10^{-13}	6.6663×10^{-11}

Table 7: Shows the absolute error for the results obtained by NSFD(9) in Example2 with $\Delta t = 0.001$, $\Delta x = 0.01, \theta = 0.0001.0.1, 0.5, \delta = 1$ at $t = 0.5$.

θ	0.0001	0.1	0.5
error	1.4990×10^{-9}	1.3487×10^{-8}	1.8100×10^{-7}

As all the Figures and Tables show, the proposed methods give very accurate results.

4. Conclusion

In this paper, the solution of the Berger-Huxley and Generalized Huxley Equations is successfully approximated by a high-order numerical NSFD method. The convergence, consistency and stability analysis for this NSFD scheme have been proved. Numerical solutions for different θ are given using tables. The absolute error of these NSFD schemes depends on θ, when θ close to zero the NSFD scheme has the lowest absolute error. The numerical results from the method have been compared with the exact solution and the results [1, 3, 5, 7, 9]. As the numerical results show, performance of the methods is in excellent agreement with the exact solution. It may be concluded that the NSFD method (9) is very powerful and efficient technique for finding an approximate solution for various kind of linear/nonlinear problems.

Note: The data used to support the findings of this study are included in the article. We have not used any extra data in this article. We have solved the Equation by a mathematical technique and all the results are inside the paper.

References

[1] I. Hashim, M.S.M. Noorani and M.R.S. AL-Hadidi, *Solving the generalized Burgers-Huxley equation using the Adomian decomposition method*, Math. Computer Model. 43 (2006) 1404–1411.

[2] M. Javidi, *A numerical solution of the generalized Burger-Huxley equation by pseudospectral method and Darvishi's preconditioning*, Appl. Math. Comput. 175 (2006) 1619–1628.

[3] B. Batiha, M.S.M. Noorani and I. Hashim, *Application of variational iteration method to the generalized Burger-Huxley equation*, Chaos, Solutions and Fractals 36 (2008) 660–663.

[4] M. Javidi and A. Gollbabai, *A new domain decomposition algorithm for generalized Burger-Huxley equation based on Chebyshev polynomials and preconditioning*, Chaos, Solutions and Fractals 39(2) (2009) 849–857.

[5] J. Biazar and F. Mahmoodi, *Application of differential transform method to the generalized Burger-Huxley equation, Applications and Applied mathematics*, Appl. Appl. Math. Int. J. 5(10) (2010) 1726–1740.

[6] I. Celik, *Haar wavelet method for solving generalized Burger-Huxley equation*, Arab J. Math. Sci. 18 (2012) 25–37.

[7] A.M. AL-Rozbayani, *Discrete Adomian decomposition method for solving Burger-Huxley equation*, Int. J. Contemp. Math. Sci. 8(13) (2013) 623–631.

[8] A.G. Bratsos, *A fourth order improved numerical scheme for the generalized Burger-Huxley equation*, Amer. J. Comput. Math. 1 (2011) 152–158.

[9] B. Inan, *finite difference methods for generalized Huxley and Burger-Huxley equations*, Kuwait J. Sci. 44(3) (2017) 20–27.

[10] J. Noye, *Numerical Solution of Partial Differential Equations Finite Difference Methods*, Honours Applied Mathematics, 1993.

[11] D. Hoff, *stability and convergence of finite difference methods for systems of nonlinear reaction-diffusion*, SIAM J. Numerical Anal. 15(6) (1978) 1161–1177.

[12] J. Noye and H.H. Tan, *Finite difference methods for solving the two-dimensional advection-diffusion equation*, Int. J. Numerical Meth. Fluids 9 (1989) 75–89.

[13] A.R. Mitchell and D.F. Griffiths, *Hand Book of Finite Difference Method in Partial Differential Equation*, Wiley-Inter Science publication, 1980.

[14] H. Naja, S. Edalatpanah and A.H. Refahi Sheikhani, *Convergence Analysis of Modified Iterative Methods to solve Linear Systems*, Mediter. J. Math. 11(3) (2014) 1019–1032.

[15] H. Saberi, H. Naja and A. Refahi, *FOM-inverse vector iteration method for computing a few smallest (largest) eigenvalues of pair* (A, B), Appl. Math. Comput. 184(3) (2007) 421–428.