High-order solution of Generalized Burgers–Fisher Equation using compact finite difference and DIRK methods

Mardan A. Pirdawood 1, Younis A. Sabawi 2
1,2 Department of Mathematics, Faculty of Science and Health, Koya University, Koya KOY45, Kurdistan Region - F.R. Iraq

Abstract. The main goal of this paper is to developed a high-order and accurate method for the solution of one-dimensional of generalized Burgers-Fisher with Numman boundary conditions. We combined between a fourth-order compact finite difference scheme for spatial part with diagonal implicit Runge Kutta scheme in temporal part. In addition, we discretized boundary points by using a compact finite difference scheme in terms of fourth order accuracy. This combine leads to ordinary differential equation which will take full advantage of method of line (MOL). Some numerical experiments presented to show that the combination give an accurate and reliable for solving the generalized Burgers-Fisher problems.

Keywords: A compact difference methods, IMEX-RK methods, Generalized Burgers–Fisher Equation, A compact Fourth-Order Diagonally Implicit Runge-Kutta Type Method.

1. Introduction

In numerous areas of engineering and physics, the most physical phenomenon can be modeled as non-linear partial differential equations (NPDEs). Finite element and Compact finite difference methods of solutions to NPDEs has attracted much interest and many numerical schemes have been suggested, see for example [1-17].

In terms of construction high order compact finite difference methods with Neumann boundary conditions are considered in [17–19]. Cao al el [17] constructed a fourth-order compact finite difference scheme for solving the convection–diffusion equation with Neumann boundary conditions. In their works, they proposed compact method of fourth-order to discrete interior and boundary points. Yao al el [18] have studied a fourth-order scheme for dealing with the model equation of simulated moving bed. They used direct method and pseudo grid point method to derive boundary condition. Fu al el [19] have used a high-order exponential scheme convection-diffusion equation with Neumann boundary conditions.

The main contribution of this work is to derive fourth-order accurate in both interior and boundary points. These difficulties are addressed by employing techniques introduced by [19] for boundary point

Consider the generalized form of the Burgers’-Fisher equation with Neumann boundary conditions:

\[
\frac{\partial u}{\partial t} + \alpha u \frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial x^2} = \beta u(1 - u^r), \quad 0 \leq x \leq 1, \text{ and } t \geq 0,
\]

The initial condition:

\[ u(x, 0) = u_0(x) \]
The boundary conditions at $x = 0$, and $x = 1$:

$$u(0, t) = \left( \frac{1}{2} + \frac{1}{2} \tanh \left( \frac{\alpha y}{2(y + 1)} \right) \right)^{\frac{1}{p}}, \quad t \geq 0,$$

$$u(1, t) = \left( \frac{1}{2} + \frac{1}{2} \tanh \left( \frac{-\alpha y}{2(y + 1)} \left( 1 - \left( \frac{\alpha}{\gamma + 1} + \frac{\beta(y + 1)}{\alpha} \right) t \right) \right) \right)^{\frac{1}{p}}, \quad t \geq 0,$$

and the Neumann boundary conditions:

$$\frac{\partial u}{\partial x}_{x=0} = \phi(t) = \frac{-\alpha t}{4(y + 1)} \sech^2 \left( \frac{\alpha y}{2(y + 1)} \left( \frac{\alpha}{\gamma + 1} + \frac{\beta(y + 1)}{\alpha} \right) t \right) \left( \frac{1}{2} \right)$$

$$+ \frac{1}{2} \tanh \left( \frac{\alpha y}{2(y + 1)} \left( \frac{\alpha}{\gamma + 1} + \frac{\beta(y + 1)}{\alpha} \right) t \right)^{\frac{1}{p-1}},$$

$$\frac{\partial u}{\partial x}_{x=1} = \phi(t) = \frac{-\alpha t}{4(y + 1)} \sech^2 \left( \frac{-\alpha y}{2(y + 1)} \left( 1 - \left( \frac{\alpha}{\gamma + 1} + \frac{\beta(y + 1)}{\alpha} \right) t \right) \right) \left( \frac{1}{2} \right)$$

$$+ \frac{1}{2} \tanh \left( \frac{-\alpha y}{2(y + 1)} \left( 1 - \left( \frac{\alpha}{\gamma + 1} + \frac{\beta(y + 1)}{\alpha} \right) t \right) \right)^{\frac{1}{p-1}}.$$

2. Compact finite difference scheme

2.1 spatial domain

In this section, we discretise on spatial domain, by a fourth-order compact difference approximation for the space part.

Go back to (1) and setting $\psi(x) = \frac{\partial u}{\partial t} - \beta u(1 - U^r)$, becomes

$$\frac{d^2 U}{dx^2} - a U^r \frac{du}{dx} = \psi(x).$$

(2)

Discretising (2) by $\delta_x U_l$ and $\delta_x^2 U_l$ gives

$$\delta_x^2 U_l - \alpha U_l^r \delta_x U_l - \tau = \psi_l,$$

where

$$\delta_x^2 U_l = \frac{U_{l+1} - 2U_l + U_{l-1}}{h^2} + \frac{h^2}{12} \delta_x^4 U_l,$$

$$\delta_x U_l = \frac{U_{l+1} - U_{l-1}}{2h} + \frac{h^2}{6} \delta_x^3 U_l,$$

$$\tau = \frac{h^2}{12} \left[ \frac{d^4 U}{dx^4} - 2a U_l^r \frac{d^3 U}{dx^3} \right] + O(h^4).$$

(4)

We seek to approximate (4) in fourth order. This accomplishes by differentiating (2) to yield:

$$\frac{d^3 U}{dx^3} = a U^r \frac{d^2 U}{dx^2} + a \gamma U^{r-1} \left( \frac{du}{dx} \right)^2 + \frac{d\psi}{dx},$$

$$\frac{d^4 U}{dx^4} = a U^r \frac{d^3 U}{dx^3} + 3a \gamma U^{r-1} \frac{dU}{dx} \frac{d^2 U}{dx^2} + \alpha U^{r-2} \frac{d^2 U}{dx^2} + a \gamma (y - 1) U^{r-2} \frac{dU}{dx} \frac{d^2 U}{dx^2} + \frac{d^2 \psi}{dx^2},$$

(5)
Substituting (7) and (8) in (6), imply
\[ \tau = \frac{h^2}{12} \left( 3\alpha U_{y=1} \delta_x U_j^2 U_i + \alpha y (y - 1) U_{y=2}^2 (\delta_x U_j)^3 - \alpha U_{y=2}^2 \delta_x U_i - \alpha^2 y U_{y=2} \delta_x U_j^2 + \delta_x^2 \psi_i \right) + O(h^4). \]

Go back to (4) and substituting above equation in (4), reads
\[ r_1 \frac{dU_{i-1}}{dt} + r_2 \frac{dU_i}{dt} + r_3 \frac{dU_{i+1}}{dt} = \left( r_4 + r_6 \beta (1 - U_i^R) \right) U_{i-1} \]
\[ + \left( r_5 + r_6 \beta (1 - U_{i+1}^R) \right) U_i + \left( r_6 + r_6 \beta (1 - U_{i+1,j}^R) \right) U_{i+1,j}, \]
(6)

where
\[ r_1 = 1 + 0.5 \text{h}a \text{h} U_i^R, \quad i = 1, ..., N, \]
\[ r_2 = 10, \]
\[ r_3 = 1 - 0.5 \text{h}a \text{h} U_i^R, \quad i = 1, ..., N, \]
\[ r_4 = \frac{12}{h^2} + \alpha U_{i,j}^2 + \frac{6}{h} \alpha U_i^R + \frac{\alpha y (y - 1)}{8h} U_{i,j}^R U_i U_{i+1}\]
\[ + \frac{1.5}{h} \alpha U_{i,j}^R U_{i-1} U_{i,1} - 0.5 \alpha U_{i,j}^R U_{i+1,1}, \]
\[ i = 1, ..., N, j = 1, ..., M - 2, \]
\[ r_5 = \frac{24}{h^2} + 2 \alpha U_{i,j}^2, \quad i = 1, ..., N, j = 2, ..., M - 1, \]
\[ r_6 = \frac{12}{h^2} + \alpha U_{i,j}^2 - \frac{6}{h} \alpha U_i^R - \frac{\alpha y (y - 1)}{8h} U_{i,j}^R U_i U_{i,1}\]
\[ - \frac{1.5}{h} \alpha U_{i,j}^R U_{i-1} U_{i,1} + 0.5 \alpha U_{i,j}^R U_{i+1,1}, \]
\[ i = 1, ..., N, j = 3, ..., M. \]

2.2 The boundary spatial points

The main goal here is to construct a fourth order accurate on the spatial points for the left and right boundaries. This may be accomplished by using techniques deduced in [19], to do this, begin with lifting Neumann boundary condition, yields,
\[ \frac{\partial U(x_0,t)}{\partial x} = \Delta_{2x} U(x_0,t) + \frac{\partial^3 U(x_0,t)}{\partial x^3} + \frac{h^3 \partial^3 U(x_0,t)}{4 \partial x^3} + O(h^4), \]
(7)
\[ \frac{\partial U(x_0,t)}{\partial x} = \Delta_{2x} U(x_n,t) + \frac{\partial^3 U(x_n,t)}{\partial x^3} + \frac{h^3 \partial^3 U(x_n,t)}{4 \partial x^3} + O(h^4), \]
(8)

where
\[ \Delta_{2x} U(x_0,t) = \frac{-3U(x_0,t) + 4U(x_1,t) - U(x_2,t)}{2h}, \]
(9)
\[ \Delta_{2x} U(x_n,t) = \frac{-3U(x_{n-2},t) + 4U(x_{n-1},t) - U(x_n,t)}{2h}. \]
Substituting (7) and (8) into (10) along with Neumann boundary conditions, this becomes

\[ \phi_0(t) = \Delta_{2x}^1 U(x_0, t) + \frac{h^2}{3} \left[ a \alpha U \Delta_{2x}^2 U(x_0, t) + a \gamma U^{\gamma-1}(\phi_0(t))^2 + \Delta_{2x}^1 \psi(x_0, t) \right] \\
+ \frac{h^2}{4} \left[ a^2 U^{2 \gamma}(x_0, t) \Delta_{2x}^1 U(x_0, t) + a^2 \gamma U^{\gamma-1}(\phi_0(t))^2 + 3a \gamma U^{\gamma-1}(x_0, t) \phi_0(t) \Delta_{2x}^1 U(x_0, t) + \right. \\
\left. a \gamma (y-1) U^{\gamma-2}(\phi_0(t))^3 + \Delta_{2x}^1 \psi(x_0, t) + a U^{\gamma}(x_0, t) \Delta_{2x}^1 \psi(x_0, t) \right]. \]

Applying \( \Delta_{2x}^1 U(x_0, t) = \frac{-3U_y + 4U_{y2} - U_2}{2h} \) and \( \Delta_{2x}^1 = \frac{U_{y3} - U_3}{h} \) in above equation, imply that

\[ r_{00} \frac{\partial U(x_0, t)}{\partial t} + r_{01} \frac{\partial U(x_1, t)}{\partial t} + r_{02} \frac{\partial U(x_2, t)}{\partial t} = b_{00} U_0(t) + b_{01} U_1(t) + b_{02} U_2(t) + \frac{\alpha U^\gamma(t)}{3d_1} U_3(t) + \phi_0(t). \]

where

\[ r_{00} = -\frac{h}{4} - \frac{ah^2}{4} U_{y2}(t), \quad r_{01} = \frac{h}{6} + \frac{ah^2}{4} U_x(t), \quad r_{02} = \frac{h}{12}, \]

\[ b_{00} = -\left( \frac{2aU_y^\gamma(t)}{3} - \frac{3h}{2h} \right) + \frac{h}{4} \left( a^2 U_{y2}^{\gamma-1}(t) + 3\alpha \gamma U_{y3}^{\gamma-2}(t) \phi_0(t) \right) \left( U_0(t) - 2U_1(t) + U_2(t) \right) + \]
\[ \frac{h^2}{4} \left( a^2 \gamma U_{y3}^{\gamma-2}(t) \phi_0(t)^2 + \frac{h}{3} \alpha \gamma (y-1) U_{y3}^{\gamma-3}(t) \phi_0(t)^3 \right) + \frac{h^2}{4} \alpha \gamma U_{y3}^{\gamma-2}(t) \phi_0(t)^2 - \beta \left( 1 - U_x^\gamma(t) \right) r_{00}, \]

\[ b_{01} = -\left( \frac{2}{h} - \frac{5aU_y(t)}{3} - \beta \left( 1 - U_x^\gamma(t) \right) r_{01} \right), \]

\[ b_{02} = -\left( \frac{4aU_y^\gamma(t)}{3} - \frac{1}{2h} \beta \left( 1 - U_x^\gamma(t) \right) r_{02} \right). \]

Go back to (11), follow with the same techniques for constructing left boundary point, this becomes

\[ r_{NM} \frac{\partial U(x_N, t)}{\partial t} + r_{N,M-1} \frac{\partial U(x_{N-1}, t)}{\partial t} + r_{N,M-2} \frac{\partial U(x_{N-2}, t)}{\partial t} = b_{NM} U_N(t) + b_{N,M-1} U_{N-1}(t) + b_{N,M-2} U_{N-2}(t) + \frac{\alpha U^\gamma(t)}{3d_1} U_{N-3}(t) + \varphi_N(t), \]

where

\[ r_{NM} = \frac{3h}{4} + \frac{ah^2}{4} U_{y2}(t), \quad r_{N,M-1} = -\frac{7h}{6} - \frac{ah^2}{4} U_x(t), \quad r_{N,M-2} = \frac{5h}{12}, \]

\[ b_{NM} = -\left( \frac{3}{2h} + \frac{2a}{3} U_y^\gamma(t) + \frac{h^2}{3} \alpha \gamma U_{y3}^{\gamma-2}(t) \phi_N(t)^2 + \frac{h^3}{4} \alpha^2 \gamma U_{y3}^{\gamma-2}(t) \phi_N(t)^2 + \frac{h}{4} \left( a^2 U_{y3}^{\gamma-1}(t) + 3\alpha \gamma U_{y3}^{\gamma-2}(t) \phi_N(t) \right) \left( U_0(t) - 2U_1(t) + U_2(t) \right) + \right. \]
\[ \frac{h^2}{4} \left( a^2 \gamma U_{y3}^{\gamma-2}(t) \phi_N(t)^2 + \frac{h}{3} \alpha \gamma (y-1) U_{y3}^{\gamma-3}(t) \phi_N(t)^3 \right) - \beta \left( 1 - U_x^\gamma(t) \right) r_{NM}, \]

\[ b_{N,M-1} = -\left( \frac{2}{h} - \frac{5a}{3} U_y^\gamma(t) - \beta \left( 1 - U_x^\gamma(t) \right) r_{N,M-1} \right), \]

\[ b_{N,M-2} = -\left( \frac{1}{2h} + \frac{4aU_y^\gamma(t)}{3} - \beta \left( 1 - U_x^\gamma(t) \right) r_{N,M-2} \right). \]

Collecting together (9), (13), and (14), we have a system of linear ordinary differential equations as follows

\[ \frac{dU(t)}{dt} = \mathbf{A}^{-1} \mathbf{BU}(t) + \mathbf{A}^{-1} \mathbf{g}(t), \]
\[ A = \begin{bmatrix} r_{00} & r_{01} & r_{02} \\ r_{1} & r_{2} & r_{3} \\ \vdots & \vdots & \vdots \\ r_{N,M-2} & r_{N,M-1} & r_{N,M} \end{bmatrix}, \quad B = \begin{bmatrix} b_{00} & b_{01} & b_{02} & \frac{aU^Y_X(t)}{3} \\ b_{1} & b_{2} & b_{3} & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ b_{N,M-2} & b_{N,M-1} & b_{N,M} \end{bmatrix}, \]

where \( U(t) = [u_0(t), u_1(t), ..., u_N(t)]^T \), \( g(t) = [\phi_0(t), 0, ..., 0, \varphi_N(t)]^T \),

- \( b_1 = \left(r_0 + r_i \beta \left(1 - U^Y_{i-1} \right) \right), \quad i = 1, ..., N, j = 1, ..., M - 2, \)
- \( b_2 = \left(-r_0 + r_i \beta \left(1 - U^Y_{i+1} \right) \right), \quad i = 1, ..., N, j = 2, ..., M - 1, \)
- \( b_3 = \left(r_0 + r_i \beta \left(1 - U^Y_{i+1} \right) \right), \quad i = 1, ..., N, j = 3, ..., M. \)

### 3. Diagonally Implicit Runge-Kutta methods (DIRK)
This sections aims to combine the high-order compact difference scheme presented in section 2 with DIRK schemes to obtain a new and high-order method for solving (1). We now extend the algorithm to the ODE system. Assume that \( U^n = (u^n_0, u^n_1, ..., u^n_N)^T \) is the numerical solution of Eq. (14) at time \( t^n \), the following algorithm is used to calculate \( U^{n+1} \):

\[
K_1 = F(U^n + r\Delta t K_{1,1}, x, t^n + r\Delta t), \\
K_2 = F \left(U^n + \left(\frac{1}{2} - r\right)\Delta t K_{2,1} + r\Delta t K_{2,2}, x, t^n + \frac{\Delta t}{2} \right), \\
K_3 = F \left(U^n + 2r\Delta t K_{1,1} + (1 - 4r)\Delta t K_{2,2} + r\Delta t K_{3,3}, x, t^n + (1 - r)\Delta t \right), \\
U^{n+1} = U^n + \Delta t \left( rK_1 + \frac{K^2_2}{2} + (1 - r)K_3 \right),
\]

where \( r = \frac{1}{2} + \frac{\sqrt{3}}{3}\cos\left(\frac{\pi}{18}\right) \), \( K_1, K_2 \) and \( K_3 \) are vectors and \( F(U, x, t) = A^{-1}BU + A^{-1}g(t) \), the matrices \( \mathcal{A} \) and \( B \) are defined in Eq. (14). Combining Eq. (15) and Eq. (16), we obtain the following scheme for solving \( K_1, K_2 \), and \( K_3 \):

\[
\mathcal{A}(U^n + r\Delta t K_{1,1}, x, t^n + r\Delta t)K_1 \]

\[
\mathcal{A}(U^n + \left(\frac{1}{2} - r\right)\Delta t K_{2,1} + r\Delta t K_{2,2}, x, t^n + \frac{\Delta t}{2})K_2 \\
+ B \left(U^n + \left(\frac{1}{2} - r\right)\Delta t K_{1,1} + r\Delta t K_{2,2}, x, t^n + \frac{\Delta t}{2} \right) \left(U^n + \left(\frac{1}{2} - r\right)\Delta t K_{1,1} + r\Delta t K_{2,2} \right)
\]

\[
\mathcal{A}(U^n + 2r\Delta t K_{1,1} + (1 - 4r)\Delta t K_{2,2} + r\Delta t K_{3,3}, x, t^n + (1 - r)\Delta t)K_3 \\
+ B \left(U^n + 2r\Delta t K_{1,1} + (1 - 4r)\Delta t K_{2,2} + r\Delta t K_{3,3}, x, t^n + (1 - r)\Delta t \right) \left(U^n + 2r\Delta t K_{1,1} + (1 - 4r)\Delta t K_{2,2} + r\Delta t K_{3,3} \right) + g(t^n + (1 - r)\Delta t). 
\]

The main challenge in the solution of such problem is having to deal with a nonlinear term. We tackle the challenge for this problem by employing modified Newton-Raphson. To do that, applying Newton-Raphson iteration on (16), gives
where

\[ u(x,0) = u_0(x) = \left( \frac{1}{2} + \frac{1}{2} \tanh \left( \frac{-\alpha y}{2(y+1)} x \right) \right)^{\frac{1}{\beta}}, \quad 0 \leq x \leq 1, \]

The boundary conditions at \( x = 0 \), and \( x = 1 \):

\[ u(0,t) = \left( \frac{1}{2} + \frac{1}{2} \tanh \left( \frac{\alpha y}{2(y+1)} \left( \frac{\alpha}{y+1} + \frac{\beta(y+1)}{\alpha} \right) t \right) \right)^{\frac{1}{\beta}}, \quad t \geq 0, \]

\[ u(1,t) = \left( \frac{1}{2} + \frac{1}{2} \tanh \left( \frac{-\alpha y}{2(y+1)} \left( 1 - \left( \frac{\alpha}{y+1} + \frac{\beta(y+1)}{\alpha} \right) t \right) \right) \right)^{\frac{1}{\beta}}. \]

The exact solution of Eq. (1) is

\[ u(x,t) = \left( \frac{1}{2} + \frac{1}{2} \tanh \left( \frac{-\alpha y}{2(y+1)} \left( x - \left( \frac{\alpha}{y+1} + \frac{\beta(y+1)}{\alpha} \right) t \right) \right) \right)^{\frac{1}{\beta}}, \quad 0 \leq x \leq 1, t \geq 0. \]

where \( \alpha, \beta \) and \( \gamma \) are constants.
Figure 1. Example 4.1. The numerical solution of the DIRK scheme in left and the exact solution for the generalized Burger’s Fisher equation in the right at time $t = 0.001$ whenever $\alpha = 50, \beta = 0.01$ and $\gamma = 3$.

Figure 2. Example 4.1. The numerical solution of the DIRK scheme in left and the exact solution for the generalized Burger’s Fisher equation in the right at time $t = 0.001$ whenever $\alpha = 1, \beta = 0.01$ and $\gamma = 2$. 
Figure 3. Example 4.1. A comparison between the DIRK solutions and the exact solution for the generalized Burger’s Fisher equation at time $t = 0.001$ whenever $\alpha = 50, \beta = 0.01$ and $\gamma = 3$.

Figure 4. Example 4.1. A comparison between the DIRK solutions and the exact solution for the generalized Burger’s Fisher equation at time $t = 0.001$ whenever $\alpha = 1, \beta = 0.01$ and $\gamma = 2$.

| $x$  | DIRK       | Exact Solution | Absolute error |
|------|------------|----------------|----------------|
| 0    | 8.47875175e-01 | 8.47875175e-01 | 0              |
| 0.2  | 8.21524761e-02 | 9.51937807e-02 | 1.30413046e-02 |
| 0.4  | 6.74762190e-03 | 7.81622827e-03 | 1.06860637e-03 |
| 0.6  | 5.53878871e-04 | 6.41595197e-04 | 8.77163262e-05 |
| 0.8  | 4.54676695e-05 | 5.26623652e-05 | 7.19469576e-06 |
| 1    | 3.81469727e-06 | 3.81469727e-06 | 0              |

Table 1. The absolute errors of the DIRK solutions and the exact solution for the generalized Burger’s Fisher equation at time $t = 0.001$ whenever $\alpha = 50, \beta = 0.01$ and $\gamma = 3$. 
Table 2. The absolute errors of the DIRK solutions and the exact solution for the generalized Burger’s Fisher equation at time $t = 0.001$ whenever $\alpha = 1, \beta = 0.01$ and $\gamma = 2$.

| $x$  | DIRK                      | Exact Solution | Absolute error |
|------|----------------------------|----------------|----------------|
| 0    | 7.07147458e-01            | 7.07147458e-01 | 0              |
| 0.2  | 6.84138881e-01            | 6.83208047e-01 | 9.3083204e-04  |
| 0.4  | 6.59517559e-01            | 6.58621381e-01 | 8.9617846e-04  |
| 0.6  | 6.34395493e-01            | 6.33535814e-01 | 8.5967914e-04  |
| 0.8  | 6.08929386e-01            | 6.08106407e-01 | 8.2297914e-04  |
| 1    | 5.82490527e-01            | 5.82490527e-01 | 0              |

Table 3. The absolute errors of the DIRK solutions and the exact solution for the generalized Burger’s Fisher equation at time $t = 0.001$ whenever $\alpha = 1, \beta = 1$ and $\gamma = 1$.

| $x$  | DIRK                      | Exact Solution | Absolute error |
|------|----------------------------|----------------|----------------|
| 0    | 5.00296875e-01            | 5.00296875e-01 | 0              |
| 0.2  | 4.75696774e-01            | 4.75316955e-01 | 3.79819116e-04 |
| 0.4  | 4.50807926e-01            | 4.50459946e-01 | 3.47980331e-04 |
| 0.6  | 4.26164317e-01            | 4.25847803e-01 | 3.16513740e-04 |
| 0.8  | 4.01884622e-01            | 4.01597683e-01 | 2.86939449e-04 |
| 1    | 3.77819776e-01            | 3.77819776e-01 | 0              |

5. Conclusion
This paper devotes to propose a high-order compact scheme for solving the one-dimensional of generalized Burgers-Fisher with Numman boundary conditions. We combine a fourth-order compact finite difference scheme to discretise the spatial derivative and diagonally implicit Runge Kutta method to the time integration, and this leads to linear system of ordinary differential equations. To address the difficulty of this, work, by using modified Newton-Raphson methods to deal with nonlinear term. Numerical experiments through Matlab programming also confirm that the proposed method are reliable and efficient for solving generalized Burgers-Fisher. This approach may be extended to tackle stochastic equations with growth model [20, 21]. Another interesting of this work is to use discontinuous Galerkin methods for estimating this type of problem in terms of $L^2(H^1)$ and $L^\infty(H^1)$ [22 – 25].

References
[1] Younis A. Sabawi, “A Posteriori $L_\infty(H^1)$- Error Bound in Finite Element Approximation of Semilinear Parabolic Problems, ”2019 First International Conference of Computer and Applied Science (CAS), Baghdad, Iraq. 2019, pp. 102-106, doi: 10.1109/CAS47993.2019.9075699.
[2] Sabawi YA. A Posteriori Error Analysis in Finite Element Approximation for Fully Discrete Semilinear Parabolic Problems. InFinite Element Methods and Their Applications 2020 Dec 10. IntechOpen
[3] Fu Y. Compact fourth-order finite difference schemes for Helmholtz equation with high wave numbers. Journal of Computational Mathematics. 2008 Jan 1:98-11
[4] Manaa SA, Moheemmeed MA, Hussien YA. A Numerical Solution for Sine-Gordon Type System. Tikrit Journal of PureScience. 2010;15(3):106-13.
[5] 2N order compact finite difference scheme with collocation method for solving the generalized Burger’s–Huxley and Burger’s–Fisher equations
[6] Yang X, Ge Y, Zhang L. A A class of high-order compact difference schemes for solving the
Burgers’ equations. Applied Mathematics and Computation. 2019 Oct 1;358:394-417.

[7] Boscarino S, Filbet F, Russo G. High order semi-implicit schemes for time dependent partial differential equations. Journal of Scientific Computing. 2016 Sep 1;68(3):975-1001.

[8] Bhatt H, Chowdhury A. A compact fourth-order implicit-explicit Runge-Kutta type scheme for numerical solution of the Kuramoto-Sivashinsky equation. arXiv preprint arXiv:1911.12183. 2019 Nov 27.

[9] Hammad DA, El-Azab MS. 2N order compact finite difference scheme with collocation method for solving the generalized Burger’s–Huxley and Burger’s–Fisher equations. Applied Mathematics and Computation. 2015 May 1;258:296-311.

[10] Hussien YA. Combination Between Single Diagonal Implicit and Explicit Runge Kutta (SDIMEX-RK) Methods for solving stiff Differential equations. Tikrit Journal of Pure Science. 2011;16(1):94-101.

[11] Pareschi L, Russo G. Implicit–explicit Runge–Kutta schemes and applications to hyperbolic systems with relaxation. Journal of Scientific computing. 2005 Oct 1;25(1):129-55.

[12] Ascher UM, Ruuth SJ, Spiteri RJ. Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations. Applied Numerical Mathematics. 1997 Nov 1;25(2-3):151-67.

[13] Pao CV, Wang YM. Numerical solutions of a three-competition Lotka–Volterra system. Applied Mathematics and Computation. 2008 Oct 1;204(1):423-40.

[14] Owolabi KM, Patidar KC. Higher-order time-stepping methods for time-dependent reaction–diffusion equations arising in biology. Applied Mathematics and Computation. 2014 Aug 1;240:30-50.

[15] Ascher UM, Ruuth SJ, Spiteri RJ. Implicit–explicit Runge–Kutta methods for time-dependent partial differential equations. Applied Numerical Mathematics. 1997 Nov 1;25(2-3):151-67.

[16] Hundsdoerfer W, Ruuth SJ. IMEX extensions of linear multistep methods with general monotonicity and boundedness properties. Journal of Computational Physics. 2007 Aug 10;225(2):2016-42.

[17] Cao HH, Liu LB, Zhang Y, Fu SM. A fourth-order method of the convection–diffusion equations with Neumann boundaryconditions. Applied Mathematics and Computation. 2011 Jul 15;217(22):9133-41.

[18] Fu Y, Tian Z, Liu Y. A Compact Exponential Scheme for Solving 1D Unsteady Convection-Diffusion Equation with Neumann Boundary Conditions. arXiv preprint arXiv:1805.05728. 2018 May 15.

[19] Yao C, Zhang Y, Chen J, Ling X, Jing K, Lu Y, Fan E. Development of a fourth-order compact finite difference scheme for simulation of simulated-moving-bed process. Scientific reports. 2020 May 8;10(1):1-3.

[20] Tesfay A, Tesfay D, Khalaf A, Brannan J. Mean exit time and escape probability for the stochastic logistic growth model with multiplicative α-stable Lévy noise. Stochastics and Dynamics. 2020 Aug 28:2150016.

[21] Khalaf AD, Abouagwa M, Mustafa, Wang X. Stochastic Volterra integral equations with jumps and the strong superconvergence of the Euler–Maruyama approximation. Journal of Computational and Applied Mathematics. 2021 Jan;382:113071.

[22] Younis A, Sabawi, Adaptive discontinuous Galerkin methods for interface problems, PhD Thesis, University of Leicester, Leicester, UK (2017).

[23] Cangiani, Andrea, Emmanuil H. Georgoulis, and Younis A. Sabawi Adaptive discontinuous Galerkin methods for elliptic interface problems, Math. Comp. 87 (2018), no. 314, 2675–2707.

[24] Andrea Cangiani, Emmanuil H. Georgoulis, and Younis A. Sabawi, Convergence of an adaptive discontinuous Galerkin method for elliptic interface problems, Journal of Computational and Applied Mathematics 367 (2020):112397.

[25] Sabawi YA. A Posteriori $L^p(L^2) + L^2(H^1)$–Error Bounds in Discontinuous Galerkin Methods For Semidiscrete Semilinear Parabolic Interface Problems. Baghdad Science Journal 2021;18(3):0522–.