Eurasian J. Sci. Eng., 2025, 11(3), 199-214
https://doi.org/10.23918/eajse.v11i3p12

Published by Tishk International University
Available at: https://eajse.tiu.edu.iq/

RESEARCH ARTICLE Open Access

# Numerical Solution of Burgers' Equation Using the Explicit and Crank-Nicolson Finite Difference Methods

Younis A. Sabawi[1*,3], Mohammed I. Sadeeq[2], and Mardan A. Pirdawood[1]

[1] Department of Mathematics, Faculty of Science and Health, Koya University, Koya 44023, Kurdistan Region - F.R. Iraq.
[2] Department of Mathematics, College of Education, Akre University for Applied Sciences, Akre, Kurdistan Region, Iraq.
[3] College of Information Technology, Imam Ja'afar AL-Sadiq University, Baghdad, Iraq.

**Article History**
*Received: 21.03.2025*
*Revised: 31.08.2025*
*Accepted: 06.12.2025*
*Published: 03.02.2026*
*Communicated by: Prof. Dr. S. A. Mohiuddine*

**Email address:*
*sabawiyounis79@gmail.com*
**Corresponding Author*

**Abstract**: The Burgers' equation is numerically studied in this paper by using the explicit and Crank-Nicolson finite difference methods. Finite difference approximations are used in place of the derivatives, resulting in a system of equations that, when solved, approximate the solutions to the Burgers' equation. Additionally, the stability by Von Neumann method and convergence analysis are discussed for both methods. Furthermore, a comparison is done between the exact solutions and the approximate solutions for two test problems. We have shown that there is a high degree of agreement between the approximate and exact solutions. It was also determined that the explicit method exhibits conditional stability, whereas the Crank-Nicolson method remains stable under all conditions.

**Keywords**: *Burgers' equation; Finite difference method; Explicit method; Crank-Nicolson method; Von Neumann stability; Convergence analysis.*

## 1. Introduction

Partial differential equations (PDEs), or systems of PDEs, provide a quantitative explanation of many major models in the social, biological, and physical sciences. The relationship between the partial derivatives with respect to independent variables and their unknown functions are both well-explained in the description. Even if the basic differential connections seem straightforward, nonlinear motion, reaction, diffusion, equilibrium, and many other complicated phenomena and mathematical problems are all governed by PDEs [1, 2, 3].

Approximating derivatives with finite differences is the numerical method known as the finite difference method (FDM) for solving differential equations. It is particularly common in solving PDEs and ordinary differential equations (ODEs). The basic idea is to replace derivatives in the differential equations with finite difference approximations, which allows the problem to be discretized and solved on a computer. The FDM is often used when analytical solutions are difficult or impossible to obtain, especially for complex or nonlinear problems. The FDM provides a solution only at specific discrete points within the domain of interest, rather than delivering a general formula or closed-form solution applicable to all points in the domain, as is typically expected in an analytical approach [1, 2, 4, 5, 6, 7, 8].

The explicit method is one of the FDMs, this method is explicit because the solutions are computed explicitly in terms of the present values, without the need to solve a system of equations, while the method is conditionally stable. The Crank-Nicolson method is unconditionally stable and more accurate. The stability property makes it a popular choice for solving time-dependent problems, especially when the accuracy and stability are crucial. However, it is computationally more expensive than explicit methods due to the need to solve a system of equations at each time step [9, 10, 11, 12].

While the Finite Difference Method (FDM) provides a more general solution, it comes with challenges related to stability and convergence. Additionally, it may require careful handling of boundary conditions, significant computational storage, and extended execution time. Addressing numerical dispersion in finite difference solutions is also a complex task. When a differential equation is replaced with a finite difference equation, errors arise due to the truncation of series approximating the derivatives. These errors can accumulate, and understanding the conditions under which they diminish from one time step to the next helps determine numerical stability. A common approach for assessing stability is the Von Neumann analysis, which is one of the most widely used methods for evaluating the stability or instability of finite difference approximations, as will be discussed later [13, 14, 15].

In the context of fluid dynamics and nonlinear waves, the phrase "Burgers' equation" usually refers to a PDEs. The Burgers' equation can be expressed in one dimension as follows [16]:

$$\begin{gathered} u_t + uu_x = \alpha u_{xx}\,; \quad a \le x \le b,\ \ t > 0, \\ u(x,0) = g(x),\ \ u(a,t) = f_1(t),\ \ u(b,t) = f_2(t), \end{gathered} \tag{1}$$

where $u = u(x,t)$ and $\alpha > 0$ is a constant of diffusivity.

Many methods have been proposed to solve the Burgers' equation for the various boundary and initial conditions. Dag et al. [17] obtained a finite element solution to Burgers' equation using both quadratic and cubic B-spline functions. Fan and Li [18] solved the two-dimensional Burgers' equations using a meshfree numerical scheme, which is a combination of the implicit Euler method, the generalized finite difference method and the fictitious time integration method (FTIM). Also, the one-dimensional Burgers' was solved by Liao and Zhu [19] using two efficient fourth-order compact finite difference algorithms, the methods are based on the HopfCole transformation, Richardson's extrapolation, and multilevel grids. They first transformed the original nonlinear Burgers' equation into a linear heat equation, the resulted heat equation was solved by the second-order accurate Crank-Nicholson algorithm. Fletcher [20] presented a comparison of finite difference and finite element approaches for solving the one- and two-dimensional Burgers' equations.

The main purpose for this work is the reformulation of FDMs for the Burgers' equation. The paper is structured in the following manner: In Section 2. and 3., the Burgers' equation is solved by using the explicit and Crank-Nicolson methods, respectively. A stability analysis by Von Neumann method and convergence analysis are included in Sections 4. and 5.. The numerical results are contained in Section 6.. In Section 7., we discussed some conclusions.

## 2. Explicit method

The discrete approximation to the exact solution $u(x_i,t_n)$ at the grid point $(x_i,t_n)$ is first represented by the symbol $U_i^n$. The forward-time difference approximation is then used to estimate the time derivative, and the central-space difference approximation is used to approximate the space derivatives [16, 21, 22]. Discreteizing the Burgers' equation (1) by using explicit finite difference approximations yields the following results:

$$\frac{U_i^{n+1}-U_i^n}{k}+U_i^n\frac{U_{i+1}^n-U_{i-1}^n}{2h}=\alpha\frac{U_{i+1}^n-2U_i^n+U_{i-1}^n}{h^2}. \tag{2}$$

When we simplify (2), we obtain

$$U_i^{n+1}=U_i^n-\mu_1U_i^n\left(U_{i+1}^n-U_{i-1}^n\right)+\alpha\mu_2\left(U_{i+1}^n-2U_i^n+U_{i-1}^n\right), \tag{3}$$

where

$$\mu_1=\frac{k}{2h},\ \ \mu_2=\frac{k}{h^2}.$$

Consequently, the computation of $U_i^n$, where $i$ takes values from 1 to $N$, can be directly derived from the information at the preceding time step for each $n\geq 0$. The values along the side boundaries and the lower boundary (initial values) are determined by specified boundary conditions and the initial condition, respectively. This computational approach is specifically termed the explicit finite difference scheme applicable to the Burgers' equation (1).

## 3. Crank-Nicolson method

The Crank-Nicolson method integrates both implicit and explicit time-stepping approaches to enhance accuracy and stability beyond that of purely explicit methods. However, it necessitates advanced algorithms to solve a system of equations at each time step. Notably, this method is unconditionally stable, making it a preferred choice for solving mathematical problems. Additionally, it offers second-order accuracy in both time and space [14, 23, 24, 25].

Using the Crank-Nicolson approach to discretize the Burgers' equation (1), the result is

$$\begin{aligned}\frac{U_i^{n+1}-U_i^n}{k}&+\frac{1}{4h}\left(U_i^n\left(U_{i+1}^{n+1}-U_{i-1}^{n+1}\right)+U_i^{n+1}\left(U_{i+1}^n-U_{i-1}^n\right)\right)\\&=\frac{\alpha}{2h^2}\left(U_{i+1}^{n+1}-2U_i^{n+1}+U_{i-1}^{n+1}+U_{i+1}^n-2U_i^n+U_{i-1}^n\right).\end{aligned}$$

Rearranging the above equation, yields

$$A_i^nU_{i+1}^{n+1}+B_i^nU_i^{n+1}-C_i^nU_{i-1}^{n+1}=\alpha r_2U_{i+1}^n+r_3U_i^n+\alpha r_2U_{i-1}^n, \tag{4}$$

where, $r_1=k/4h$, $r_2=k/2h^2$, $A_i^n=r_1U_i^n-\alpha r_2$, $B_i^n=1+r_1U_{i+1}^n-r_1U_{i-1}^n+2\alpha r_2$, $C_i^n=r_1U_i^n+\alpha r_2$, and $r_3=1-2\alpha r_2$. This method requires the solution of simultaneous equations in order to calculate the value of $U_i^n$. The left side of (4) contains unknown and the right side known values. If there are $N$ internal mesh points along each time row, then for $n=0$ and $i=1,\ldots,N$, the difference scheme (4) gives $N$ simultaneous equations for $N$ unknown pivotal values $\left(U_1^{n+1},U_2^{n+1},\ldots,U_N^{n+1}\right)$ along the first time row in terms of known initial and boundary values. Similarly, $n=1$ expresses $N$ unknown pivotal values along the second time row regarding the calculated values along the first time row, etc. Therefore, equation (4) can be written as a multi-diagonal linear system as follows:

$$\mathbb{D}\mathbb{U}^{n+1}=\mathbb{E}\mathbb{U}^n+\mathbb{B}, \tag{5}$$

where

$$\mathbb{D}=\begin{bmatrix} B_1^n & A_1^n & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -C_2^n & B_2^n & A_2^n & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -C_3^n & B_3^n & A_3^n & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & -C_4^n & B_4^n & A_4^n & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & -C_{n-1}^n & B_{n-1}^n & A_{n-1}^n \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & -C_n^n & B_n^n \end{bmatrix},$$

$$\mathbb{E}=\begin{bmatrix} r_3 & \alpha r_2 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \alpha r_2 & r_3 & \alpha r_2 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & \alpha r_2 & r_3 & \alpha r_2 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \alpha r_2 & r_3 & \alpha r_2 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & \alpha r_2 & r_3 & \alpha r_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & \alpha r_2 & r_3 \end{bmatrix}$$

$$\mathbb{U}^{n+1}=\begin{bmatrix} U_1^{n+1} \\ U_2^{n+1} \\ U_3^{n+1} \\ \vdots \\ U_{N-1}^{n+1} \\ U_N^{n+1} \end{bmatrix}, \quad \mathbb{U}^{n}=\begin{bmatrix} U_1^{n} \\ U_2^{n} \\ U_3^{n} \\ \vdots \\ U_{N-1}^{n} \\ U_N^{n} \end{bmatrix}, \quad B=\begin{bmatrix} \alpha r_2 f_1(t_n)+C_1^n f_1(t_{n+1}) \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \alpha r_2 f_2(t_n)-A_n^n f_2(t_{n+1}) \end{bmatrix}.$$

In order to obtain the numerical solution at the grid points $(x_i,t_n)$, along with initial and boundary conditions, we must finally solve the linear system (5) for each $n \geq 0$, consecutively using any appropriate method.

## 4. Stability analysis

Stability analysis is an important aspect of numerical methods, including the finite difference methods. The stability of a numerical scheme is related to its ability to produce accurate and reliable results over time. For finite difference scheme, stability is often analyzed by using the concept of the Von Neumann method. The Von Neumann stability approach has been developed for linear, one-dimensional, and multi-dimensional problems. It is based on a Fourier analysis in the space domain. The amplification factor can be easily achieved with this technique, which is the most generally used one. It is important to remember that this approach is limited to initial value problems using periodic initial data and only applies to linear equations. This method's general idea involves substituting the amplification factor for the numerical solution [12, 13, 22, 26]. In other words, this method uses a Fourier series to express $u_{i,j}$ in the following sense:

$$U_i^n=\lambda^n e^{l\beta ih}, \tag{6}$$

where $l$ denotes the complex number, $l=\sqrt{-1}$. The finite difference equation will be stable if $|U_i^n|$ remains bounded for all $n$ as $h,k \to 0$ and for all values of $\beta$ (where $\beta$ is real number) needed to satisfy the initial conditions. When there isn't an exponential rise in the exact solution of the difference equations over time, a necessary and sufficient condition for stability is given by

$$|\lambda| \leq 1.$$

To examine the stability of the explicit scheme (2) by Von Neumann technique, the nonlinear term $uu_x$ in the Burgers' equation has been linearized by treating the expression $\hat{u}u_x$ a local constant. Then, using (6) into (3), we get

$$\begin{aligned}\lambda^{n+1}e^{l\beta ih} = \lambda^{n}e^{l\beta ih} - \mu_1\hat{u}(\lambda^{n}e^{l\beta(i+1)h} - \lambda^{n}e^{l\beta(i-1)h}) \\ + \alpha\mu_2(\lambda^{n}e^{l\beta(i+1)h} - 2\lambda^{n}e^{l\beta ih} + \lambda^{n}e^{l\beta(i-1)h}).\end{aligned}$$

Dividing both sides of the above equation by the term $\lambda^n e^{l\beta ih}$, gives the following equation:

$$\lambda = 1 - \mu_1\hat{u}(e^{l\beta h} - e^{-l\beta h}) + \alpha\mu_2(e^{l\beta h} - 2 + e^{-l\beta h}).$$

Using the Euler's formula

$$e^{l\theta} = \cos\theta + l\sin\theta,$$

and after that equating the real and imaginary components, we obtain

$$\lambda = 1 - 4\alpha\mu_2\sin^2(\frac{\beta h}{2}).$$

We have derived an expression for $\lambda$. We now merely need to see what requirements are involved such that $|\lambda| \leq 1$, so, we need to consider the conditions:

$$-1 \leq 1 - 4\alpha\mu_2\sin^2(\frac{\beta h}{2}) \leq 1,$$

the upper bound 1 is achieved automatically, since $\alpha, \mu_2 > 0$, so, we need to

$$-1 \leq 1 - 4\alpha\mu_2\sin^2(\frac{\beta h}{2}),$$

or equivalently,

$$\mu_2 \leq \frac{1}{2\alpha\sin^2(\frac{\beta h}{2})},$$

noting the behavior of $\sin^2(\frac{\beta h}{2})$ for any choice of $\beta h$, we see that in the worst instance [i.e. $\sin^2(\frac{\beta h}{2}) = 1$], we have

$$\mu_2 \leq \frac{1}{2\alpha}.$$

The explicit scheme (3) is conditionally stable subject to a stability bound of

$$0 < \mu_2 \leq \frac{1}{2\alpha}.$$

In a similar manner, when examining the stability of the Crank-Nicolson approach, we arrive at the following equation:

$$\begin{aligned}\lambda - 1 + \mu_1\lambda\hat{u}(e^{l\beta h} - e^{-l\beta h}) + \mu_1\hat{u}(e^{l\beta h} - e^{-l\beta h}) \\ = \alpha\mu_2\lambda(e^{l\beta h} - 2 + e^{-l\beta h}) + \alpha\mu_2(e^{l\beta h} - 2 + e^{-l\beta h}),\end{aligned}$$

this implies that

$$\lambda = \frac{1-4\alpha r_2 \sin^2(\frac{\beta h}{2}) - 2lr_1\hat{u}\sin(\beta h)}{1+4\alpha r_2 \sin^2(\frac{\beta h}{2}) + 2lr_1\hat{u}\sin(\beta h)}. \tag{7}$$

It is easy to show that $|\lambda| \leq 1$, by taking the modulus of (7), so that the Crank-Nicolson method is unconditionally stable.

## 5. Convergence analysis

This section examines the accuracy of the present numerical schemes for Burgers' equation (1), employing a Taylor series expansion together with the concept of truncation error. Suppose that $u$ is twice continuously differentiable in time and four times continuously differentiable in space, i.e.,

$$u \in C^2([0,T];L^2(\Omega)) \cap C^4(\Omega).$$

The explicit finite difference scheme reads

$$\frac{u_i^{n+1}-u_i^n}{k} + u_i^n \frac{u_{i+1}^n - u_{i-1}^n}{2h} = \alpha \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{h^2},\ i=1,\dots,N,\ n \geq 0. \tag{8}$$

The local truncation error $\tau_i^n$ of (8) is defined as:

$$\begin{aligned}\tau_i^n = {} & \frac{u(x_i,t_{n+1})-u(x_i,t_n)}{k} + u(x_i,t_n)\frac{u(x_{i+1},t_n)-u(x_{i-1},t_n)}{2h} \\ & - \alpha\frac{u(x_{i+1},t_n)-2u(x_i,t_n)+u(x_{i-1},t_n)}{h^2} - u_t(x_i,t_n) - u(x_i,t_n)u_x(x_i,t_n) + \alpha u_{xx}(x_i,t_n).\end{aligned}$$

**Lemma 1.** *Let $u$ be a sufficiently smooth with $u \in C_t^2 \cap C_x^4$. Then the local truncation error of scheme* (8) *satisfies*

$$\tau_i^n = O(k) + O(h^2).$$

**Proof.** Using a Taylor expansion in time, we find that

$$\frac{u(x_i,t_{n+1})-u(x_i,t_n)}{k} - u_t(x_i,t_n) = \frac{k}{2}u_{tt}(x_i,\xi_t), \quad \xi_t \in (t_n,t_{n+1}),$$

so that the time discretization error is $O(k)$. Similarly, expanding in space,

$$\frac{u(x_{i+1},t_n)-u(x_{i-1},t_n)}{2h} - u_x(x_i,t_n) = \frac{h^2}{6}u_{xxx}(x_j,\xi_1),$$

$$\frac{u(x_{i+1},t_n)-2u(x_i,t_n)+u(x_{i-1},t_n)}{h^2} - u_{xx}(x_i,t_n) = \frac{h^2}{12}u_{xxxx}(x_i,\xi_2),$$

for some $\xi_1,\xi_2 \in (x_{i-1},x_{i+1})$. The spatial discretization error is of order $O(h^2)$. By combining these estimates, we obtain the desired result. Hence, the scheme (8) is first-order accuracy in time and second-order accuracy in space, provided the solution is smooth. □

**Theorem 1** (Convergence of the explicit scheme)**.** *Consider the exact solution $u$ of the Burgers' equation (1) with $u \in C^2([0,T];C^4(\bar{\Omega}))$. Let $U_i^n$ denote the approximate solution given by the explicit finite difference scheme (2) with $U_i^0 = u(x_i,0)$ and appropriate boundary conditions. Then, there exists a constant $C > 0$, independent of $h$ and $k$, such that for all $n$ with $t_n = nk \leq T$,*

$$\|U^n - u^n\|_{\ell^2} \leq C\left(k+h^2\right).$$

*So that the scheme converges with global error of order $O(k+h^2)$ in the discrete $L^2$-norm.*

**Proof.** Let $e_i^n = U_i^n - u_i^n$ denote the error, and let

$$\langle v, w\rangle_h := h\sum_i v_i w_i, \qquad \|v\|_h^2 := \langle v, v\rangle_h.$$

Subtracting the explicit scheme associated with the exact solution, including the truncation error $\tau_i^n$, from the numerical one yields

$$\frac{e_i^{n+1}-e_i^n}{k} + \left(U_i^n D_0 U_i^n - u_i^n D_0 u_i^n\right) = \alpha\Delta_h e_i^n - \tau_i^n,$$

where $D_0 v_i := \frac{v_{i+1}-v_{i-1}}{2h}$ and $\Delta_h v_i := \frac{v_{i+1}-2v_i+v_{i-1}}{h^2}$. Expanding the nonlinear term, as follows:

$$U_i^n D_0 U_i^n - u_i^n D_0 u_i^n = e_i^n D_0 U_i^n + u_i^n D_0 e_i^n.$$

Multiplying the error equation by $e_i^{n+1}$, summing in $i$, and using discrete integration by parts, one obtains

$$\begin{aligned}\frac{1}{2k}\left(\|e^{n+1}\|_h^2 - \|e^n\|_h^2\right) \ \le\ & C\|e^n\|_h^2 + C\|e^{n+1}\|_h^2 - \alpha\|D_+ e^n\|_h^2 \\ & + \langle \tau^n, e^{n+1}\rangle_h,\end{aligned}$$

where $D_+ v_i := (v_{i+1}-v_i)/h$. The truncation error satisfies

$$\tau_i^n = O(k) + O(h^2),$$

hence

$$|\langle \tau^n, e^{n+1}\rangle_h| \le \tfrac{1}{4k}\|e^{n+1}\|_h^2 + Ck\big(k^2+h^4\big).$$

Collecting terms, we obtain a recursive inequality of the form

$$\|e^{n+1}\|_h^2 \ \le\ (1+Ck)\,\|e^n\|_h^2 + Ck\big(k^2+h^4\big).$$

Using the discrete Grönwall inequality, together with the fact that $\|e^0\|_h = O(h^2)$ (since the initial data is projected exactly or with second-order accuracy), we conclude that

$$\|e^n\|_h \ \le\ C\big(k+h^2\big), \qquad 0 \le t_n \le T.$$

This proves convergence of order $O(k+h^2)$. □

In the same way, way we can show that the convergence of the Crank-Nicolson scheme (4) is second order in both $k$ and $h$, that is

$$\|U^n - u^n\|_{\ell^2} \ \le\ C\left(k^2+h^2\right).$$

## 6. Numerical example and results

In this section, we applied the present methods to calculate the numerical solutions of the Burgers' equation given by (1). The numerical results are compared to the exact solution. All computations were performed using MATLAB R2023a.

**Problem 1**: Suppose that the exact solution of the Burgers' equation is given in the reference [27], as follows:

$$u(x,t) = \frac{2x}{1+2t}. \tag{9}$$

The initial condition at $t = 0$ is given by

$$u(x,0) = 2x,$$

and the boundary conditions are

$$u(a,t) = \frac{2a}{1+2t}, \;\; u(b,t) = \frac{2b}{1+2t}.$$

The precision of the methods is evaluated by calculating the absolute error $L_{abs}$, which is defined by

$$L_{abs}(u) = |u^n - U^n|.$$

Although we have considered only the special case when $\alpha = 1$, with the spatial domain $x$-domain $0 \leq x \leq 5$ and the temporal domain $t$-domain is $0 \leq t \leq 1$, using $h = 0.25$ and $k = 0.02$. By computing the $L_{abs}$ error, as presented in Tables 1 and 2, and examining Figures 1-9, it becomes clear that the numerical results obtained using the proposed methods closely match the exact solutions. Additionally, the Crank-Nicolson method yielded superior results and demonstrated greater effectiveness compared to the explicit method. While at each time step, the Crank-Nicolson approach requires solving a set of equations, the explicit method directly calculates the solutions.

**Problem 2**: Assume that the Burgers' equation has an exact solution the form [17]

$$u(x,t) = \frac{x/t}{1 + (t/t_0)^{\frac{1}{2}} \exp(x^2/4\alpha t)}, \; t \geq 1, \; 0 \leq x \leq 1, \tag{10}$$

where $t_0 = \exp(\frac{1}{8\alpha})$. This solution represent the propagation of the shocks which becomes slightly smoother as time progresses. The maximum absolute error $L_\infty$ is used to measure the versatility and accuracy of the method for this problem. We use the following formula [17]:

$$L_\infty = \|u^n - U^n\|_\infty = \max_{0 \leq j \leq N} \left| u^n - U^n \right|.$$

Initial condition which is taken when $t = 1$ in (10) is used, and the boundary conditions are taken as $u(0,t) = u(1,t) = 0$. Parameters $h = k = 0.01$ and different values of $\alpha$ are selected. Computation is done up to time $t = 3$ over the problem domain $[0,1]$. Tables 3 and 4 show the error norm $L_\infty$ for explicit and Crank-Nicolson methods, respectively, for various time levels and different values of $\alpha$. The tables demonstrate that the proposed approach provides results in close agreement with the exact solutions. It is evident from the tables that a decrease in the parameter $\alpha$ leads to an improvement in the accuracy of both methods.

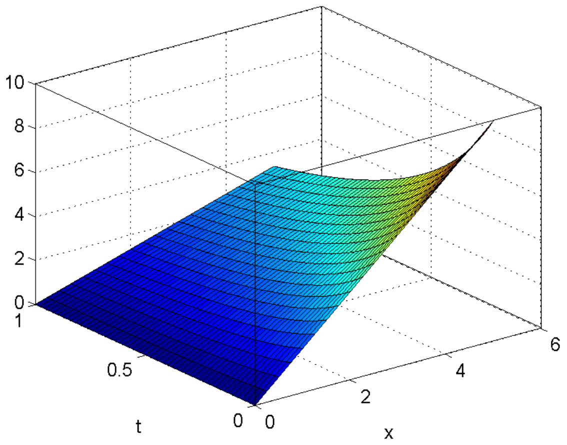

Figure 1: Exact solution for the Burgers' equation (1) in Problem (1), when $0 \leq x \leq 5$, $0 \leq t \leq 1$, $h = 0.25$, $k = 0.02$, and $\alpha = 1$.

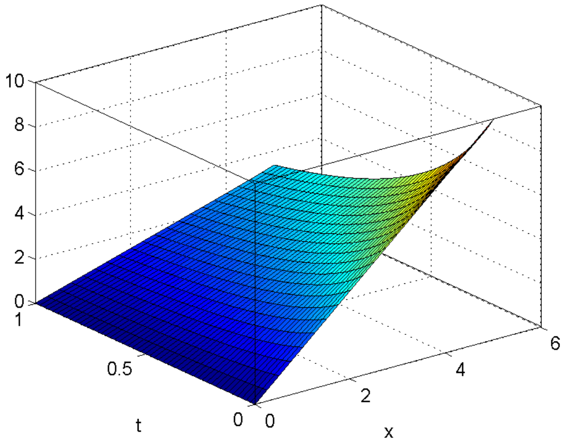

Figure 2: Approximate solution for the Burgers' equation (1) obtained by the explicit method in Problem (1), when $0 \leq x \leq 5$, $0 \leq t \leq 1$, $h = 0.25$, $k = 0.02$, and $\alpha = 1$.

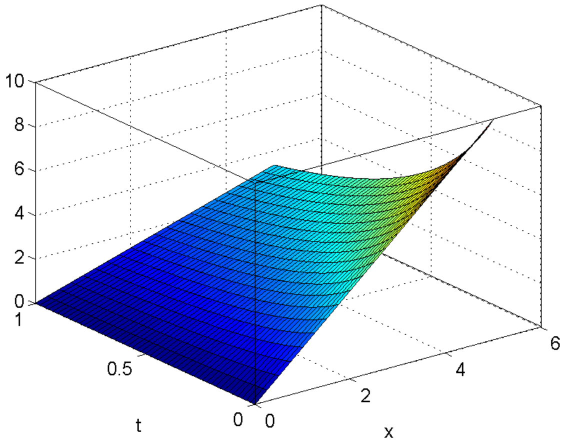

Figure 3: Approximate solution for the Burgers' equation (1) obtained by the Crank-Nicolson method in Problem (1), when $0 \leq x \leq 5$, $0 \leq t \leq 1$, $h = 0.25$, $k = 0.02$, and $\alpha = 1$.

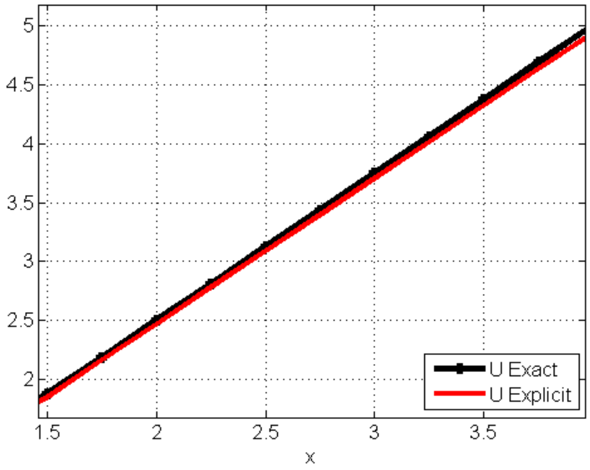

Figure 4: $u(x,0.3)$: Exact with explicit for Problem (1).

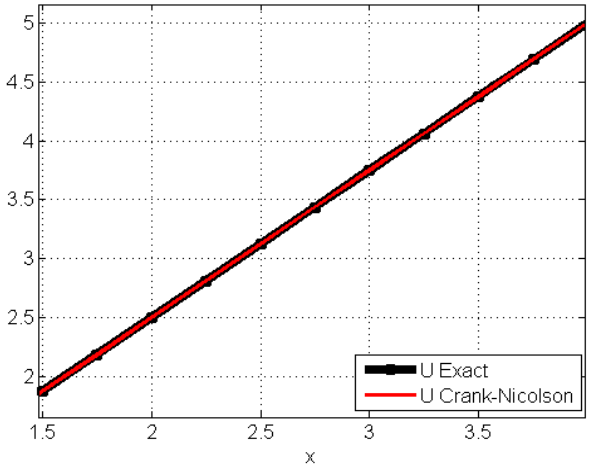

Figure 5: $u(x,0.3)$: Exact with Crank-Nicolson for Problem (1).

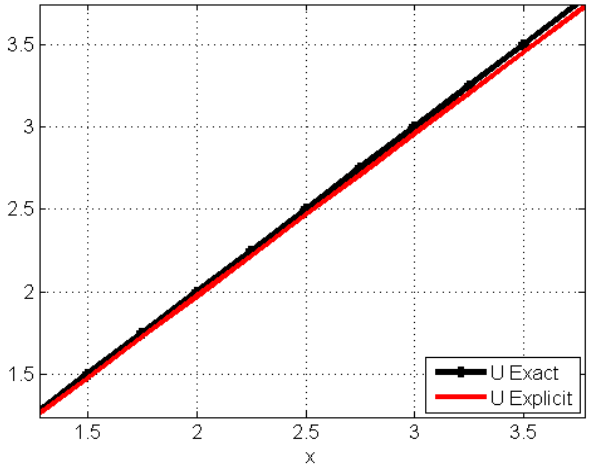

Figure 6: $u(x,0.5)$: Exact with explicit for Problem (1).

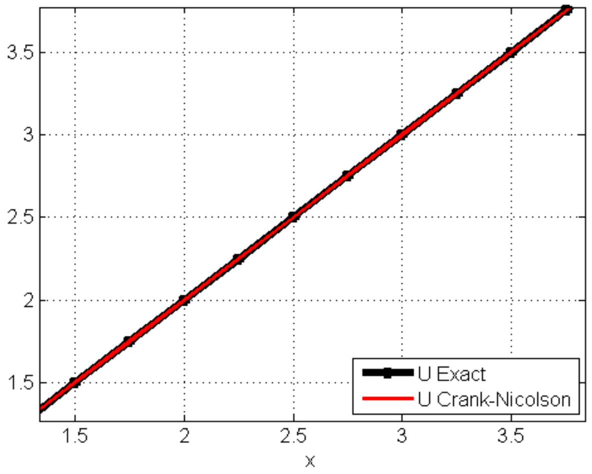

Figure 7: $u(x,0.5)$: Exact with Crank-Nicolson for Problem (1).

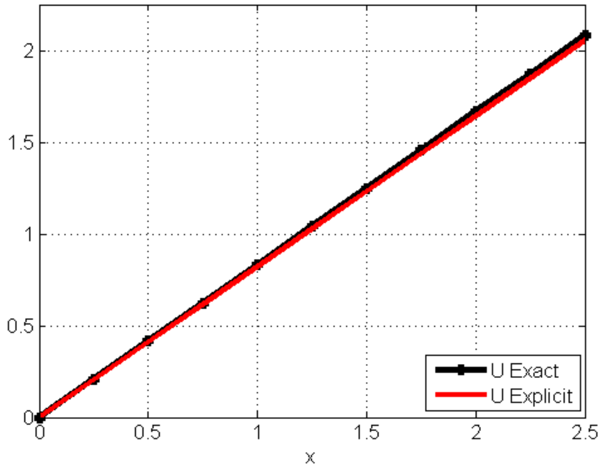

Figure 8: $u(x,0.7)$: Exact with explicit for Problem (1).

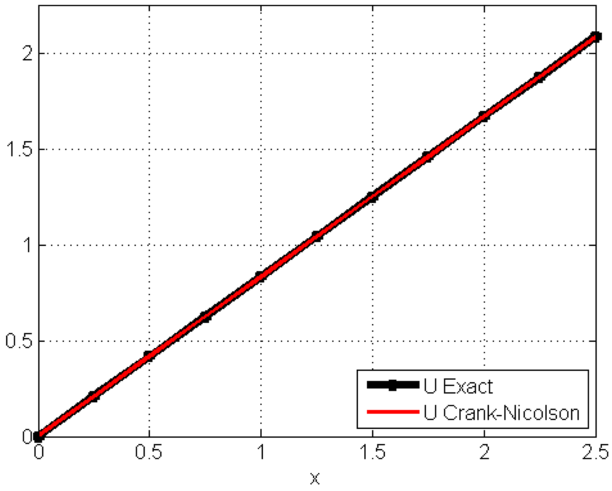

Figure 9: $u(x,0.7)$: Exact with Crank-Nicolson for Problem (1).

Table 1: $L_{abs}$ errors for the results obtained by explicit method in Problem (1), when $0 \leq x \leq 5$ and for a fixed time $t$, as in the table, with $h = 0.25$ and $k = 0.02$.

| $x$ | $t = 0.1$ | $t = 0.3$ | $t = 0.5$ | $t = 0.7$ | $t = 0.9$ |
|---|---|---|---|---|---|
| 0.0000 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 |
| 0.2500 | 2.6542328E-03 | 3.7831998E-03 | 3.5390382E-03 | 3.0877737E-03 | 2.6589814E-03 |
| 0.5000 | 5.3084656E-03 | 7.5663996E-03 | 7.0780763E-03 | 6.1755473E-03 | 5.3179626E-03 |
| 0.7500 | 7.9626984E-03 | 1.1349599E-02 | 1.0617114E-02 | 9.2633209E-03 | 7.9769427E-03 |
| 1.0000 | 1.0616931E-02 | 1.5132799E-02 | 1.4156153E-02 | 1.2351094E-02 | 1.0635920E-02 |
| 1.2500 | 1.3271164E-02 | 1.8915999E-02 | 1.7695191E-02 | 1.5438868E-02 | 1.3294889E-02 |
| 1.5000 | 1.5925397E-02 | 2.2699199E-02 | 2.1234229E-02 | 1.8526639E-02 | 1.5953834E-02 |
| 1.7500 | 1.8579630E-02 | 2.6482398E-02 | 2.4773267E-02 | 2.1614406E-02 | 1.8612714E-02 |
| 2.0000 | 2.1233862E-02 | 3.0265598E-02 | 2.8312305E-02 | 2.4702152E-02 | 2.1271410E-02 |
| 2.2500 | 2.3888095E-02 | 3.4048798E-02 | 3.1851341E-02 | 2.7789830E-02 | 2.3929607E-02 |
| 2.5000 | 2.6542328E-02 | 3.7831998E-02 | 3.5390368E-02 | 3.0877270E-02 | 2.6586452E-02 |
| 2.7500 | 2.9196561E-02 | 4.1615198E-02 | 3.8929344E-02 | 3.3963907E-02 | 2.9239675E-02 |
| 3.0000 | 3.1850794E-02 | 4.5398397E-02 | 4.2468067E-02 | 3.7047864E-02 | 3.1883271E-02 |
| 3.2500 | 3.4505026E-02 | 4.9181589E-02 | 4.6005568E-02 | 4.0123022E-02 | 3.4501469E-02 |
| 3.5000 | 3.7159259E-02 | 5.2964683E-02 | 4.9537335E-02 | 4.3169623E-02 | 3.7053096E-02 |
| 3.7500 | 3.9813492E-02 | 5.6746668E-02 | 5.3042914E-02 | 4.6124504E-02 | 3.9431219E-02 |
| 4.0000 | 4.2467718E-02 | 6.0517651E-02 | 5.6431352E-02 | 4.8787330E-02 | 4.1359264E-02 |
| 4.2500 | 4.5121828E-02 | 6.4188531E-02 | 5.9304327E-02 | 5.0526531E-02 | 4.2124189E-02 |
| 4.5000 | 4.7786760E-02 | 6.6999689E-02 | 5.9936142E-02 | 4.9359044E-02 | 3.9890861E-02 |
| 4.7500 | 5.1029887E-02 | 6.2684292E-02 | 5.0891480E-02 | 3.9063261E-02 | 2.9932638E-02 |
| 5.0000 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 |
| CPU time | 4.345 sec | 4.354 sec | 4.362 sec | 4.382 sec | 4.413 sec |

Table 2: $L_{abs}$ errors for the results obtained by Crank-Nicolson method in Problem (1), when $0 \leq x \leq 5$ and for a fixed time $t$, as in the table, with $h = 0.25$ and $k = 0.02$.

| $x$ | $t = 0.1$ | $t = 0.3$ | $t = 0.5$ | $t = 0.7$ | $t = 0.9$ |
|---|---|---|---|---|---|
| 0.0000 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 |
| 0.2500 | 2.2204460E-16 | 3.8857806E-16 | 5.8286709E-16 | 6.1062266E-16 | 6.3837824E-16 |
| 0.5000 | 4.4408921E-16 | 7.7715612E-16 | 1.2767565E-15 | 1.2212453E-15 | 1.2767565E-15 |
| 0.7500 | 8.8817842E-16 | 1.1102230E-15 | 1.6653345E-15 | 1.8873791E-15 | 1.7763568E-15 |
| 1.0000 | 1.9984014E-15 | 1.9984014E-15 | 1.9984014E-15 | 2.3314684E-15 | 2.3314684E-15 |
| 1.2500 | 1.7763568E-15 | 2.4424907E-15 | 2.6645353E-15 | 3.3306691E-15 | 3.2196468E-15 |
| 1.5000 | 8.8817842E-16 | 3.1086245E-15 | 3.1086245E-15 | 3.5527137E-15 | 3.7747583E-15 |
| 1.7500 | 1.3322676E-15 | 3.5527137E-15 | 3.5527137E-15 | 3.9968029E-15 | 4.4408921E-15 |
| 2.0000 | 2.2204460E-15 | 3.9968029E-15 | 4.6629367E-15 | 4.6629367E-15 | 4.8849813E-15 |
| 2.2500 | 2.2204460E-15 | 4.8849813E-15 | 5.3290705E-15 | 4.8849813E-15 | 5.5511151E-15 |
| 2.5000 | 3.5527137E-15 | 4.8849813E-15 | 5.7731597E-15 | 5.7731597E-15 | 5.9952043E-15 |
| 2.7500 | 3.5527137E-15 | 5.3290705E-15 | 6.2172489E-15 | 6.6613381E-15 | 5.7731597E-15 |
| 3.0000 | 2.6645353E-15 | 5.7731597E-15 | 7.1054274E-15 | 7.1054274E-15 | 6.2172489E-15 |
| 3.2500 | 2.6645353E-15 | 6.2172489E-15 | 7.9936058E-15 | 7.9936058E-15 | 7.1054274E-15 |
| 3.5000 | 3.5527137E-15 | 7.1054274E-15 | 7.5495166E-15 | 8.4376950E-15 | 7.9936058E-15 |
| 3.7500 | 4.4408921E-15 | 7.1054274E-15 | 7.1054274E-15 | 8.8817842E-15 | 8.8817842E-15 |
| 4.0000 | 4.4408921E-15 | 7.1054274E-15 | 6.2172489E-15 | 9.7699626E-15 | 8.8817842E-15 |
| 4.2500 | 3.5527137E-15 | 7.9936058E-15 | 7.1054274E-15 | 9.7699626E-15 | 9.3258734E-15 |
| 4.5000 | 3.5527137E-15 | 8.8817842E-15 | 7.9936058E-15 | 9.3258734E-15 | 9.3258734E-15 |
| 4.7500 | 4.4408921E-15 | 9.7699626E-15 | 7.1054274E-15 | 7.5495166E-15 | 7.5495166E-15 |
| 5.0000 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 | 0.0000000E+00 |
| CPU time | 10.234 sec | 10.267 sec | 10.311 sec | 11.234 sec | 11.472 sec |

Table 3: $L_\infty$ errors for the results obtained by explicit method in Problem (2) at various time levels when $0 \leq x \leq 1$ with $h = k = 0.01$ and different values of $\alpha$.

| $t$ | $\alpha = 1$ | $\alpha = 0.5$ | $\alpha = 0.1$ | $\alpha = 0.01$ | $\alpha = 0.001$ |
|---|---|---|---|---|---|
| 1 | 8.3461517E-03 | 6.2713962E-03 | 5.9035084E-04 | 5.2527475E-04 | 5.8305718E-05 |
| 1.25 | 8.7373945E-03 | 6.6982179E-03 | 6.2340884E-04 | 5.4157402E-04 | 6.1545395E-05 |
| 1.5 | 8.3046178E-03 | 6.5296910E-03 | 6.0583013E-04 | 5.9586585E-04 | 5.8813139E-05 |
| 1.75 | 7.5878211E-03 | 6.0948040E-03 | 5.6431533E-04 | 5.5552189E-04 | 5.4561450E-05 |
| 2 | 6.8174110E-03 | 5.5887553E-03 | 5.1496783E-04 | 4.1040723E-04 | 5.0070766E-05 |
| 2.25 | 6.0824791E-03 | 5.1416846E-03 | 4.6936553E-04 | 4.2345919E-04 | 4.5447392E-05 |
| 2.5 | 5.4149016E-03 | 4.7037502E-03 | 4.2670328E-04 | 4.1825492E-04 | 4.1018329E-05 |
| 2.75 | 4.8419589E-03 | 4.2911947E-03 | 3.8684968E-04 | 3.6635530E-04 | 3.6922992E-05 |
| 3 | 4.4680283E-03 | 3.9106389E-03 | 3.5035236E-04 | 2.3830366E-04 | 3.3217430E-05 |
| CPU time | 5.775 sec | 5.823 sec | 6.456 sec | 6.567 sec | 6.781 sec |

Table 4: $L_\infty$ errors for the results obtained by Crank-Nicolson method in Problem (2) at various time levels when $0 \leq x \leq 1$ with $h = k = 0.01$ and different values of $\alpha$.

| $t$ | $\alpha = 1$ | $\alpha = 0.5$ | $\alpha = 0.1$ | $\alpha = 0.01$ | $\alpha = 0.001$ |
|---|---|---|---|---|---|
| 1 | 1.0658141E-08 | 3.5527137E-09 | 2.8817842E-09 | 7.9936058E-10 | 1.7763568E-11 |
| 1.25 | 7.9936058E-08 | 6.2172489E-09 | 5.6645353E-09 | 6.2172489E-10 | 2.6645353E-11 |
| 1.5 | 1.0658141E-08 | 6.2172489E-09 | 5.6645353E-09 | 7.1054274E-10 | 8.8817842E-11 |
| 1.75 | 1.0658141E-08 | 7.9936058E-09 | 5.8817842E-09 | 8.8817842E-10 | 2.2204460E-11 |
| 2 | 9.7699626E-08 | 7.1054274E-09 | 5.7763568E-09 | 9.7699626E-10 | 2.6645353E-11 |
| 2.25 | 1.1546319E-08 | 5.3290705E-09 | 3.7763568E-09 | 9.7699626E-10 | 1.7763568E-11 |
| 2.5 | 9.3258734E-08 | 5.7731597E-09 | 3.2204460E-09 | 8.8817842E-10 | 1.3322676E-11 |
| 2.75 | 8.8817842E-08 | 4.8849813E-09 | 2.7763568E-09 | 9.7699626E-10 | 8.8817842E-11 |
| 3 | 9.3258734E-08 | 5.7731597E-09 | 4.7763568E-09 | 9.3258734E-10 | 8.8817842E-11 |
| CPU time | 12.675 sec | 12.687 sec | 12.968 sec | 13.789 sec | 13.996 sec |

## 7. Conclusions

In this paper, the explicit and Crank-Nicolson FDMs have been used to solve the Burgers' equation (1). The Crank-Nicolson method was shown to be unconditionally stable, while the explicit technique was proven to be conditionally stable with a stability bound of $0 < \mu_2 \leq 1/2\alpha$. Furthermore, we have seen that the order of convergence of the explicit method is first order in $k$ and second order in $h$, whereas for the Crank-Nicolson method is of second order in both $k$ and $h$. A comparison is also made between the approximate and exact solutions. The computational findings indicate that these methods serve as an effective and efficient approach for obtaining an approximate solution to the Burgers' equation, demonstrating remarkable consistency with the exact solution. In conclusion, the Crank-Nicolson method gave optimal results and was better than the explicit method, but the explicit method required less CPU time. In the future, we intend to broaden the application of the Haar wavelet collocation method to the Burgers' problems, for more details see [6, 7].

## Author's Contribution

We confirm that all named authors have read and approved the manuscript. We also confirm that each author has the same contribution to the paper. We further confirm that all authors have approved the order of authors listed in the manuscript.

## Conflict of Interest

The authors declare that there is no conflict of interest.

## AI Usage Declaration

The authors declare that AI tools were limited to language and readability improvement and were not used to generate scientific content, data, analyses, or conclusions. Responsibility for the manuscript remains with the authors.

## Funding

Not applicable.

## Data Availability

Not applicable.

## Acknowledgment

The authors gratefully acknowledge the academic and institutional support provided by Koya University and Akre University for Applied Sciences.

## References

[1] Logan JD. Applied Mathematics: A Contemporary Approach. New York: J. Wiley & Sons; 1987.

[2] Tadmor E. A review of numerical methods for nonlinear partial differential equations. Bull Am Math Soc. 2012;49(4):507-54. https://doi.org/10.1090/S0273-0979-2012-01379-4

[3] Sadeeq MI, Sabawi YA. Cubic B-spline finite element method for solving Whitham-Broer-Kaup shallow water model. Int J Comput Methods. 2025. https://doi.org/10.1142/S021987622550032X

[4] Gumel AB, Twizell EH, Arigu MA, Fakhir F. Numerical methods for a non-linear system arising in chemical kinetics. Pertanika J Sci Technol. 1997;5(2):191-200. http://psasir.upm.edu.my/id/eprint/3335

[5] Lapidus L, Pinder GF. Numerical solution of partial differential equations in science and engineering. New York: J. Wiley & Sons; 1999.

[6] Mohammad NA, Sabawi YA, Hasso MS. A compact finite-difference and Haar wavelets collocation technique for parabolic Volterra integro-differential equations. Phys Scr. 2024;99(12):125251. https://doi.org/10.1088/1402-4896/ad8d3d

[7] Mohammad NA, Sabawi YA, Hasso MS. Numerical solution based on the Haar wavelet collocation method for partial integro-differential equations of Volterra type. Arab J Basic Appl Sci. 2024;31(1):614-28. https://doi.org/10.1080/25765299.2024.2419145

[8] Mohammad NA, Sabawi YA, Hasso MS. Error estimation and approximate solution of nonlinear Fredholm integro-differential equations. Palest J Math. 2024;13(3):354-70. https://pjm.ppu.edu/paper/1786-error-estimation-and-approximate-solution-nonlinear-fredholm-integro-differential

[9] Pirdawood MA, Sabawi YA. High-order solution of generalized Burgers'-Fisher equation using compact finite difference and DIRK methods. J Phys Conf Ser. 2021;1999(1):012088. https://doi.org/10.1088/1742-6596/1999/1/012088

[10] Sabawi YA, Hussen BO. A cubic B-spline finite element method for Volterra-integro differential equation. Palest J Math. 2024;13(3):571-83. https://pjm.ppu.edu/paper/1802-cubic-b-spline-finite-element-method-volterra-integro-differential-equation

[11] Sabawi YA, Pirdawood MA, Sadeeq MI. A compact fourth-order implicit-explicit Runge-Kutta type method for solving Diffusive Lotka–Volterra system. J Phys Conf Ser. 2021;1999:012103. https://doi.org/10.1088/1742-6596/1999/1/012103

[12] Smith GD. Numerical solution of partial differential equations: finite difference methods. 3rd ed. Oxford: Oxford University Press; 1985.

[13] Cushman-Roisin B. Analytical, linear stability criteria for the LeapFrog, Dufort-Frankel method. J Comput Phys. 1984;53(2):227-39. https://doi.org/10.1016/0021-9991(84)90039-1

[14] Fadugba SE, Nwozo CR. Crank-Nicolson finite difference method for the valuation of options. Pac J Sci Technol. 2013;14(2):136-46. https://www.akamai.university/uploads/1/2/7/7/127725089/pjst14_2_136.pdf

[15] Manna SA, Saeed RK, Easif FH. The finite difference methods and its stability for Glycolysis model in one dimension. J Math Comput Sci. 2012;2(6):1634-45. https://scik.org/index.php/jmcs/article/view/484

[16] Kutluay S, Bahadir AR, Ozdes A. Numerical solution of one-dimensional Burgers' equation: explicit and exact-explicit finite difference methods. J Comput Appl Math. 1999;103:251-61. https://doi.org/10.1016/S0377-0427(98)00261-1

[17] Dag I, Saka B, Boz A. B-spline Galerkin methods for numerical solutions of the Burgers' equation. Appl Math Comput. 2005;166:506-22. https://doi.org/10.1016/j.amc.2004.06.078

[18] Fan CM, Li PW. Generalized finite difference method for solving two-dimensional Burgers' equations. Procedia Eng. 2014;79:55-60. https://doi.org/10.1016/j.proeng.2014.06.310

[19] Liao W, Zhu J. Efficient and accurate finite difference schemes for solving one-dimensional Burgers' equation. Int J Comput Math. 2011;88(12):2575-90. https://doi.org/10.1080/00207160.2010.548519

[20] Fletcher CAJ. A comparison of finite element and finite difference solutions of the one- and two-dimensional Burgers' equations. J Comput Phys. 1983;51(1):159-88. https://doi.org/10.1016/0021-9991(83)90085-2

[21] Evseev EG, Morozov EV. Explicit finite difference method in the dynamic analysis of composite structures. Compos Struct. 1997;39(3-4):215-21. https://doi.org/10.1016/S0263-8223(97)00115-3

[22] Ucar Y, Yagmurlu NM, Tasbozan O. Numerical solutions of the modified Burgers' equation by finite difference methods. J Appl Math Stat Inform. 2017;1(3):19-30. https://doi.org/10.1515/jamsi-2017-0002

[23] Cleophas K, Benjamin N, John W. Hybrid Crank-Nicolson-DuFort and Frankel (CN-DF) scheme for the numerical solution of the 2-D coupled Burgers' system. Appl Math Sci. 2014;8(48):2353-61. http://dx.doi.org/10.12988/ams.2014.310596

[24] Duffy DJ. A critique of the Crank-Nicolson scheme: strengths and weaknesses for financial instrument pricing. Wilmott J. 2004;4:68-76. https://doi.org/10.1002/wilm.42820040417

[25] Wani SS, Thakar SH. Crank-Nicolson type method for Burgers' equation. Int J Appl Phys Math. 2013;3(5):324-8. https://doi.org/10.7763/IJAPM.2013.V3.230

[26] Bieniasz LK. The von Neumann stability of finite-difference algorithms for the electrochemical kinetic simulation of diffusion coupled with homogeneous reactions. J Electroanal Chem. 1993;345(1-2):13-25. https://doi.org/10.1016/0022-0728(93)80466-U

[27] Biazar J, Aminikhah H. Exact and numerical solutions for non-linear Burgers' equation by VIM. Math Comput Model. 2009;49:1394-400. https://doi.org/10.1016/j.mcm.2008.12.006