Research Article

Finite Volume Method for a Time-Dependent Convection-Diffusion-Reaction Equation with Small Parameters

Uzair Ahmed, Daoud Suleiman Mashat, and Dalal Adnan Maturi

Department of Mathematics, King Abdulaziz University, Jeddah, Saudi Arabia

Correspondence should be addressed to Uzair Ahmed; uabdulkhalimahmed@stu.kau.edu.sa

Received 24 January 2022; Accepted 2 April 2022; Published 17 May 2022

Academic Editor: Sining Zheng

Copyright © 2022 Uzair Ahmed et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Convection, diffusion, and reaction mechanisms are characteristics of transient mass-transfer phenomena that occur in natural and industrial systems. In this article, we contemplate a passive scalar transport governed by the convection-diffusion-reaction (CDR) equation in 2D flow. The efficiency of solving computationally partial differential equations can be illustrated by using a precise numerical method that yields remarkable precision at a low cost. The accuracy and computational efficiency of two second-order finite difference methods were investigated. The results were compared to a finite volume technique, which has a memory advantage and conserves mass, momentum, and energy even on coarse grids. For various diffusion coefficient values, numerical simulation of unsteady CDR equation are also performed. The techniques were examined for consistency and convergence. The effectiveness and accuracy of these approaches for solving CDR equations are demonstrated by simulation results. Efficiency is measured using $L_2$ and $L_\infty$, and the estimated results are compared to the corresponding analytical solution.

1. Introduction

Different types of partial differential equations govern mathematical models of physical, chemical, biological, and environmental events. These models illustrate the conservation laws, whose solutions can be invariably discontinuous and have a large gradient, making precise numerical solutions extremely difficult to find. CDR equations appear frequently to model the phenomena in the field of biology, chemistry, ecology, physics, and engineering. The basic convection-diffusion-reaction model exhibits the time evolution of chemical or biological species. The mathematical equations representing this evolution are partial differential equations that can be derived from mass balances [1]. Convection-diffusion-reaction equations appear in modeling of the phenomenon of cancer and tumor [2–6]. Water above the ground, in pores and crevices of the ground, and the unsaturated zone are all affected by advection and diffusion, which have an impact on the transfer of contaminants. It can be significantly influenced by the reaction. The numerical features of modeling advection diffusion transport have been the subject of extensive research, which is still ongoing. As a result, a huge amount of literature are available for advection diffusion transport [7]. The mathematical models which are used to examine the pollution in water of river, stream, and lakes are also based on the convection-diffusion-reaction equation [8–10].

For 2D transient scalar CDR problem, the governing partial differential equation is expressed as follows:

$$\phi_t + \overrightarrow{u} \left( \phi_x + \phi_y \right) - \varepsilon \left( \phi_{xx} + \phi_{yy} \right) + \sigma \phi = h. \quad (1)$$

Reaction, diffusion, and convection are terms in the governing (1) containing the unknown variable’s zero, second, and first order derivatives. Electrophoresis segregation process simulation or the operation of many chemical reactors may be modeled using this type of equation as a elementary model. Both the intensity and the heating rate are unknown variables in such mechanisms. Another interesting use is the simulating fluid movement in a non-inertial environment. This event may be expressed in a diffusive-convective-reactive form, with inertial forces within the convection, viscous effects in diffusive section, and Coriolis forces in the reaction term [11]. The precise
solution of the CDR equation may include abrupt strata where the gradient of the solution varies quickly whether the system is substantially governed by advection or reaction. In such instances, some traditional numerical methods produce erroneous oscillations that pollute the entire domain [12]. Because of the convection term, numerical simulation is challenging for forecasting transport processes based on transient CDR equation. The flow characteristics are frequently characterized by high gradients, necessitating particular numerical treatment. The majority of classical schemes have spurious oscillations, which result in excessive numerical dispersion [13–18].

The study of CDR equation has garnered considerable attention recently [1–24]. Idelsohn et al. [11] used a Petrov–Galerkin methodology to solve CDR scenarios in 1995. Later in 1999, a semi-discretization approach and an ADI scheme, proposed by Polezhaev, were applied to inspect the CDR equation in 1D and 2D, respectively, by Sheu et al. [1]. It is possible for finite difference approximations to provide cyclic results when used to model convective-diffusive-reactive mass transfer. Akram Hossain [7] presented an explicit scheme based on finite difference approach to investigate advective-dispersive transport with reaction in 1999.

Kachiashvili et al. [9] used finite difference approximations to analyze the river Kobistskali contaminated by NO₃ in 1D, 2D, and 3D CDR equation. Phongthanapanich and Dechaumphai presented a hybrid finite element and finite volume model to discuss a 2D CDR equation. The CDR equation was discretized using the FVM, and the gradient variables at cell sides were defined using the finite element scheme [16]. Moreover, in 2010, Phongthanapanich and Dechaumphai [19] suggested an explicit scheme of FVM to study the CDR system with triangle based meshes. Furthermore, Sarra [6] worked on complexly shaped domains based on advection-diffusion-reaction model using a local radial basis function method in 2012.

Kaya [12] also proposed a FDM to solve CDR equations. Using the self-cleansing system of the river, Simon and Koya developed a 1D Streeter–Phelps model. This model also counts the amount of pollutant downstream since the pollutants move downstream with the stream velocity. Based on a few practical presumptions, the Streeter–Phelps demonstrate is adjusted and split into a system of coupled advection-diffusion-reaction equations expressing biochemical oxygen demand and the concentrations of dissolved oxygen [8]. Subsequently, in 2017, a mathematical model describing contaminants and suspended particles in a river or waterway was examined by Samalerk and Pochai [10]. The proposed water quality model was governed by advection-diffusion-reaction equation using the Saulyev finite difference scheme. In 2018, Maghdadi et al. [4] implemented a customized cancer progression forecasting model based on CDR equation. In same year, Al-Bayati and Wrobel [20] introduced a new scheme for an unsteady convection-diffusion-reaction equation relying on the element method. Soon after, they introduced another new scheme built on the formulation of radial integration BEM in 2019 [21]. Yousefi et al. [22] devised a novel and more efficient numerical approach for solving advection-diffusion equations. The approach was based on the integration of the advection-diffusion equation. Following that, the integral equation would be converted to a system of linear algebraic equations using a HCBFs. Later, Yousefi et al. [23] used Chebyshev polynomials to solve two-dimensional transient advection-diffusion equations using the spectral integral equation approach. To approximate the solution of one- and two-dimensional conservation laws, Yousefi et al. [24] presented a higher-order scheme based on a spectral volume technique. The study revealed that the Chebyshev spectral volume approach leads to the conventional spectral volume method when used in conjunction with the novel limiter. Recently, in 2020, an unconditionally convergent implicit forward time central space scheme of FDM was applied to study the 2D advection-diffusion-reaction equation by Pananu et al. [25].

The explicit and implicit approaches are two types of computational strategies for solving the PDEs. The earlier method is prevalent due to its simplicity. However, CFL stability criteria restrict the technique, so a minimal time step must be taken to maintain the stability of computation process [19]. When compared to the explicit method, the implicit method’s solution accuracy degrades with time, thus larger time steps cannot be used. It takes long time to invert the coefficient matrix, which is another issue with this problem. The amount of memory available is a critical factor to consider when simulating large-scale industrial operations.

2. Governing Equation

For the unsteady scalar CDR problem in two dimensions, the governing partial differential equation is expressed as follows:

\[ u_t + A(u_x + u_y) - \epsilon(u_{xx} + u_{yy}) + \sigma u = h, \]  

(2)

where \((x, y, t) \in \Gamma \times (0, T]\) alongside initial conditions \(u(x, y, 0) = u(0)(x, y), (x, y) \in \Gamma\). Given the Dirichlet boundary condition can be denoted by the following expression:

\[ u(a, y, t) = f_1(x, y, t), u(b, y, t) = f_2(x, y, t) \]
\[ u(x, c, t) = g_1(x, y, t), u(x, d, t) = g_2(x, y, t), \]

where \((x, y) \in \Gamma \times (0, T]\), \(\Gamma = (x, y) : a \leq x \leq b, c \leq y \leq d\) in \(R^2\), time interval is \([0, T]\) . It is considered that the functions \(f_1, f_2, u_0, g_1, \text{ and } g_2\) are smooth enough to simulate the conduction, diffusion, and so on. \(u_0, f_1, f_2, g_1, \text{ and } g_2\) are the given sufficiently smooth functions and \(u(x, y, t)\) may represent heat, diffusion, etc. Here, \(h = dxt = dy\); diffusion coefficient is denoted by \(\epsilon\) while reaction coefficient is represented by \(\sigma\) and \(A\) denotes the convection coefficient, respectively.

2.1. Analytical Solution. Analytical solution [26, 27] of the two dimensional CDR (2) is as follows:
The diffusion coefficient is selected such that the impact of diffusion coefficient on the solution is approximated. In this section, numerical schemes will be explored theoretically. Initially, we determine \( \Gamma \) from the exact solution. Here, \( \Gamma \times T = (x, y, t): 0 \leq x, y \leq 1, 0 < t \leq T \), \( A = (2, 3)^T \) and \( \sigma = 1 \). Moreover, to see the impact of diffusion coefficient on the solution, values of diffusion coefficient is selected such that \( \epsilon = 10, 10^{-1}, 10^{-2}, 10^{-3} \).

### 3. Numerical Methods

In this section, numerical schemes, to investigate the approximate solution of 2D CDR equation in a finite domain, will be explored theoretically. Initially, we determine integers \( P \) and \( Q \) to express step sizes, \( dx = (b - a)/P \) and \( dy = (d - c)/Q \) in respective horizontal and vertical directions. Dividing the interval \([a, b]\) into \( P \) equidistant parts of width \( dx \) and the interval \([c, d]\) into \( Q \) equidistant parts of width \( dy \). Illustrate a grid by mapping horizontal and vertical axes connecting the coordinate points \((x_p, y_q)\), where \( x_p = a + p dx \) for each \( p = 0, 1, 2, \ldots, P \) and \( y_q = c + q dy \) for each \( q = 0, 1, 2, \ldots, Q \), also the lines \( x = x_p \) and \( y = y_q \) are grid lines, and their junctions are grid’s mesh. For every work point interior of the lattice, \((x_p, y_q)\), for \( p = 1, 2, \ldots, P - 1 \) and \( q = 1, 2, \ldots, Q - 1 \). We apply different calculations to approximate the numerical arrangement to the issue in this equation (2). We expect \( tn = nk, n = 0, 1, \ldots, NT \) where \( t \) represents the time.

#### 3.1. 2nd Order Crank–Nicolson Implicit Scheme

By applying the Crank–Nicolson method to (2), by integrating in the compact way, we obtain the following:

\[
\begin{align*}
U(t, x, y) &= 16 \sin(\pi t) (1 - x)(1 - y) \\
&\times \left\{\frac{1}{2} + \arctan \left( \frac{2e^{(-1/2)}(0.25^2 - (x - 0.5)^2 - (y - 0.5)^2)}{\pi} \right) \right\}.
\end{align*}
\]

(3)

Within the computations the constrain term \( h = dx = dy \), the initial condition \( u_0 \) and the boundary conditions are set from the exact solution. Here, \( \Gamma \times T = (x, y, t): 0 \leq x, y \leq 1, 0 < t \leq T \), \( A = (2, 3)^T \) and \( \sigma = 1 \). Moreover, to see the impact of diffusion coefficient on the solution, values of diffusion coefficient is selected such that \( \epsilon = 10, 10^{-1}, 10^{-2}, 10^{-3} \).

By substituting the above expressions into (2), the 2nd order implicit method is given by the following:

\[
\begin{align*}
\delta_x u_t &= \frac{u_{p+1,q}^{n+1} - u_{p-1,q}^{n+1} + u_{p+1,q}^{n} - u_{p-1,q}^{n}}{4h} , \\
\delta_y u_t &= \frac{u_{p,q+1}^{n+1} - u_{p,q-1}^{n+1} + u_{p,q+1}^{n} - u_{p,q-1}^{n}}{4h} , \\
\delta_x^2 u &= \frac{u_{p+1,q}^{n+1} - 2u_{p,q}^{n+1} + u_{p-1,q}^{n+1}}{2h} + \frac{u_{p+1,q}^{n} - 2u_{p,q}^{n} + u_{p-1,q}^{n}}{2h} , \\
\delta_y^2 u &= \frac{u_{p,q+1}^{n+1} - 2u_{p,q}^{n+1} + u_{p,q-1}^{n+1}}{2h} + \frac{u_{p,q+1}^{n} - 2u_{p,q}^{n} + u_{p,q-1}^{n}}{2h}.
\end{align*}
\]

(4)

by substituting the above expressions into (2), the 2nd order implicit method is given by the following:

\[
\begin{align*}
&u_{p,q}^{n+1} - u_{p,q}^{n} = P_1 \left[ \left( u_{p+1,q}^{n+1} - 2u_{p,q}^{n+1} + u_{p-1,q}^{n+1} \right) + \left( u_{p+1,q}^{n} - 2u_{p,q}^{n} + u_{p-1,q}^{n} \right) \right] - \\
&P_2 \left[ \left( u_{p+1,q}^{n+1} - 2u_{p,q}^{n+1} + u_{p-1,q}^{n+1} \right) + \left( u_{p,q+1}^{n+1} - 2u_{p,q}^{n+1} + u_{p,q-1}^{n+1} \right) \right] + \\
&P_3 \left[ \left( u_{p+1,q}^{n} - 2u_{p,q}^{n} + u_{p-1,q}^{n} \right) + \left( u_{p,q+1}^{n} - 2u_{p,q}^{n} + u_{p,q-1}^{n} \right) \right] + \\
&P_4 \left( u_{p,q}^{n+1} - u_{p,q}^{n} \right) = 0
\end{align*}
\]

\[
\begin{align*}
&\text{where } P_1 = \frac{\epsilon}{2kh^2}, P_2 = \frac{A_x}{4h}, P_3 = \frac{A_y}{4h}.
\end{align*}
\]

(5)
The implicit scheme appears that the accuracy is of $O(k^2 + h_x^2 + h_y^2)$. Moreover, we use the notation $U_{p,q}^n$ and $u_{p,q}^n$ representing the exact and approximate solution of mesh points $(x_p, y_q, t_n)$ respectively. Furthermore, we utilize the von Neumann stability investigation of method (4) to determine the stability of the second order implicit method, which reveals the unconditional stability of scheme. A network is presently pentadiagonal, even though results are good, but tragically because of huge iteration size, it amplifies from the inclining at slightest $n$ sections absent in each path, but an alternative strategy is that schemes can be utilized to deal similar issues, because of the huge transfer speed, expanding network focuses the calculation ended up utilizing to deal similar issues, because of the huge transfer speed, expanding network focuses the calculation ended up more troublesome. To overcome this trouble, another numerical scheme is needed due to large number of calculations. The iterative method is utilized for a linear system. The CN scheme is computationally expensive.

3.2. Computationally Efficient Implicit Scheme. To find a time effective substitute, we investigated that the CN implicit scheme for 2D (2) and discovered that efficiency in term of time is not a prominent feature for this scheme. To get excellent results along with time effectiveness, the efficient ADI scheme can be applied. Here, the finite difference conditions are composed at two levels. In any case, two diverse limited contrast approximations were utilized then again, initially to perform operations from $n$ to $n^*$, and later from $(n^*)$ to $(n + 1)$. Similar constraints are utilized for this scheme as portrayed before. For the determination of the alternating direction implicit method, following steps have been taken:

**Horizontal Sweep**

\[
\frac{2}{k} [u_{p,q}^{n+1} - u_{p,q}^n] = Q_1 [u_{p+1,q}^{n+1} - 2u_{p+1,q}^n + u_{p-1,q}^n] + Q_2 [u_{p,q+1}^{n+1} - 2u_{p,q+1}^n + u_{p,q-1}^n] + \frac{Q_3}{2} [u_{p+1,q}^n - u_{p-1,q}^n] - Q_4 [u_{p+1,q+1}^n - u_{p,q+1}^n - u_{p,q-1}^n] + G(u^n)
\]

In simplified form, the (6) is as follows:

\[
-Q_1 u_{p-1,q}^n + \frac{2}{k} + Q_2} u_{p,q}^n - Q_1 u_{p+1,q}^n = Q_1 [u_{p+1,q}^n - 2u_{p,q}^n + u_{p-1,q}^n] + Q_2 [u_{p,q+1}^n - 2u_{p,q}^n + u_{p,q-1}^n] + Q_3 [u_{p+1,q}^n - u_{p-1,q}^n] - Q_4 [u_{p+1,q+1}^n - u_{p,q+1}^n - u_{p,q-1}^n] + G(u^n)
\]

**Initial Node in Horizontal Direction**

\[
\frac{2}{k} + Q_2} u_{p,q}^n - Q_1 u_{p+1,q}^n = Q_1 [u_{p+1,q}^n - 2u_{p,q}^n + u_{p-1,q}^n] + Q_3 [u_{p+1,q}^n - u_{p-1,q}^n] - Q_4 [u_{p,q+1}^n - u_{p,q-1}^n] + G(u^n) + Q_1 u_{p-1,q}^n
\]

**Final Node in Horizontal Direction**

\[
-Q_1 u_{p-1,q}^n + \frac{2}{k} + Q_2} u_{p,q}^n - Q_1 u_{p+1,q}^n = Q_1 [u_{p+1,q}^n - 2u_{p,q}^n + u_{p-1,q}^n] + Q_3 [u_{p+1,q}^n - u_{p-1,q}^n] - Q_4 [u_{p,q+1}^n - u_{p,q-1}^n] + G(u^n) + Q_1 u_{p+1,q}^n
\]

where $Q_1 = (\Gamma/2h^2)$, $Q_2 = (\Gamma/h^2)$, $Q_3 = (A_x/2h)$, $Q_4 = (A_y/2h)$ and $G(u^n) = [(2/k) - \sigma]u_{p,q}^n$.
\[
\begin{align*}
\frac{2}{k} [u_{p,q}^{n+1} - u_{p,q}^*] &= Q_1 [u_{p,q+1}^{n+1} - 2u_{p,q}^{n+1} + u_{p,q-1}^{n+1} + u_{p,q+1}^* - 2u_{p,q}^* + u_{p,q-1}^*] + \\
Q_2 [u_{p+1,q}^* - 2u_{p,q}^* + u_{p-1,q}^*] - Q_3 [u_{p+1,q}^* - u_{p-1,q}^*] - Q_4 [u_{p,q+1}^* - u_{p,q-1}^*] - \sigma u_{p,q}^*
\end{align*}
\]

(10)

In simplified form the (10) is as follows:

\[
\begin{align*}
-Q_1 u_{p,q}^{n+1} + & \left[ \frac{2}{k} + Q_2 \right] u_{p,q+1}^{n+1} - Q_1 u_{p,q-1}^{n+1} = Q_3 [u_{p+1,q}^* - 2u_{p,q}^* + u_{p-1,q}^*] + \\
Q_2 [u_{p+1,q}^* - 2u_{p,q}^* + u_{p-1,q}^*] - Q_3 [u_{p+1,q}^* - u_{p-1,q}^*] - Q_4 [u_{p,q+1}^* - u_{p,q-1}^*] + & \left[ \frac{2}{k} - \sigma \right] u_{p,q}^*
\end{align*}
\]

(11)

Initial Node in Vertical Direction

\[
\begin{align*}
\left[ \frac{2}{k} + Q_2 \right] u_{p,q}^{n+1} - Q_1 u_{p,q-1}^{n+1} = Q_3 [u_{p+1,q}^* - 2u_{p,q}^* + u_{p-1,q}^*] + \\
Q_2 [u_{p+1,q}^* - 2u_{p,q}^* + u_{p-1,q}^*] - Q_3 [u_{p+1,q}^* - u_{p-1,q}^*] - Q_4 [u_{p,q+1}^* - u_{p,q-1}^*] + & \left[ \frac{2}{k} - \sigma \right] u_{p,q}^*
\end{align*}
\]

(12)

Final Node in Vertical Direction

\[
\begin{align*}
-Q_1 u_{p,q}^{n+1} + & \left[ \frac{2}{k} + Q_2 \right] u_{p,q+1}^{n+1} - Q_1 u_{p,q-1}^{n+1} = Q_3 [u_{p+1,q}^* - 2u_{p,q}^* + u_{p-1,q}^*] + \\
Q_2 [u_{p+1,q}^* - 2u_{p,q}^* + u_{p-1,q}^*] - Q_3 [u_{p+1,q}^* - u_{p-1,q}^*] - Q_4 [u_{p,q+1}^* - u_{p,q-1}^*] + & \left[ \frac{2}{k} - \sigma \right] u_{p,q}^* + Q_1 u_{p,q+1}^{n+1}
\end{align*}
\]

(13)

where \( Q_1 = (\Gamma/2h^2), Q_2 = (\Gamma/h^2), Q_3 = (A_y/2h), Q_4 = (A_y/h^2) \) and \( G(u^*) = [(2/k) - \sigma]u_{p,q}^* \)

This scheme has accuracy of 2nd order both in space and time along with the unconditional stability.

The linear system in the horizontal direction is as follows:

\[
ar_{11} u_{x,m}^{n+1} = b_{1m} + F(k u_{x,m}^*)
\]

(14)

where \( m = 1, \ldots, M - 1 \)

The linear system in the vertical direction is as follows:

\[
c_{11} u_{y,m}^{n+1} = d_{1m} + G(k u_{y,m}^*)
\]

(15)

where \( l = 1, \ldots, L - 1 \)

where \( a_{11}, c_{11} \) create tridiagonal matrix and the array \( b_{1m}, d_{1m} \) are dependent upon \( p \) and \( q \)

The reactive term in horizontal-direction

\[
G(k u_{x,m}^*) = [kG(u_{x,m}^*, \ldots, kG(u_{x,L-1}^*)]
\]

So, also for the reactive term in the vertical direction, it is as follows:

\[
G(k u_{y,m}^*) = [kG(u_{y,m}^*, M-1, \ldots, kG(u_{y,L-1}^*)]
\]

The notations \( u_{x,m}^* \) and \( u_{y,m}^* \) are used in same manner. The scheme gives us 2nd order accuracy in time and space \( O(k^2 + h^2 + h_y^2) \). For stability analysis, this scheme exhibits unconditional stability. The stability of alternating direction implicit strategy (6) and (10) can be obtained by using the von Neumann criteria for stability. The regular stability investigation criteria indicate toward the amplification factor \( G \) of (6) and (10) may be observed as by substituting:

\[
u^n = \lambda^n e^{i(k_x h_x + k_y h_y + k_z h_z)}
\]

(16)

It is obvious that the amplification factor is always less of equal to \( |G| \leq 1 \). Hence, it has unconditional stability. Eventually, the methodology results in a tridiagonal linear system. Iterative strategies were applied to unravel the abovementioned framework. The trick utilized in developing the ADI scheme is to divide the time step into two, and apply...
two diverse stencils in each half time step; hence, to increase time by one time step in grid point. Initially, we determine the values of these stencils, which are selected so that the arising linear framework is tridiagonal [28–32]. To get the approximate solution, we have to find the solution of a nonlinear tridiagonal framework with each passing step of time.

3.3. Finite Volume Method. The finite volume method has a lot of flexibility in solving the transport phenomena equations [33–39]. Whenever fluid flow appears, finite volume becomes more and more useful. When we have a fluid flow, we must have the corresponding velocity field satisfying the continuity equation. By solving (2) using the finite volume method, we define the quantities as algebraic values of the continuity equation. By solving (2) using the finite volume method, we define the quantities as algebraic values of the variables \( u \) at the main grid points s.t. \((W, E, P, N, S)\). To do that, we need profile assumptions. For applying the finite volume method, following steps should be needed.

where

\[
\begin{align*}
W &= \text{ west}, \\
E &= \text{ east}, \\
P &= \text{ center point}, \\
N &= \text{ north}, \\
S &= \text{ south}.
\end{align*}
\]

First, divide the domain logically into a number of finite size sub-domains. Each sub-domains is represented by a finite number of grid points. Subsequently, it converts and integrates the partial differential equation into balancing equations for a single element. Using an integrated quadrature of a given order of precision, the process converts volumetric and area integrals to algebraic variables across components and their surfaces. As a consequence, a collection of semi-discrete equations arises. Also, at this step, interpolation profiles are defined to estimate the fluctuation components and their surfaces. As a consequence, a collection of semi-discrete equations arises. Also, at this step, interpolation profiles are defined to estimate the fluctuation components and their surfaces.

3.3.1. 2nd Order Finite Volume Scheme. By solving (2) using the finite volume method, we have the following:

\[
\begin{align*}
\begin{cases}
\frac{u_{P}^{n+1} - C_{1}u_{P}^{n} + C_{2} [D_{e}(u_{E}^{n} - u_{P}^{n}) - D_{w}(u_{P}^{n} - u_{W}^{n})] + C_{3} [D_{N}(u_{N}^{n} - u_{P}^{n}) - D_{S}(u_{P}^{n} - u_{S}^{n})]} - }{C_{4}[u_{P}^{n} - u_{W}^{n}] - C_{5}[u_{N}^{n} - u_{S}^{n}]} \\
\end{cases},
\end{align*}
\]

where

\[
\begin{align*}
C_{1} &= (1 - \sigma \Delta x \Delta y), \\
C_{2} &= \frac{\Delta t}{\Delta x}, \\
C_{3} &= \frac{\Delta t}{\Delta y}, \\
C_{4} &= \frac{\Delta t \Delta x}{2 \Delta x}, \\
C_{5} &= \frac{\Delta t \Delta y}{2 \Delta y}
\end{align*}
\]

\[
\begin{align*}
D_{e} &= \left(\frac{\epsilon}{\Delta x}\right), \\
D_{w} &= \left(\frac{\epsilon}{\Delta x}\right), \\
D_{n} &= \left(\frac{\epsilon}{\Delta y}\right), \\
D_{s} &= \left(\frac{\epsilon}{\Delta y}\right),
\end{align*}
\]

Bottom left corner boundary cell

\[
\begin{align*}
\frac{u_{P}^{n+1} - C_{1}u_{P}^{n} + C_{2} [D_{e}(u_{E}^{n} - u_{P}^{n}) - 2D_{w}(u_{P}^{n} - u_{W}^{n})] + C_{3} [D_{N}(u_{N}^{n} - u_{P}^{n}) - 2D_{S}(u_{P}^{n} - u_{S}^{n})]} - }{C_{4}[u_{P}^{n} - u_{W}^{n}] - C_{5}[u_{N}^{n} - u_{S}^{n}]} \\
\end{align*}
\]

Top left corner boundary cell

\[
\begin{align*}
\frac{u_{P}^{n+1} - C_{1}u_{P}^{n} + C_{2} [D_{e}(u_{E}^{n} - u_{P}^{n}) - 2D_{w}(u_{P}^{n} - u_{W}^{n})] + C_{3} [2D_{N}(u_{N}^{n} - u_{P}^{n}) - 2D_{S}(u_{P}^{n} - u_{S}^{n})]} - }{C_{4}[u_{P}^{n} - u_{W}^{n}] - C_{5}[u_{N}^{n} - u_{S}^{n}]} \\
\end{align*}
\]

Bottom right corner boundary cell

\[
\begin{align*}
\frac{u_{P}^{n+1} - C_{1}u_{P}^{n} + C_{2} [D_{e}(u_{E}^{n} - u_{P}^{n}) - 2D_{w}(u_{P}^{n} - u_{W}^{n})] + C_{3} [D_{N}(u_{N}^{n} - u_{P}^{n}) - 2D_{S}(u_{P}^{n} - u_{S}^{n})]} - }{C_{4}[u_{P}^{n} - u_{W}^{n}] - C_{5}[u_{N}^{n} - u_{S}^{n}]} \\
\end{align*}
\]

Top right corner boundary cell
\[
\begin{align*}
\mathbf{u}^{n+1} &= C_1 \mathbf{u}^n + C_2 \left[ 2D \left( \mathbf{u}_{\text{right}} - \mathbf{u}^n \right) - D_w \left( \mathbf{u}_{\text{left}}^n - \mathbf{u}^n \right) \right] + C_3 \left[ 2D_n \left( \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right) - D_s \left( \mathbf{u}_{\text{bottom}}^n - \mathbf{u}^n \right) \right], \\
C_4 \left[ \mathbf{u}_{\text{right}} - \mathbf{u}^n \right] - C_5 \left[ \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right],
\end{align*}
\]

(23)

Bottom non-corner boundary cell

\[
\begin{align*}
\mathbf{u}^{n+1} &= C_1 \mathbf{u}^n + C_2 \left[ D \left( \mathbf{u}_{\text{right}}^n - \mathbf{u}^n \right) - D_w \left( \mathbf{u}_{\text{left}}^n - \mathbf{u}^n \right) \right] + C_3 \left[ D_n \left( \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right) - 2D_s \left( \mathbf{u}_{\text{bottom}}^n - \mathbf{u}^n \right) \right], \\
C_4 \left[ \mathbf{u}_{\text{right}}^n - \mathbf{u}^n \right] - C_5 \left[ \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right],
\end{align*}
\]

(24)

Left non-corner boundary cell

\[
\begin{align*}
\mathbf{u}^{n+1} &= C_1 \mathbf{u}^n + C_2 \left[ D \left( \mathbf{u}_{\text{right}}^n - \mathbf{u}^n \right) - 2D_w \left( \mathbf{u}_{\text{left}}^n - \mathbf{u}^n \right) \right] + C_3 \left[ D_n \left( \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right) - D_s \left( \mathbf{u}_{\text{bottom}}^n - \mathbf{u}^n \right) \right], \\
C_4 \left[ \mathbf{u}_{\text{right}}^n - \mathbf{u}^n \right] - C_5 \left[ \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right],
\end{align*}
\]

(25)

Top non-corner boundary cell

\[
\begin{align*}
\mathbf{u}^{n+1} &= C_1 \mathbf{u}^n + C_2 \left[ D \left( \mathbf{u}_{\text{right}}^n - \mathbf{u}^n \right) - D_w \left( \mathbf{u}_{\text{left}}^n - \mathbf{u}^n \right) \right] + C_3 \left[ 2D_n \left( \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right) - D_s \left( \mathbf{u}_{\text{bottom}}^n - \mathbf{u}^n \right) \right], \\
C_4 \left[ \mathbf{u}_{\text{right}}^n - \mathbf{u}^n \right] - C_5 \left[ \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right],
\end{align*}
\]

(26)

Right non-corner boundary cell

\[
\begin{align*}
\mathbf{u}^{n+1} &= C_1 \mathbf{u}^n + C_2 \left[ 2D \left( \mathbf{u}_{\text{right}}^n - \mathbf{u}^n \right) - D_w \left( \mathbf{u}_{\text{left}}^n - \mathbf{u}^n \right) \right] + C_3 \left[ D_n \left( \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right) - D_s \left( \mathbf{u}_{\text{bottom}}^n - \mathbf{u}^n \right) \right], \\
C_4 \left[ \mathbf{u}_{\text{right}}^n - \mathbf{u}^n \right] - C_5 \left[ \mathbf{u}_{\text{top}}^n - \mathbf{u}^n \right],
\end{align*}
\]

(27)

The scheme gives us an accuracy of second order. For stability criteria, we know that \( P_e = \text{advection strength} / \text{diffusion strength} \). Figure 1. To define the stability of 2nd order finite volume scheme, \( P_e \) must be less than or equal to 2. The method in (18) demonstrates that despite its conservativeness and convergence, it is conditionally stable. Since the abovementioned inequality establishes a straight maximum limit for the limit step, improving spatial precision can be costly at times.

3.4. Algorithm. Clearly, the framework is tridiagonal and can be fathomed with the Thomas algorithm. The measurement of \( H \) is \( p \times q \). In common a tridiagonal framework can be composed as follows:

\[
a_p x_{p-1} + b_p x_p + c_p x_{p+1} = S, \text{ with } a_1 = c_1 = 0,
\]

(28)

The above system can be expressed as follows in a matrix-vector form:

\[
Hu = S,
\]

(29)

where \( H \) represents a coefficient matrix. The column vector on the right side is known. Our primary objective is to discover the resultant vector \( u \). Presently, we have the following:

\[
H = \begin{bmatrix}
  b_1 & c_1 & 0 & 0 & 0 & \cdots & 0 \\
  a_2 & b_2 & c_2 & 0 & 0 & \cdots & 0 \\
  \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
  \vdots & \vdots & \vdots & \vdots & \cdots & a_{n-1} & b_{n-1} & c_{n-1} \\
  \vdots & \vdots & \vdots & \vdots & \cdots & \cdots & a_n & b_n \\
  \end{bmatrix},
\]

(30)

\[
\mathbf{u} = [u_1, u_2, u_3, \ldots, u_n]^t,
\]

\[
\mathbf{S} = [s_1, s_2, s_3, \ldots, s_n]^t.
\]
The method is clarified within the taking after steps,

\[ H = LU, \]

\[
L = \begin{bmatrix}
\mu_1 & 0 & 0 & 0 & 0 & \ldots & 0 \\
0 & \mu_2 & 0 & 0 & 0 & \ldots & 0 \\
\alpha_3 & \beta_3 & \mu_3 & 0 & 0 & \ldots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
\ldots & \alpha_{n-1} & \beta_{n-1} & \mu_{n-1} & 0 & \ldots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \alpha_n & \beta_n & \mu_n
\end{bmatrix},
\]

and

\[
U = \begin{bmatrix}
1 & \delta_1 & \lambda_1 & 0 & 0 & \ldots & 0 \\
0 & 1 & \delta_2 & \lambda_2 & 0 & \ldots & 0 \\
0 & 0 & 1 & \delta_3 & \lambda_3 & \ldots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
\ldots & \ldots & 0 & 1 & \delta_{n-2} & \lambda_{n-2} & 0 & \ldots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ldots & \ldots & \ldots & 0 & \ldots & 0 & 1
\end{bmatrix}.
\]

By comparing both sides of the \( Hu = S \), it gives us the components of the matrices \( L \) and \( U \). The computational steps for the execution of the Thomas calculation are appeared in comes about, taken from particular illustrations.

### 4. Error Norms

The precision and consistency of the methods is studied in terms of error norms specifically \( L_2 \) and \( L_\infty \) which can be expressed as follows:

\[
\text{RMS Error} = \sqrt{\frac{\sum_{p,q=1}^{L} (U_{p,q} - u_{p,q})^2}{L \times L}},
\]

\[
L_\infty = \max_{1 \leq p \leq P} \sum_{q=1}^{P} \| U_{p,q} - u_{p,q} \|_1, \tag{33}
\]

\[
L_2 = \sqrt{\rho \left( \frac{U_{p,q} - u_{p,q}}{U_{p,q} - u_{p,q}} \right)},
\]

where \( u(x, y, t) \) and \( U(x, y, t) \) represent the approximate and exact solutions at the grid point \( (x_p, y_q, t_n) \). In this method, \( \rho (U_{p,q} - u_{p,q}) = \max(\lambda) \) and \( \lambda \) is an eigen value of \( (U_{p,q} - u_{p,q}) \).

### 5. Results and Discussion

In several application areas, such as electromagnetics, biomathematics, precise modeling of electric signals, and water modeling takes more time. Finite volume and second order implicit schemes have been devised to solve this problem and to lower the error norm that best matches the accuracy effectively. To calculate numerical derivatives effectively, we utilized a short step size along the grid line. MATLAB algorithms for the 2D CDR (2) were also constructed.

The ADI scheme is used to reduce computing costs while increasing efficiency, and the FVM is applied to ensure conservation over the entire scope of the problem. The convective and diffusive fluxes at the control volume faces are calculated using a finite volume approach that employs consistent formulations. The article’s main goal is to identify the accuracy and efficiency of second order numerical methods. The algorithms developed are compared to (3) analytical findings. The parameters \( \sigma = 1, \ A = (2, 3)^T \), and changing \( \epsilon = 10, 10^{-1}, 10^{-2} \) for the 2D CDR equation are used to achieve this goal. To emphasize the relevance of second order schemes on such phenomenological systems, numerical and analytical solutions are contrasted and justified in terms of error norms.

The calculated results are realized by changing the grid space, time space and by varying \( \epsilon \) values. Furthermore, utilizing 2nd-order finite difference schemes, the comparison of analytic and numerical results with \( \epsilon = 0.01 \) linked to two variables \( (p, q) \) at \( t = 0.1 \) with a fixed grid size of \( 10 \times 10 \) is found in Table 1. The truncation error is computed using the \( L_2 \) and \( L_\infty \) norms with a fixed grid size of \( 10 \times 10 \) in Table 2. Table 2 demonstrates that modifying the time \( (t = 2, 1, 0.5) \) with the same grid size at \( k = 0.0001, \epsilon_{\text{grid}} = 0.001 \), improves the accuracy of the \( L_\infty \) error norm. Table 3 shows that error decreases as grid size increases at \( k = 0.001 \). We also discovered that the alternating direction implicit method is more time efficient. In comparison to the Crank–Nicolson method, the findings show that ADI has accurate and excellent convergence. It is evident that as \( \epsilon \) decreases, (2) becomes convection dominated and as a result, we’ll require extra components to deal with the associated inconsistencies. Table 4 shows that if treated correctly, FVM may produce results that are comparable to the analytical solution for higher and lower values.

**Figure 1: Peclet number.**
of \( \epsilon \). On the other hand, all these approaches have computational problems for smaller values of \( \epsilon \leq 10^{-3} \).

Furthermore, it is observed that by reducing step sizes \( h \) values in Table 5 with grid size \( 20 \times 20 \) and \( \epsilon = 0.001 \) can prevent turbulence. In this situation, applying error norms to reduce the space step size improves the accuracy of the finite volume technique findings. By decreasing the space step, the root mean square value contributes to accuracy. It is a great strategy for dealing with conservative rules. In addition, the comparison of three techniques will be taken into account in Table 6. The spatial step size \( h \) and time level \( t = 0.1, 0.001, 0.5 \) with a fixed grid size of \( 50 \times 50 \) is presented using error analysis norm. Results can be seen using the \( L_{\infty} \) norm. Variation of the \( h \) parameter toward smaller values results in an increase in accuracy. When we changed the value of \( t \) from high to low, we saw that the error dropped. Moreover, the use of a finite volume method yielded conservative findings that fully support the precise solution. The CN and ADI schemes solve the problem effectively because of this excellent characteristic of the solution, but the second order finite volume scheme also shows a compatible solution with exactitude. In Table 7, CPU time shows efficiency of the proposed scheme with respect to self-time, or time invested in its child unit relative to the algorithm’s overall run time.
which is the time spent in its child function to the total time spent in the algorithm. When the two machines are compared by changing the grid size, it is clear that finite volume produces good results with tiny intervals. Using two machines, HP and Dell, it was discovered that Dell is more efficient than HP.

**Table 7: Performance of CPU time efficiency using FVM.**

| Grid size | Dell Total-time | Dell Self-time | HP Total-time | HP Self-time |
|-----------|----------------|----------------|---------------|--------------|
| 25 × 25   | 99.218         | 0.175          | 113.821       | 0.188        |
| 50 × 50   | 224.501        | 0.206          | 284.866       | 0.229        |
| 75 × 75   | 486.160        | 0.346          | 640.087       | 0.390        |
| 100 × 100 | 2308.526       | 0.805          | 3010.264      | 1.009        |

**Figure 2:** Analytical and approximate results of the CN scheme at 10 × 10 grid size, time \( t = 2 \), time step \( \Delta t = 0.0001 \), and \( \epsilon = 0.1 \).

**Figure 3:** Numerical solution of the CN scheme at 30 × 30 grid size, \( k = 0.01 \), and \( \epsilon = 0.01 \).
On a two-dimensional partial differential system, the graphical representation of numerical schemes is seen. A comparison was performed between analytic and numerical findings using a second order finite difference and finite volume methodology. For $k = 0.0001$, $t = 2$, $\epsilon = 0.1$ and grid size $= 10 \times 10$ can be seen from Figure 2. A study of CN’s approximate and analytical findings will be taken into account in this figure. The approximate results of the CN and ADI schemes are shown in Figures 3 and 4 for $k = 0.001$ and $k = 0.0001$, respectively. Figure 5 shows more accurate and refined observations than Figure 2 while using the same time step. An example is shown in Figure 6, which compares the exact and numerical solution of second order finite volume scheme at time $\epsilon = 0.1$ and grid size $= 20 \times 20$. A detailed examination of Figure 6 reveals good agreement with the exact answer, as evidenced by the data points at various places. Figure 7 compares the exact and numerical findings of a second order finite volume method at $h = 0.2$ with smaller time space. Figure 8 shows efficient and reliable numerical results using the ADI technique with grid size $= 75 \times 75$, $\epsilon = 0.01$ and $k = 0.001$. At a higher grid size, the numerical results of the FVM are contrasted to the exact values in Figure 9. While, two-dimensional numerical methods at different time and epsilon along the $\rho, q$ axis may
be seen in all figures. It demonstrates that the degree of numerical solutions in different grid places approximates the analytic results. The findings demonstrate that these schemes are reliable for solving PDEs.

In conclusion, the numerical experiments demonstrated the correctness and effectiveness of the numerical methods. However, the finite volume scheme is more suitable because it outperforms the finite difference method in the way that it...
does not require a structured mesh, but it can be used if desired. Despite the fact that structured grids require less memory to store than unstructured grids due to their easier connection, FVM surpass the finite difference methods for extremely complicated geometries. For the stated finite volume formulation of the diffusive flow, flux consistency assures conservation throughout the whole domain. It is aspirant that the analytical and numerical solutions are strongly aligned with generation encrypting, as shown in the figures and tables.

6. Conclusion

The 2D transient scalar CDR problem is solved numerically in this paper. Theoretical formulations of the proposed schemes were fully addressed. The three scheme’s efficiency,
precision, and stability are shown by a convincing example. Increasing the grid size or reducing the space step size resulted in error norms convergent toward zero. Although all of the schemes are consistent, the alternating direction implicit and finite volume schemes are less computationally expensive than the Crank–Nicolson (CN) scheme. The FVM, on the other hand, ensures that conservation is maintained across the domain. The stated techniques might be used to address similar mathematical challenges in biology, physics, and engineering.

Data Availability
The article contains the supporting data for the study’s findings.

Conflicts of Interest
The authors declare that they have no conflicts of interest.

Authors’ Contributions
All authors contributed equally.

References
[1] T. W. H. Sheu, S. K. Wang, and R. K. Lin, "An implicit scheme for solving the convection-diffusion-reaction equation in two dimensions," *Journal of Computational Physics*, vol. 164, no. 1, pp. 123–142, 2000.
[2] T. Hillen and K. J. Painter, "A user’s guide to PDE models for chemotaxis," *Journal of Mathematical Biology*, vol. 58, no. 1-2, pp. 183–217, 2009.
[3] E. Di Costanzo, V. Ingangi, C. Angelini, M. F. Carfora, M. V. Carriero, and R. Natalini, “A macroscopic mathematical model for cell migration assays using a real-time cell analysis,” *PLoS One*, vol. 11, no. 9, Article ID e0162553, 2016.
[4] N. Meghdadi, M. Soltani, H. Niroomand-Oscuii, and N. Yamani, "Personalized image-based tumor growth prediction in a convection-diffusion-reaction model," *Acta Neurologica Belgica*, vol. 120, no. 1, pp. 49–57, 2020.
[5] W. Alt, “Biased random walk models for chemotaxis and related diffusion approximations,” *Journal of Mathematical Biology*, vol. 9, no. 2, pp. 147–177, 1980.
[6] S. A. Sarra, “A Local Radial Basis Function method for Advection-Diffusion-Reaction equations on complexly shaped domains,” *Applied Mathematics and Computation*, vol. 218, no. 19, pp. 9853–9865, 2012.
[7] Md. A. Hossain, “Modeling advective-dispersive transport with reaction: An accurate explicit finite difference model,” *Applied Mathematics and Computation*, vol. 102, no. 2-3, pp. 101–108, 1999.
[8] T. Simon and P. R. Koya, “Numerical simulation of system of TimeDependent advection-diffusion-reaction equations describing river pollution,” *JSIPR*, vol. 30, no. 2, 2016.
[9] K. Kachiashvili, D. Gordeziani, R. Lazarov, and D. Mielkizhmanian, “Modeling and simulation of pollutants transport in rivers,” *Applied Mathematical Modelling*, vol. 31, no. 7, pp. 1371–1396, 2007.
[10] P. Samalerk and N. Pochai, “Numerical simulation of A one-dimensional water-quality model in A stream using a saulyev technique with quadratic interpolated initial-boundary conditions,” *Abstract and Applied Analysis*, vol. 2018, Article ID 1926519, 7 pages, 2018.
[11] S. Idelsohn, N. Nigro, M. Storti, and G. Buscaglia, “A Petrov-Galerkin formulation for advection-reaction-diffusion problems,” *Computer Methods in Applied Mechanics and Engineering*, vol. 136, no. 1-2, pp. 27–46, 1996.
[12] A. Kaya, “A finite difference scheme for multidimensional convection-diffusion-reaction equations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 278, pp. 347–360, 2014.
[13] A. Kurganov and E. Tadmor, “New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations,” *Journal of Computational Physics*, vol. 160, no. 1, pp. 241–282, 2000.
[14] R. J. Leveque, *Finite Volume Methods for Hyperbolic Problems*, Vol. 31, Cambridge University Press, Cambridge, UK, 3rd edition, 20.
[15] A. Malatip, N. Wansophark, and P. Dechaumphai, “Combined Streamline Upwind Petrov Galerkin method and segregated finite element algorithm for conjugate heat transfer problems,” *Journal of Mechanical Science and Technology*, vol. 20, no. 10, pp. 1741–1752, 2006.
[16] S. Phongthanapanich and P. Dechaumphai, “Combined finite volume and finite element method for convection-diffusion-reaction equation,” *Journal of Mechanical Science and Technology*, vol. 23, no. 3, pp. 790–801, 2009.
[17] J. Principe and R. Codina, “On the stabilization parameter in the subgrid scale approximation of scalar convection-diffusion-reaction equations on distorted meshes,” *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 21-22, pp. 1386–1402, 2010.
[18] H. Wang and J. Liu, “A semi-implicit finite-difference approach for two-dimensional coupled burgers equations,” *SIAM Journal on Scientific Computing*, vol. 23, no. 4, pp. 1418–1438, 2006.
[19] S. Phongthanapanich and P. Dechaumphai, “Finite volume method for convection–diffusion–reaction equation on triangular meshes,” *Int. J. Numer. Meth. Biomed. Engng*, vol. 26, pp. 716–727, 2010.
[20] S. A. Al-Bayati and L. C. Wrobel, “A novel dual reciprocity boundary element formulation for two-dimensional transient convection-diffusion-reaction problems with variable velocity,” *Engineering Analysis with Boundary Elements*, vol. 94, pp. 60–68, 2018.
[21] S. A. Al-Bayati and L. C. Wrobel, “Radial integration boundary element method for two-dimensional non-homogeneous convection-diffusion-reaction problems with variable source term,” *Engineering Analysis with Boundary Elements*, vol. 101, pp. 89–101, 2019.
[22] M. H. N. Yousefi, S. H. G. Najafabadi, and E. Tohidi, “A fast and efficient numerical approach for solving advection–diffusion equations by using hybrid functions,” *Computational and Applied Mathematics*, vol. 38, no. 4, pp. 1–9, 2019.
[23] M. H. N. Yousefi, S. H. G. Najafabadi, and E. Tohidi, “A New Spectral Integral Equation Method for Solving Two-Dimensional Unsteady Advection-Diffusion Equations via Chebyshev Polynomials,” *Engineering Computations*, vol. 36, no. 7, 2019.
[24] M. Hadadian Nejad Yousefi, S. H. Ghoreishi Najafabadi, and E. Tohidi, “A new WENO based Chebyshev Spectral Volume method for solving one- and two-dimensional conservation laws,” *Journal of Computational Physics*, vol. 403, Article ID 109055, 2020.
K. Pananu, S. Sungnul, S. Sirisubtawee, and S. Phongthanapanich, “Convergence and applications of the implicit finite difference method for advection-diffusion-reaction equations,” *IAENG International Journal of Computer Science*, vol. 47, pp. 645–663, 2020.

H. Fendoglu, C. Bozkaya, and M. Tezer-Sezgin, “DBEM and DRBEM solutions to 2D transient convection-diffusion-reaction type equations,” *Engineering Analysis with Boundary Elements*, vol. 93, pp. 124–134, 2018.

V. John and E. Schmeyer, “Finite element methods for time-dependent convection-diffusion-reaction equations with small diffusion,” *Computer Methods in Applied Mechanics and Engineering*, vol. 198, no. 3-4, pp. 475–494, 2008.

J. D. Cole, “On a quasi-linear parabolic equation occurring in aerodynamics,” *Quarterly of Applied Mathematics*, vol. 9, no. 3, pp. 225–236, 1951.

K. Pandey and L. Verma, “A note on crank-nicolson scheme for burgers’ equation,” *Applied Mathematics*, vol. 02, no. 07, pp. 883–889, 2011.

J. N. Noye, “Nonlinear Partial Differential Equations in Engineering,” in *Proceedings of the Conference In Queen’s College, University of Melbourne, Sydney, Australia*, 1981.

S. Mazumder, *Numerical Methods for Partial Differential Equations: Finite Difference and Finite Volume Methods*, 484 pages, Academic Press, Cambridge, MA, USA, 2015.

H. K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, Pearson Education, London, UK, 2007.

F. Moukalled, L. Mangani, and M. Darwish, *The Finite Volume Method in Computational Fluid Dynamics*, Vol. 113, Springer, Berlin, Germany, 2016.

S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, CRC Press, Florida, FL, USA, 2018.

D. A. Calhoun and C. Helzel, “A finite volume method for solving parabolic equations on logically cartesian curved surface meshes,” *SIAM Journal on Scientific Computing*, vol. 31, no. 6, pp. 4066–4099, 2010.