Compact finite difference schemes for solving partial differential equations

A Dissertation Submitted

to

NATIONAL INSTITUTE OF TECHNOLOGY KURUKSHETRA, INDIA

of

Master of Science

in

Mathematics

by

MANOJ SOLANKI

(723410009)

under the guidance of

Dr. MADDURI MANI HARSHITA
(Assistant Professor)

DEPARTMENT OF MATHEMATICS

NATIONAL INSTITUTE OF TECHNOLOGY

KURUKSHETRA-136119 (INDIA)

July 2025

Department of Mathematics
National Institute of Technology Kurukshetra

Certificate

I hereby certify that the work which is being presented in this M.Sc. Dissertation entitled "Compact finite difference schemes for solving partial differential equations", in partial fulfillment of the requirements for the award of the Master of Science in Mathematics is an authentic record of my own work carried out during a period of 2024-25 under the supervision of Dr. Harshita Madduri, Assistant Professor, Mathematics Department.

The matter presented in this thesis has not been submitted for the award of any other degree elsewhere.

Signature of the Candidate
Manoj Solanki
Roll No: 723410009

Date:
Place: Kurukshetra
This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Signature of the Supervisor
Dr. MADDURI MANI HARSHITA
Assistant Professor, Department of Mathematics,
National Institute of Technology Kurukshetra,
Kurukshetra-136119 (India)

Date:
Place: Kurukshetra

ACKNOWLEDGEMENT

I would like to express my deepest gratitude and appreciation to several individuals who have contributed to the successful completion of this thesis. Their support, guidance, and encouragement have been invaluable throughout my academic journey.

Above all, I want to express my gratitude to my supervisor, Dr. Madduri Mani Harshita, who is an assistant professor at the Department of Mathematics, NIT Kurukshetra. Her proficiency with mathematical modelling and numerical techniques, as well as her understanding and patient guidance, have greatly influenced the focus of this study. She has continuously pushed me to investigate fresh angles in applied mathematics, especially when it comes to applying numerical methods to tackle practical issues, since the start of this project. It has been a remarkable and intellectually stimulating experience to work under her direction.

I would also like to extend my heartfelt thanks to the Department of Mathematics, NIT Kurukshetra, for providing a research-conducive environment and all the academic support needed to complete this work. The department's commitment to fostering research excellence and its encouragement of independent inquiry have greatly contributed to my growth as a scholar.

A special word of thanks goes to my mother, whose constant mental support and quiet strength have always been a source of encouragement in my academic journey. Additionally, I would like to thank my family and friends for their continuous support, encouragement, and belief in me throughout this journey. Their presence has been a source of strength and motivation. Last but not least, I thank God, the Almighty, for granting me the strength, patience, and wisdom to carry out and complete this research successfully.

Manoj Solanki

ABSTRACT

A higher order numerical approach for solving 1-D parabolic PDEs, such as the Diffusion and Advection-Diffusion-Reaction (ADR) equations, is developed and studied in this work. The proposed approach uses the Crank–Nicolson traditional finite difference method (FDM) for time discretization and a fourth-order compact finite difference method (CFDM) for spatial discretization. The accuracy and efficiency limits of the explicit, implicit, and Crank-Nicolson finite difference methods (FDM) are highlighted in this we use operator-based and Taylor series techniques, a compact study is created to overcome these issues, producing a solution that uses fewer grid points while achieving higher accuracy. The stability of the scheme is analyzed using Von Neumann (Fourier analysis). A comparative study of CFDM is conducted through Python implementation on a variety of problems involving diffusion, advection, and reaction with constant coefficients. The results show that CFDM yields more accurate and efficient solutions than traditional methods. The work concludes by outlining future directions, particularly the extension of the scheme to PDEs with variable coefficients, for which initial developments are underway.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The equations involving partial derivatives of function of several variables is known as Partial Differential Equation. PDEs play on important role in physical phenomena like modeling dynamics, describing spatial and temporal variations, conservation laws, describing diffusion and transport processes, wave propagation etc.

Higher order PDEs can be classified based on the form of the equation and the nature of their solutions. General form of n^{th} order PDEs in two independent variables,

$$(a_0 D^n + a_1 D^{n-1} D^{'} + a_2 D^{n-1} D^{'2} + ... + a_{n-1} D D^{'n-1} + a_n D^{'})u = f(x,y) \tag{1.0.1}$$

where $D^n = \frac{\partial^n}{\partial x^n}, D^{'n} = \frac{\partial^n}{\partial t^n}$ with $a_0, a_n \neq 0$.

1.1 A general form of Quasi linear / Semi linear PDEs

Second-order quasi-linear or semi-linear PDEs govern a large number of physical problems in mathematical modelling. These equations maintain linearity with regard to the highest-order derivatives, although they are not entirely linear in all terms.

Definition

A quasi-linear or semi-linear PDEs is a second-order PDEs in which the highest-order derivative terms appear linearly, although their coefficients may depend on the independent variables.
The general form of such an equation in two variables is,

$$Au_{xx} + Bu_{xy} + Cu_{yy} + F(x,y,u_x,u_y,u) = G(x,y) \tag{1.1.1}$$

here, A, B, C, F, and G are functions of x, y. The function $u = u(x,y)$ is unknown function of variable x and y.
The classification of second-order PDEs given in equation (1.1.1) is based on the discriminant Δ of the quadratic form which is given by,

$$\Delta = B^2 - 4AC \tag{1.1.2}$$

Second order PDE are classified into three categories using equation (1.1.2),

- PDE is hyperbolic if $\Delta > 0$. For example

$$u_t = c^2 u_{xx} \quad \text{(One-dimensional heat equation)} \tag{1.1.3}$$

- PDE is parabolic if $\Delta = 0$. For example

$$u_{tt} = c^2 u_{xx} \text{ (One-dimensional wave equation)} \tag{1.1.4}$$

- PDE is elliptic if $\Delta < 0$. For example

$$u_{xx} + u_{yy} = 0 \quad \text{(Two-dimensional Laplace equation)} \tag{1.1.5}$$

In order to verify that the problem is well-posed and has a unique solution, suitable beginning and boundary conditions must be specified for the diffusion equation (1.1.3). These conditions typically include,

- An initial condition specifying the state of the system at initial time $t = 0$,

$$u(x,0) = \phi(x) \tag{1.1.6}$$

- Boundary conditions that define the nature of the solution at the ends of the spatial domain $x = 0$ and $x = L$,

$$u(0,t) = \psi_L(t), \quad u(L,t) = \psi_R(t) \tag{1.1.7}$$

Here, L represents the length of the domain (for example, a rod). These prescribed conditions allow us to apply finite difference methods for obtaining a numerical solution, which we shall explore in the following sections.

1.2 Importance of Parabolic PDEs

Parabolic PDEs play a foundational role in the mathematical modeling of time-dependent processes involving diffusion, transport, and transformation phenomena. These equations describe how quantities such as temperature, concentration, or population density evolve over time under the influence of spatial gradients and localized interactions.

The one-dimensional general parabolic model studied in this thesis is given by,

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

where,

- $u(x,t)$ is the scalar field of interest (e.g., temperature, chemical concentration),
- $\alpha > 0$ is the diffusion coefficient,
- β is the advection or convection velocity,
- γ is a reaction or absorption coefficient.

The parabolic equation has three fundamental sub-classes of parabolic PDEs depending on the values of β and γ, each corresponding to different types of physical behavior as follows

1.2.1 Diffusion Equation ($\beta = 0$, $\gamma = 0$)

When both the advection and reaction terms vanish, equation (4.1.1) reduces to the traditional diffusion equation,

$$\frac{\partial u}{\partial t}(x,t) = \alpha \frac{\partial^2 u}{\partial x^2}(x,t) \tag{1.2.1}$$

This equation (1.2.1) models pure spreading behavior where the quantity $u(x,t)$ diffuses from regions of high concentration to low concentration. It arises in,

- Heat conduction in solids,

- Molecular diffusion of solutes in fluids,
- Population dispersal in ecological systems,
- Smoothing in image processing.

Solutions exhibit the characteristic behavior of flattening over time - disturbances in the system are smoothed out as they propagate.

1.2.2 Advection-Diffusion Equation ($\gamma = 0$, $\beta \neq 0$)

When a nonzero velocity β is present but no reaction occurs, the PDE becomes the advection-diffusion equation,

$$\frac{\partial u}{\partial t}(x,t) + \beta \frac{\partial u}{\partial x}(x,t) = \alpha \frac{\partial^2 u}{\partial x^2}(x,t) \tag{1.2.2}$$

This equation (1.2.2) models combined transport and spreading, where the quantity is both carried by a flow and diffused due to concentration gradients. It arise in,

- Pollutant transport in rivers or atmospheric flows,
- Heat convection in fluids,
- Drug delivery in bloodstreams,
- Contaminant dispersion in porous media.

The nature of the solution is governed by the Peclet number $P_e = |\frac{\beta}{\alpha}|$, High P_e cases (advection-dominated) create sharp fronts and boundary layers, requiring accurate schemes to avoid artificial diffusion and oscillations.After classifying the parabolic PDE, Now in this work our primary objective is to determine the solution $u(x,t)$ over the domain $[0,L] \times [0,T]$. In some idealized cases, where the coefficients α, β, and γ are constant, the domain is regular, and the boundary and initial conditions are simple and compatible, exact analytical solutions may be derived using traditional methods such as separation of variables, integral transforms, or Green's functions. These solutions are valuable for understanding the qualitative behavior of the system and serve as benchmarks for validating numerical schemes.

However, exact solutions are typically limited to highly simplified scenarios. In most real-world applications, the governing equations involve spatially or temporally varying parameters, nonlinear reaction terms, non-uniform geometries, or complex boundary conditions, which render analytical approaches intractable. Even when exact solutions do exist, they may involve special functions or infinite series that are difficult to evaluate efficiently or converge too slowly to be practical. These challenges make exact solutions inadequate as general-purpose tools for modeling and simulation in science and engineering.

Consequently, it becomes necessary to use on numerical methods to obtain approximate solutions with controllable accuracy. Numerical techniques allow us to solve parabolic equation, under general conditions, variable coefficients, irregular domains, and nonlinearity. Among these methods, finite difference methods (FDM) are particularly attractive due to their conceptual simplicity, ease of implementation, and efficiency on structured grids. traditional finite difference schemes such as Forward-Time Central-Space (FTCS) discussed in the section 1.4.1, Backward-Time Central-Space (BTCS) discussed in the section 1.4.2, and Crank-Nicolson discussed in the section 1.4.3, are widely used. However, they are generally limited to second-order accuracy in space and may suffer from numerical diffusion or instability, especially in advection-dominated problems.

To overcome these limitations and achieve high-order accuracy without significantly increasing computational cost, we focus on solving the equation using compact finite difference methods (CFDM).

These methods achieve higher spatial accuracy by incorporating additional information from neighboring points through compact stencils, resulting in fourth-order accuracy using only three-point spatial approximations. Unlike wide-stencil high-order schemes, CFDMs are efficient, easy to implement near boundaries, and better suited for problems where fine spatial resolution is required but grid refinement is computationally expensive.

In this work, we construct two compact schemes tailored to the general parabolic PDE.The first is an operator-based compact scheme designed for the diffusion equation discussed in the section 2.2. The second is a generalized fourth-order compact scheme applicable to the full advection-diffusion-reaction (ADR) equation discussed in the section 3.1.

The proposed methods are analyzed for stability and convergence and are validated through numerical experiments that demonstrate their superior performance compared to traditional second-order schemes.

1.2.3 Advection-Diffusion-Reaction Equation ($\gamma \neq 0$, $\beta \neq 0$)

The most general case occurs when both advection and reaction mechanisms are active. Equation (4.1.1) then represents a full advection-diffusion-reaction (ADR) model,

$$\frac{\partial u}{\partial t}(x,t) + \beta \frac{\partial u}{\partial x}(x,t) + \gamma u(x,t) = \alpha \frac{\partial^2 u}{\partial x^2}(x,t) \tag{1.2.3}$$

Here, the reaction term $\gamma u(x,t)$ may represent,

- Linear decay ($\gamma > 0$),
- Growth or source terms ($\gamma < 0$),
- Coupling to more complex nonlinear reaction kinetics in extended models.

ADR equations appear in:

- Combustion and flame propagation,
- Biological and ecological dynamics (e.g., predator-prey, disease spread),
- Reactive transport in chemical reactors,
- Electrochemical systems and population models.

These systems are typically more challenging to simulate due to the interaction between transport and localized transformations, often leading to stiff dynamics or pattern formation.

1.3 Important definitions

Finite Difference Method (FDM)

FDM is a numerical technique that transforms a differential equation into a system of algebraic equations by discretizing the continuous domain (such as space or time) into a finite set of grid or mesh points. At these discrete points, derivatives in the original equation are approximated using finite differences-that is, differences between function values at neighboring grid points. In essence, FDM replaces continuous derivatives with discrete difference quotients, typically derived by truncating the Taylor series expansion of a function $u(x,y)$, as shown in equation (1.3.1),
Finite difference methods are categorized as follows:

1. **Explicit methods:** These compute the solution at each spatial point for the current time level t^j directly using known values from previous time levels, involving only simple arithmetic operations.

2. **Implicit methods:** These require solving a system of algebraic equations involving unknown values at the new time level t^{j+1}, often using matrix-based or iterative solvers.

Additionally, depending on how many time levels are involved in the discretization, the methods are classified as:

- **Two-level methods:** Involve values at two consecutive time levels (e.g., t^j and t^{j+1}).
- **Three-level methods:** Use values at three time levels (e.g., t^{j-1}, t^j, and t^{j+1}).

Here, we describe the numerical technique to approximate the derivatives by truncation of Taylor series expansion of a function $u(x,y)$ given in equation (1.3.1),

$$u(x,t) \approx u(x_0,t_0) + \left.\frac{\partial u}{\partial x}\right|_{(x_0,t_0)} (x-x_0) + \left.\frac{\partial u}{\partial t}\right|_{(x_0,y_0)} (t-t_0) + \frac{1}{2!}\left[\left.\frac{\partial^2 u}{\partial x^2}\right|_{(x_0,t_0)} (x-x_0)^2 + 2\left.\frac{\partial^2 u}{\partial x \partial t}\right|_{(x_0,t_0)} (x-x_0)(t-t_0) + \left.\frac{\partial^2 u}{\partial t^2}\right|_{(x_0,t_0)} (t-t_0)^2\right] + \cdots \quad (1.3.1)$$

By using this we can construct the finite difference scheme for PDEs. Now if we have five points in such a way that shown in the Figure (1.3.1),

Let $u(x,t)$ denote the temperature at position $x \in (0,L)$ and time t, along a thin rod of length L, where heat flows with diffusion constant α. The spatial grid points are defined as $x_i = ih$ for $i = 0,1,...,M$, with $x_0 = 0$, $x_M = L$, and $M = \frac{L}{h}$. Time is discretized as $t_j = jk$ for $j = 0,1,...,N$.

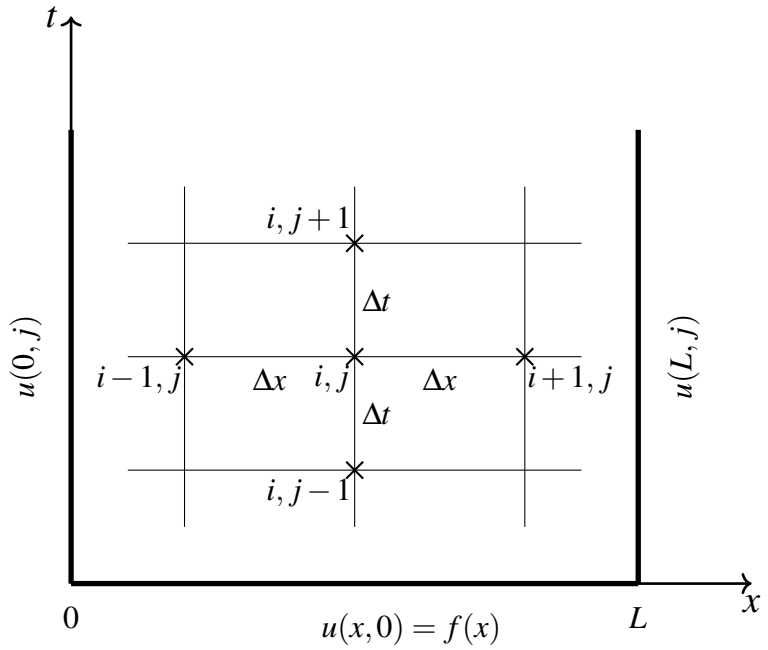


Figure 1.3.1: Five point stencils

- First Order Forward Difference operator for first derivative at (x_i,t_j)

$$\left.\delta_x u\right|_i^j = \frac{u_{i+1}^j - u_i^j}{h} + \mathcal{O}(h) \quad (1.3.2)$$

$$\left.\delta_t u\right|_i^j = \frac{u_i^{j+1} - u_i^j}{k} + \mathcal{O}(k) \quad (1.3.3)$$

- First Order Backward Difference operator for first derivative at (x_i, t_j)

$$\delta_x u\Big|_i^j = \frac{u_i^j - u_{i-1}^j}{h} + \mathscr{O}(h) \tag{1.3.4}$$

$$\delta_t u\Big|_i^j = \frac{u_i^j - u_i^{j-1}}{k} + \mathscr{O}(k) \tag{1.3.5}$$

- Second Order Central Difference operator for first derivative at (x_i, t_j)

$$\delta_x u\Big|_i^j = \frac{u_{i+1}^j - u_{i-1}^j}{2(h)} + \mathscr{O}(h)^2 \tag{1.3.6}$$

$$\delta_t u\Big|_i^j = \frac{u_i^{j+1} - u_i^{j-1}}{2(k)} + \mathscr{O}(k)^2 \tag{1.3.7}$$

- Second Order Central Difference operator for second derivative at (x_i, t_j)

$$\delta_{xx} u\Big|_i^j = \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{h^2} + \mathscr{O}(h)^2 \tag{1.3.8}$$

$$\delta_{tt} u\Big|_i^j = \frac{u_i^{j+1} - 2u_i^j + u_i^{j-1}}{k^2} + \mathscr{O}(k)^2 \tag{1.3.9}$$

Three traditional finite difference methods are commonly used: Forward-Time Central-Space method (see 1.4.1), Backward-Time Central-Space method (see 1.4.2), and the Crank-Nicolson method (see 1.4.3). These methods differ in how they discretize the time derivative and are categorized as explicit, implicit, and Crank-Nicolson scheme, respectively.

Stability of Numerical scheme

A finite difference scheme is defined to be stable if the numerical solution stays bounded as the number of time steps grow (or as the grid gets refined), for a fixed time interval. i.e., If small perturbations in the data or round-off errors do not increases as the computation proceeds, the scheme is stable or If errors grow exponentially with time, the scheme is unstable.

Why stability is very important for any numerical method?
When solving PDEs (for instance diffusion equation), we iteratively calculate the solution forward in time. If the method is unstable, then even a small error at one time step can grow fast enough to render the entire solution completely meaningless.

Truncation Error

Local truncation error (LTE) is the error associated with a single time step in the finite difference scheme when the exact solution is substituted into the finite difference scheme. It is a measure of how close the finite difference approximation is to the original differential equation. Let us consider diffusion equation

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$$

and a difference scheme $F(u)$, the truncation error is:

$$\text{LTE} = F(u(x_i, t_j)) - \left(\frac{\partial u}{\partial t} - \alpha \frac{\partial^2 u}{\partial x^2} \right) \tag{1.3.10}$$

Consistency

Consistency refers to how accurately a finite difference scheme approximates the original PDEs. A numerical method is said to be consistent with the PDE if the local truncation error tends to zero as the grid spacings h (in space) and k (in time) approach zero. Mathematically, this is expressed as,

$$\lim_{h\to 0,\ k\to 0} \tau_i(h,k) = 0 \tag{1.3.11}$$

where $\tau_i(h,k)$ denotes the local truncation error at the i^{th} grid point.

Numerical Convergence

A finite difference method is said to be convergent if the numerical solution obtained from the scheme approaches the exact solution of the PDEs as the grid spacings $h \to 0$ and $k \to 0$. That is, the global error between the numerical solution and the exact solution tends to zero as the discretization becomes finer. i.e.,

$$\lim_{h\to 0,\ k\to 0} \left| u(x_i,t_j) - U_i^j \right| = 0 \tag{1.3.12}$$

for all grid points (x_i,t_j) in the domain.

Statement: Lax Equivalence Theorem

The Lax Equivalence Theorem is a fundamental result in numerical analysis of finite difference methods for linear initial value problems.
For a well posed linear initial value problem and a consistent finite difference approximation to it, stability is a necessary and sufficient condition for convergence.
In simple terms:

- If a method is consistent (i.e., the finite difference approximation correctly represents the PDE), and
- If the method is stable (i.e., errors do not grow uncontrollably in the iteration),
- Then the method will converge (i.e., the numerical solution approaches the exact solution as the grid is refined).

Order of Convergence

The order of convergence of a finite difference method describes how fast the numerical solution approaches the exact solution as the grid spacings h (space step) and k (time step) tend to zero. i.e.,

$$|u(x_i,t_j) - U_i^j| = \mathscr{O}(h^p + k^q) \tag{1.3.13}$$

This means:

- The error decreases proportionally to h^p as the spatial grid size h decreases.
- The error decreases proportionally to k^q as the time step k decreases.

A higher order of convergence implies a more accurate method for the same grid size. The values of p and q are typically determined from truncation error analysis.
For the Crank-Nicolson method, we derived in equation (1.4.15),

$$\tau_i = \mathscr{O}(k^2 + h^2) \tag{1.3.14}$$

So, Order of convergence in time and space is two.

In this work we are computing the numerical error with respect to the exact solution by applying the methods to a range of parameter values including the grid sizes h and k, and the coefficients α, β, and γ. The numerical error was calculated using the maximum absolute error, defined by (1.3.15),

$$E = \max_{i,j} \left| u(x_i, t_j)_{\text{exact}} - u_i^j \right|, \tag{1.3.15}$$

where $u(x_i,t_j)_{\text{exact}}$ is the exact solution and u_i^j is the numerical approximation at the grid point (x_i,t_j). In addition, we assessed the computational order of convergence for each scheme using the formula,

$$\mathscr{O} = \frac{\log\left(\frac{E_1}{E_2}\right)}{\log\left(\frac{h_1}{h_2}\right)}, \tag{1.3.16}$$

where E_1 and E_2 are the maximum errors corresponding to spatial step sizes h_1 and h_2, respectively.

1.4 Finite Difference Schemes for Parabolic PDE

In this section, we discuss the traditional finite difference method (FDMs) that are used to solve PDEs numerically. By discretising time and space and approximating derivatives using finite differences, these techniques offer workable answers to issues that would otherwise be challenging to resolve analytically. We focus on three popular schemes as follows,

1.4.1 Forward-Time Central-Space scheme (Explicit Method)

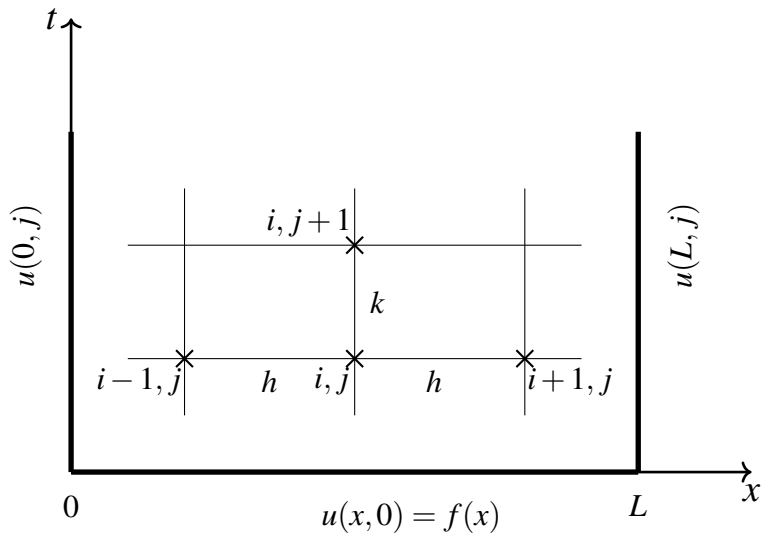


Figure 1.4.1: Forward-Difference Stencils

The scheme given in equation (1.4.1) is an explicit method (because the scheme is for direct solution and we are not using the previous time level to calculate the solution) or two level difference method scheme or smith Method.

$$\frac{u_i^{j+1} - u_i^j}{k} = \alpha \left[\frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{h^2} \right]$$

taking $\lambda = \frac{\alpha k}{h^2}$

$$u_i^{j+1} = u_i^j + \lambda \left[u_{i+1}^j - 2u_i^j + u_{j+1}^j \right] \tag{1.4.1}$$

If $\lambda = \frac{1}{2}$ in this scheme (1.4.1), is called Bender Smith method i.e $u_i^{j+1} = \frac{1}{2} \left[u_{i+1}^j + u_{i-1}^j \right]$.

The Explicit scheme (1.4.1) is conditionally stable for $0 < \lambda \leq \frac{1}{2}$.
Now, the Von-Neumann stability condition helps to ensure that errors do not grow uncontrollably as the computation progresses.The explicit finite difference scheme is given by equation (1.4.1), we are using m and n for representing the discretization of space and time respectively.

$$u_m^{n+1} = u_m^n + \lambda \left[u_{m+1}^n - 2u_m^n + u_{m-1}^n \right] \tag{1.4.2}$$

Let u_m^nbe the numerical solution of the difference equation, the

$$u_m^n = u_m^n + \varepsilon_m^n$$

where ε_m^n is error due to rounding off or computational mistake.

$$\varepsilon_m^{n+1} = \varepsilon_m^n + \lambda \left[\varepsilon_{m+1}^n - 2\varepsilon_m^n + \varepsilon_{m-1}^n \right]$$

$$\varepsilon_m^{n+1} = [1 - 2\lambda] \varepsilon_m^n + \lambda \left[\varepsilon_{m+1}^n - \varepsilon_{m-1}^n \right] \tag{1.4.3}$$

we assume a group of error in introduced at the initial level $t = 0$ which propagates forward in accordance with above equation. For a differential equation with constant coefficient, the error may be expanded in a finite Fourier series, The error can be written as

$$\varepsilon_m^0 = \sum_j A_j \xi^0 \exp u_m^n = u_m^n + \varepsilon_m^n$$

where ε_m^n is error due to rounding off or computational mistakes $(i\beta mh)$, and $\beta = \frac{j\pi}{b-a}$

$$\varepsilon_m^n = \sum_j A_j \xi^n(j) \exp(i\beta mh)$$

It is for a Linear Combination, the it is true for one term then

$$\varepsilon_m^n = A\xi^n \exp(i\beta mh) \tag{1.4.4}$$

here if $|\xi| > 1$,then it will be large. but $|\xi| \leq 1$,converge.

$$\begin{aligned} A\xi^{n+1} \exp(i\beta mh) = A\xi^n \exp(i\beta mh) + \lambda \left[A\xi^n \exp(i\beta(m+1)h) \right. \\ \left. - 2A\xi^n \exp(i\beta mh) + A\xi^n \exp(i\beta(m-1)h) \right] \end{aligned}$$

$$\frac{\xi^{n+1}}{\xi^n} \left[A \exp(i\beta mh) \right] = (1 - 2\lambda) A \exp(i\beta mh) + \lambda A \left[\exp i\beta h(m-1) + \exp i\beta h(m+1) \right]$$

$$\frac{\xi^{n+1}}{\xi^n} \left[A \exp(i\beta mh) \right] = (1 - 2\lambda) \exp(i\beta mh) + \lambda \left[\exp(i\beta mh) . \exp(-i\beta h) + \exp(i\beta mh) \exp(i\beta h) \right]$$

$$\frac{\xi^{n+1}}{\xi^n} = 1 - 2\lambda(1 - \cos\theta)$$

where $\theta = \beta h$ and taking $\frac{\xi^{n+1}}{\xi^n} = \xi$

$$\xi = 1 - 4\lambda \sin^2 \frac{\theta}{2}$$

Accordin to the condition -

$$\left|1-4\lambda\sin^2\frac{\theta}{2}\right| \leq 1$$

$$-1 \leq 1-4\lambda\sin^2\frac{\theta}{2} \leq 1$$

if $\lambda > 0$ then $-1 \leq 1-4\lambda\sin^2\frac{\theta}{2}$, i.e. it is true for only $\lambda \leq \frac{1}{2}$ i.e. $\frac{\alpha k}{h^2} \leq \frac{1}{2}$, So by this we can conclude that it is a conditionally stable numerical scheme.

1.4.2 Backward-time central-space scheme (Implicit Method)

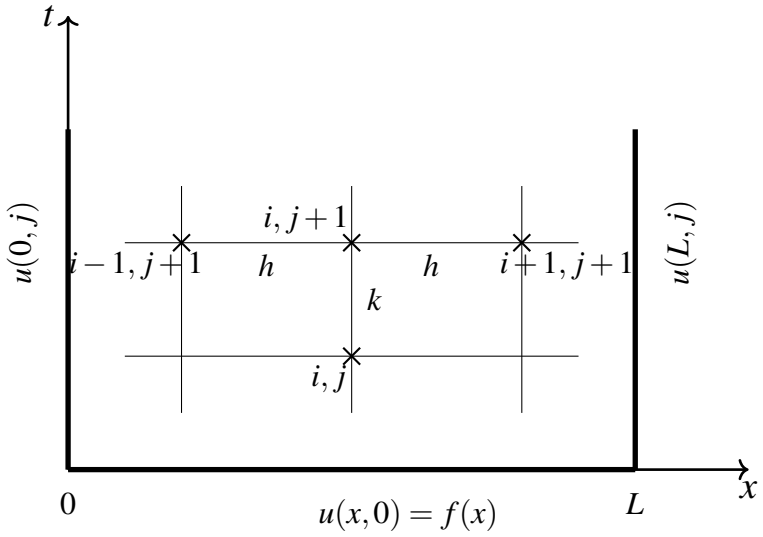


Figure 1.4.2: The Implicit stencils

The scheme given in equation (1.4.5) is an implicit method (because the scheme is a recursion formula for system of equation and we are using the previous time level to calculate the solution). .

$$\frac{u_i^{j+1}-u_i^j}{k} = \alpha\left[\frac{u_{i+1}^{j+1}-2u_i^{j+1}+u_{i-1}^{j+1}}{h^2}\right]$$

$$u_i^{j+1} = \frac{\alpha k}{h^2} + \left[u_{i+1}^{j+1}-2u_i^{j+1}+u_{i-1}^{j+1}\right] - u_i^j$$

taking $\frac{\alpha k}{h^2} = \lambda$

$$u_i^j = -\lambda u_{i-1}^{j+1} + [1+2\lambda]\,u_i^{j+1} - \lambda u_{i+1}^{j+1} \tag{1.4.5}$$

It is unconditionally stable, Now the Von-Neumann stability condition helps to ensure that errors do not grow uncontrollably as the computation progresses, we are using m and n for representing the discretization of space and time respectively. The backward times difference scheme or implicit scheme is given by equation (1.4.5),

$$u_m^n = -\lambda u_{m-1}^{n+1} + [1+2\lambda]\,u_m^{n+1} - \lambda u_{m+1}^{n+1} \tag{1.4.6}$$

Let u_m^nbe the numerical solution of the difference equation, the

$$u_m^n = u_m^n + \varepsilon_m^n$$

where ε_m^n is error due to rounding off or computational mistake.

$$\varepsilon_m^n = -\lambda\varepsilon_{m-1}^{n+1} + [1+2\lambda]\,\varepsilon_m^{n+1} - \lambda\varepsilon_{m+1}^{n+1} \tag{1.4.7}$$

we assume a group of error at the initial time level $t = 0$ which moves forward in accordance with equation (1.4.7). In a finite Fourier series the error may be expanded, So the error term can be written as

$$\varepsilon_m^0 = \sum_j A_j \xi^0 \exp(i\beta mh)$$

where $\beta = \frac{j\pi}{b-a}$

$$\varepsilon_m^n = \sum_j A_j \xi^n(j) \exp(i\beta mh)$$

It is for a Linear Combination, the it is true for one term then

$$\varepsilon_m^n = A\xi^n \exp(i\beta mh)$$

here if $|\xi| > 1$,then it will be large. but $|\xi| \leq 1$,converge.

$$A\xi^n \exp(i\beta mh) = -\lambda A\xi^{n+1} \exp(i\beta(m-1)h) + [1+2\lambda]A\xi^{n+1} \exp(i\beta mh) - \lambda A\xi^{n+1} \exp(i\beta(m+1)h)$$

$$A\xi^n \exp(i\beta mh) = -\lambda A\xi^{n+1} \exp(i\beta hm) \exp(-i\beta h) + [1+2\lambda] \times A\xi^{n+1} \exp(i\beta mh) - \lambda A\xi^{n+1} \exp(i\beta hm) \exp(i\beta h)$$

$$\frac{\xi^n}{\xi^{n+1}} = -2\lambda \cos\theta + 1 + 2\lambda$$

where $\theta = \beta h$ and taking $\frac{\xi^{n+1}}{\xi^n} = \xi$, According to the condition,

$$\left| 1 + 4\lambda \sin^2 \frac{\theta}{2} \right| \geq 1$$

$$-1 \leq 1 + 4\lambda \sin^2 \frac{\theta}{2}$$

This condition is true for every value of λ, so it is unconditionally stable.

1.4.3 Crank-Nicolson scheme

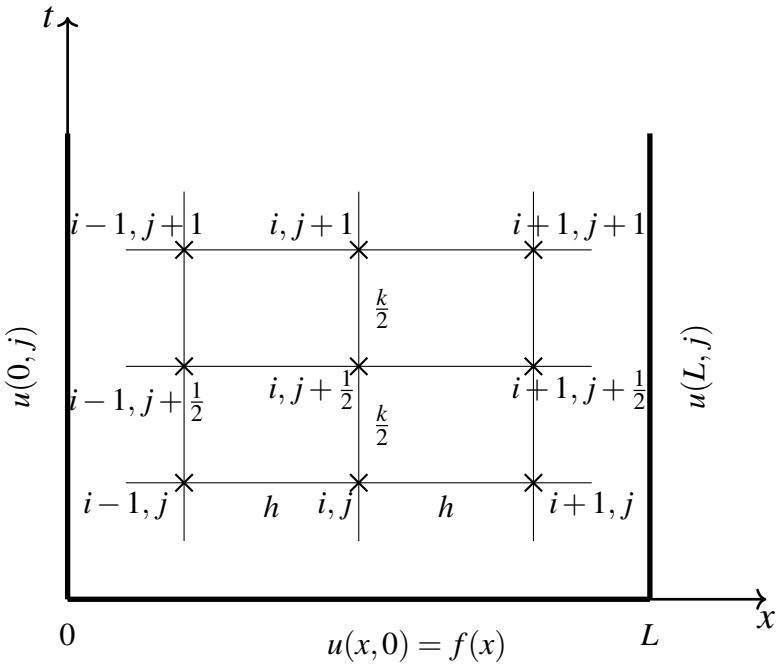


Figure 1.4.3: The Crank-Nicolson stencils

This scheme given in equation (1.4.8) is called Crank Nicolson Method.

$$\frac{u_i^{j+1}-u_i^j}{k}=\frac{\alpha}{2}\left[\frac{u_{i+1}^{j+1}-2u_i^{j+1}+u_{i-1}^{j+1}}{h^2}+\frac{u_{i+1}^{j}-2u_i^{j}+u_{i-1}^{j}}{h^2}\right]$$

$$u_i^{j+1}=\frac{\alpha k}{2h^2}\left[u_{i+1}^{j+1}-2u_i^{j+1}+u_{i-1}^{j+1}+u_{i+1}^{j}-2u_i^{j}+u_{i-1}^{j}\right]+u_i^j$$

where, $\lambda=\frac{\alpha k}{h^2}$

$$-u_{i-1}^{j+1}\left(\frac{\lambda}{2}\right)+u_i^{j+1}(1+\lambda)-u_{i+1}^{j+1}\left(\frac{\lambda}{2}\right)=u_{i-1}^{j}\left(\frac{\lambda}{2}\right)+u_i^{j}(1-\lambda)+u_{i+1}^{j}\left(\frac{\lambda}{2}\right). \tag{1.4.8}$$

It is unconditionally stable, Now, the Von-Neumann stability condition helps to ensure that errors do not grow uncontrollably as the computation progresses, we are using m and n for representing the discretization of space and time respectively. The Crank-Nicolson scheme is given by equation (1.4.8),

$$u_m^{n+1}=u_m^n+\frac{\lambda}{2}\left(u_{m+1}^{n+1}-2u_m^{n+1}+u_{m-1}^{n+1}+u_{m+1}^{n}-2u_m^{n}+u_{m-1}^{n}\right) \tag{1.4.9}$$

Let the numerical solution contain an error:

$$u_m^n=u_m^{\text{true},n}+\varepsilon_m^n$$

Then the error also satisfies the same Crank-Nicolson equation

$$\varepsilon_m^{n+1}=\varepsilon_m^n+\frac{\lambda}{2}\left(\varepsilon_{m+1}^{n+1}-2\varepsilon_m^{n+1}+\varepsilon_{m-1}^{n+1}+\varepsilon_{m+1}^{n}-2\varepsilon_m^{n}+\varepsilon_{m-1}^{n}\right) \tag{1.4.10}$$

Assume error in the form of Fourier modes:

$$\varepsilon_m^n=\xi^n e^{i\beta mh},\quad \text{where } h=\Delta x$$

$$\xi^{n+1}e^{i\beta mh}=\xi^n e^{i\beta mh}+\frac{\lambda}{2}\left[\xi^{n+1}(e^{i\beta h(m+1)}-2e^{i\beta hm}+e^{i\beta h(m-1)})+\xi^n(e^{i\beta h(m+1)}-2e^{i\beta hm}+e^{i\beta h(m-1)})\right]$$

$$\xi^{n+1}=\xi^n+\frac{\lambda}{2}\left[\xi^{n+1}(e^{i\beta h}+e^{-i\beta h}-2)+\xi^n(e^{i\beta h}+e^{-i\beta h}-2)\right]$$

Note: $e^{i\beta h}+e^{-i\beta h}=2\cos(\beta h)$

$$\xi^{n+1}=\xi^n+\lambda\left[\xi^{n+1}(\cos\theta-1)+\xi^n(\cos\theta-1)\right]\quad \text{where } \theta=\beta h$$

$$\xi^{n+1}\left[1-\lambda(1-\cos\theta)\right]=\xi^n\left[1+\lambda(\cos\theta-1)\right]$$

Thus

$$\frac{\xi^{n+1}}{\xi^n}=\frac{1-\lambda(1-\cos\theta)}{1+\lambda(1-\cos\theta)}$$

Let this ratio be the amplification factor G, then

$$G=\frac{1-\lambda(1-\cos\theta)}{1+\lambda(1-\cos\theta)}$$

$$|G|=\left|\frac{1-\lambda(1-\cos\theta)}{1+\lambda(1-\cos\theta)}\right|$$

$\cos\theta\in[-1,1]\Rightarrow 1-\cos\theta\in[0,2]$, $\lambda>0$, Then

$$0<|G|\leq 1\quad\Rightarrow\quad \text{Crank-Nicolson is unconditionally stable}$$

The scheme derived in section 1.4.3 is unconditionally stable.

We now move on to analysis of the Crank-Nicolson scheme's truncation error. The local truncation error will be measured by using Taylor series and we replace the exact solution with finite difference scheme. In addition to verifying the scheme 2nd order accuracy in temporal and space. The Crank-Nicolson scheme is given by section 1.4.3,

$$\frac{u_i^{j+1}-u_i^j}{k}=\frac{\alpha}{2}\left(\frac{u_{i+1}^j-2u_i^j+u_{i-1}^j}{h^2}+\frac{u_{i+1}^{j+1}-2u_i^{j+1}+u_{i-1}^{j+1}}{h^2}\right) \tag{1.4.11}$$

Let us consider the the exact solution of diffusion equation is given by $u(x,t)$, we on substituting into this scheme and expand both sides using Taylor series to find the truncation error.

$$u_i^{j+1}=u(x_i,t_j+k)=u+ku_t+\frac{k^2}{2}u_{tt}+\mathscr{O}(k^3)$$

$$\Rightarrow \frac{u_i^{j+1}-u_i^j}{k}=u_t+\frac{k}{2}u_{tt}+\mathscr{O}(k^2) \tag{1.4.12}$$

At time level j,

$$\frac{u_{i+1}^j-2u_i^j+u_{i-1}^j}{h^2}=u_{xx}+\frac{h^2}{12}u_{xxxx}+\mathscr{O}(h^4)$$

At time level $j+1$,

$$\frac{u_{i+1}^{j+1}-2u_i^{j+1}+u_{i-1}^{j+1}}{h^2}=u_{xx}+\frac{h^2}{12}u_{xxxx}+\mathscr{O}(h^4)$$

Average of both:

$$\text{RHS}=\alpha\left(u_{xx}+\frac{h^2}{12}u_{xxxx}\right)+\mathscr{O}(h^4) \tag{1.4.13}$$

Now by equation (1.4.12) and equation (1.4.13), we get,

$$u_t+\frac{k}{2}u_{tt}=\alpha\left(u_{xx}+\frac{h^2}{12}u_{xxxx}\right)+\mathscr{O}(k^2,h^4) \tag{1.4.14}$$

But from the PDE $u_t=\alpha u_{xx}\Rightarrow u_{tt}=\alpha^2 u_{xxxx}$

$$\Rightarrow u_t+\frac{k}{2}\alpha^2 u_{xxxx}=\alpha u_{xx}+\frac{\alpha h^2}{12}u_{xxxx}$$

$$\tau_i=\left(\frac{k}{2}\alpha^2-\frac{\alpha h^2}{12}\right)u_{xxxx}+\mathscr{O}(k^2,h^4) \tag{1.4.15}$$

Which implies

$$\tau_i=\mathscr{O}(k^2)+\mathscr{O}(h^2)$$

Therefore, the Crank-Nicolson defined in section method is second-order accurate in both time and space. $k\to 0$, with order $\mathscr{O}(k^2+h^2)$ i.e. it is second-order accurate in both time and space.

1.5 Motivation

In the previous sections, we have introduced the fundamental concepts of finite difference methods (FDMs) for solving parabolic PDEs, including traditional schemes such as the Forward-Time Central-Space (FTCS) in section 1.4.1, Backward-Time Central-Space (BTCS) in section 1.4.2, and the Crank-Nicolson method in section (1.4.3). These methods have been widely used due to their simplicity and

structured formulation on uniform grids.

However, traditional FDMs are subject to several important limitations. like second-order accuracy in space when using standard three-point central difference formulas. To handle limitations, compact finite difference methods (CFDMs) are developed.

"Compact Finite Difference Methods (CFDMs) are high-order finite difference schemes that use implicit, compact stencils involving only neighboring points to approximate derivatives with greater accuracy, typically fourth order or higher, while maintaining computational efficiency."

The need for CFDMs arises from the demand for high-resolution solutions in problems where traditional FDMs fail to deliver the desired accuracy without excessive grid refinement. In applications involving diffusion, advection-diffusion, or advection-diffusion-reaction (ADR) equations, it is crucial to resolve small-scale structures and sharp gradients efficiently. CFDMs offer a promising solution by providing fourth-order accuracy while retaining a compact form, making them particularly attractive for time-dependent and multiscale PDEs.
In the subsequent chapters, two distinct CFDM approaches for solving one-dimensional parabolic PDEs. In Chapter (2), we present a compact operator-based scheme, which constructs a fourth-order spatial discretization using a differential operator applied to the diffusion equation. In Chapter (3), we introduce a generalized compact finite difference method, which derives high-order approximations directly using Taylor series expansions of the first and second derivatives, enabling its application to ADR equations. These two schemes demonstrate the flexibility and power of CFDMs in addressing the limitations of classical finite difference approaches.

1.6 Literature Survey

The numerical treatment of one-dimensional parabolic PDEs, particularly diffusion and advection-diffusion-reaction (ADR) equations, plays a pivotal role in simulating physical processes such as heat transfer, mass transport, and fluid flow. traditional finite difference methods (FDM), including explicit, implicit, and Crank-Nicolson schemes, are widely used for such problems. However, their limited order of accuracy and inefficiencies, especially in convection-dominated scenarios, motivate the development of higher-order schemes. Compact finite difference methods (CFDM) have emerged as an efficient alternative, providing enhanced accuracy using a smaller stencil and achieving higher convergence rates while maintaining computational simplicity.

Dehghan [1] developed and analyzed weighted finite difference schemes for the one-dimensional advection-diffusion equation using constant coefficients. His work employed the modified equivalent partial differential equation (MEPDE) approach, which offers a systematic way to derive and assess the accuracy of numerical schemes by relating them to the underlying continuous models. This framework revealed that new weighted schemes could eliminate numerical diffusion while outperforming conventional two-level approximations in both accuracy and computational efficiency. Earlier, Spalding [3] contributed to this area by formulating novel finite difference approximations that treat convective and diffusive terms jointly rather than separately. He showed that upwind schemes were advantageous for problems with large Peclet numbers and emphasized that better analogues arise from approximating the full differential operator rather than its individual terms.

Building upon these foundational works, Dehghan extended his methods to the three-dimensional advection-diffusion equation [4], again employing the MEPDE technique. His analysis underscored the ability of high-order explicit and implicit schemes to maintain stability and accuracy, even in complex multi-dimensional domains. The study demonstrated that the combination of higher-order accuracy and numerical robustness could be achieved by careful design, further justifying the transition to compact schemes.

One of the most relevant works in the direction of CFDM is by Mohebbi and Dehghan [19], who

proposed a fourth-order compact finite difference method for solving the one-dimensional heat and advection-diffusion equations. They coupled the spatial discretization with a cubic spline collocation method for time integration, resulting in a method that was not only fourth-order accurate in space and time but also unconditionally stable. This paper forms a theoretical backbone for the present study, which employs a fourth-order CFDM in space and Crank-Nicolson in time.

Chemeda et al. [5] further advanced this methodology by solving the nonlinear Fisher's equation using a fourth-order CFDM combined with Richardson extrapolation, thus elevating spatial accuracy to sixth order. The Crank-Nicolson scheme was used for temporal discretization, while the nonlinear term was linearized using a lagging technique. Their findings confirmed unconditional stability and significant improvements in both accuracy and efficiency, making their method particularly suitable for nonlinear parabolic PDEs.

In terms of rigorous analysis, Sun and Zhang [26] presented a linearized compact difference scheme for a class of nonlinear delay PDEs. They established unconditional stability and convergence in the L_∞-norm, with an accuracy order of $O(\tau^2 + h^4)$. Their results contribute significantly to the theoretical underpinnings of CFDM, especially for time-dependent nonlinear systems with delay effects, and validate the convergence behavior observed in high-order schemes like those used in the present study.

Another notable contribution comes from Roul and Kumari [28], who proposed a nonuniform mesh CFDM to solve singular boundary value problems in electrohydrodynamic flow (EHF). By constructing a graded mesh near the singular point and adapting the compact scheme accordingly, they achieved fourth-order spatial accuracy even in the presence of rational nonlinearities and singularities. Their study demonstrates the adaptability of CFDM to complex geometries and boundary behaviors, aligning well with the proposed future extension of this work to variable coefficient PDEs.

In the domain of financial mathematics, Roul et al. [29] applied a high-order compact finite difference method to the generalized Black-Scholes equation. Their approach involved a semi-discrete Crank-Nicolson scheme in time followed by CFDM in space, resulting in highly accurate solutions that matched exact benchmarks. This further illustrates the versatility and strength of CFDM in handling both physical and financial parabolic PDEs.

A comprehensive foundation for CFDM is provided in the work of Spotz [30], who developed a general class of high-order compact (HOC) finite difference schemes that achieve higher accuracy while using a compact stencil. His approach leverages the governing differential equations to cancel out leading-order truncation error terms in central difference schemes. Spotz first applied these methods to steady convection-diffusion problems and later extended them to solve the streamfunction-vorticity formulation of the steady 2D incompressible Navier-Stokes equations. Even on coarser grids, the HOC schemes achieved results comparable to those from standard methods on finer grids, and they exhibited reduced spurious oscillations. Importantly, he extended the HOC framework to nonuniform grids through coordinate transformation techniques and further generalized the schemes to transient problems, nonlinear Poisson equations, and 3D linear Poisson problems. Despite higher condition numbers, Spotz demonstrated that certain iterative solvers converged faster for HOC systems. He also analyzed the implications of grid mapping accuracy and proposed strategies for computing compact grid metrics when grid generation is governed by differential equations. His findings confirm that HOC schemes, especially when combined with implicit time integration methods like Crank-Nicolson, offer an unconditionally stable and computationally advantageous alternative for complex PDE problems.

In summary, these studies establish compact finite difference methods as superior alternatives to traditional schemes, especially when combined with time integration methods like Crank-Nicolson. The literature confirms that CFDM offers improved accuracy, stability, and computational efficiency, making it a robust tool for solving one-dimensional parabolic PDEs. The present work extends this foundation by implementing a fourth-order CFDM combined with the Crank-Nicolson scheme in Python for solving diffusion and ADR equations with constant coefficients. The stability of the scheme is analyzed via Von Neumann analysis, and its performance is validated through comparative studies. Future work will aim to generalize the method for PDEs with variable coefficients, inspired by the flexibility and success of nonuniform CFDM and HOC schemes seen in the literature.

Higher-order numerical schemes are critical in finite difference methods because they offer more accuracy with fewer grid points, which improves computational efficiency. The higher the order, the more rapidly the truncation error is reduced when the grid is made finer. For example, decreasing the grid size by half in a second-order scheme decreases the error by four, whereas a fourth-order scheme decreases it by sixteen. This implies that higher-order schemes can match the accuracy of a finer grid using coarser grids, which is memory and computationally cheaper. They are particularly useful for more accurate capture of smooth solution features, like waves or sharp gradients, without too much refinement. Even though fourth-order schemes will have more complicated formulas and marginally higher computational effort per step, their efficiency and accuracy in the long run usually make them more desirable than lower-order schemes.

Now, we present in the following chapters several fourth order finite difference techniques, and one class, compact finite difference schemes. These schemes have been constructed to achieve high accuracy using small stencils. Thus, they are very efficient and can be used to obtain approximate solutions for a variety of PDEs, such as the diffusion equation, the advection equation, the advection-diffusion equation, and the reaction-diffusion equation. Every type of equation has unique numeric challenges, and compact schemes will present powerful tools for solving them with high precision and stability.

Chapter 2

Linearized Compact Finite Difference scheme (Operator Method)

In this chapter, we discuss the Compact Finite Difference Method (CFDM), which is a method of achieving higher-order accuracy using compact stencils. Unlike the traditional finite difference methods that need wide stencils for higher order approximations, CFDM accomplishes the same or more accuracy using fewer grid points, by re substituting the higher derivatives in differential equations.

In this chapter particularly we discuss the fourth order CFDM using operator, the operator is derived from the higher order truncation of Taylor Series of the function.

2.1 A fourth-order compact finite difference scheme

Higher-order compact finite difference schemes begin with traditional approximations and enhance accuracy by constructing compact representations of derivatives through algebraic manipulations and Taylor series expansion. We first construct the compact difference operator for the spatial term to achieve fourth-order spatial accuracy. This operator is obtained independently of the time discretization and results in a linear system involving second derivatives at neighboring points.
After establishing the spatial discretization, we then apply the Crank-Nicolson scheme for time integration. The Crank-Nicolson method is a second-order, unconditionally stable time-stepping scheme that averages the temporal derivative at two successive time levels. By combining this time discretization with the compact spatial operator, we obtain a fully discrete scheme that is fourth-order accurate in space.

2.1.1 Derivation of Operator

In this section We derive the Operator to approximated the second order partial derivative to get fourth order compact difference scheme, the idea of derivation is take from the Lemma (2.1.1) and it is based on truncation of Taylor series discussed in [26]

Lemma 2.1

Suppose $S(x)$ is a smooth function belonging to $C^6[0,1]$. Then the following relation holds

$$\frac{1}{12}\left[S''(x_{i+1})+10S''(x_i)+S''(x_{i-1})\right]-\frac{1}{h^2}\left[S(x_{i+1})-2S(x_i)+S(x_{i-1})\right]=\frac{h^4}{240}S^{(6)}(\xi_i), \qquad (2.1.1)$$

where $\xi_i \in (x_{i-1}, x_{i+1})$.

Proof

Now, starting from the Taylor series expansion around x_i for a sufficiently smooth function $S(x)$:

$$S(x_{i+1}) = S(x_i) + hS'(x_i) + \frac{h^2}{2}S''(x_i) + \frac{h^3}{6}S^{(3)}(x_i) + \frac{h^4}{24}S^{(4)}(x_i) + \frac{h^5}{120}S^{(5)}(x_i) + \frac{h^6}{120}\int_0^1 S^{(6)}(x_i + vh)(1-v)^5\,dv$$

$$S(x_{i-1}) = S(x_i) - hS'(x_i) + \frac{h^2}{2}S''(x_i) - \frac{h^3}{6}S^{(3)}(x_i) + \frac{h^4}{24}S^{(4)}(x_i) - \frac{h^5}{120}S^{(5)}(x_i) + \frac{h^6}{120}\int_0^1 S^{(6)}(x_i - vh)(1-v)^5\,dv.$$

Adding these two expressions gives

$$\frac{1}{h^2}\left[S(x_{i+1}) - 2S(x_i) + S(x_{i-1})\right] = S''(x_i) + \frac{h^2}{12}S^{(4)}(x_i) + \frac{h^4}{120}\int_0^1 \left[S^{(6)}(x_i + vh) + S^{(6)}(x_i - vh)\right](1-v)^5\,dv. \tag{2.1.2}$$

expanding $S''(x)$ at the neighboring points

$$S''(x_{i+1}) = S''(x_i) + hS^{(3)}(x_i) + \frac{h^2}{2}S^{(4)}(x_i) + \frac{h^3}{6}S^{(5)}(x_i) + \frac{h^4}{6}\int_0^1 S^{(6)}(x_i + vh)(1-v)^3\,dv$$

$$S''(x_{i-1}) = S''(x_i) - hS^{(3)}(x_i) + \frac{h^2}{2}S^{(4)}(x_i) - \frac{h^3}{6}S^{(5)}(x_i) + \frac{h^4}{6}\int_0^1 S^{(6)}(x_i - vh)(1-v)^3\,dv$$

$$\frac{1}{12}\left[S''(x_{i+1}) + 10S''(x_i) + S''(x_{i-1})\right] = S''(x_i) + \frac{h^2}{12}S^{(4)}(x_i) + \frac{h^4}{72}\int_0^1 \left[S^{(6)}(x_i + vh) + S^{(6)}(x_i - vh)\right](1-v)^3\,dv \tag{2.1.3}$$

Subtracting equation (2.1.2) from (2.1.3) leads to:

$$\frac{1}{12}\left[S''(x_{i+1}) + 10S''(x_i) + S''(x_{i-1})\right] - \frac{1}{h^2}\left[S(x_{i+1}) - 2S(x_i) + S(x_{i-1})\right] = \frac{h^4}{360}\int_0^1 \left[S^{(6)}(x_i + vh) + S^{(6)}(x_i - vh)\right](1-v)^3\left(5 - 3(1-v)^2\right)dv.$$

Applying the mean value theorem for integrals yields:

$$= \frac{h^4}{240}S^{(6)}(\xi_i), \quad \text{for some } \xi_i \in (x_{i-1}, x_{i+1}).$$

Let us consider the derivative as operator of the form,

$$\mathscr{Z}(S_i) = \frac{1}{12}\left[S''(x_{i+1}) + 10S''(x_i) + S''(x_{i-1})\right] \tag{2.1.4}$$

2.2 The Proposed Method

Let us consider the diffusion equation model defined in equation (1.1.3), We define a uniform discretization of the computational domain $\Omega = (x_{\min}, x_{\max}) \times (0,T)$. Domain Ω is divided into a grid of points (x_i, t_j), where:

- Spatial nodes: $x_i = x_{\min} + ih$, for $i = 0,1,2,\ldots,m$.
- Temporal nodes: $t_j = jk$, for $j = 0,1,2,\ldots,n$.

Applying the Crank-Nicolson scheme (1.4.8) to the system discretized with the compact finite difference operator (2.1.4), the diffusion equation (1.1.3) is time-discretized as:

$$\delta_k u_i^{j+\frac{1}{2}} = \frac{1}{2}\alpha\left(\frac{\partial^2 u}{\partial x^2}\Big|_i^{j+1} + \frac{\partial^2 u}{\partial x^2}\Big|_i^{j}\right) \tag{2.2.1}$$

Now on applying the operator $\mathscr{Z}$, from equation (2.1.4) to equation (2.2.1) We get,

$$\mathscr{Z}\left(\delta_k u_i^{j+\frac{1}{2}}\right) = \frac{1}{2}\alpha\left(\mathscr{Z}\frac{\partial^2 u}{\partial x^2}\Big|_i^{j+1} + \mathscr{Z}\frac{\partial^2 u}{\partial x^2}\Big|_i^{j}\right) \tag{2.2.2}$$

$$\mathscr{Z}\left(\frac{u_i^{j+1}-u_i^j}{k}\right) = \frac{\alpha}{2}\left(\mathscr{Z}\left(\frac{u_{i+1}^{j+1}-2u_i^{j+1}+u_{i-1}^{j+1}}{h^2}\right) + \mathscr{Z}\left(\frac{u_{i+1}^{j}-2u_i^{j}+u_{i-1}^{j}}{h^2}\right)\right) \tag{2.2.3}$$

The expansion can be written as

$$Wu_{i+1}^{j+1} + Xu_i^{j+1} + Yu_{i-1}^{j+1} = \tilde{W}u_{i+1}^{j} + \tilde{X}u_i^{j} + \tilde{Y}u_{i-1}^{j} \tag{2.2.4}$$

where

$$Y = \frac{1}{12k} - \frac{\alpha}{2h^2}, X = \frac{10}{12k} + \frac{\alpha}{h^2}, W = \frac{1}{12k} - \frac{\alpha}{2h^2}, \tilde{Y} = \frac{1}{12k} + \frac{\alpha}{2h^2}, \tilde{X} = \frac{10}{12k} - \frac{\alpha}{h^2}, \tilde{W} = \frac{1}{12k} + \frac{\alpha}{2h^2}$$

For space $i = 1,2,...,m-1$ and for time $j = 0,1,...n$, This construct a system of $m-1$ unknowns $u_1^{j+1}, u_2^{j+1}, ..., u_{m-1}^{j+1}$ at the $(j+1)^{th}$ time level to the fully discrete problem (1.1.3) with conditions (1.1.6) and (1.1.6), which can be written in the matrix form, as follows,

$$\mathscr{A}U^{j+1} = \mathscr{B}U^{j} + \mathscr{C} \tag{2.2.5}$$

where,

$$\mathscr{A} = \begin{bmatrix} X & W & 0 & 0 & \cdots & 0 & 0 \\ Y & X & W & 0 & \cdots & 0 & 0 \\ 0 & Y & X & W & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & Y & X & W \\ 0 & \cdots & \cdots & \cdots & \cdots & Y & Q \end{bmatrix}_{m-1\times m-1}, \mathscr{B} = \begin{bmatrix} \tilde{X} & \tilde{W} & 0 & 0 & \cdots & 0 & 0 \\ \tilde{Y} & \tilde{X} & \tilde{W} & 0 & \cdots & 0 & 0 \\ 0 & \tilde{Y} & \tilde{X} & \tilde{W} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & \tilde{Y} & \tilde{X} & \tilde{W} \\ 0 & \cdots & \cdots & \cdots & \cdots & \tilde{Y} & \tilde{X} \end{bmatrix}_{m-1\times m-1}$$

$$\mathscr{C} = \begin{bmatrix} \tilde{Y}u_0^j - Yu_0^{j+1} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \tilde{W}u_m^j - Wu_m^{j+1} \end{bmatrix}_{m-1\times 1}, U^{j+1} = \begin{bmatrix} u_1^{j+1} \\ u_2^{j+1} \\ \vdots \\ \vdots \\ \vdots \\ u_{m-2}^{j+1} \\ u_{m-1}^{j+1} \end{bmatrix}_{m-1\times 1}, U^{j} = \begin{bmatrix} u_1^{j} \\ u_2^{j} \\ \vdots \\ \vdots \\ \vdots \\ u_{m-2}^{j} \\ u_{m-1}^{j} \end{bmatrix}_{m-1\times 1}$$

This is a fourth order accurate method, by using the derivative operator $\mathscr{Z}$, called Linearized compact finite difference method.

2.3 Numerical Example

Now, we present numerical examples of the proposed method in the section 2.2 and also the result compared with the scheme derived in the section 1.4.
All computations are performed using **Python** on a system with an **Intel® Core™ i7-8265U CPU** @ 1.60GHz (1.80GHz), **8.00 GB** of installed RAM, and a 64-bit operating system with an x64-based processor. Additionally.. The computational maximum absolute error using the proposed schemes, denoted as **CN error**, **CF-1 error** for the proposed method in section 1.4, section 2.2 Respectively.

we discuss the solution for diffusion equation given by the equation (1.1.3), with initial and boundary condition given by the equations (1.1.6)-(1.1.7).

Example-1 Consider 1-D diffusion equation given in (1.1.3), with $\alpha = 1$,$L = 1$ and $t > 0$. The exact solution is for this example is given by

$$u(x,t) = e^{-\pi^2 t}\sin(\pi x). \tag{2.3.1}$$

and initial and boundary condition are given by

$$\begin{cases} u(x,0) = \sin(\pi x) & \textit{Intial Condition} \\ u(0,t) = u(1,t) = 0 & \textit{Boundary Condition} \end{cases} \tag{2.3.2}$$

we solve this problem for $L = 1$ at the final time $T = 1$, using various values of the time step $k = h^2$ and h is the spatial mesh size. We can see in Table (2.1), the scheme given in equation (1.4.8), is a second order scheme and the Proposed scheme in section 2.2 is a fourth order scheme. Fig. (2.3.1) comparing the Exact and Numerical Solution at $h = 1/40$ and $k = h^2$ while Fig. (2.3.2) and Fig. (2.3.3) showing the behavior of Error and Order of convergence with different h and k values.

h	CN-Error	CN-Order	CF-1 Error	CF-1 Order
$1/5$	3.9412×10^{-5}	$*$	4.3147×10^{-3}	$*$
$1/10$	9.6487×10^{-6}	2.03	2.8390×10^{-4}	3.92
$1/20$	2.3484×10^{-6}	2.03	1.7730×10^{-5}	3.99
$1/40$	5.8290×10^{-7}	2.01	1.1082×10^{-6}	4.00
$1/80$	1.4545×10^{-7}	2.00	6.9261×10^{-8}	4.00
$1/160$	3.6347×10^{-8}	2.00	4.3289×10^{-9}	4.00

Table 2.1: Max Absolute error and Order of Convergence for Example-1

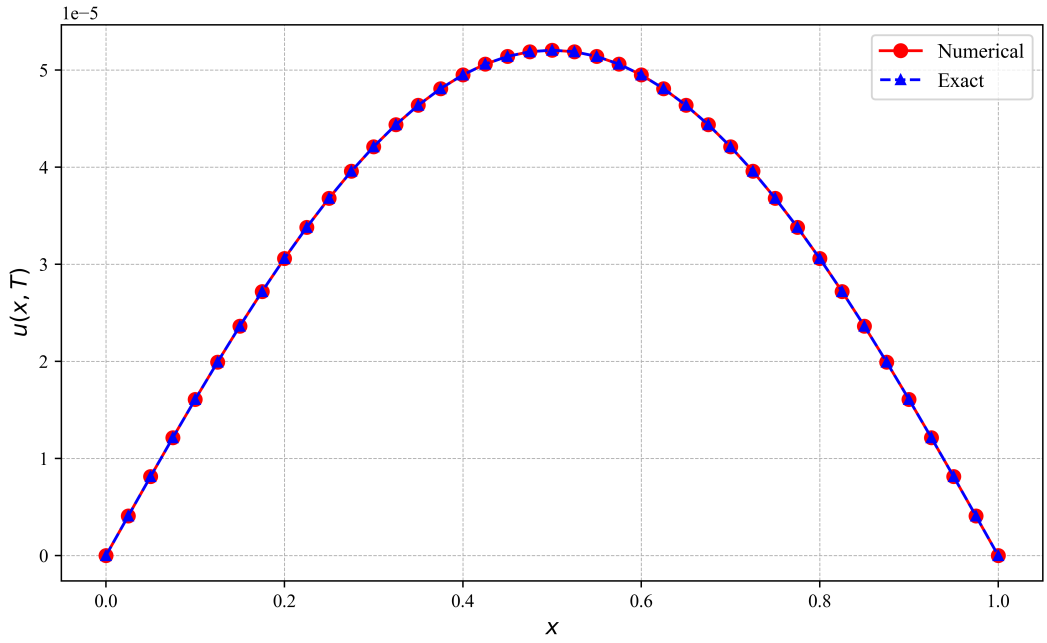


Figure 2.3.1: Exact and Numerical Solution for Example-1 at $h = 1/40$ and $k = h^2$

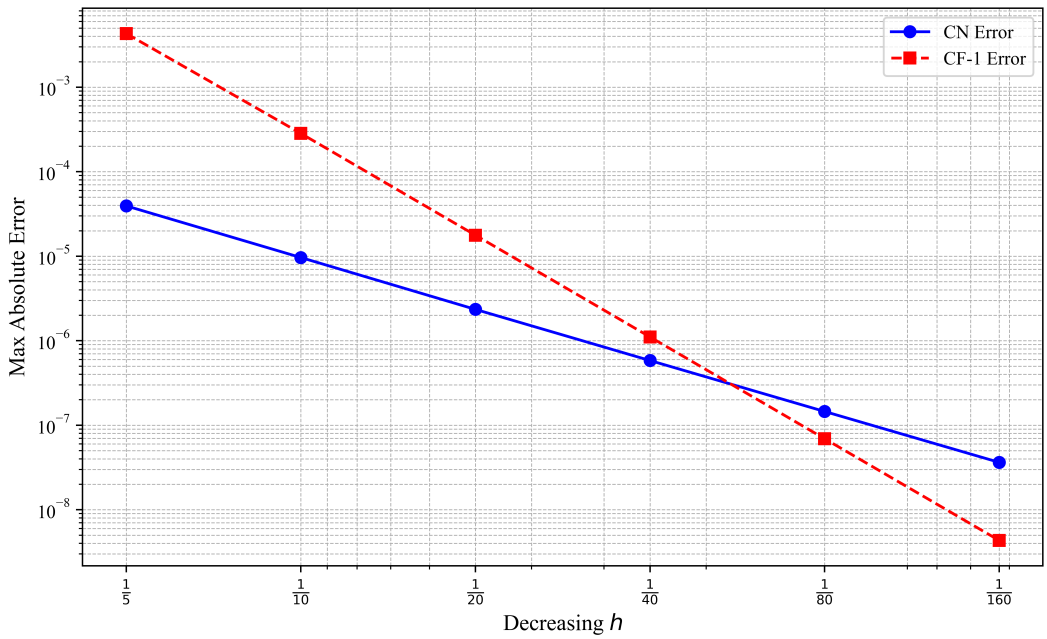


Figure 2.3.2: Behavior of CN-Error and CF-1 Error for Example-1 with Different h values

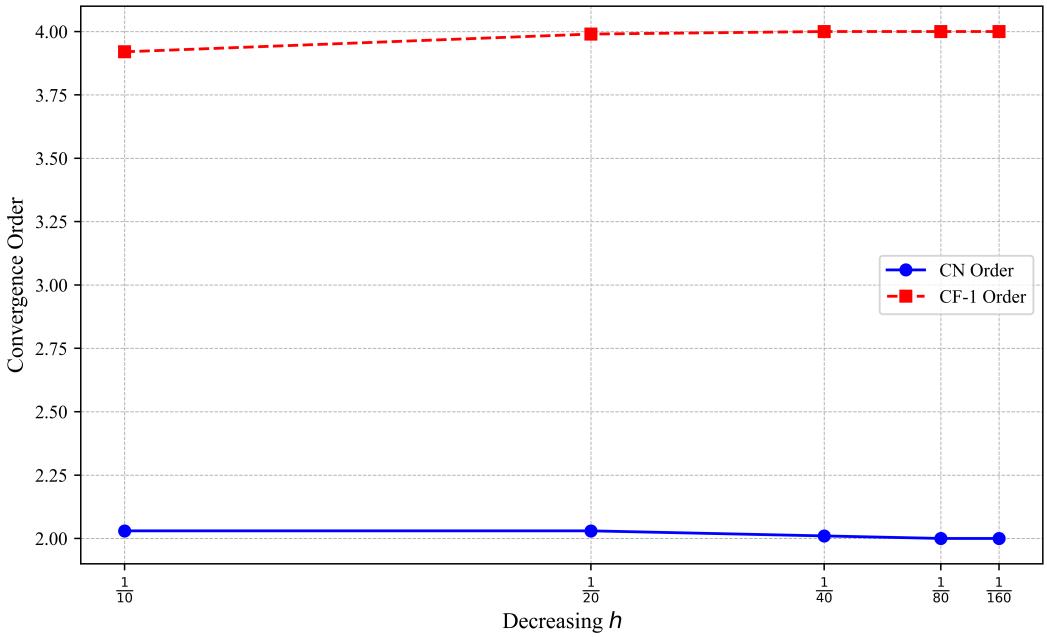


Figure 2.3.3: Behavior of CN-order and CF-1 Order for Example-1 with different h values

It is clear from the Table (2.1) that the operator-based compact finite difference scheme demonstrates excellent agreement with the exact solutions and exhibits a rapid decay in error as the grid is refined. The computed error confirm that the scheme achieves a convergence rate consistent with fourth-order accuracy in space, thereby validating the theoretical predictions. This high accuracy is further supported by the truncation error analysis derived in the section , where it is shown that the compact operator for the second derivative introduces an error term of order $\mathscr{O}(h^4)$, ensuring fourth-order spatial accuracy. In addition, the stability analysis performed in section (4.1) verifies that the scheme is unconditionally stable when applied with Crank-Nicolson time discretization.

2.4 Limitations of scheme

In this method, we use a higher-order finite difference operator denoted by $\mathscr{Y}$ as derived in section 2.2, which provides a fourth-order accurate approximation of the second-order spatial derivative. To achieve higher-order accuracy in time, we incorporate the Crank-Nicolson scheme. However, when dealing with more general parabolic equations, such as advection-diffusion-reaction equations, this approach faces limitations. The reason is that the operator $\mathscr{Y}$ is specifically constructed to approximate only the second-order spatial derivative, and cannot be directly used to approximate lower-order terms like the reaction term u and the advection term u_x. Therefore, the main limitation of this method is its inapplicability to equations containing first-order or zeroth-order spatial derivatives.

Chapter 3

General Compact Finite Difference scheme for Advetion-Diffusion-Reaction equation

The general CFDM approach builds fourth-order proper approximations for both the first and second derivatives by directly applying Taylor series expansions, in contrast to operator-based approaches. It works especially well for resolving advection-diffusion-reaction (ADR) equations. Transport of physical quantities like diffusion, mass, or concentration in a medium is regulated by the Advection - Diffusion-reaction equation. The combined effects of advection, i.e., describing the bulk transport of the material due to a flow imposed on it from the outside, and diffusion, representing the random diffusion of particles based on molecular collisions, are quantified by the equation. The one-dimensional advection-diffusion-reaction equation is given by equation (1.2.3), with initial condition specifies the distribution of $u(x,t)$ at $t = 0$:

$$u(x,0) = \phi(x), \quad x \in [0,L]. \tag{3.0.1}$$

Additionally, boundary conditions describe the behavior of $u(x,t)$ at the domain edges:

$$u(0,t) = \Psi_L(t), \quad u(L,t) = \Psi_R(t), \quad t \in [0,T]. \tag{3.0.2}$$

These conditions ensure a well-posed problem for practical applications in engineering, fluid dynamics, and environmental modeling.

3.1 A fourth-order compact finite difference scheme

Higher order compact finite difference scheme begin with a traditional compact difference approximation and extend precision by the use of additional terms through differentiation by repeated subtraction and compact difference. Compact difference schemes are in frequent use throughout computational fluid mechanics, acoustics, and electromagnetic due to high precision as well as the level of efficiency it provides. For this reason, we write in this section the fourth-order finite compact difference formulation for equation (1.2.3) space derivatives, In order to derive the scheme for equation (1.2.3) first we discretize the system in space and then we apply time discretization, so for discretization of space we analyze differential equation form of the equation (1.2.3),

$$\alpha \frac{d^2V}{dx^2} - \beta \frac{dV}{dx} - \gamma V = f(x), \quad 0 < x < l. \tag{3.1.1}$$

with boundary conditions:

$$V(0) = v_0, \quad V(l) = v_l. \tag{3.1.2}$$

A fourth-order accurate compact finite difference scheme for equation (1.2.3) can be derived as follows [23]

We approximate the first and second derivatives at a grid point x_i using central finite difference schemes. Using a second-order central difference formula:

$$\frac{d^2V}{dx^2}\Big|_{x_i} = \frac{V_{i+1} - 2V_i + V_{i-1}}{h^2} + \mathscr{O}(h^2). \tag{3.1.3}$$

Thus, second-order finite difference operator:

$$\delta_x^2 V_i = \frac{V_{i+1} - 2V_i + V_{i-1}}{h^2}. \tag{3.1.4}$$

Using a second-order central difference formula:

$$\frac{dV}{dx}\Big|_{x_i} = \frac{V_{i+1} - V_{i-1}}{2h} + \mathscr{O}(h^2). \tag{3.1.5}$$

Thus, we define:

$$\delta_x V_i = \frac{V_{i+1} - V_{i-1}}{2h}. \tag{3.1.6}$$

Substituting equation (3.1.4) and equation (3.1.6) these into the differential equation (3.1.1) at x_i:

$$\alpha\delta_x^2 V_i - \beta\delta_x V_i - \gamma V_i = f_i + \mathscr{T}_i, \tag{3.1.7}$$

where τ_i is the truncation error.

3.1.1 Higher order derivative approximation

To obtain a higher-order compact finite difference scheme, we expand the function $V(x)$ about x_i using Taylor series.

$$V_{i+1} = V_i + hV_i' + \frac{h^2}{2}V_i'' + \frac{h^3}{6}V_i''' + \frac{h^4}{24}V_i^{(iv)} + \mathscr{O}(h^5). \tag{3.1.8}$$

$$V_{i-1} = V_i - hV_i' + \frac{h^2}{2}V_i'' - \frac{h^3}{6}V_i''' + \frac{h^4}{24}V_i^{(iv)} + \mathscr{O}(h^5). \tag{3.1.9}$$

On adding equation (3.1.8) and equation (3.1.9), we get

$$V_{i+1} + V_{i-1} = 2V_i + h^2V_i'' + \frac{h^4}{12}V_i^{(iv)} + \mathscr{O}(h^6).$$

$$\frac{V_{i+1} - 2V_i + V_{i-1}}{h^2} = V_i'' + \frac{h^2}{12}V_i^{(iv)} + \mathscr{O}(h^4).$$

Thus,

$$\delta_x^2 V_i = V_i'' + \frac{h^2}{12}V_i^{(iv)} + \mathscr{O}(h^4). \tag{3.1.10}$$

On subtracting equation (3.1.8) and equation (3.1.9), we get

$$V_{i+1} - V_{i-1} = 2hV_i' + \frac{h^3}{3}V_i''' + \mathscr{O}(h^5).$$

$$\frac{V_{i+1} - V_{i-1}}{2h} = V_i' + \frac{h^2}{6}V_i''' + \mathscr{O}(h^4).$$

Thus,

$$\delta_x V_i = V_i' + \frac{h^2}{6}y_i''' + \mathscr{O}(h^4). \tag{3.1.11}$$

Expressing V_i''' and $V_i^{(iv)}$ in terms of known quantities From the given equation (3.1.1), on differentiating both sides of equation (3.1.1), we get

$$\alpha V_i''' - \beta V_i'' - \gamma V_i = f_i'.$$

$$V_i''' = \frac{1}{\alpha}\left(f_i' + \beta V_i'' + \gamma V_i'\right). \tag{3.1.12}$$

Approximating equation (3.1.12) equation derivatives using central differences, we get

$$V_i''' = \frac{1}{\alpha}\left(\delta_x f_i + \beta\,\delta_x^2 V_i + \gamma V_i'\right) + \mathscr{O}(h^2). \tag{3.1.13}$$

on differentiating equation (3.1.13), we get

$$\alpha V_i^{(iv)} - \beta V_i''' - \gamma V_i'' = f_i''.$$

$$V_i^{(iv)} = \frac{1}{\alpha}\left(f_i'' + \beta V_i''' + \gamma V"_i\right). \tag{3.1.14}$$

Substituting the value of V_i''' from equation (3.1.13) into equation (3.1.14), we get

$$V_i^{(iv)} = V_i''\left(\frac{\beta^2}{\alpha^2} + \frac{\gamma}{\alpha}\right) + V_i'\left(\frac{\gamma\beta}{\alpha^2}\right) + f_i'\left(\frac{\beta}{\alpha^2}\right) + \frac{f_i''}{\alpha^2} + \mathscr{O}(h^2). \tag{3.1.15}$$

Approximating equation (3.1.15) using central differences, we get

$$V_i^{(iv)} = \delta_x^2 V_i\left(\frac{\beta^2}{\alpha^2} + \frac{\gamma}{\alpha}\right) + \delta_x V_i\left(\frac{\gamma\beta}{\alpha^2}\right) + \delta_x f_i\left(\frac{\beta}{\alpha^2}\right) + \frac{\delta_x^2 f_i}{\alpha^2} + \mathscr{O}(h^2). \tag{3.1.16}$$

3.1.2 Derivation of Error term

To find the truncation error $\mathscr{T}_i$ of the scheme derived in section 3.1.1, consider the equation (3.1.7), where

$$\mathscr{T}_i = \frac{h^2}{12}\left(\alpha V^{(iv)} - 2\beta V'''\right) + \mathscr{O}(h^4) \tag{3.1.17}$$

so, on putting the values of higher order derivatives from the section 3.1.1, we get

$$\begin{aligned}\mathscr{T}_i = \frac{h^2}{12}\Bigg\{\alpha\left[\delta_x^2 V_i\left(\frac{\beta^2}{\alpha^2} + \frac{\gamma}{\alpha}\right) + \delta_x V_i\left(\frac{\gamma\beta}{\alpha^2}\right) + \delta_x f_i\left(\frac{\beta}{\alpha^2}\right) + \frac{\delta_x^2 f_i}{\alpha^2}\right] \\ -2\beta\left[\frac{1}{\alpha}\left(\delta_x f_i + \beta\,\delta_x^2 V_i + \gamma V_i'\right)\right]\Bigg\} + \mathscr{O}(h^4)\end{aligned} \tag{3.1.18}$$

on solving we get,

$$\mathscr{T}_i = \delta_x^2 V_i\left(\frac{h^2}{12} - \frac{\beta^2 h^2}{12\alpha}\right) + \delta_x V_i\left(-\frac{\gamma\beta h^2}{12\alpha}\right) + \delta_x f_i\left(-\frac{\beta h^2}{12\alpha}\right) + \delta_x^2 f_i\left(\frac{h^2}{12}\right) + \mathscr{O}(h^4) \tag{3.1.19}$$

This is the truncation error due to derivative approximation and ensures that the derivatives is fourth order accurate in space discretization.

Now on substituting the value from section 3.1.1 and section 3.1.2, in equation (3.1.7), we get the is the fourth-order compact finite difference scheme, incorporating higher-order corrections to improve accuracy.

$$\delta_x^2 V_i\left(\alpha - \frac{h^2\gamma}{12} + \frac{\beta h^2}{12\alpha}\right) + \delta_x V_i\left(-\beta + \frac{\gamma\beta h^2}{12\alpha}\right) - \gamma V_i = f_i + \frac{h^2}{12}\left(\delta_x^2 f_i - \frac{\beta}{\alpha}\delta_x f_i\right) + \mathscr{O}(h^4), \tag{3.1.20}$$

on approximate derivatives in equation (3.1.20), by central finite difference, we get

$$\left[\frac{V_{i+1}-2V_i+V_{i-1}}{h^2}\right]\left(\alpha-\frac{h^2\gamma}{12}+\frac{\beta h^2}{12\alpha}\right)+\left[\frac{V_{i+1}-V_{i-1}}{2h}\right]\left(-\beta+\frac{\gamma\beta h^2}{12\alpha}\right)+(-\gamma)V_i = f_i+\frac{h^2}{12}\left(f''-\frac{\beta}{\alpha}f'\right)+\mathscr{O}(h^4) \tag{3.1.21}$$

on re-arranging in a tri-diagonal systems and replacing V with u, we get

$$-(K_1+K_2)u_{i+1}(t)+(2K_1-\gamma)u_i(t)-(K_1-K_2)u_{i-1}(t) = (K_3+K_4)u'_{i+1}(t)+(1-2K_3)u'_i(t)+(K_3-K_4)u'_{i-1}(t) \tag{3.1.22}$$

where:

- $u_i(t)$, $u_{i+1}(t)$, and $u_{i-1}(t)$ are the function values at different spatial grid points at time t.
- $u'_i(t)$, $u'_{i+1}(t)$, and $u'_{i-1}(t)$ are their time derivatives.
- $K_1 = -\left(\frac{\alpha}{h^2}+\frac{\beta^2}{12\alpha}-\frac{\gamma}{12}\right)$, $K_2 = \frac{\beta}{2h}-\frac{\gamma\beta h}{24\alpha}$, $K_3 = \frac{1}{12}$, $K_4 = -\frac{\beta h}{24\alpha}$

3.2 The Proposed Method

To get the higer order compact finite difference scheme for Advection-diffusion-reaction equation (1.2.3), now we apply the crank-nicolson scheme define in equation (1.4.8) and [1]

We define a uniform discretization of the computational domain $\Omega = (x_{\min}, x_{\max}) \times (0, T)$. The domain is divided into a grid of node points (x_i, t_j), where:

- Spatial nodes: $x_i = x_{\min} + ih$, for $i = 0, 1, 2, \ldots, m$.
- Temporal nodes: $t_j = jk$, for $j = 0, 1, 2, \ldots, n$.

On using Crank-Nicolson scheme (1.4.8), on discretized system of equations (3.1.22), on using

$$u(x) = \frac{u^{j+1}(x)+u^j(x)}{2},\ u_t(x) = \frac{u^{j+1}(x)-u^j(x)}{k} \tag{3.2.1}$$

The equation (3.2.1) shows the weighted average of the space discretization and the Central difference for the derivative on time derivative.we get,

$$Fu_{i+1}^{j+1}+Gu_i^{j+1}+Hu_{i-1}^{j+1} = \tilde{F}u_{i+1}^j+\tilde{G}u_i^j+\tilde{H}u_{i-1}^j \tag{3.2.2}$$

Where,

$F = -\frac{(K_1+K_2)}{2}-\frac{(K_3+K_4)}{k}$, $\tilde{F} = \frac{(K_1+K_2)}{2}-\frac{(K_3+K_4)}{k}$

$G = \frac{2K_1-\gamma}{2}-\frac{(1-2K_3)}{k}$, $\tilde{G} = -\frac{2K_1-\gamma}{2}-\frac{(1-2K_3)}{k}$

$H = -\frac{(K_1-K_2)}{2}-\frac{(K_3-K_4)}{k}$, $\tilde{H} = \frac{(K_1-K_2)}{2}-\frac{(K_3-K_4)}{k}$

For $i = 1,2,...,m-1$ and $j = 0,1,...n$ we have,

$$Fu_2^{j+1} + Gu_1^{j+1} + Hu_0^{j+1} = \tilde{F}u_2^j + \tilde{G}u_1^j + \tilde{H}u_0^j$$

$$\vdots$$

$$Fu_m^{j+1} + Gu_{m-1}^{j+1} + Hu_{m-2}^{j+1} = \tilde{F}u_m^j + \tilde{G}u_{m-1}^j + \tilde{H}u_{m-2}^j \tag{3.2.3}$$

with the help of the boundary condition given in equation (3.0.2). We get the following two Relations

$$u^{j+1}(x_0) = \Psi_L(t^{j+1}) \tag{3.2.4}$$

$$u^{j+1}(x_m) = \Psi_R(t^{j+1}) \tag{3.2.5}$$

The resulting equation (3.2.3) for $i = 2,3,...m-1$ yield a system of $m-1$ unknowns $u_1^{j+1}, u_2^{j+1}, ..., u_{m-1}^{j+1}$ at the $(j+1)^{th}$ time level to the fully discrete scheme for equation (1.2.3) with conditions (3.0.1)-(3.0.2), which can be written in the matrix form, as follows,

$$\mathscr{A}U^{j+1} = \mathscr{B}U^j + \mathscr{C} \tag{3.2.6}$$

where,

$$\mathscr{A} = \begin{bmatrix} G & F & 0 & 0 & \cdots & 0 & 0 \\ H & G & F & 0 & \cdots & 0 & 0 \\ 0 & H & G & F & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & H & G & F \\ 0 & \cdots & \cdots & \cdots & \cdots & H & G \end{bmatrix}_{m-1\times m-1}, \mathscr{B} = \begin{bmatrix} \tilde{G} & \tilde{F} & 0 & 0 & \cdots & 0 & 0 \\ \tilde{H} & \tilde{G} & \tilde{F} & 0 & \cdots & 0 & 0 \\ 0 & \tilde{H} & \tilde{G} & \tilde{F} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & \tilde{H} & \tilde{G} & \tilde{F} \\ 0 & \cdots & \cdots & \cdots & \cdots & \tilde{H} & \tilde{G} \end{bmatrix}_{m-1\times m-1}$$

$$\mathscr{C} = \begin{bmatrix} \tilde{H}u_0^j - Hu_0^{j+1} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \tilde{F}u_m^j - Fu_m^{j+1} \end{bmatrix}_{m-1\times 1}, U^{j+1} = \begin{bmatrix} u_1^{j+1} \\ u_2^{j+1} \\ \vdots \\ \vdots \\ \vdots \\ u_{m-2}^{j+1} \\ u_{m-1}^{j+1} \end{bmatrix}_{m-1\times 1}, U^j = \begin{bmatrix} u_1^j \\ u_2^j \\ \vdots \\ \vdots \\ \vdots \\ u_{m-2}^j \\ u_{m-1}^j \end{bmatrix}_{m-1\times 1}$$

3.3 Numerical Examples

Now, we present numerical examples to demonstrate the applicability and accuracy of the proposed method in the section 3.2 and also the result compared with the scheme derived in the section 1.4 and section 2.2.

Example-1 Consider 1-D diffusion equation given in equation (1.1.3), with $\alpha = \frac{1}{\pi^2}, L = 1$ and $t > 0$. The exact solution is for this example is given by

$$u(x,t) = e^{-t}\sin(\pi x). \tag{3.3.1}$$

and initial and boundary condition are given by

$$\begin{cases} u(x,0) = \sin(\pi x) & \textit{Intial Condition} \\ u(0,t) = u(1,t) = 0 & \textit{Boundary Condition} \end{cases} \tag{3.3.2}$$

This test example was originally considered in [17], To evaluate the accuracy of our proposed method,we solve this example for $L = 1$ at the final time $T = 1$, using various values of the time step $k = h^2$ and h is the spatial mesh size. Here in Example-1, we are also applying the higher order compact finite difference scheme for adection-diffusion-reaction equation, proposed in section 3.2 with the Advection and reaction coefficients $\beta = \gamma = 0$. We can see in Table (3.1),The scheme given in equation (1.4.8), is a second order scheme and the Proposed scheme in section 2.2 and section 3.2 are fourth order scheme with different values of h and by taking $k = h^2$.

Fig (3.3.2) comparing the Exact and Numerical Solution at $h = 1/40$ and $k = h^2$ while Fig. (3.3.2) and Fig. (3.3.3) showing the behavior of Error and Order of convergence with different h and k values.

h	CN-Error	CN-Order	CF-1 Error	CF-1 Order	CF-2 Error	CF-2 Order
$1/5$	2.5764×10^{-2}	$*$	1.8398×10^{-4}	$*$	1.8399×10^{-4}	$*$
$1/10$	6.7221×10^{-3}	1.94	1.1923×10^{-5}	3.95	1.1923×10^{-5}	3.95
$1/20$	1.6772×10^{-3}	2.00	7.4250×10^{-7}	4.00	7.4250×10^{-7}	4.00
$1/40$	4.1909×10^{-4}	2.00	4.6364×10^{-8}	4.00	4.6364×10^{-8}	4.00
$1/80$	1.0476×10^{-4}	2.00	2.8970×10^{-9}	4.00	2.8970×10^{-9}	4.00
$1/160$	2.6190×10^{-2}	2.00	1.8059×10^{-10}	4.00	1.8059×10^{-10}	4.00

Table 3.1: Max Absolute error and Order of Convergence for Example-1

by this Table (3.1) we can observe, the results by scheme given in section 3.2 is exactly same as scheme given in section 2.2 for $\beta = \gamma = 0$ in Advection-Diffusion-Reaction equation (1.2.3).

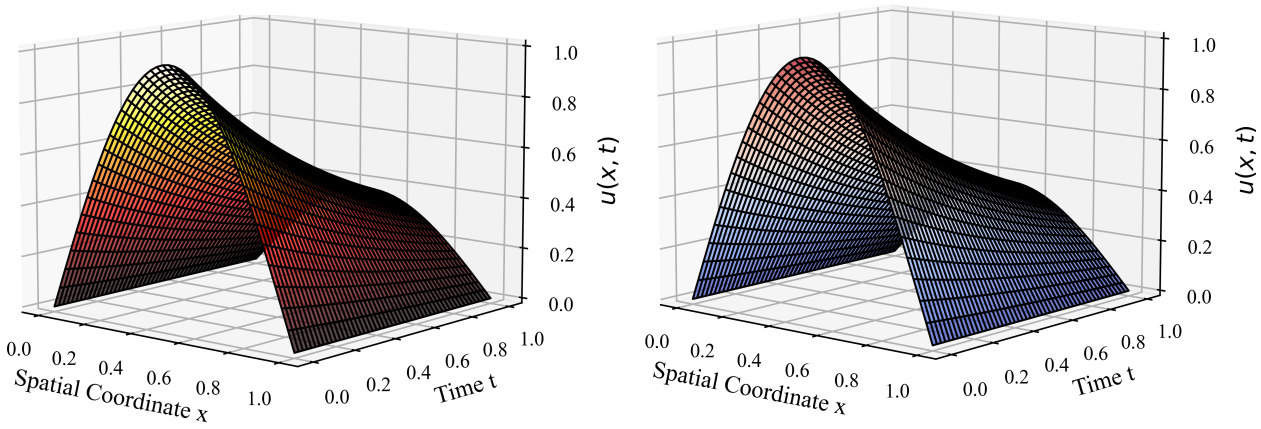


Figure 3.3.1: Numerical and Exact Solutions of Example-1 at $h = 1/40$ and $k = h^2$

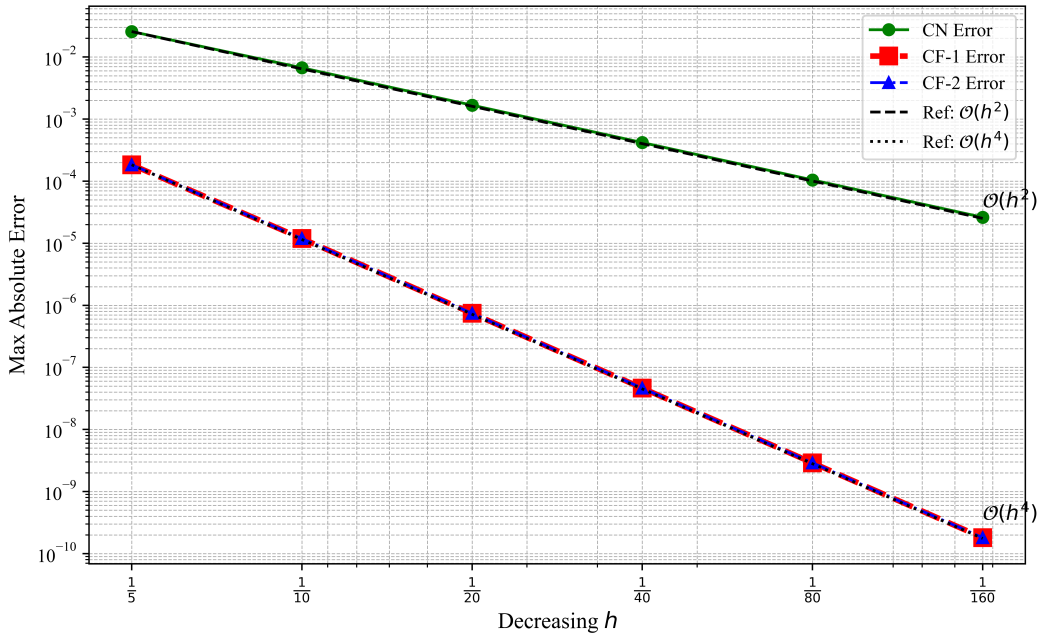


Figure 3.3.2: Behavior of CN-Erro, CF-1 Error and CF-2 Error for Example-1 with Different h values

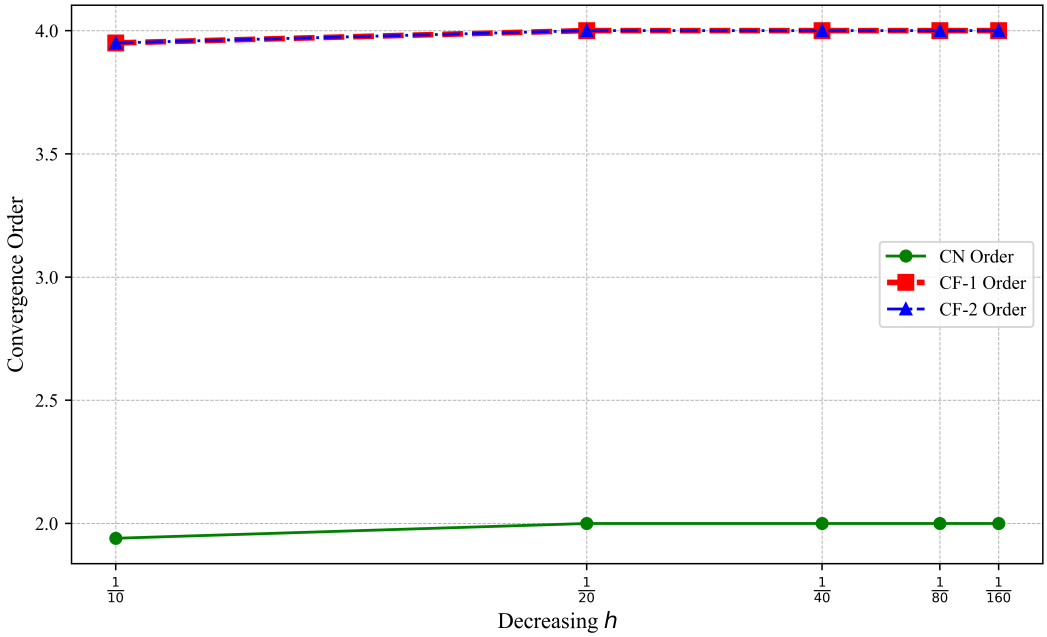


Figure 3.3.3: Behavior of CN-order, CF-1 Order and CF-2 Order for Example-1 with different h values

Now, we discuss the solution for Advection-Diffusion equation given by the equation (1.2.3) by using the scheme given in section 3.2, with reaction coefficient $\gamma = 0$, initial and boundary condition given by the equation (3.0.1)-(3.0.2), we solve the Advection-Diffusion equation with different values of α and β.

Example-2 Consider the advection-diffusion equation given by (1.2.3) with initial Condition given by

$$\phi(x) = \exp(5x)\left[\cos\left(\frac{\pi}{2}x\right) + 0.25\sin\left(\frac{\pi}{2}x\right)\right] \tag{3.3.3}$$

The exact solution is for this example is given by

$$u(x,t) = \exp(5(x-\frac{t}{2}))\exp\left(-\frac{\pi^2}{40}t\right)\left[\cos\left(\frac{\pi}{2}x\right)+0.25\sin\left(\frac{\pi}{2}x\right)\right] \tag{3.3.4}$$

This example is based on the advection-diffusion equation discussed in [22]. The boundary conditions are obtained from the exact solution. To evaluate the accuracy of our proposed method,we solve this example for $L=1$ at the final time $T=2$, using various values of the time step $k=h^2$ and h is the spatial mesh size. Here in Example-2, we apply the higher order compact finite difference scheme, proposed in section 3.2 with the reaction coefficients $\gamma = 0.$ and $\alpha = 0.1, \beta = 1.$ We can see in Table (3.2),The Proposed scheme in section 3.2 is fourth order scheme.

Fig. (3.3.4) comparing the Exact and Numerical Solution at $h = 1/64$ and $k = h^2$ while Fig. (3.3.5) and Fig. (3.3.6) showing the behavior of Error and Order of convergence with different h and k values.

h	CF-2 Error	CF-2 Order
$1/16$	3.0308×10^{-4}	$*$
$1/32$	1.9066×10^{-5}	3.98
$1/64$	1.1941×10^{-6}	4.00
$1/128$	7.4672×10^{-8}	4.00

Table 3.2: Max Absolute Error and Order of Convergence for Example-2

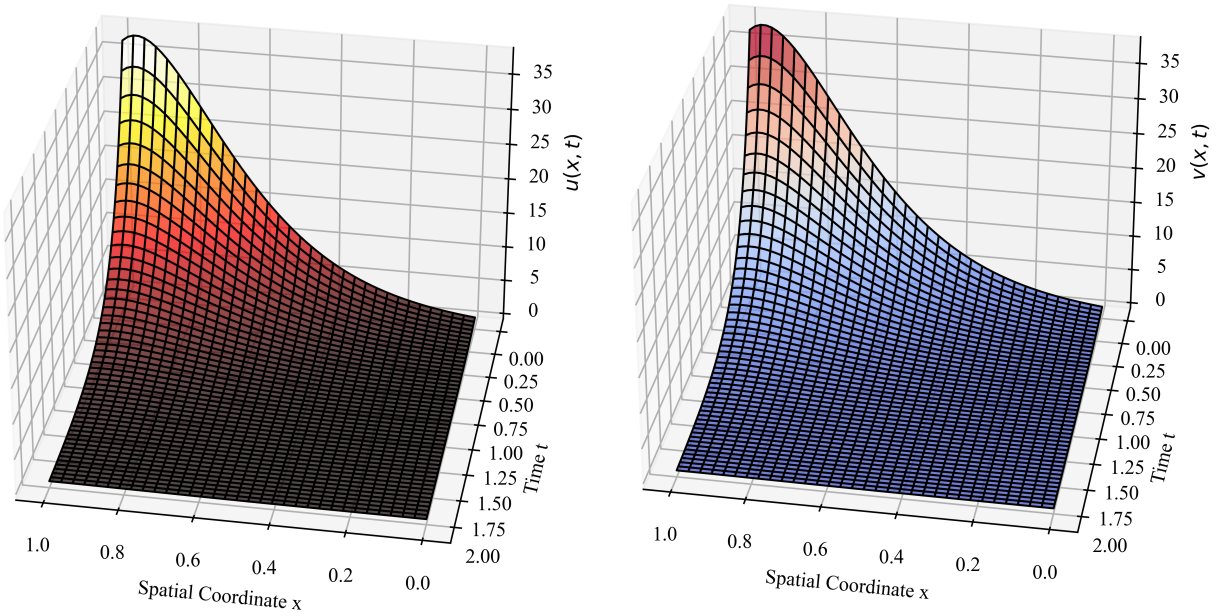


Figure 3.3.4: Numerical Solution and Exact Solution of Example-2 at $h = 1/64, k = h^2$

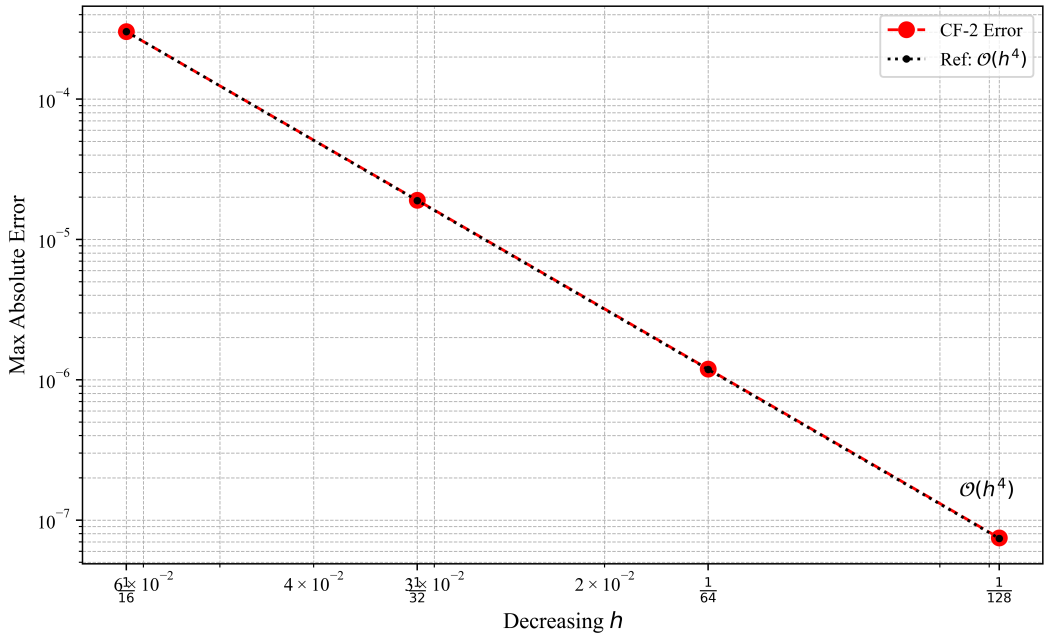


Figure 3.3.5: Behavior of CF-2 Error for Example-2 with Different h values

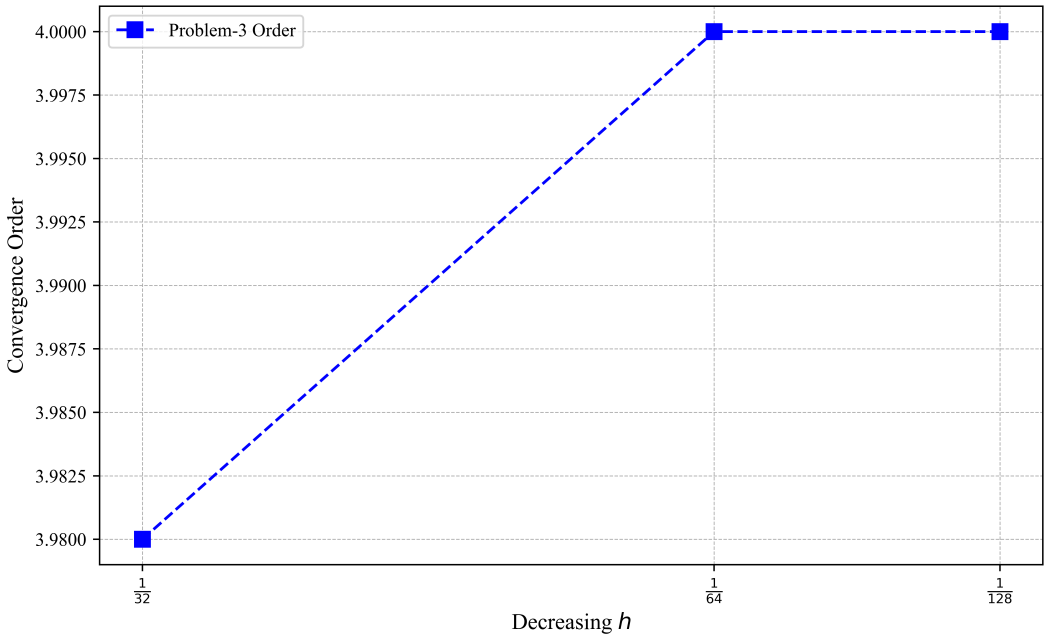


Figure 3.3.6: Behavior of CF-2 Order for Example-2 with different h values

Now, We also see the behavior of solution of Advection-Diffusion equation with Péclet number P_e defined as
$P_e = \left|\frac{\beta}{\alpha}\right|$
with different initial and boundary conditions in the following example.

Example-3 Consider the advection-diffusion equation given by equation (1.2.3) with the initial condition:

$$\phi(x) = a\exp(-\bar{c}x). \tag{3.3.5}$$

The exact solution is for this example is given by

$$u(x,t) = a\exp(bt - \bar{c}x), \tag{3.3.6}$$

where $\bar{c} = \frac{-\beta+\sqrt{\beta^2+4\alpha b}}{2\alpha}$
and Boundary Conditions are given as follow

$$u(0,t) = a\exp(bt), \quad u(1,t) = a\exp(bt-\bar{c}) \tag{3.3.7}$$

This example is based on the advection-diffusion equation discussed in [20]. To evaluate the accuracy of our proposed method,we solve this example for $L = 1$ at the final time $T = 1$, using various values of the time step $k = h^2$ and h is the spatial mesh size. Here in Example-3, we apply the higher order compact finite difference scheme, proposed in section 3.2 with the reaction coefficients $\gamma = 0$ and For this example, we use $a = 1$, $b = 0.1$, $\beta = -1$. When the Péclet number (P_e) increases, meaning the convection term dominates over the diffusion term, the analytical solution exhibits a sharp discontinuity near the left boundary. We can see in Table (3.3) and Table (3.4),The order reducing by increasingP_e.

Fig. (3.3.7) comparing the Exact and Numerical Solution at $h = 1/128$ and $k = h^2$ with different values of P_e.while Fig. (3.3.5) and Fig. (3.3.6) showing the behavior of Error and Order of convergence with different h and k values.

h	$P_e = 10$		$P_e = 20$	
	CF-2 Error	CF-2 Order	CF-2 Error	CF-2 Order
$1/16$	8.8239×10^{-5}	$*$	1.4970×10^{-3}	$*$
$1/32$	5.5792×10^{-6}	3.98	8.7430×10^{-5}	4.00
$1/64$	3.4715×10^{-7}	4.00	5.5135×10^{-6}	4.00
$1/128$	2.1691×10^{-8}	4.00	3.4308×10^{-7}	4.00

Table 3.3: Maximum error and Order of Convergence for Example-3 with $Pe = 10, 20$

h	$P_e = 100$		$P_e = 1000$	
	CF-2 Error	CF-2 Order	CF-2 Error	CF-2 Order
$1/16$	1.6741×10^{-1}	$*$	7.5202×10^{-1}	$*$
$1/32$	3.3796×10^{-2}	2.38	5.1300×10^{-1}	0.55
$1/64$	3.4567×10^{-3}	3.30	2.4073×10^{-1}	1.09
$1/128$	2.1290×10^{-4}	4.00	6.1102×10^{-2}	2.00

Table 3.4: Maximum error and Order of Convergence for Example-3 with $Pe = 100, 1000$

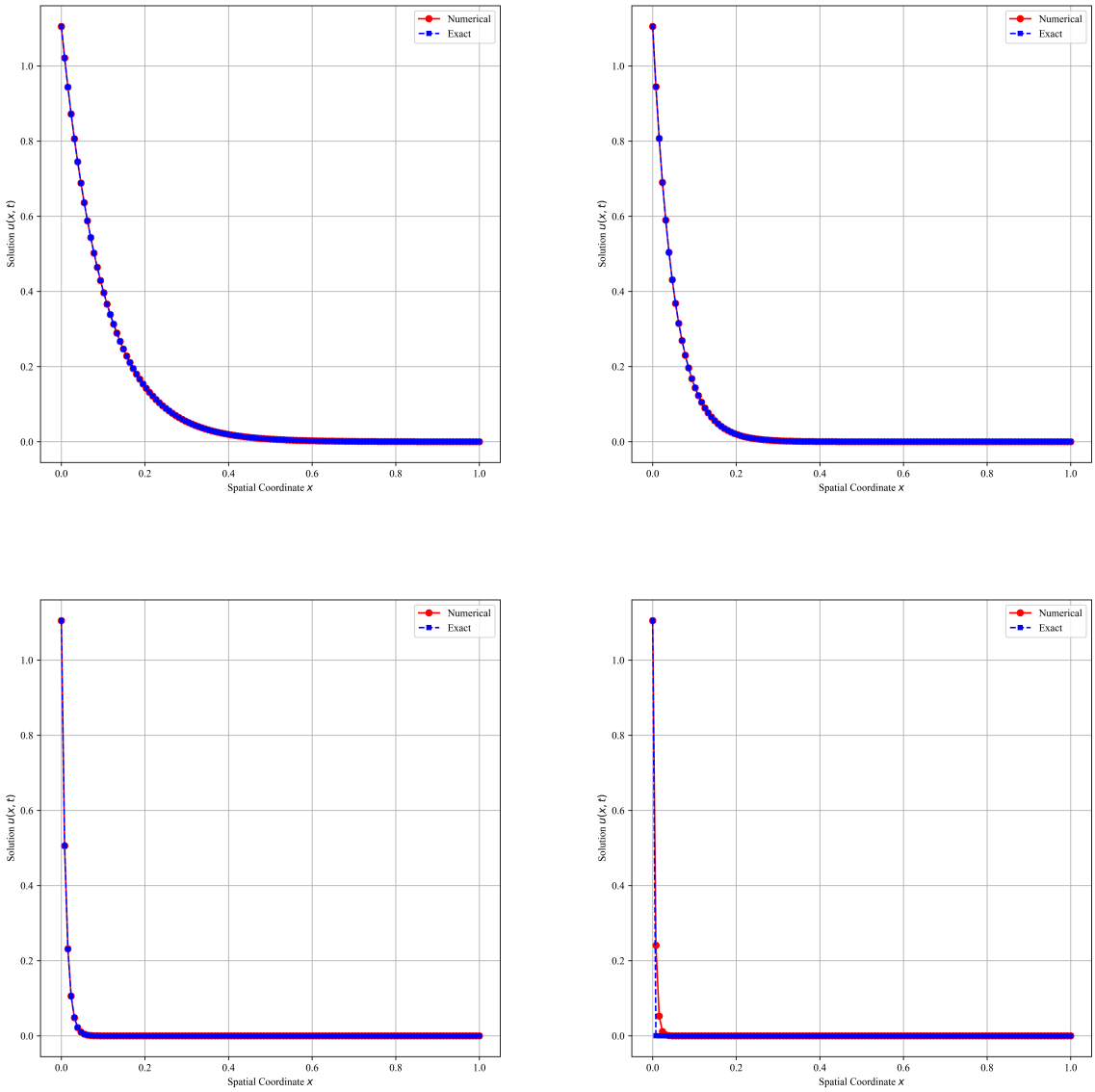


Figure 3.3.7: Exact and Numerical Solution of Example-3 for $P_e = 10, 20, 100, 1000$ at $h = 1/128$

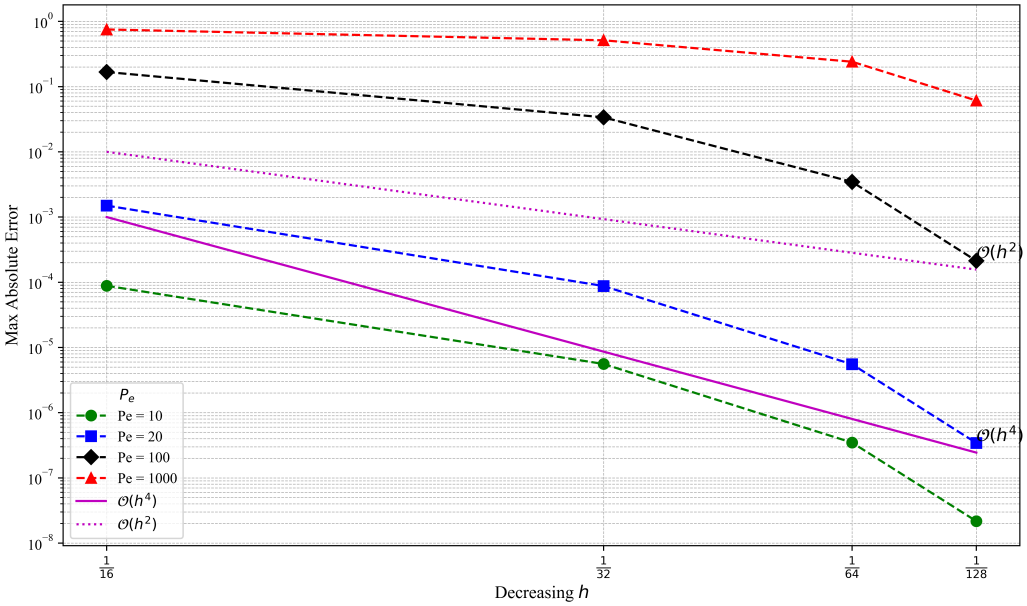


Figure 3.3.8: Behavior of CF-2 Error for Example-3 with Different P_e values

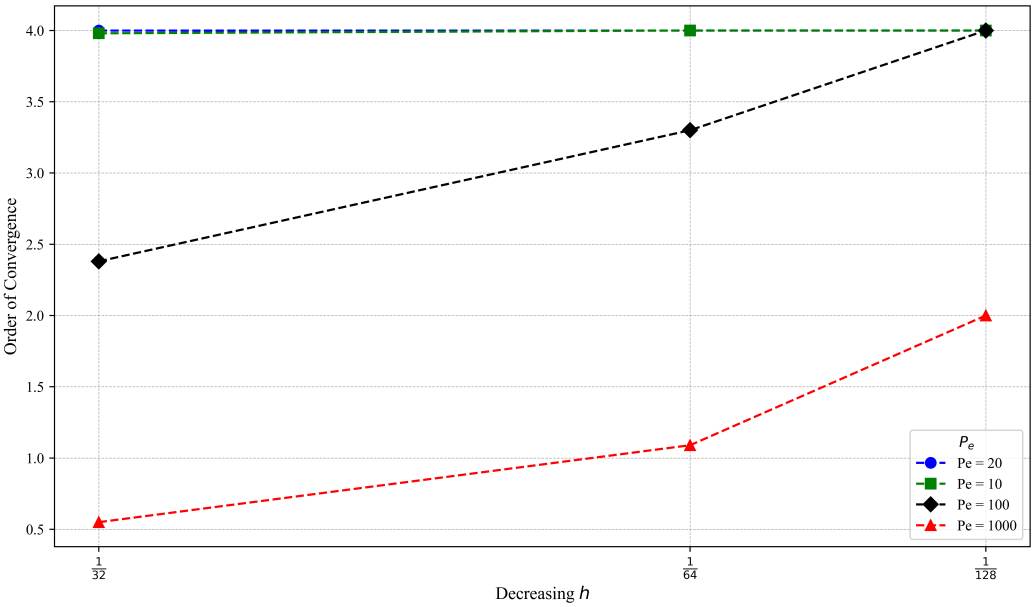


Figure 3.3.9: Behavior of CF-2 Order for Example-3 with different P_e values

Here, we discuss the solution for Advection-Diffusion-Reaction equation given by the equation (1.2.3) by using the scheme given in section 3.2,with initial and boundary condition given by the equations (3.0.1)-(3.0.2), we solve the Advection-Diffusion-Reaction equation with different values of α, β and γ.

Example-4 Consider equation given in (1.2.3), as

$$\frac{\partial u}{\partial t}(x,t) + \frac{\partial u}{\partial x}(x,t) - 3u(x,t) = \frac{\partial^2 u}{\partial x^2}(x,t), \quad (x,t) \in [0,2\pi] \times \left[0, \frac{1}{2}\right], \tag{3.3.8}$$

with the initial condition $u(x,0) = \sin(x)$. Boundary condition $u(0,t) = e^{2t}\sin(-t)$ and $u(2\pi,t) = e^{2t}\sin(2\pi - t)$.

The exact solution is given by $u(x,t) = e^{2t}\sin(x-t)$.
we solve this example for $L = 2\pi$ at the final time $T = \frac{1}{2}$, using various values of the time step $k = h^2$ and h is the spatial mesh size. We can see in Table (3.5),the Proposed scheme in section 2.2 is a fourth order scheme.

Fig (3.3.10) comparing the Exact and Numerical Solution at $h = 1/40$ and $k = h^2$ while Fig. (3.3.11) and Fig. (3.3.12) showing the behavior of Error and Order of convergence with different h and k values.

h	CF-2 Error	CF-2 Order
$1/8$	3.0497×10^{-4}	$*$
$1/16$	1.9066×10^{-5}	3.98
$1/32$	1.1919×10^{-6}	4.00
$1/64$	7.4495×10^{-8}	4.00
$1/128$	4.6559×10^{-9}	4.00

Table 3.5: Max Absolute Error and Order of Convergence for Example-4

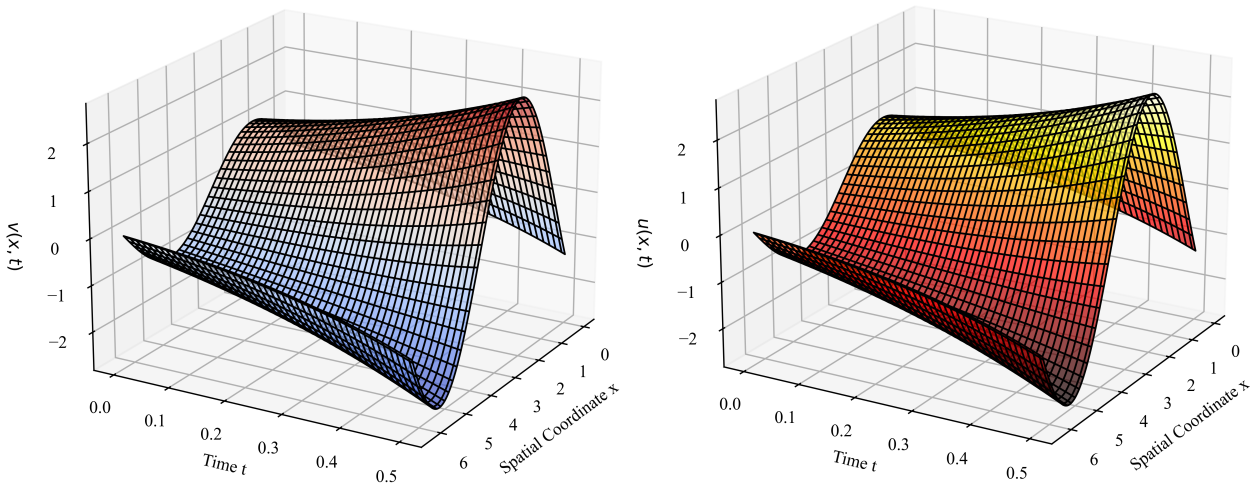


Figure 3.3.10: Numerical Solution and Exact Solution of Example-4 at $h = 1/32, k = h^2$

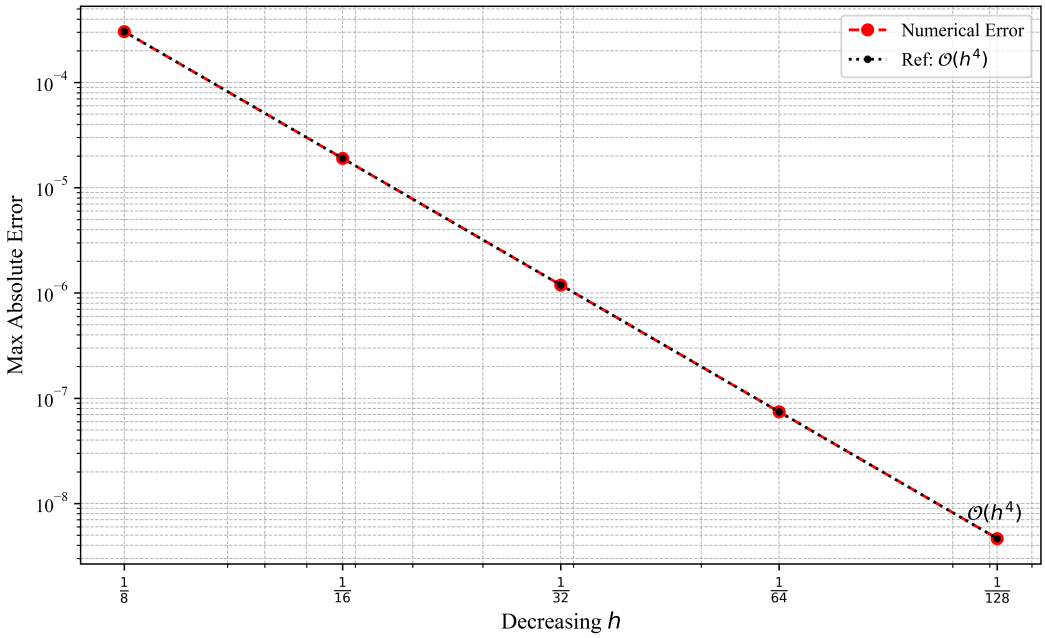
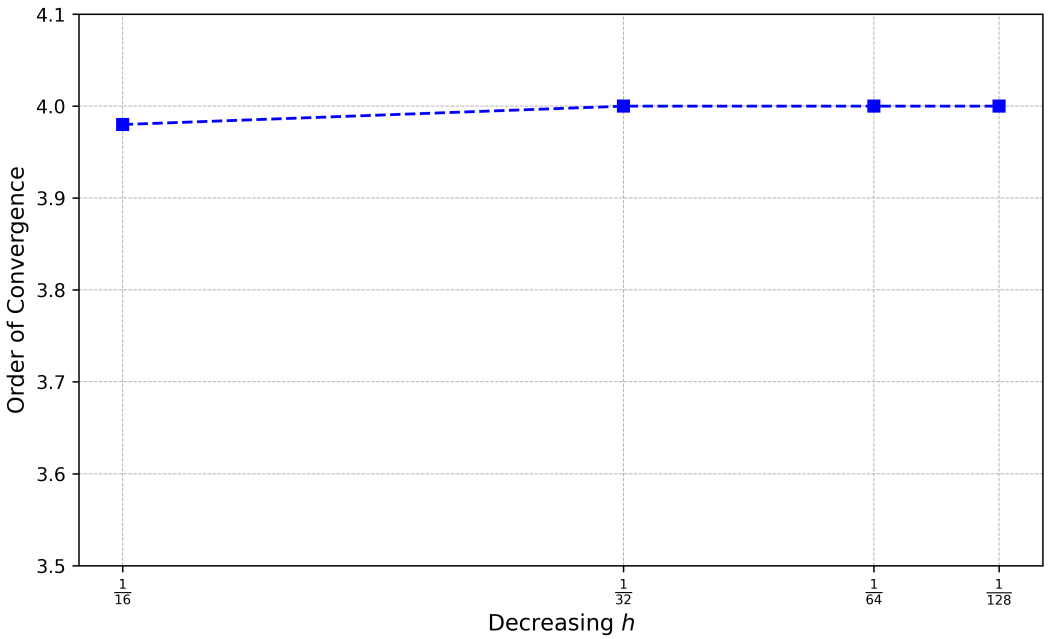


Figure 3.3.11: Behavior of CF-2 Error for Example-4 with Different h values



4.00

Figure 3.3.12: Behavior of CF-2 Order for Example-4 with different h values

It is clear from the above results that the scheme demonstrates excellent agreement with the exact solutions and exhibits a rapid decay in error as the grid is refined. The computed error confirm that the scheme achieves a convergence rate consistent with fourth-order accuracy in space, thereby validating the theoretical predictions. The stability analysis performed in section 4.2 verifies that the scheme is unconditionally stable when applied with Crank-Nicolson time discretization.

Chapter 4

Stability of Proposed Schemes

The stability defined in 1.3,plays a critical role in ensuring that the computed solution remains bounded and physically meaningful throughout the simulation. For time-dependent PDEs, particularly parabolic equations, a stable scheme ensures that small perturbations or discretization errors introduced during the computation do not grow uncontrollably with time. Stability, together with consistency, is a necessary condition for convergence as per the Lax equivalence theorem defined in 1.3, which forms the a fundamental principle in the analysis of finite difference methods.

In this chapter, we examine the stability properties of the two proposed compact finite difference schemes developed in Chapter 2 and Chapter 3. Both schemes are derived using Crank-Nicolson in time discretization, which is known to be unconditionally stable for linear diffusion problems when applying second-order central differences.

To assess the stability of these schemes, we are using Von Neumann (Fourier) stability analysis, a widely used method for analyzing linear constant-coefficient finite difference schemes. This technique is particularly effective in evaluating how individual Fourier modes of the numerical error evolve over time. For both schemes, we derive the amplification factor and determine the conditions under which its magnitude remains less than or equal to one, ensuring bounded growth of the numerical solution.

The stability analysis for the scheme which derived in Chapter 2 derived in section 4.1. In the section 4.2, we analyze stability scheme derived in Chapter 3.

4.1 Stability Analysis for Operator Method

In this section we discuss the stability of the scheme derived in Chapter 2, we are using m and n for representing the discretization of space and time respectively.
Now, consider the numerical approximation u_m^n of the solution $u(x_m,t_n)$ at spatial point $x_m = x_0 + mh$ and time $t_n = nk$. The Crank-Nicolson scheme combined with the compact finite difference operator leads to a system of equations at each time step by equation (2.2.5), we have

$$\mathscr{A}U^{n+1} = \mathscr{B}U^n + \mathscr{C},$$

where

$U^n = [u_1^n, u_2^n, \ldots, u_{M-1}^n]^T$ is the vector of unknowns,

$\mathscr{A}$ and $\mathscr{B}$ are tridiagonal matrices defined by coefficients W, X, Y and $\tilde{W}, \tilde{X}, \tilde{Y}$,

$\mathscr{C}$ is a vector incorporating boundary conditions.

The Von Neumann method assumes solutions of the form

$$u_m^n = \xi^n e^{i\beta mh},$$

where, $i = \sqrt{-1}$ is the imaginary unit, β is the spatial wave number, ξ is the amplification factor, which controls the growth or decay of the mode.

Lemma 1

Fourier Mode Stability Condition For stability, it is required that

$$|\mathscr{X}| \leq 1, \quad \forall \xi$$

Applying the Crank-Nicolson scheme to the spatially discretized equation yields a relation involving matrices $\mathscr{A}$ and $\mathscr{B}$, both of which are tridiagonal with constant coefficients. When applied to a Fourier mode u_m^n, these matrices act as scalar multipliers, i.e., their action corresponds to multiplication by their respective symbols.

$$(\mathscr{A}U^{n+1})_m = Yu_{m-1}^{n+1} + Xu_m^{n+1} + Wu_{m+1}^{n+1},$$

and similarly for $\mathscr{B}U^n$. Substituting the Fourier mode form, we get

$$\begin{aligned} &Y\xi^{n+1}e^{i\beta(m-1)h} + X\xi^{n+1}e^{i\beta mh} + W\xi^{n+1}e^{i\beta(m+1)h} \\ &\quad = \tilde{Y}\xi^n e^{i\beta(m-1)h} + \tilde{X}\xi^n e^{i\beta mh} + \tilde{W}\xi^n e^{i\beta(m+1)h}. \end{aligned}$$

$$\xi\left(Ye^{-i\beta h} + X + We^{i\beta h}\right) = \tilde{Y}e^{-i\beta h} + \tilde{X} + \tilde{W}e^{i\beta h}. \tag{4.1.1}$$

$$\begin{aligned} W = Y = \frac{1}{12k} - \frac{\alpha}{2h^2}, \quad X = \frac{10}{12k} + \frac{\alpha}{h^2}, \\ \tilde{W} = \tilde{Y} = \frac{1}{12k} + \frac{\alpha}{2h^2}, \quad \tilde{Q} = \frac{10}{12k} - \frac{\alpha}{h^2}. \end{aligned}$$

Because $W = Y$ and $\tilde{W} = \tilde{Y}$, we can write

$$\xi = \frac{\tilde{X} + 2\tilde{W}\cos(\beta h)}{X + 2W\cos(\beta h)}. \tag{4.1.2}$$

Note: $\cos(\beta h) \in [-1, 1]$ as β varies over all possible wave numbers.

$$\begin{aligned} N &= \frac{10}{12k} - \frac{\alpha}{h^2} + \frac{2}{12k}\cos(\beta h) + \frac{\alpha}{h^2}\cos(\beta h), \\ &= \frac{10 + 2\cos(\beta h)}{12k} - \frac{\alpha}{h^2}(1 - \cos(\beta h)). \\ D &= \frac{10}{12k} + \frac{\alpha}{h^2} + \frac{2}{12k}\cos(\beta h) - \frac{\alpha}{h^2}\cos(\beta h), \\ &= \frac{10 + 2\cos(\beta h)}{12k} + \frac{\alpha}{h^2}(1 - \cos(\beta h)). \end{aligned}$$

$$\mathscr{I}_1 = \frac{10 + 2\cos(\beta h)}{12k} > 0, \quad \mathscr{I}_2 = \frac{\alpha}{h^2}(1 - \cos(\beta h)) \geq 0,$$

since $1 - \cos(\beta h) \geq 0$.Then

$$\xi = \frac{\mathscr{I}_1 - \mathscr{I}_2}{\mathscr{I}_1 + \mathscr{I}_2}.$$

Stability condition $|\xi| \leq 1$,Since $\mathscr{I}_1 > 0$ and $\mathscr{N} \geq 0$, ξ is real, and

$$|\xi| = \left|\frac{\mathscr{I}_1 - \mathscr{I}_2}{\mathscr{I}_1 + \mathscr{I}_2}\right| \leq 1,$$

because $N \leq D$ and both positive. For $\beta h = 0$, $\cos(0) = 1 \Rightarrow \mathscr{I}_2 = 0$, so $\xi = 1$, meaning no amplification or damping of the zero mode. For all other modes, $1 - \cos(\beta h) > 0$, so $|\xi| < 1$. the method is unconditionally stable for all mesh sizes h and time steps k.

Truncation Error

Now we discuss the truncation error of the scheme derived in Chapter 2, The goal is to analyze how well the discrete scheme approximates the continuous diffusion equation as the mesh sizes tend to zero, So consider the scheme from equation (2.2.3), we have

$$\mathscr{Z}\frac{u_m^{n+1}-u_i^n}{k}=\frac{\alpha}{2}\left(\mathscr{Z}\delta_x^2u_m^{n+1}+\mathscr{Z}\delta_x^2u_m^n\right)$$

where,

$$\delta_x^2u_m^n=\frac{u_{m+1}^n-2u_m^n+u_{m-1}^n}{h^2}$$

local truncation error τ_m^n is defined by substituting the exact solution $u(x_m,t_n)$ into the discrete scheme:

$$\tau_m^n=\mathscr{Z}\frac{u(x_m,t_{n+1})-u(x_m,t_n)}{k}-\frac{\alpha}{2}\left(\mathscr{Z}\delta_x^2u(x_m,t_{n+1})+\mathscr{Z}\delta_x^2u(x_m,t_n)\right) \tag{4.1.3}$$

We want to find τ_m^n in terms of powers of k and h. At fixed x_m, expand u about time t_n,

$$u(x_m,t_{n+1})=u+ku_t+\frac{k^2}{2}u_{tt}+\frac{k^3}{6}u_{ttt}+\mathscr{O}(k^4)$$

where derivatives are evaluated at (x_m,t_n).

$$\frac{u(x_m,t_{n+1})-u(x_m,t_n)}{k}=u_t+\frac{k}{2}u_{tt}+\frac{k^2}{6}u_{ttt}+\mathscr{O}(k^3)$$

Expand u at neighboring points,

$$u(x_{m\pm1},t_n)=u\pm hu_x+\frac{h^2}{2}u_{xx}\pm\frac{h^3}{6}u_{xxx}+\frac{h^4}{24}u_{xxxx}+\mathscr{O}(h^5)$$

$$\delta_x^2u_m^n=\frac{u_{m+1}^n-2u_m^n+u_{m-1}^n}{h^2}=u_{xx}+\frac{h^2}{12}u_{xxxx}+\mathscr{O}(h^4)$$

$$\mathscr{Z}\delta_x^2u_m^n=u_{xx}+\mathscr{O}(h^4)$$

$$u_{xx}(x_m,t_{n+1})=u_{xx}+ku_{xxt}+\frac{k^2}{2}u_{xxtt}+\mathscr{O}(k^3)$$

So,

$$\mathscr{Z}\delta_x^2u_m^{n+1}=u_{xx}+ku_{xxt}+O(k^2)+\mathscr{O}(h^4)$$

At t_n:

$$\mathscr{Z}\delta_x^2u_m^n=u_{xx}+\mathscr{O}(h^4)$$

$$\tau_m^n=\left(u_t+\frac{k}{2}u_{tt}+\frac{k^2}{6}u_{ttt}+\mathscr{O}(k^3)+\mathscr{O}(h^4)\right)-\frac{\alpha}{2}\left((u_{xx}+ku_{xxt}+\mathscr{O}(k^2)+\mathscr{O}(h^4))+(u_{xx}+\mathscr{O}(h^4))\right)$$

$$\tau_m^n=u_t+\frac{k}{2}u_{tt}+\frac{k^2}{6}u_{ttt}-\alpha u_{xx}-\frac{\alpha k}{2}u_{xxt}+\mathscr{O}(k^2)+\mathscr{O}(h^4) \tag{4.1.4}$$

From the diffusion equation,

$$u_t=\alpha u_{xx} \tag{4.1.5}$$

$$u_{tt}=\alpha u_{xxt},\quad u_{ttt}=\alpha u_{xxtt} \tag{4.1.6}$$

on substituting values from equation (4.1.6)into τ_m^nin equation (4.1.4),

$$\tau_m^n=\alpha u_{xx}+\frac{\alpha k}{2}u_{xxt}+\frac{\alpha k^2}{6}u_{xxtt}-\alpha u_{xx}-\frac{\alpha k}{2}u_{xxt}+\mathscr{O}(k^2)+\mathscr{O}(h^4)$$

$$\tau_m^n=\frac{\alpha k^2}{6}u_{xxtt}+\mathscr{O}(k^2)+\mathscr{O}(h^4)$$

The truncation error at node (m,n) satisfies, $\tau_m^n=\mathscr{O}(k^2)+\mathscr{O}(h^4)$, showing that the scheme is second-order accurate in time and fourth-order accurate in space, hence the scheme is consistent.

4.2 Stability Analysis for General Method

In this section we discuss the stability of the scheme derived in Chapter 2. We assume the solution at grid points (x_m, t_n) is represented as a Fourier mode,

$$u_m^n = \hat{u}^n e^{i\xi x_m} \tag{4.2.1}$$

where,$\hat{u}^n$ is the amplitude at time step n, ξ is the wave number.,$i = \sqrt{-1}$ is the imaginary unit. Since we analyze error propagation, we define the growth factor G as:

$$u_m^{n+1} = \mathscr{X} u_m^n \tag{4.2.2}$$

which implies

$$u_m^{n+1} = \mathscr{X} \hat{u}^n e^{i\xi x_m}$$

Lemma 1

Fourier Mode Stability Condition For stability, it is required that

$$|\mathscr{X}| \leq 1, \quad \forall \xi$$

The Crank-Nicolson scheme applied to the spatially discretized equation gives by equation (3.2.2),

$$F u_{m+1}^{n+1} + G u_m^{n+1} + H u_{m-1}^{n+1} = \tilde{F} u_{m+1}^n + \tilde{G} u_m^n + \tilde{H} u_{m-1}^n \tag{4.2.3}$$

in which the coefficients are

$$F = -\frac{(K_1+K_2)}{2} - \frac{(K_3+K_4)}{k}, \quad \tilde{F} = \frac{(K_1+K_2)}{2} - \frac{(K_3+K_4)}{k}$$

$$G = \frac{2K_1-\gamma}{2} - \frac{(1-2K_3)}{k}, \quad \tilde{G} = -\frac{2K_1-\gamma}{2} - \frac{(1-2K_3)}{k}$$

$$H = -\frac{(K_1-K_2)}{2} - \frac{(K_3-K_4)}{k}, \quad \tilde{H} = \frac{(K_1-K_2)}{2} - \frac{(K_3-K_4)}{k}$$

On substituting the Fourier Mode Solution, Using $u_m^n = \hat{u}^n e^{i\xi x_m}$, we substitute into the equation 4.2.3,

$$F\hat{u}^{n+1} e^{i\xi x_{m+1}} + G\hat{u}^{n+1} e^{i\xi x_m} + H\hat{u}^{n+1} e^{i\xi x_{m-1}} = \tilde{F}\hat{u}^n e^{i\xi x_{m+1}} + \tilde{G}\hat{u}^n e^{i\xi x_m} + \tilde{H}\hat{u}^n e^{i\xi x_{m-1}}$$

$$F\hat{u}^{n+1} e^{i\xi h} + G\hat{u}^{n+1} + H\hat{u}^{n+1} e^{-i\xi h} = \tilde{F}\hat{u}^n e^{i\xi h} + \tilde{G}\hat{u}^n + \tilde{H}\hat{u}^n e^{-i\xi h}$$

Since $\hat{u}^{n+1} = F\hat{u}^n$, we substitute $\mathscr{X}$:

$$F\mathscr{X}\hat{u}^n e^{i\xi h} + G\mathscr{X}\hat{u}^n + H\mathscr{X}\hat{u}^n e^{-i\xi h} = \tilde{F}\hat{u}^n e^{i\xi h} + \tilde{G}\hat{u}^n + \tilde{H}\hat{u}^n e^{-i\xi h}$$

$$F\mathscr{X} e^{i\xi h} + G\mathscr{X} + H\mathscr{X} e^{-i\xi h} = \tilde{F} e^{i\xi h} + \tilde{G} + \tilde{H} e^{-i\xi h}$$

Now we are solving for $\mathscr{X}$,

$$\mathscr{X} = \frac{\tilde{F} e^{i\xi h} + \tilde{G} + \tilde{H} e^{-i\xi h}}{F e^{i\xi h} + G + H e^{-i\xi h}} \tag{4.2.4}$$

On using Euler's identities $e^{i\xi h} = \cos(\xi h) + i\sin(\xi h), \quad e^{-i\xi h} = \cos(\xi h) - i\sin(\xi h)$ in equation (4.2.4), we have

$$\mathscr{X} = \frac{(\tilde{F}+\tilde{H})\cos\xi h + i(\tilde{F}-\tilde{H})\sin\xi h + \tilde{G}}{(F+H)\cos\xi h + i(F-H)\sin\xi h + G} \tag{4.2.5}$$

Lemma 2

The Crank-Nicolson method is unconditionally stable if the modulus of the growth factor satisfies. $|\mathscr{X}| \leq 1, \forall h,k$ *which ensures the numerical solution does not amplify errors over time.*

Now we are computing $|\mathscr{X}|^2$for all h^2, β^2, α, k all are positive and here we are taking $\gamma = 0$, i.e. firstly we check the stability for equation (1.2.3) , Now on putting all the values in equation 4.2.5, we get

$$|\mathscr{X}|^2 = \frac{N}{D} = \frac{\mathscr{J}_1 - \mathscr{J}_2}{\mathscr{J}_1 + \mathscr{J}_2} \tag{4.2.6}$$

Where,

$$\begin{aligned} N = {} & 204h^4\alpha^2 - 432h^2k\alpha^3 + 432k^2\alpha^4 + h^6\beta^2 - 24h^4k\alpha\beta^2 + 108h^2k^2\alpha^2\beta^2 + 3h^4k^2\beta^4 \\ & + 80h^4\alpha^2\cos(h\xi) + 384h^2k\alpha^3\cos(h\xi) - 576k^2\alpha^4\cos(h\xi) + 32h^4k\alpha\beta^2\cos(h\xi) - 96h^2k^2\alpha^2\beta^2\cos(h\xi) \\ & - 4h^4k^2\beta^4\cos(h\xi) + 4h^4\alpha^2\cos(2h\xi) + 48h^2k\alpha^3\cos(2h\xi) + 144k^2\alpha^4\cos(2h\xi) - h^6\beta^2\cos(2h\xi) \\ & - 8h^4k\alpha\beta^2\cos(2h\xi) - 12h^2k^2\alpha^2\beta^2\cos(2h\xi) + h^4k^2\beta^4\cos(2h\xi) \end{aligned}$$

$$\begin{aligned} D = {} & 204h^4\alpha^2 + 432h^2k\alpha^3 + 432k^2\alpha^4 + h^6\beta^2 + 24h^4k\alpha\beta^2 + 108h^2k^2\alpha^2\beta^2 + 3h^4k^2\beta^4 \\ & + 80h^4\alpha^2\cos(h\xi) - 384h^2k\alpha^3\cos(h\xi) - 576k^2\alpha^4\cos(h\xi) - 32h^4k\alpha\beta^2\cos(h\xi) - 96h^2k^2\alpha^2\beta^2\cos(h\xi) \\ & - 4h^4k^2\beta^4\cos(h\xi) + 4h^4\alpha^2\cos(2h\xi) - 48h^2k\alpha^3\cos(2h\xi) + 144k^2\alpha^4\cos(2h\xi) - h^6\beta^2\cos(2h\xi) \\ & + 8h^4k\alpha\beta^2\cos(2h\xi) - 12h^2k^2\alpha^2\beta^2\cos(2h\xi) + h^4k^2\beta^4\cos(2h\xi) \end{aligned}$$

and

$$\begin{aligned} \mathscr{J}_1 = {} & 204h^4\alpha^2 + 432k^2\alpha^4 + h^6\beta^2 + 108h^2k^2\alpha^2\beta^2 + 3h^4k^2\beta^4 + 80h^4\alpha^2\cos(h\xi) - 576k^2\alpha^4\cos(h\xi) \\ & - 96h^2k^2\alpha^2\beta^2\cos(h\xi) - 4h^4k^2\beta^4\cos(h\xi) + 4h^4\alpha^2\cos(2h\xi) + 144k^2\alpha^4\cos(2h\xi) - h^6\beta^2\cos(2h\xi) \\ & - 12h^2k^2\alpha^2\beta^2\cos(2h\xi) + h^4k^2\beta^4\cos(2h\xi) \end{aligned}$$

$$\begin{aligned} \mathscr{J}_2 = {} & 432h^2k\alpha^3 + 24h^4k\alpha\beta^2 - 384h^2k\alpha^3\cos(h\xi) - 32h^4k\alpha\beta^2\cos(h\xi) - 48h^2k\alpha^3\cos(2h\xi) \\ & + 8h^4k\alpha\beta^2\cos(2h\xi) \end{aligned}$$

now to prove the stability condition from 4.2, we need to show that B is always positive and that the ratio $\frac{\mathscr{J}_1 - \mathscr{J}_2}{\mathscr{J}_1 + \mathscr{J}_2}$ is always less than one. This can be achieved by applying the trigonometric identity $\cos(2h\xi) = 2\cos^2(h\xi) - 1$, which simplifies the expression for B. Using this identity, we can confirm that $\mathscr{J}_2$ remains positive and that the ratio $\frac{\mathscr{J}_1 - \mathscr{J}_2}{\mathscr{J}_1 + \mathscr{J}_2}$ is less than one, thereby ensuring the stability of the numerical method, we get

$$\mathscr{J}_2 = 16\left((30h^2k\alpha^3 + hk\alpha\beta^2) - 2(12h^2k\alpha^3 + h^4k\alpha\beta^2)\cos(h\xi) + (-6h^2k\alpha^3 + h^4k\alpha\beta^2)\cos^2(h\xi)\right)$$

The constant term, coefficient of $\cos(h\xi)$ and coefficient of $\cos^2(h\xi)$ are positive as h^2, β^2, k and α all are positive, now we get $\mathscr{J}_2 = C_0 - 2C_1\cos(h\xi) + C_2\cos^2(h\xi)$, where,$C_0 = 16(30h^2k\alpha^3 + hk\alpha\beta^2), C_1 = 16(12h^2k\alpha^3 + h^4k\alpha\beta^2)$ and $C_2 = 16(-6h^2k\alpha^3 + h^4k\alpha\beta^2)$, As can find the term B is always positive so the ratio $\frac{\mathscr{J}_1 - \mathscr{J}_2}{\mathscr{J}_1 + \mathscr{J}_2}$ will always be less than or equal to 1. the ratio of numerator and denominator are equal in magnitude, leading to $|\mathscr{X}| \leq 1, \quad \forall \xi$ .Now we are computing $|\mathscr{X}|^2$for all h^2, β^2, α, k all are positive and here we are taking $\gamma \neq 0$, i.e. we check the stability for equation (1.2.3) , Now on putting all the values in equation 4.2.5, we get

$$|\mathscr{X}|^2 = \frac{N}{D} \tag{4.2.7}$$

where,

$$
\begin{aligned}
N = -\{&-816h^4\alpha^2 + 1728h^2k\alpha^3 - 1728k^2\alpha^4 + 864h^3k\alpha^2\beta - 1728hk^2\alpha^3\beta - 4h^6\beta^2 + 96h^4k\alpha\beta^2 \\
&- 864h^2k^2\alpha^2\beta^2 - 144h^3k^2\alpha\beta^3 - 12h^4k^2\beta^4 - 1104h^4k\alpha^2\gamma + 1440h^2k^2\alpha^3\gamma + 72h^5k\alpha\beta\gamma \\
&+ 576h^3k^2\alpha^2\beta\gamma + 48h^4k^2\alpha\beta^2\gamma - 12h^5k^2\beta^3\gamma - 396h^4k^2\alpha^2\gamma^2 + 60h^5k^2\alpha\beta\gamma^2 - 3h^6k^2\beta^2\gamma^2 \\
&+ 4\Big(576k^2\alpha^4 + 576hk^2\alpha^3\beta + h^6k^2\beta^2\gamma^2 + 4h^5k\beta\gamma(k\beta^2 - 4\alpha(1+k\gamma)) - 48h^2k\alpha^2 \\
&(-5k\beta^2 + 8\alpha(1+k\gamma)) - 48h^3k\alpha\beta(-k\beta^2 + \alpha(4+3k\gamma)) + 4h^4\Big(k^2\beta^4 - 2k\alpha\beta^2(4+k\gamma) \\
&+ \alpha^2(-20 - 4k\gamma + 7k^2\gamma^2)\Big)\Big)\cos(h\xi) - \Big(576k^2\alpha^4 + 576hk^2\alpha^3\beta + 48h^3k\alpha\beta(2\alpha + k\beta^2) \\
&+ h^6\beta^2(-4 + k^2\gamma^2) + 4h^5k\beta\gamma(k\beta^2 + \alpha(2 - k\gamma)) - 96h^2k\alpha^2(-k\beta^2 + \alpha(-2 + k\gamma)) \\
&+ 4h^4\Big(k^2\beta^4 + 4k\alpha\beta^2(-2 + k\gamma) + \alpha^2(-2 + k\gamma)^2\Big)\Big)\cos(2h\xi)\Big\}
\end{aligned}
$$

and

$$
\begin{aligned}
D = &\, 816h^4\alpha^2 + 1728h^2k\alpha^3 + 1728k^2\alpha^4 + 864h^3k\alpha^2\beta + 1728hk^2\alpha^3\beta + 4h^6\beta^2 + 96h^4k\alpha\beta^2 \\
&+ 864h^2k^2\alpha^2\beta^2 + 144h^3k^2\alpha\beta^3 + 12h^4k^2\beta^4 + 816h^4k\alpha^2\gamma + 864h^2k^2\alpha^3\gamma + 72h^5k\alpha\beta\gamma \\
&+ 576h^3k^2\alpha^2\beta\gamma + 144h^4k^2\alpha\beta^2\gamma + 12h^5k^2\beta^3\gamma + 204h^4k^2\alpha^2\gamma^2 + 36h^5k^2\alpha\beta\gamma^2 + 3h^6k^2\beta^2\gamma^2 \\
&- 4\Big(576k^2\alpha^4 + 576hk^2\alpha^3\beta + h^6k^2\beta^2\gamma^2 + 4h^5k\beta\gamma(k\beta^2 + 2\alpha(2+k\gamma)) + 48h^2k \\
&\alpha^2(5k\beta^2 + 4\alpha(2+k\gamma)) + 48h^3k\alpha\beta(k\beta^2 + \alpha(4+3k\gamma)) - 4h^4\Big(-k^2\beta^4 + 5\alpha^2(2+k\gamma)^2 \\
&- 2k\alpha\beta^2(4+5k\gamma)\Big)\Big)\cos(h\xi) + \Big(576k^2\alpha^4 + 576hk^2\alpha^3\beta + 48h^3k\alpha\beta(-2\alpha + k\beta^2) \\
&+ h^6\beta^2(-4 + k^2\gamma^2) + 4h^5k\beta\gamma(k\beta^2 - \alpha(2+k\gamma)) - 96h^2k\alpha^2(-k\beta^2 + 4h^4 \\
&\Big(-k^2\beta^4 + 5\alpha^2(2+k\gamma)^2 - 2k\alpha\beta^2(4+5k\gamma)\Big)\Big)\cos(h\xi) + \Big(576k^2\alpha^4 + 576hk^2\alpha^3\beta \\
&+ 48h^3k\alpha\beta(-2\alpha + k\beta^2) + h^6\beta^2(-4 + k^2\gamma^2) + 4h^5k\beta\gamma(k\beta^2 - \alpha(2+k\gamma)) - \\
&96h^2k\alpha^2(-k\beta^2 + \alpha(2+k\gamma)) + 4h^4\Big(k^2\beta^4 + 4k\alpha\beta^2(2+k\gamma) + \alpha^2(2+k\gamma)^2\Big)\Big) \\
&\cos(2h\xi)\alpha(2+k\gamma)) + 4h^4\Big(k^2\beta^4 + 4k\alpha\beta^2(2+k\gamma) + \alpha^2(2+k\gamma)^2\Big)\Big)\cos(2h\xi)
\end{aligned}
$$

Here $N \le D$,i.e. $|\mathscr{X}| \le 1, \quad \forall\xi$. Fig. (4.2.1) represents by experimental value of $|\mathscr{X}|^2 \le 1$, at the $h = \frac{1}{32}$ and $k = h^2$ Thus, the Proposed scheme in section 3.2 is unconditionally stable.

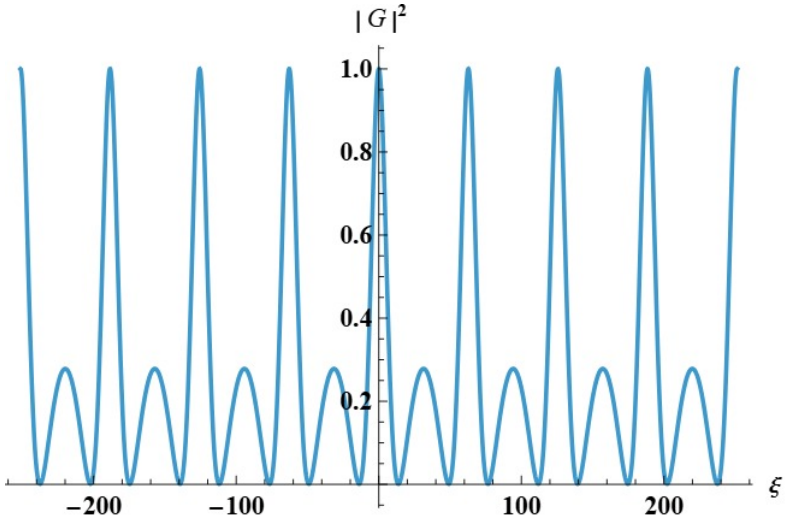


Figure 4.2.1: Plot of Stability with, $-80\pi \le \xi \le 80\pi$

This chapter presented the stability analysis of the proposed compact finite difference schemes. The operator-based scheme for the diffusion equation was shown to be unconditionally stable, while the

generalized scheme for the ADR equation exhibited conditional stability, depending on the advection and reaction terms. These theoretical findings are consistent with the numerical results already demonstrated in the respective chapters, confirming the effectiveness of both schemes.

Future Work

Application to Variable Coefficients Problems

In this work, we have developed and implemented a fourth-order compact finite difference scheme combined with the Crank-Nicolson method, successfully applied to one-dimensional parabolic PDEs with constant coefficients. Building on this foundation, we have extended the scheme to handle variable-coefficient problems of the form,

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(a(x)\frac{\partial u}{\partial x}\right) - b(x)\frac{\partial u}{\partial x} + c(x)u,$$

where spatially varying diffusion $a(x)$, advection $b(x)$, and reaction $c(x)$ terms introduce additional complexity. Although the extended scheme retains the same mathematical structure, current numerical results fall short of achieving the desired fourth-order spatial accuracy. This highlights the need for further refinement in discretizing variable coefficients and ensuring stability. Ongoing efforts include the analytical derivation of truncation error expressions and improved matrix formulations tailored to variable coefficients. Resolving these challenges is critical for attaining consistent, high-order accuracy in practical computations.

Moreover, we are actively exploring applications to Fokker-Planck equations and Black-Scholes-type models, which naturally involve variable coefficients and complex boundary conditions. Successfully adapting the compact scheme to these important classes of PDEs would demonstrate its broader utility in statistical physics, population dynamics, and financial mathematics. Looking ahead, future research will focus on extending the method to multidimensional problems, incorporating nonlinearities, and integrating advanced techniques such as adaptive mesh refinement (AMR) and operator splitting. These developments aim to enhance the efficiency and robustness of the high-order compact schemes, enabling their use in complex, real-world multiphysics simulations.

The Idea of scheme

In [27], the authors developed a high-order compact (HOC) scheme for time-fractional Fokker-Planck equations with variable convection terms. Their method utilizes the L1 scheme for time discretization and handles the variable coefficients through simple modifications of the coefficient matrices. The scheme is rigorously analyzed in matrix form using the energy method and is proven to be stable and convergent with order $\tau^{2-\alpha} + h^4$, which is higher than that of several recently studied schemes. Numerical experiments confirm the theoretical accuracy and stability.

In contrast, our current work adopts the Crank-Nicolson method for time discretization, aiming to achieve fourth-order accuracy in both time and space for the traditional (non-fractional) case where $\alpha = 1$. The development and detailed analysis of this Crank-Nicolson-based scheme are ongoing. However, the approach is conceptually similar to the general compact finite difference method described previously. By integrating the Crank-Nicolson scheme with the high-order spatial discretization, we expect to improve temporal accuracy while maintaining stability, thus advancing the numerical solution of the Fokker-Planck equation from [27].

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} + p(x)\frac{\partial u}{\partial x} + p'(x)u = f(x,t), \quad x \in [0,L], \quad t > 0.$$

We discretize the spatial domain $[0,L]$ into m uniform subintervals of size $h = \frac{L}{m}$ with grid points $x_i = ih$, $i = 0,1,\ldots,m$. For a grid function u_i^j approximating $u(x_i,t_j)$, To achieve fourth-order accuracy in space, the scheme introduces correction coefficients defined as

$$\mathscr{H}_i = 1 + \frac{h^2}{12}\left(\delta_x^2 - p_i\,\delta_x\right),$$
$$\mathscr{A}_i = 1 + \frac{h^2}{12}\left(p_i^2 - 2\delta_x\widehat{p}_i\right),$$
$$\mathscr{C}_i = p_i + \frac{h^2}{12}\left(\delta_x^2 p_i - p_i\,\delta_x\widehat{p}_i\right),$$

where $p_i = p(x_i)$ and $\delta_x\widehat{p}_i$, $\delta_x^2 p_i$ are discrete approximations of the first and second derivatives of $p(x)$ at x_i.

The temporal domain $[0,T]$ is divided into n uniform steps of size $\tau = \frac{T}{n}$, with time levels$t_j = j\tau$, $j = 0,1,\ldots,n$. The Crank-Nicolson method approximates the time derivative by a centered difference and evaluates spatial terms at the midpoint between time levels t_j and t_{j+1}, providing second-order accuracy in time and unconditional stability. For each interior spatial node $i = 1,2,\ldots,m-1$ and time step $j = 0,1,\ldots,n-1$, the scheme reads:

$$\mathscr{H}_i\frac{u_i^{j+1}-u_i^j}{\tau} - \frac{1}{2}\mathscr{A}_i\,\delta_x^2(u_i^{j+1}+u_i^j) + \frac{1}{2}\mathscr{C}_i\,\delta_x(u_i^{j+1}+u_i^j) + \frac{1}{2}\mathscr{H}_i p_i'(u_i^{j+1}+u_i^j)$$
$$= \mathscr{H}_i\frac{f_i^{j+1}+f_i^j}{2}.$$

This equation states:

- The time derivative $\partial u/\partial t$ is approximated by the backward difference quotient $\frac{u_i^{j+1}-u_i^j}{\tau}$, multiplied by $\mathscr{H}_i$ to incorporate spatial correction terms.
- The diffusion term u_{xx} is discretized using a high-order compact approximation δ_x^2, applied to the average $\frac{u_i^{j+1}+u_i^j}{2}$.
- The convection term $p(x)u_x$ is discretized similarly with δ_x and coefficient $\mathscr{C}_i$.
- The reaction term $p'(x)u$ is also discretized at the temporal midpoint.
- The source term $f(x,t)$ is approximated by the average $\frac{f_i^{j+1}+f_i^j}{2}$ to maintain second-order accuracy in time.

The solution must satisfy the prescribed boundary conditions at each time level:

$$u_0^k = \varphi_1^j, \quad u_M^j = \varphi_2^j, \quad \text{for } j = 0,1,\ldots,n,$$

and initial condition at time $t = 0$:

$$u_i^0 = \phi_i, \quad \text{for } i = 0,1,\ldots,m.$$

This scheme couples a fourth-order spatial discretization with the second-order Crank-Nicolson time stepping method. It is implicit and requires solving a linear system at each time step.The high-order spatial corrections ensure improved accuracy and better approximation of derivatives, especially useful when convection terms and variable coefficients are present. The Crank-Nicolson method provides stability and accuracy in time.

Application to Nonlinear Problems

In this work, we focused on linear parabolic PDEs with constant coefficients, However, many real-world systems are governed by nonlinear PDEs, where the solution behavior is more complex and analytical tools are limited.

A canonical example is the Fisher KPP equation, a nonlinear reaction diffusion model,

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2} + ru(1-u),$$

where the nonlinear term $ru(1-u)$ models logistic population growth with diffusion. Another example is the viscous Burgers' equation:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2},$$

which combines advection, nonlinearity, and diffusion, Extending compact schemes to such nonlinear equations involves challenges such as,

- The spatial operator becomes solution-dependent.
- Time-stepping may require nonlinear solvers (e.g., Newton Raphson iteration).
- Stability is no longer easily addressed via Fourier analysis; energy methods or discrete maximum principles may be required.

The Idea of scheme

One possible approach is to linearize the nonlinear terms via semi-implicit discretization, for example,

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \nu\delta_{xx}u_i^{n+1} - u_i^n\delta_x u_i^n,$$

where the diffusion term is treated implicitly using the compact operator, and the nonlinear advection term is treated explicitly. Future work may focus on deriving nonlinear compact schemes, analyzing their behavior under various discretization strategies, and applying them to physically meaningful models in biology, fluid dynamics, or chemical kinetics.

Extension to Non-Uniform Grids

In this work, the compact finite difference schemes have been formulated and implemented on uniform spatial grids, where the grid spacing $h = x_{i+1} - x_i$ remains constant. While this is standard in many numerical studies due to its simplicity and ease of analysis, many practical problems governed by advection-diffusion-reaction (ADR) equations exhibit steep gradients, localized fronts, or boundary layers that cannot be resolved efficiently using uniform grids.

In future work, the proposed compact finite difference schemes can be extended to non-uniform spatial grids, which are more effective for solving problems with localized features, sharp gradients, or boundary layers. A non-uniform mesh refers to a grid in which the spacing between consecutive points varies across the domain. Mathematically, for a spatial domain $[a,b]$, a non-uniform grid is represented by a sequence,

$$a = x_0 < x_1 < \cdots < x_N = b, \quad \text{with} \quad h_i = x_{i+1} - x_i, \text{ and } h_i \neq h_{i+1}.$$

This approach contrasts with uniform grids, where the grid spacing is constant. Non-uniform grids are particularly advantageous for capturing phenomena like boundary layers, internal layers, or reaction fronts without resorting to globally refined grids that increase computational cost unnecessarily.

Idea of Non-uniform Mesh

For example, consider the ADR equation given by (1.2.3), where spatial variation in $\beta(x)$, $\gamma(x)$, or $\alpha(x)$ can cause the solution to develop steep transitions. Solving such equations efficiently on uniform grids would require very small step sizes throughout the domain. Instead, non-uniform grids concentrate points only where needed.
A typical mesh transformation used to generate a non-uniform mesh is:

$$x_i = a + (b-a) \cdot \frac{\tanh\left(\sigma\left(\frac{2i}{n} - 1\right)\right)}{2\tanh(\sigma)}, \quad 0 \leq i \leq n,$$

where the parameter $\sigma > 0$ controls the clustering of points near boundaries or layers [31]. To apply compact finite difference schemes to such grids, classical finite difference approximations must be modified to account for variable spacing.

Conclusion

In this work, we developed and studied a high-order accurate numerical method for solving one-dimensional parabolic PDEs such as the diffusion equation and the advection-diffusion-reaction (ADR) equation. The proposed scheme uses a fourth-order compact finite difference method (CFDM) for spatial discretization and the Crank-Nicolson scheme for time discretization, which is second-order accurate in time and unconditionally stable.

We began by revisiting the traditional finite difference methods: the explicit (Forward Time Central Space, FTCS), implicit (Backward Time Central Space, BTCS), and Crank-Nicolson methods. We analyzed their stability, consistency, and truncation errors using Taylor series expansions and Von Neumann analysis. We observed that traditional methods are limited to second-order spatial accuracy and may require finer grids to achieve better results.

To improve accuracy, we derived a compact finite difference operator based on a sixth-order Taylor expansion and established a discrete relation using operator techniques. This allowed us to approximate the second derivative $\frac{\partial^2 u}{\partial x^2}$ with fourth-order accuracy using only three grid points. The resulting scheme forms a tridiagonal linear system at each time step, which is efficiently solvable.

For time integration, we used the Crank-Nicolson method, which is a two-level scheme that averages the spatial discretization at time levelst^j and t^{j+1}. The combined method was shown to have global accuracy of $\mathcal{O}(k^2+h^4)$, where k is the time step and h is the spatial step size.

We implemented both traditional and compact methods using Python and tested them on several model problems, including:

- Pure diffusion equations: $\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$
- Advection-diffusion equations: $\frac{\partial u}{\partial t} + \beta \frac{\partial u}{\partial x} = \alpha \frac{\partial^2 u}{\partial x^2}$
- Advection-diffusion-reaction equations: $\frac{\partial u}{\partial t} + \beta \frac{\partial u}{\partial x} + \gamma u = \alpha \frac{\partial^2 u}{\partial x^2}$

For all these problems, the compact scheme gave better results compared to traditional FDM. It produced smoother, more accurate solutions and required fewer spatial nodes to achieve the same accuracy, thereby reducing computational cost. As an extension of this work, we are currently focusing on applying the same fourth-order compact scheme to PDEs with variable coefficients. Although the scheme has been successfully developed for such cases, the numerical results obtained so far do not fully align with the theoretical expectations. Therefore, further refinement and analysis are ongoing.

In particular, we are investigating the application of this method to *Fokker-Planck equations* and *Black-Scholes-type equations*, which involve non-constant coefficients and are important in fields such as statistical physics and financial mathematics. This remains an active area of research, and we aim to improve the formulation to obtain stable and accurate solutions in these more complex scenarios.

Appendix

The pseudocode for the main numerical techniques covered and used in this paper, such as the suggested fourth-order compact finite difference approach in conjunction with the Crank-Nicolson time discretisation and traditional finite difference schemes, is provided in this appendix. The computational framework for numerically solving one-dimensional diffusion and advection-diffusion-reaction (ADR) equations is described by these pseudocodes.

For full implementation details, including complete and executable Python code, visualization scripts, and test cases, readers are encouraged to access the associated GitHub repository:
GitHub Repository: https://github.com/manojms3063/High-Order-Compact-Scheme-for-1D-Heat-and-ADR-Equations

Appendix - A : Python Pseudocode

A1: Pseudo code for Crank-Nicolson scheme

Input: Diffusion coefficient α, domain length L, final time T, grid spacing Δx
Compute time step $\Delta t = \Delta x^2$ and set $r = \dfrac{\alpha \Delta t}{\Delta x^2}$
Define number of spatial points $N_x = \dfrac{L}{\Delta x}$ and time steps $N_t = \dfrac{T}{\Delta t}$
Generate spatial grid $x \in [0, L]$ and initialize $u(x,0) = \sin(\pi x)$
Define exact solution $u_{\text{exact}}(x,t) = e^{-\pi^2 t}\sin(\pi x)$
Construct tridiagonal matrix A for Crank-Nicolson scheme:

- Lower diagonal: $-\dfrac{r}{2}$
- Main diagonal: $1 + r$
- Upper diagonal: $-\dfrac{r}{2}$

for each time step $n = 0$ to $N_t - 1$ **do**
    Compute right-hand side vector b:

$$b_i = \frac{r}{2}u_{i-1} + (1-r)u_i + \frac{r}{2}u_{i+1}$$

    Solve $A \cdot u^{n+1} = b$
end for
Compute numerical solution u_{num} and exact solution u_{exact} at $t = T$
Calculate pointwise absolute error $|u_{\text{num}} - u_{\text{exact}}|$ and display max error
Plot both u_{num} and u_{exact} over spatial grid
(Optional) Plot absolute error versus x

A2: Pseudocode for Operator CFDM scheme

Input: Spatial step h, time step $k = h^2$, domain $[x_0, x_m]$, final time T
Define grid points $x_i = x_0 + ih$, $i = 0, \ldots, m$, and time steps $t_j = jk$, $j = 0, \ldots, n$
Set α value
Define constants:

$$R = \frac{1}{12k} - \frac{\alpha}{2h^2}, \qquad Q = \frac{10}{12k} + \frac{\alpha}{h^2}, \qquad P = \frac{1}{12k} - \frac{\alpha}{2h^2}$$
$$Rb = \frac{1}{12k} + \frac{\alpha}{2h^2}, \qquad Qb = \frac{10}{12k} - \frac{\alpha}{h^2}, \qquad Pb = \frac{1}{12k} + \frac{\alpha}{2h^2}$$

Initialize solution matrix $u[i,0] = \sin(\pi x_i)$, $\forall i$
Apply boundary conditions: $u[0,j] = u[m,j] = 0$, for all j
Construct tridiagonal matrix A with diagonals:

$$\text{Lower} = R, \quad \text{Main} = Q, \quad \text{Upper} = P$$

for each time step $j = 0$ to $n-1$ **do**
    Compute vector C with:

$$C[0] = R \cdot u[0,j], \quad C[-1] = P \cdot u[m,j]$$

    for each spatial node $i = 1$ to $m-1$ **do**
        Compute RHS:

$$b[i-1] = Pb \cdot u[i+1,j] + Qb \cdot u[i,j] + Rb \cdot u[i-1,j] - C[i-1]$$

    end for
    Solve $A \cdot v = b$ for v
    Update $u[i, j+1] = v[i-1]$ for $i = 1$ to $m-1$
end for
Compute exact solution $v[i, j]$(exact solution of Problem)
Compute absolute error $e[i, j] = |u[i, j] - v[i, j]|$
Output $\max(e)$
Plot $u(x,T)$ vs exact $v(x,T)$
(Optional) Plot 3D surfaces for $u(x,t)$ and $v(x,t)$

A3: Pseudocode for General CFDM scheme

Input: Domain $[x_0, x_m]$, time T, step size h, $k = h^2$

Set α, β

Discretize $x_i = x_0 + ih$, $i = 0, 1, \ldots, m$ and $t_j = jk$, $j = 0, 1, \ldots, n$

Define coefficients:

$$S_1 = -\left(\frac{\alpha}{h^2} + \frac{\beta^2}{12\alpha}\right), \quad S_2 = \frac{\beta}{2h}, \quad S_3 = \frac{1}{12}, \quad S_4 = -\frac{\beta h}{24\alpha}$$
$$a_1 = S_3 - S_4, \quad a_2 = 1 - 2S_3, \quad a_3 = S_3 + S_4$$
$$a_4 = -(S_1 - S_2), \quad a_5 = 2S_1, \quad a_6 = -(S_1 + S_2)$$

Matrix coefficients:

$$P = -\frac{(S_1 + S_2)}{2} - \frac{S_3 + S_4}{k}, \qquad Pb = \frac{(S_1 + S_2)}{2} - \frac{S_3 + S_4}{k}$$
$$Q = S_1 - \frac{1 - 2S_3}{k}, \qquad Qb = -S_1 - \frac{1 - 2S_3}{k}$$
$$R = -\frac{(S_1 - S_2)}{2} - \frac{S_3 - S_4}{k}, \qquad Rb = \frac{(S_1 - S_2)}{2} - \frac{S_3 - S_4}{k}$$

Initialize $u(x,0) = e^{5x}\left[\cos\left(\frac{\pi}{2}x\right) + 0.25\sin\left(\frac{\pi}{2}x\right)\right]$

Apply boundary conditions for $u(0,t)$ and $u(1,t)$ using exact solution form

Construct tridiagonal matrix A with diagonals:

$$\text{Lower} = R, \quad \text{Main} = Q, \quad \text{Upper} = P$$

for each time level $j = 0$ to $n-1$ **do**

  Compute $C[0] = R \cdot u[0, j+1]$, $C[m-2] = P \cdot u[m, j+1]$

  for $i = 1$ to $m-1$ **do**

    Compute RHS:

$$b[i-1] = Pb \cdot u[i+1, j] + Qb \cdot u[i, j] + Rb \cdot u[i-1, j] - C[i-1]$$

  end for

  Solve $Av = b$

  Update $u[i, j+1] = v[i-1]$ for $i = 1$ to $m-1$

end for

Compute exact solution: $v(x,t)$(exact solution of Problem)

Compute absolute error: $err[i, j] = |u[i, j] - v[i, j]|$

Output: $\max(err)$

(Optional) Plot 3D surfaces for $u(x,t)$ and $v(x,t)$

Appendix - B : Tridiagonal System Solution

B1: Properties of Tridiagonal Matrices

The mathematical characteristics of tridiagonal matrices, which are commonly encountered in the discretisation of PDEs using finite differences, are covered in this appendix. Analysing the stability, effectiveness, and solvability of the numerical schemes employed in this thesis requires an understanding of these characteristics.

A matrix $A \in \mathbb{R}^{n \times n}$ is called tridiagonal if:

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

Only the main diagonal and the first sub- and super-diagonals contain non-zero elements.

1. Efficient Solvability: A tridiagonal system can be solved in $\mathcal{O}(n)$ time using the Thomas algorithm.

2. Storage Efficiency: Requires storing only three vectors: $a = [a_2, \ldots, a_n]$, $b = [b_1, \ldots, b_n]$, $c = [c_1, \ldots, c_{n-1}]$.

3. Stability under Diagonal Dominance: If $|b_i| > |a_i| + |c_i|$ for all i, then the matrix is strictly diagonally dominant, which implies nonsingularity and stability of the numerical method.

4. Symmetric Positive Definite (SPD) Case: If A is symmetric and all its eigenvalues are positive, then A is SPD. This is often true for matrices arising in diffusion problems with Neumann or Dirichlet conditions.

5. Eigenvalue Bounds: Every eigenvalue λ of a tridiagonal matrix lies within at least one Gershgorin disc:
$$|\lambda - b_i| \leq |a_i| + |c_i|$$

This can be used to estimate the spectral radius and stability.

B2: Thomas Algorithm for Tridiagonal Systems

In this appendix, we present the Thomas algorithm, a simplified form of Gaussian elimination specifically designed to solve tridiagonal linear systems.When solving differential equations using the Finite Difference Method (FDM) , the continuous domain is replaced with a discrete grid, converting the differential equation into a system of linear equations. This system is often tridiagonal, where each equation involves three adjacent unknowns (variables), corresponding to the neighboring grid points. To solve this tridiagonal system, we use a specialized algorithm, such as the Thomas algorithm, The algorithm is an efficient direct method designed specifically for tridiagonal matrices, reducing the computational complexity from $O(n^3)$ (in general Gaussian elimination) to $O(n)$, where n is the number of grid points or unknowns.

The basic steps of the Thomas algorithm are:

1. Forward Elimination: This step modifies the coefficients of the system to eliminate the terms below the main diagonal, transforming the system into an upper-triangular form.

2. Back Substitution: After forward elimination, the system is solved by back-substitution, starting from the last equation and moving upward.

By using this method, we can efficiently solve the system of linear equations arising from the finite difference discretization of the differential equation. A tridiagonal matrix looks like this:

$$A = \begin{bmatrix} a_1 & b_1 & 0 & 0 & \dots & 0 \\ c_1 & a_2 & b_2 & 0 & \dots & 0 \\ 0 & c_2 & a_3 & b_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & c_{n-1} & a_n & b_{n-1} \\ 0 & \dots & 0 & 0 & c_{n-1} & a_n \end{bmatrix}$$

The system of equations associated with the matrix is:

$$A\mathbf{x} = \mathbf{b}$$

Where:

$\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ is the vector of unknowns.

$\mathbf{b} = [b_1, b_2, \dots, b_n]^T$ is the constant vector.

To solve a tridiagonal system efficiently, we use the Thomas algorithm. The Thomas algorithm is a specialized form of Gaussian elimination for tridiagonal systems, which reduces computational complexity to $O(n)$.

Thomas Algorithm Steps Given a system:

$$a_1 x_1 + b_1 x_2 = d_1$$

$$c_1 x_1 + a_2 x_2 + b_2 x_3 = d_2$$

$$\vdots$$

$$c_{n-1} x_{n-2} + a_{n-1} x_{n-1} + b_{n-1} x_n = d_{n-1}$$

$$c_{n-1} x_n = d_n$$

Forward Elimination: first eliminate the variables below the main diagonal, transforming the system into an upper-triangular form. For each row i (from 2 to n):

$$w_i = \frac{a_i}{c_{i-1}}, \quad c_i' = c_i - w_i b_{i-1}$$

$$d_i' = d_i - w_i d_{i-1}$$

Back Substitution: Once the system is in upper-triangular form, we can back-substitute to find the unknowns:

$$x_n = \frac{d_n'}{c_n'}$$

$$x_i = \frac{d_i' - b_i x_{i+1}}{c_i'} \quad \text{for } i = n-1, n-2, \dots, 1$$

This will provide the solution of Tri-diagonal System of equations.

References

1. M. Dehghan, Weighted finite difference techniques for the one-dimensional advection-diffusion equation, Appl. Math. Comput. 147 (2004) 307-319

2. P. Roache, Computational Fluid Dynamics, Hermosa Press, Albuquerque, NM, 1972.

3. D.B. Spalding, A novel finite difference formulation for differential expressions involving both first and second derivatives, Int. J. Numer. Methods Eng.4 (1972) 551-559.

4. M. Dehghan, Numerical solution of the three-dimensional advection-diffusion equation, Appl. Math. Comput. 150 (2004) 5-19

5. Chemeda, Hailu Muleta, Negassa, Ayana Deressa, Merga, Feyisa Edosa, Compact Finite Difference scheme Combined with Richardson Extrapolation for Fisher's Equation, Mathematical Problems in Engineering, 2022, 7887076, 7 pages, 2022.

6. J.Y. Parlarge, Water transport in soils, Ann. Rev. Fluids Mech. 2 (1980) 77-102.

7. Q.N. Fattah, J.A. Hoopes, Dispersion in anisotropic homogeneous porous media, J. Hydraul. Eng. 111 (1985) 810-827

8. P.C. Chatwin, C.M. Allen, Mathematical models of dispersion in rivers and estuaries, Ann. Rev. Fluid Mech. 17 (1985) 119-149.

9. F.M. Holly, J.M. Usseglio-Polatera, Dispersion simulation in two-dimensional tidal flow, J. Hydraul. Eng. 111 (1984) 905-926

10. J.R. Salmon, J.A. Liggett, R.H. Gallager, Dispersion analysis in homogeneous lakes, Int. J. Numer. Methods Eng. 15 (1980) 1627-1642

11. Z. Zlatev, R. Berkowicz, L.P. Prahm, Implementation of a variable step-size variable formula in the time- integration part of a code for treatment of longrange transport of air pollutants, J. Comput. Phys. 55 (1984) 278-301.

12. L. Lapidus, N.R. Amundston, Mathematics of absorption in beds, J. Phys. Chem. 56 (8) (1952) 984-988.

13. C.R. Gane, P.L. Stephenson, An explicit numerical method for solving transient combined heat conduction and convection problems, Int. J. Numer. Methods Eng. 14 (1979) 1141-1163.

14. M.H. Chaudhry, D.E. Cass, J.E. Edinger, Modelling of unsteady-flow water temperatures, J. Hydraul. Eng. 109 (5) (1983) 657-669

15. H. Karahan, Unconditional stable explicit finite difference technique for the advection-diffusion equation using spreadsheets, Adv. Eng. Software 38(2007) 80-86.

16. H. Karahan, Implicit finite difference techniques for the advection-diffusion equation using spreadsheets, Adv. Eng. Software 37 (2006) 601-608

17. H. Sun, J. Zhang, A high-order compact boundary value method for solving one-dimensional heat equa- tions, Numer. Methods Partial Differ. Eq. 19 (2003) 846-857.

18. B.J. Noye, A new third-order finite difference method for transient one-dimensional advection-diffusion, Commun. Appl. Numer. Methods 6 (1990) 279-288.

19. High-order compact solution of the one-dimensional heat and advection-diffusion equations Akbar Mo- hebbi, Mehdi Dehghan.

20. M. Zerroukat, K. Djidjel, A. Charafi, Explicit and implicit meshless methods for linear advection diffusion- type partial differential equations, Int. J.Numer. Methods Eng. 48 (2000) 9-35.

21. M.M. Cecchi, M.A. Pirozzi, High order finite difference numerical methods for time-dependent convection- dominated problems, Appl. Numer. Math. 55 (2005) 334-356.

22. M.M. Chawla, M.A. Al-Zanaidi, M.G. Al-Aslab, Extended one-step time-integration schemes for convection- diffusion equations, Comput. Math. Appl. 39 (2000) 71-84.

23. W.F. Spotz, High-order Compact Finite Difference schemes for Computational Mechanics, Ph.D. Thesis, University of Texas at Austin, Austin, TX, 1995

24. T. Han, J.A.C. Humphrey, B.E. Laundev, A comparison of hybrid and quadratic-upstream differencing in high Reynolds number elliptic flows, Comput. Methods Appl. Mech. Eng. 29 (1981) 81-95.

25. B.P. Leonard, Simple high-accuracy resolution program for convective modelling of discontinuities, Int. J. Numer. Methods Fluids 8 (1988) 1291-1318.

26. Z.-z. Sun and Z.-b. Zhang, "A linearized compact difference scheme for a class of nonlinear delay partial differential equations," \textit{Applied Mathematical Modelling}, vol. 37, no. 3, pp. 742--752, 2013.

27. S. Vong and Z. Wang, "A high order compact finite difference scheme for time fractional Fokker-Planck equations," \textit{Applied Mathematics Letters}, vol. 43, pp. 38--43, 2015.

28. Roul, Pradip & Kumari, Trishna. (2023). Design of a novel computational procedure for solving electrohydrodynamic flow equation. Soft Computing. 28. 1-17. 10.1007/s00500-023-08156-2.

29. Pradip Roul, V.M.K. Prasad Goura,A new higher order compact finite difference method for generalised Black-Scholes partial differential equation: European call option, Journal of Computational and Applied Mathematics,Volume 363,2020,

30. Spotz, William. (1996). High-Order Compact Finite Difference schemes for Computational Mechanics. Thesis for: Ph.D. Aerospace EngineeringAdvisor: Graham F. Carey

31. P. Moin, Fundamentals of Engineering Numerical Analysis, 2nd ed., Cambridge University Press, 2010.