NUMERICAL SOLUTION OF ADVECTION-DIFFUSION EQUATION WITH AND WITHOUT A SOURCE TERM USING FINITE DIFFERENCE METHODS

MSc THESIS

ZERIHUN MEKOYA DEMISSIE

JUNE 2019

HARAMAYA UNIVERSITY, HARAMAYA

Numerical Solution of Advection-Diffusion Equation with and without a Source Term Using Finite Difference Methods

A Thesis Submitted to Department of Mathematics

Postgraduate Programs Directorate

HARAMAYA UNIVERSITY

In Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE IN MATHEMATICS

(NUMERICAL ANALYSIS)

Zerihun Mekoya

June 2019

Haramaya University, Haramaya

HARAMAYA UNIVERSITY

Postgraduate Programs Directorate

I hereby certify that I have read and evaluated this Thesis entitled **‘‘Numerical Solution of Advection-Diffusion Equation with and without a Source Term Using Finite Difference Methods’’** which has been prepared under my guidance by Zerihun Mekoya. All feedback given to the student has been incorporated in the Thesis. Therefore, I recommend that it be submitted as fulfilling the Thesis requirements.

Melisew Tefera (PhD)	___________________	____________________
Major Advisor	Signature	Date
Seleshi Demie (PhD)	___________________	____________________
Co-advisor	Signature	Date

As a member of the Board of Examiners of the MSc Thesis Open Defense Examination, I certify that I have read and evaluated the Thesis prepared by Zerihun Mekoya and examined the candidate. I recommend that the Thesis be accepted as fulfilling the Thesis requirements for the Degree of Master of Science in Mathematics (Numerical Analysis).

________________________	_________________	__________________
Chairperson	Signature	Date
________________________	_________________	__________________
Internal Examiner	Signature	Date
________________________	_________________	__________________
External Examiner	Signature	Date

Final approval and acceptance of the Thesis is contingent upon submission of its final copy to the Council of Graduate Studies (CGS) through the candidate’s School Graduate Committee (SGC).

STATEMENT OF THE AUTHOR

By my signature below, I declare and affirm that this Thesis is my own work. I have followed all ethical and technical principles of scholarship in the preparation, and compilation of the Thesis. Any scholarly matter that is included in the Thesis has been given recognition through citation.

This Thesis is submitted in partial fulfillment of the requirements for Master of Science in Mathematics at Haramaya University. The Thesis is deposited in the Haramaya University Library and made available to borrowers under the rules of the Library. I solemnly declare that this Thesis has not been submitted to any other institution anywhere for the award of any academic degree, diploma or certificate.

Brief quotations from this Thesis may be allowed without special permission provided that an accurate and complete acknowledgement of the source is made. Requests for permission for extended quotations from or reproduction of this Thesis in whole or in part may be granted by the Head of the School or Department when in his or her judgment the proposed use of the material is for a scholarly interest. In all other instances, however, permission must be obtained from the author.

Name: Zerihun Mekoya Signature: __________________

Date of Submission: ____________________

Department: Mathematics (Numerical Analysis)

BIOGRAPHICAL SKETCH

The author, Zerihun Mekoya was born on 19 June, 1996 at Sinana 3rd, Bale Zone, Oromia Region, Ethiopia, to his father Mekoya Demissie and mother Fantanesh Gezahagn. He attended his primary education at Tullu Shenen primary school. Then after, he joined Meliyu Burka secondary school to attend his secondary education and Goro to attend his preparatory education. He then joined Jimma University in 2013 and received Bachelor of Education degree in Mathematics in June, 2016. By the year he graduated, he has been employed as a Graduate, Assistant Mathematics Instructor by Ministry of Education to work at Debark University and got direct chance to attend his postgraduate program at Haramaya University, College of Natural and Computational Sciences, Department of Mathematics and pursued a program of study for MSc degree in Mathematics with specialization Numerical Analysis.

ACKNOWLEDGEMENT

First of all, I am indebted to the son of holy Virgin Mary God who gave me long life, strength and helped me to reach this precise time.

I am profoundly grateful to my major advisor, Dr. Melisew Tefera for his immense devotion, professional assistance, guidance and constructive comment throughout my research work and Thesis manuscript write-up. Without the encouragement, insight and professional expertise of him, the completion of this work would not have been possible. Grateful appreciation is also extended to my co-advisor Dr. Seleshi Demie for all his useful comments.

I wish to extend my deepest and heartfelt gratitude to all of my families. Specially, I enormously appreciate the contribution of my mother, Fantanesh Gezahagn and my father, Mekoya Demissie who cultivated me from the very beginning of my childhood and enabled me to take this position.

I am indebted to all of my best friends for their great contribution during my research work. Their great advice and discussion regarding the life issues was unforgettable and brought about glorious time memorial of my stay at Haramaya University. Specially, my sincere thanks goes to Mr. Gashu Gidisa, Medawalbu University staff member for his unlimited support and assistance. I am also indebted to the many of those names are not mentioned here for encouraging and supporting me morally during my study period.

Finally, very special thanks go to the minister of education (Debark University) for the permission of my study leave and financial support. I am also thankful to Haramaya University for providing me quality education with the help of different facilities.

ABBREVIATIONS

ADE	Advection-Diffusion Equation
BTCS	Backward time Centered Space
FDM	Finite Difference Method
FTCS	Forward Time Centered Space
FDS	Finite Difference scheme
PDE	Partial Differential Equation
Cr	Courant Number
pe	Peclet Number

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Numerical Solution of Advection-Diffusion Equation with and without a Source Term Using Finite Difference Methods

ABSTRACT

In this Thesis, new finite difference schemes are developed for solving one-dimensional advection-diffusion equation (ADE) with and without a source term. The developed schemes are based on the weighted finite difference method and higher order finite difference approximation. Firstly, by changing the values of weighted parameters, explicit forward in time centred in space (FTCS) and implicit backward in time centred in space (BTCS) schemes are derived and applied in both Dirichlet and Neumann boundary conditions. Discretization of ADEs resulted in a tri-diagonal system of linear algebraic equations which are solved by Thomas algorithm. For both FTCS and BTCS, the consistency and the stability of the schemes have been investigated. The Von-Neumann stability analysis is used to analyse the stabilities of the developed schemes. Convergence rates of the schemes are also determined numerically and it is found that they coincide with the theoretical result. Secondly, a new higher order finite difference scheme is derived using Taylor series expansion. In developing this scheme, a sixth order central difference in space and a first order forward difference in time were employed. The efficiency of the developed schemes was illustrated by solving different ADEs with both Dirichlet and Neumann boundary conditions. It is concluded that the developed schemes are in very good agreement with the analytical solutions.

Keywords: Advection-Diffusion Equation, Weighted Finite Difference, Higher Order Finite Difference, Consistency, Stability, Convergence Rate

1. INTRODUCTION

1.1. Background of the Study

Mathematics is referred as the language of science because almost every physical system and indeed any phenomenon in nature may be modelled using mathematical equations. Thus solving the mathematical models can help in efficient analysis and careful examination of the physical and dynamical properties and characteristics of the systems, and so on (Soomro *et al*., 2013). There are three important steps in the computational modelling of any physical process: (i) problem definition, (ii) mathematical model, and (iii) computer simulation. The first natural step is to define an idealization of our problem of interest in terms of a set of relevant quantities which we would like to measure. In defining this idealization we expect to obtain a well-posed problem, this is one that has a unique solution for a given set of parameters. The second step of the modelling process is to represent our idealization of the physical reality by a mathematical model: the governing equations of the problem. These are available for many physical phenomena. After the selection of an appropriate mathematical model, together with suitable boundary and initial conditions, we can proceed to its solution (Peiró and Sherwin, 2005).

Whenever we consider an initial-boundary value problem in unknown function, we have three different well-known boundary condition types: Dirichlet, Neumann, and Robin. A Dirichlet, condition specifies the value of unknown function on the boundary. A Neumann, condition, on the other hand, specifies the solution of derivative of the unknown function along the boundary. A Robin, the condition involves both the value of unknown function and its derivative, specifying an equation that must be valid along the boundary (Miyaoka *et al*., 2017).

The advection-diffusion equation (ADE) is a mathematical model that has been used to model the concentration of pollutants. It gives the amount of pollutant concentration fields after input of the velocity data from the hydrodynamic model which is derived from mass balances. The advection-diffusion equation is a partial differential equation (PDE), which describes physical phenomena where particles, energy or other physical quantities are transformed inside a physical system due to two processes: advection and diffusion (Su *et al*., 2015). The term advection means the movement of molecules within fluids, whereas, diffusion describes the spread of particles through random motion from regions of higher concentration to regions of lower concentration (Siddiqi and Arshad, 2013).

The general advection-diffusion equation with source term has the following form (Li *et al.*, 2017):

$$\frac{\partial u(x,t)}{\partial t} + \nabla\big(\alpha u(x,t)\big) = \nabla\cdot\big(\beta\nabla u(x,t)\big) + f(x,t). \tag{1.1}$$

In the above equation, four terms represents transient, advection, diffusion and source term respectively. Where $u(x,t)$ is the variable of interest, α is the average velocity that the quantity is moving, β is the diffusivity (also called diffusion coefficient), $f(x,t)$ is source term represents capacity of internal sources, ∇ represents gradient and $\nabla\cdot$ represents divergence. However, this thesis is devoted to one-dimensional ADE with and without a source term.

From a mathematical point of view, solving a differential equation is essentially a process of removing derivatives, or an integration process, and thus the solution of a differential equation typically involves arbitrary constants. Due to the advancements in the field of computational mathematics, numerical methods are widely utilized to solve equations arising in the field of applied medical science, engineering and technology, and so on (Soomro *et al.*, 2013). Numerical analysis plays a significant role when difficulties encountered in finding the exact solution of an equation using a direct method and when it becomes very difficult or impossible to apply theoretical methods to find the exact solution (Gerald and Wheatley, 2004). For the vast majority of geometries and problems, equation (1.1) cannot be solved with analytical methods.

Instead, an approximation of the equations can be constructed, typically based upon different types of discretizations. These discretization methods approximate the PDE with numerical model equations, which can be solved using numerical methods. Apart from very special cases, PDEs can only be solved numerically; the construction of their numerical solutions is a fundamental task in science and engineering (Gerdt and Robertz, 2010). The solution to the numerical model equations is, in turn, an approximation of the real solution to the PDE. In any numerical method, a discrete approximation for the solution is computed by discretizing the given domain into a set of subdomains.

In numerical analysis, three different approaches are commonly used: the finite difference, the finite element and the finite volume methods (Peiró and Sherwin, 2005). However, in this Thesis among the aforementioned three classical numerical methods that are widely applied for solving PDEs the finite difference method (FDM) was considered. The finite difference method is based upon the application of a local Taylor expansion to approximate the differential equations by difference ones defined on the chosen computational grid. The difference equations that approximate differential equations in the system of PDEs form its finite difference approximation which together with a discrete approximation of initial or/and boundary conditions is called finite difference scheme (FDS) (Strikwerda, 2004).

Many techniques are available for numerical simulation work and in order to quantify how well a particular numerical scheme performs in generating a solution to a problem, there are three fundamental criteria that can be compared and contrast different methods. The concepts are consistency, stability and convergence (Neill and Hashemi, 2018). Before numerical computations are made, these three important properties of finite difference equations must be considered.

1.2. Statement of the Problem

The study of the ADE continues to be an active field of research due to the subject has important applications to fluid dynamics as well as many other branches of science and engineering (Prieto *et al*., 2010). The solution of ADE has been a great challenge due to the fact that its general analytical solution is not available (Sanjaya and Mungkasi, 2017). Especially, when the advection becomes dominant, the analytical solution becomes ill-behaved and harder to evaluate (Mojtabi and Deville, 2015). To overcome this challenge, we need a numerical method like FDM in which we can find an approximate solution for the PDE to solve the ADE for the general case and it also requires numerical simulation. However, due to the importance and versatility of ADE in the mathematical modelling of processes in various application fields, the numerical approximation of such equations has been growing more and more. The increasing desire for the numerical solutions to such mathematical problems, which are more difficult or impossible to solve analytically, has become the present-day scientific research.

Many researchers have been conducted the numerical solution of ADE without a source term. For instance, Sanjaya and Mungkasi (2017) have investigated the numerical solution of ADE without a source term in $1D$. In their study, an explicit FDM was applied to obtain the solution and numerical solution of ADE with source term using FDM was stated as a gap. Mohammadi *et al*. (2011) have proposed the numerical solution of ADE without a source term using simultaneously temporal and special weighted parameters. In their study, they only performed the numerical solution for Dirichlet boundary conditions.

To the very best of our knowledge, there has been no investigation conducted on the numerical solution of ADE with a source term using FDM. Therefore, there is the knowledge gap on an implementation of ADE with a source term using FDM. Accordingly, obtaining stable, accurate, and fast numerical solutions for ADEs with a source term has great importance due to its wide applications in science and engineering as they are difficult to solve analytically (Mojtabi *et al*., 2015). In this Thesis, we developed finite difference method for solutions of ADEs with and without a source term.

1.3. Objectives of the study

The general objective of this Thesis is to develop new finite difference methods for the numerical solution of advection-diffusion equations with and without a source term.

The specific objectives of this Thesis were to:

1. Develop the numerical schemes of FDM based on weighted parameters for solving ADE.
2. Imposing Dirichlet and Neumann boundary conditions on the aforementioned schemes.
3. Investigate the stability and convergence of the developed schemes for solving ADE.
4. Derive the sixth order finite difference approximations of first and second derivatives of ADE terms.
5. Demonstrate the efficiency of the developed schemes by solving different ADEs with and without a source term.

2. REVIEW OF RELATED LITERATURE

2.1. Advection Diffusion Equation

The mathematical model describing the transport and diffusion processes is the one dimensional ADE. Mathematical modelling of heat transport, pollutants, and suspended matter in water and soil involves the numerical solution of ADE. The combination of advection and diffusion is important for mass transport in fluids. Whenever we consider mass transport of dissolved (solute) species or a component in a gas mixture, concentration gradients will cause advection. If there is a bulk fluid motion, diffusion will also contribute to the flux of chemical species. Therefore, we are often interested in solving for the combined effect of both advection and diffusion. Several procedures have been tried to interpret separately the advection and diffusion pollutant transport (Su *et al*., 2015).

Numerical models are important engineering tools when considering the prediction of pollution transport in a body of water. Such a prediction is achieved by the solution of the advection-diffusion equation (Gane 2000). The advection-diffusion equation has wide applications, especially for those involving heat transfer and fluid flows. Examples of using ADE model to solve practical engineering problems include heat transfer problems in a draining film (Isenberg and Gutfinger, 1973) or a Nano fluid filled enclosure (Sheikholeslami et al., 2014), radial transport in a porous medium (Veling, 2012), water transport in soils (Parlange, 1980), etc.

According to Johari *et al*. (2018) they proposed the one-dimensional ADE for predicting the water pollution concentration transport. The numerical modelling will be produced in order to predict the transportation of water pollution concentration. Advection-diffusion equation is a description of contaminant transport in pollution models. This equation reflects physical phenomena where in the advection process particles are moving with certain velocity form higher concentration to lower concentration. Advection-diffusion equation is particularly the description of the transport of pollutants in water bodies and other environments. Advection causes translation of the solute field by moving the solute with the flow velocity and diffusion causes spreading of the solute plume in such environments.

An analytical integration in two and three dimensional and time dependent case for ADEs are mostly not easy, and hence mathematical manipulations are needed. But in one dimensional case where the terms in y and z directions are constant and depend only on the longitudinal co-ordinate x are much simpler and are universally considered (Benedini, 2011). The analytical integration of a one dimensional model leads to a well-known formulation, which is currently applied and is very useful to focus on the role of the basic terms involved in the process of pollution transport, like the velocity of the fluid and the diffusion coefficient (Runkel, 1996).

During the last decades, the progress in numerical calculus has promoted the development of numerical procedures, which have been intensively applied for the integration of the fundamental equations. The process starts from the transformation of the equation into a discrete expression. In the numerical approach we should clearly emphasis that the expression contains two terms that behave in a different way according to the basic mathematical outlines, namely, a hyperbolic term, determined by the discretization of the first order derivative of the equation, and a parabolic term, which comes from the second order derivative. These two terms follow a different way of proceeding that normally should be done one another independently (Benedini, 2011).

In the discretization process, the general differential equation is represented by an expression of finite terms, both in space with intervals of specified size, Δx, Δy and Δz along the co-ordinate directions, and in time, with a Δt interval (Fernandes and Karney, 2001). Normally the integration of fundamental equation is brought to the solution of linear equations, or linear equation systems, with many steps and variables, and involves the concepts of linear systems of mathematics. An analytical solution of the one dimensional ADE is obtained by reducing the original ADE into a diffusion equation by introducing another dependent variable. Analytical solution of 1D ADE with variable coefficients is presented in a finite domain by using Laplace transformation technique. In that process new independent space and time variables have been introduced.

2.2. Finite Difference Method

Finite difference methods were made during the period of, and immediately following, the Second World War, when large-scale practical applications became possible with the aid of computers. A major role was played by the work of von Neumann, partly reported in O′Brien, Hyman and Kaplan (1951). Finite difference methods are always a convenient choice for solving boundary value problems because of their simplicity. The methods are one of the most widely used numerical schemes to solve differential equations and their application in sciences and technology (Thomée, 2001).

The major goal of FDM is to approximate solutions to differential equations. That is to find a function (or some discrete approximation to this functions) which satisfies a given relationship between several of its derivatives on some given region of space and/ or time, along with some boundary conditions along the edges of this domain. A finite difference method proceeds by replacing the derivatives in the differential equation by the finite difference approximations. This gives a large algebraic system of linear equations to be solved by Thomas Algorithm or other methods in place of the differential equation to give the solution value at the grid points and hence the solution is obtained at grid points. Some of the FDM include forward, backward and central difference methods, etc. Also the methods are essential for the application by decomposing the domain into finite steps with respect to space and time. There are different procedures for the achievement of final expressions, like the Forward Euler, Backward Euler, Crank-Nicolson methods. When we use smaller space and time intervals the discretisations should vanish and the final result would be closer to that of the analytical solution, but it increases a computational burden (Li *et al*., 2017).

An appropriate numerical method can give accurate information on the concentration of pollution at different times and at different location efficiently and quickly (Ahsan, 2012). Therefore, many researchers have studied the numerical methods to solve the ADE without a source term in predicting the water pollution problem. One of the methods to predict water pollution concentration is FDM. According to Igboekwe and Achi (2011) FDM have been used to solve the flow equations in order to obtain the flow rate and flow direction of water pollution concentration.

The main idea behind the FDMs for obtaining the solution of a given PDE is to approximate the derivatives appearing in the equation by a set of values of the function at a selected number of points. The most usual way to generate these approximations is through the use of Taylor series. The numerical techniques developed to approximate the derivatives are based on the modified equivalent PDE as described by Warming and Hyett (1974). This approach allows the simple determination of the theoretical order of accuracy, thus allowing methods to be compared with one another. Also from the truncation error of the modified equivalent equation, it is possible to eliminate the dominant error terms associated with the finite difference equations that contain free parameters (weights), thus leading to more accurate methods (Dehghan, 2004).

Taylor series based finite difference approximations are efficient numerical procedures of approximating the derivatives of a function, at a reference mesh point, by using the values of the function at the neighbouring mesh points. Automatic differentiation (Griewank, 2000) is another efficient and accurate differentiation technique that can be used for the functions, which can be represented by a computer code, but cannot be used for the inputs for which generating functions are not known, as is the case in most of the real time applications. In contrast, finite difference approximations do not need to know the generating function of the data, although in some cases this information might be helpful in choosing the best approximation to use. In addition to differentiation, finite difference approximations can be used in finding the numerical solutions of differential and PDE (Simos, 1999). For advection-diffusion problems, only few cases with special initial conditions have analytical solutions. Therefore, development of stable, accurate and efficient numerical methods for solving ADE is of vital importance.

2.3. Numerical Solution of Advection-Diffusion Equation

Efficient and accurate numerical methods for numerical approximations of PDEs appearing in science and engineering have been a goal of mathematicians, engineers, physicists, and other scientists for decades. In the last fifty years, many numerical approaches, which include finite difference method, finite element method, spectral method and finite volume method, have dominated the numerical approximations of PDEs. However, the FDM remains as a fundamental technique in solving PDE that appeared in diverse physical fields such as quantum mechanics, electro-magnetics, and fluid mechanics (Wang, 2016).

Johari *et al*. (2018) have proposed the numerical solution of a one-dimensional ADE without a source term for predicting the water pollution concentration transport using Crank-Nicolson method. However, there is a fundamental difficulty in the solution of the transport equation, which results from the fact that, while advection and diffusion are simultaneous processes as both components move the pollutant from one place to another, but they promote pollutant transport very differently: in the case of advection, transport is in one way, while in the case of diffusion it is in both ways. Mathematically, this means the need to treat simultaneously hyperbolic terms associated with advection and parabolic terms associated with diffusion terms, a problem that no numerical method has yet fully overcome.

Many popular FDMs, such as Noye and Tan, used a weighted discretization with the modified equivalent PDE for solving one-dimensional ADE. Later, the authors extended their method to solve two-dimensional ADE. The upwind scheme of Spalding and the flux-corrected scheme are available for the solution of the depth-averaged form of the ADE. Another widely used approach is split-operator approach, in which the advection and diffusion terms are solved by two different numerical methods.

Many authors deal with the solution of this equation without a source term numerically. For instance, Mojtabi and Deville (2015) have conducted a study to solve time dependent $1D$ linear ADE without the source term with Dirichlet homogeneous boundary conditions using finite element method. The method of a sixth-order compact FDS in space and a fourth-order Runge-Kutta scheme in time was presented (Gurarslan *et al*., 2013). Appadu (2013) has presented the numerical solution of one-dimensional ADE without a source term for small values of the coefficient of diffusivity using standard and nonstandard FDSs.

Ahmed (2012) has derived a new finite difference equation to solve linear ADE with both constant and variable coefficients. The derivation based on a mathematical combination between spatial and time discretisation from previous numerical methods. Also a new numerical scheme is developed to track the unknowns in the problem under consideration. Due to the practical importance of the ADE, solves and analyses these problems using a new finite difference equation as well as a numerical scheme.

Numerical Solution of the one-dimension ADE without source term using standard and non-standard FDSs are done by using constant coefficients. He discusses the numerical methods for the equation with constant coefficients. He consider the Lax-Wendrof scheme which is explicit, the Crank-Nicolson scheme which is implicit, and a non-standard FDS with specified initial and boundary conditions, for which the exact solution is known to test all the three methods. He concludes that the Lax-Wendrof and non-standard finite difference are quite good methods to approximate the $1D$ advection-diffusion equations (Appadu, 2013).

Ahsan in 2012 investigated and developed a numerical method to solve the one-dimensional ADE for the prediction of the quality of water in rivers. In this study, he uses the procedure that time variable is eliminated first by the Laplace transformation, and then a finite analytical method is applied in space. Since the Laplace transformation has been used for temporal approximation, an efficient and accurate inverse Laplace transform method is employed to obtain the solution in real time. The proposed method is compared against analytical solutions and two FDSs. The results of his proposed method agree with analytical solutions without numerical oscillation or diffusion.

According to Bulut *et al*. (2013) the difficulties arising in numerical solutions of ADE result from the dominant advection, which is for relatively high Peclet number. As a result the present methods will be applied to steady and unsteady flows and it also provides flexibility for uniform and non-uniform grid spacing and for a wide range of Peclet numbers. Mojtabi and Deville (2015) presented a time dependent one-dimensional linear ADE with Dirichlet homogeneous boundary conditions and an initial sine function is solved analytically by separation of variables and numerically by the finite element method. It is observed that when the advection becomes dominant, the analytical solution becomes ill-behaved and harder to evaluate. Therefore, another approach is designed where the solution is decomposed in a simple wave solution and a viscous perturbation. It is shown that an exponential layer builds up close to the downstream boundary.

The effects of operator splitting methods to the solution of ADE are examined. Within the context of this work two operator splitting methods, Lie-Trotter and Strang splitting methods were used and comparisons were made through various Courant numbers. These methods have been implemented to ADE in $1D$. Numerical solutions of advection and dispersion processes were carried out by a characteristics method with cubic spline interpolation and Crank-Nicolson finite difference scheme, respectively (Bahar and Gurarslan, 2017). The authors studied on the $1D$ mathematical expression of the ADE without the source term. Numerical treatment of the mathematical model is studied for water pollution. This study was examined by various mathematical models involving water pollutant. The authors used the implicit centered difference scheme in space and a forward difference scheme in time for the evaluation of the generalized transport equation (Agusto and Bamingbola, 2007).

2.4. Convergence Analysis of Finite Difference Method

Love and Rider (2013) have presented an issue in the verification of order of convergence for finite difference approximations to PDEs discretized in time and space. When one refines both space and time together, the convergence rate for a correct implementation will match that expected from an analysis of the asymptotic convergence rate (or local truncation error) of the method. Sometimes, however, only the time-step size is reduced, while the spatial mesh is held fixed. The observed rate of convergence in this case may then differ from the formal order of accuracy of the method in space–time. In particular, one class of methods, second-order Lax–Wendroff time-differencing, produces only first-order when the time step is refined at fixed spatial mesh resolution because the time and space differencing are intrinsically linked. Their method was arrived at this result is through analysing the error at a defined point in time associated with many time steps.

Morton (1971) has presented the stability and convergence in fluid flow problems. Stability analysis of finite difference schemes for the Navier-Stokes equations was obtained (Rigal, 1979). Stability analysis of finite difference schemes for the ADE was presented (Chan, 1984). A comparison of some numerical methods for the ADE is presented (Thongmoon and Mckibbin, 2006). An analytical solution of the advection-diffusion equation for a ground level finite area source is presented (Park and Baik, 2008). An explicit finite difference scheme for solving the ADE is studied. Numerical results for both the schemes are compared in terms of accuracy by error estimation with respect to exact solution of the ADE and also the numerical features of the rate of convergence are presented graphically (Azad *et al.*, 2015).

Present-day scientific research concerns the methods of numerical solutions to mathematical problems which are simpler to use and solve difficult problems. Accordingly, obtaining the numerical solution of ADE problem has great importance due to its wide applications in science and engineering research, since they are difficult or impossible to solve analytically even if numerically. Recent research in computational mathematics is not only devoted to the proposition of new numerical methods but is also primarily concerned with the comparison of available methods on the basis of their accuracy, reliability, stability and consistency of solving the given problems (Soomro *et al.*, 2013). Owing to this, this study presents numerical solutions and simulations of ADE by using different FDS.

3. MATERIALS AND METHODS

This chapter outlined the methods that have been used to succeed in the general and specific objectives of the study, and materials used in the study. This study has been employed mixed-design that are documentary review design and experimental design on the numerical solution of ADE with and without a source term using FDM. The relevant sources of information for this study were books, published articles and related studies from the internet. The experimental result has been obtained by writing new MATLAB code for the developed numerical schemes. Hence, in order to achieve the stated objectives, the study has been followed by the following procedures;

1. Defining the problem.
2. Discretizing the given domain/interval.
3. Replacing the given equation by the finite difference approximations and obtaining the systems of equations.
4. Reduce the obtained systems of equations in tri-diagonal systems after imposing the Dirichlet and Neumann boundary conditions on the systems.
5. Investigating the consistency of the schemes obtained for solving ADE with a source term.
6. Investigating the stability of the schemes obtained for solving ADE with a source term using Von-Neumann stability analysis.
7. Investigating the convergence of the schemes using Lax-Equivalence theorem.
8. Deriving the sixth order finite difference approximations of the first and second derivatives of ADE terms based on the Taylor's series expansion.
9. Writing MATLAB code for the tri-diagonal systems obtained using Thomas algorithm and considered sixth order FDSs.
10. Validating the schemes using numerical examples of ADE with and without a source term.

4. PRELIMINARY

4.1. Basis of Numerical Techniques for Solving Partial Differential Equations

The mathematical formulation of most problems in science involving rates of change with respect to two or more independent variables, usually representing time, length or angle, leads either to a PDE or to set of such equation. Partial differential equations are a differential equation where the unknown is a function of more than one independent variable. Partial differential equations form the basis of very many mathematical models of physical, chemical and biological phenomena, and more recently their use has spread into economics, financial forecasting, image processing and other fields. The general form of linear second-order PDE is:

$$a\frac{\partial^2 u}{\partial x^2}+b\frac{\partial^2 u}{\partial x\partial t}+c\frac{\partial^2 u}{\partial t^2}+d\frac{\partial u}{\partial x}+e\frac{\partial u}{\partial t}+fu=g \tag{4.1}$$

Here $u=u(x,t)$, and a,b,c,d,e,f and g are functions of x and t only they do not depend of u. The first three terms containing the second derivatives are called the Principal part of the PDE because they determine the nature of the general solution to the equation. In fact, the coefficients of the Principal part can be used to classify the PDE. Classified as the elliptic, parabolic and hyperbolic PDEs (Yang *et al*., 2005).

The vast majority of PDEs model cannot be solved analytically. So, to investigate the predictions of PDE models of such phenomena it is often necessary to approximate their solution numerically. In most cases, the approximate solution is represented by functional values at certain discrete points (grid points or mesh points). There seems a bridge between the derivatives in the PDE and the functional values at the grid points. The numerical technique is such a bridge, and the corresponding approximate solution is termed the numerical solution (Kaur and Kang, 2016).

The finite difference techniques are based upon the approximations that permit replacing differential equation by finite difference equation. There finite difference approximations are algebraic in form, and the solutions are related to grid points. Thus, a finite difference solution basically involves three steps (Sadiku, 2000):-

1) Dividing the solution into grids of notes.

2) Approximating the given differential equation by finite difference equivalence that relates the solutions to grid points.
3) Solving the difference equations subject to the prescribed boundary conditions and/or initial conditions.

Depending on the pattern of data samples used to approximate the derivative, Taylor series based finite difference approximations can be divided into three major categories, namely forward, backward and central difference approximations. These approximations use the function values at a set of equally spaced mesh points, to approximate the value of the derivative at the left most, right most and the central mesh points, respectively. The mesh point at which the derivative is approximated is called reference mesh point, and the number of mesh points used in approximation, excluding the reference mesh point defines the order of the approximation (Khan and ohba, 2003).

Finite difference replace the infinitesimal limiting process of derivative calculation

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

with a limiting process, i.e.,

$$f'(x) = \frac{f(x+\Delta x) - f(x)}{\Delta x} + O(\Delta x)$$

The term $O(\Delta x)$ gives an indication of the magnitude of the error as a function of the mesh spacing.

Forward difference formula

$$f'(x_0) = \frac{f(x_0+\Delta x) - f(x_0)}{\Delta x}$$

Backward difference formula

$$f'(x_0) = \frac{f(x_0) - f(x_0-\Delta x)}{\Delta x}$$

Central difference formula

$$f'(x_0) = \frac{f(x_0+\Delta x) - f(x_0-\Delta x)}{2\Delta x}$$

The approach used for obtaining above finite difference equations is Taylor's series:-

$$f(x_0+\Delta x) = f(x_0) + \Delta x f'(x_0) + \frac{1}{2!}(\Delta x)^2 f''(x_0) + \frac{1}{3!}(\Delta x)^3 f'''(x_0) + O(\Delta x)^4 \quad (4.2)$$

$$f(x_0 - \Delta x) = f(x_0) - \Delta x f'(x_0) + \frac{1}{2!}(\Delta x)^2 f''(x_0) - \frac{1}{3!}(\Delta x)^3 f'''(x_0) + O(\Delta x)^4 \quad (4.3)$$

where $O(\Delta x)^4$ is the error introduced by truncating the series.

Subtracting equation (4.2) from equation (4.3), we obtain

$$f(x_0 + \Delta x) - f(x_0 - \Delta x) = 2\Delta x f'(x_0) + O(\Delta x)^3,$$

which can be re-written as

$$f'(x_0) = \frac{f(x_0+\Delta x) - f(x_0 - \Delta x)}{2\Delta x} + O(\Delta x)^2$$

i.e. the central difference formula. Note that the $O(\Delta x)^2$ means the truncation error is the order of $O(\Delta x)^2$ for central difference.

The forward difference and backward difference formula could be obtained by rearranging (4.2) and (4.3) respectively, and we have

$$f'(x_0) = \frac{f(x_0+\Delta x) - f(x_0)}{\Delta x} + O(\Delta x),$$

for forward difference, and

$$f'(x_0) = \frac{f(x_0) - f(x_0 - \Delta x)}{\Delta x} + O(\Delta x),$$

for backward difference. We can find the truncation errors of their two formulas are of order Δx. On adding equation (4.2) and equation (4.3), we have

$$f(x_0 + \Delta x) + f(x_0 - \Delta x) = 2f(x_0) + (\Delta x)^2 f''(x_0) + O(\Delta x)^4,$$

and we have

$$f''(x_0) = \frac{f(x_0 + \Delta x) - 2f(x_0) + f(x_0 - \Delta x)}{(\Delta x)^2} + O(\Delta x)^2.$$

A better approximation can be obtained using the central difference scheme. Higher order finite difference approximation can be obtained by taking more terms in Taylor's series expansion.

4.2. Properties of Finite Difference Schemes

The ultimate goal of using FDS is to find the accurate approximation of the analytic solution of a given boundary value problem. Finite difference schemes approximating PDE's are analysed according to three important properties: consistency, stability, and convergence (Causon and Mingham, 2010).

4.2.1. Consistency

The notion of consistency addresses the problem of whether the finite difference approximation is really representing the PDE. We say that a finite difference approximation is consistent with a differential equation if the finite difference equations converge to the original equation as the time and space grids are refined. Hence, if the truncation error goes to zero as time and space grids are refined we conclude that the scheme is consistent (Neill and Hashemi, 2018).

Local truncation error: Local truncation error represents the difference between original differential equation and its finite difference approximations at a point in space and time (at grid points). Local truncation errors measures how well a finite difference discretization approximates the differential equation. This is to the same as to say the local truncation error provides a basis for comparing local accuracies of various difference schemes. In particular, if the PDE satisfied by the exact solution u is written $u(x_i, t_j)$ and if $F(u(i,j))$ is the equation satisfied by the discrete approximation, then local truncation error at the (x_i, t_j) mesh point is $T(x_i, t_j) = F\left(u(x_i, t_j)\right)$. Better approximations can be obtained by increasing the order of the truncation error of the FDS. This is commonly accomplished by including more points in the stencil of the numerical schemes (Randall, 1998).

4.2.2. Stability and Convergence

The stability of a numerical scheme is associated with propagation of numerical error. A finite difference scheme is stable if the error stays constant or decrees as the iterative process goes on. This is the same as saying, a FDS is stable if the scheme do not allows the growth of error in the solution with different time level. On contrary, if the error grows with time, the scheme is said to be unstable (Neill and Hashemi, 2018).

There are many approaches to analyse whether a FDS is stable or unstable. In this Thesis, we use Von-Neumann stability analysis to determine the stability of the schemes as it has merits of simplicity and intuition in practical applications, but only restricted to problems of linear equations with constant coefficients and periodic boundary conditions (Li and Yang, 2013). The Von-Neumann method is the most well-known classical method to determine necessary and sufficient stability conditions. In the study of stability using Von-Neumann stability analysis, the following procedures have been carried out (Li *et al*., 2017):

1) Set $u(i,j) = e^{ijh\xi}$ and substitute it into the finite difference scheme;
2) Express $u(i,j+1)$ as $u(i,j+1) = g(\xi)e^{ijh\xi}$, $u(i+1,j+1)$ as $u(i+1,j+1) = g(\xi)e^{i(j+1)h\xi}$ and $u(i-1,j+1)$ as $u(i-1,j+1) = g(\xi)e^{i(j-1)h\xi}$;
3) Solve for $g(\xi)$ and determine whether or when $|g(\xi)| \leq 1$ (for stability); but note that
4) If there is some ξ such that $|g(\xi)| > 1$, the method is unstable.

Note that if a numerical scheme is stable for all values of ξ, then that scheme is unconditionally stable. Clearly, the Von-Neumann condition is very important both practically and theoretically. Even for variable coefficient problems, it can be applied locally (with local values of the coefficients) and because instability is a local phenomenon, due to the high-frequency modes being the most unstable, it gives necessary stability conditions which can often be shown to be sufficient. The following important points should be noted concerning the Von-Neumann method of examining stability:

- The method which is based on Fourier analysis applies only if the coefficients of the linear difference equation are constant.
- If the difference equation has variable coefficients, the method can still be applied locally and it might be expected that a method will be stable if the Von-Neumann condition, derived as though the coefficients were constant, is satisfied at every point of the field.
- Boundary conditions are neglected by the Von-Neumann method which applies in theory only to pure initial value problems with periodic initial data. It does however provide necessary conditions for stability of constant coefficient problems regardless of type of boundary conditions.

Theorem 4.1 (Gershgorin's Theorem) (Marquis, 2016): The eigenvalues of the matrix A lie in the union of circles

$$|z - a_{ii}| \leq \sum_{\substack{i=1 \\ j \neq i}}^{M} |a_{ij}| = R_i,$$

where z is a complex number, a_{ii} are the diagonal entries, a_{ij} are the non-diagonal entries and $|z - a_{ii}|$ is a closed disc centered at a_{ii} with radius R. If all discs of the matrix A are contained in the unit circle of the complex plane, so do the eigenvalues and we will have stability.

Definition (Jayanti, 2018): A numerical scheme is said to be convergent if the computed solution of the discretized equation leads to the exact solution of the differential equation as the mesh sizes approaches to zero.

This will have definition as shown below. The computed solution $u(x_i, t_j)$ must approach the exact solution U of the differential equation at any point $x_i = i\Delta x$ and $t_j = j\Delta t$ when Δx and Δt lead to zero while keeping x_i and t_i constant. In other hand, the error

$$\varepsilon_{i,j} = u_{i,j} - U_{i,j}.$$

Satisfying the following convergence condition

$$\lim_{\Delta t, \Delta x \to 0} \left|\varepsilon_{i,j}\right| \to 0 \text{ at fixed } x_i = i\Delta x \text{ and } t_j = j\Delta t.$$

This is to say that, a solution of a set of algebraic equations is convergent if the approximate solution approaches the exact solution of the set of PDEs for each value of the independent variable as the mesh sizes approaches zero that is the grid spacing and time step goes to zero. As a result, convergence is difficult to prove directly, we use an equivalence result known as the Lax-Equivalence theorem.

Theorem 4.1 (Lax-Equivalence Theorem) (Thomas, 1995): For a well-posed initial and boundary value problem, if a FDS is consistent with the PDE, then the stability is the necessary and sufficient condition for convergence that is for linear PDE's, the fact that consistency plus stability is equivalent to convergence is known as the Lax-Equivalence Theorem.

4.3. Thomas Algorithm

The Thomas algorithm is an efficient way of solving tri-diagonal matrix systems (Quarteroni *et al.*, 2006). The Thomas algorithm was used because it is fast and tri-diagonal matrices often occur in practice. It is based on LU decomposition in which the matrix system $Mx = r$ is rewritten as $LUx = r$ where L is a lower triangular matrix and U is an upper triangular matrix. The system can be efficiently solved by setting $Ux = \rho$ and then solving first $L\rho = r$ for ρ and then $Ux = \rho$ for x. The Thomas algorithm consists of two steps. In step 1 decomposing the matrix into $M = LU$ and solving $L\rho = r$ are accomplished in a single downwards sweep, taking us straight from $Mx = r$ to $Ux = \rho$. In step 2 the equation $Ux = \rho$ is solved for x in an upwards sweep (Lee, 2011).

5. FINITE DIFFERENCE METHOD FOR ADVECTION-DIFFUSION EQUATION WITH AND WITHOUT A SOURCE TERM

5.1. Mathematical Model of Advection-Diffusion Equation with a Source Term

The mathematical model of the one-dimensional ADE with a source term is given by:

$$\frac{\partial u(x,t)}{\partial t} + \alpha \frac{\partial u(x,t)}{\partial x} = \beta \frac{\partial^2(x,t)}{\partial x^2} + f(x,t), \quad 0 < x < L, 0 < t \leq T, \tag{5.1.1}$$

where α and β are advection and diffusion coefficients, respectively. There are many functions u which satisfies equation (5.1.1) for the given source term f. Thus, to obtain a unique solution, we need to specify more than just the governing differential equation (introduce some auxiliary constraints besides the given equation).

Now, we assume appropriate initial condition and boundary conditions so that a unique solution exists and use to determine the finite difference solution. Thus, to solve equation (5.1.1), it is required a specific initial condition at $t = 0$, given:

$$u(x,0) = h(x), \qquad 0 \leq x \leq L, \tag{5.1.2}$$

and boundary conditions at $x = 0$ and $x = L$. The general form of boundary conditions are:

$$\delta_1 u(0,t) + \gamma_1 u_x(0,t) = g_1(t), \qquad 0 \leq t \leq T, \tag{5.1.3}$$

$$\delta_2 u(L,t) + \gamma_2 u_x(L,t) = g_2(t), \qquad 0 \leq t \leq T, \tag{5.1.4}$$

where $h(x)$ and $g_k(t)$ $(k = 1,2)$ are known continuous functions, while the function $u(x,t)$ is unknown. For this study, we have considered the aforementioned ADE model given in equation (5.1.1) for constant parameters (that is α and β are considered to be constant values). The solution of equation (5.1.1) with equations (5.1.2) - (5.1.4) is to find $u(x,t)$, satisfying the boundary conditions as follows:

1) If $\delta_n \neq 0$ and $\gamma_m = 0$ $(n, m = 1,2)$, then equations (5.1.3) and (5.1.4) gives Dirichlet boundary conditions.
2) If $\delta_n = 0$ and $\gamma_m \neq 0$,then equations (5.1.3) and (5.1.4) gives Neumann boundary conditions.
3) If $\delta_1 \neq 0$ and $\gamma_1 = 0$, and $\delta_2 = 0$ and $\gamma_2 \neq 0$ or If $\delta_1 = 0$ and $\gamma_1 \neq 0$ and $\delta_2 \neq 0$ and $\gamma_2 = 0$ equations (5.1.3) and (5.1.4) gives Robin (mixed) boundary conditions.

The presence of the source term in equation (5.1.1) does not modify the analysis given on the boundary conditions (Miyaoka *et al.*, 2017). In the following sections, we present the numerical schemes used for solving equation (5.1.1) with both Dirichlet and Neumann boundary conditions and additionally the stability and convergence analysis of the schemes have been investigated. The laborious tasks of the derivation of sixth order finite difference approximation was derived from Tayler series expansion.

5.2. Numerical Approximation of ADE with a Source Term by Weighted Based Finite Difference Method

In this section, we have proposed a numerical scheme based on simultaneously temporal and spatial weighted parameters for the numerical solutions of equation (5.1.1) – (5.1.4). These schemes are derived from the truncated Taylor's series where a given ADE with source term and initial and boundary conditions are replaced by a set of algebraic equations that are then solved by several well-known numerical techniques. The domain of integration of the given equation (5.1.1) – (5.1.4) or its solution domain is $D = [0 \leq x \leq L] \times [0 \leq t \leq T]$. To use the FDM, first let us take into account $x - t$ plane such that x is represented by horizontal axis and time t by vertical axis. A rectangular mesh is formed in domain D by drawing lines parallel to the axes. We discretize the computational domain through selecting a step size $h = \Delta x$ as the spatial step in x and step size $k = \Delta t$ as the temporal step in t and divide the spatial domain $[0, L]$ into M sections, each of length $\Delta x = \frac{L}{M}$, and divide the time domain $[0, T]$ into N segments, each of duration $\Delta t = \frac{T}{N}$. Therefore, the solution of the problem is covered by a mesh of grid-lines:

$x_i = i\Delta x, \qquad i = 0,1,2, \dots, M,$

$t_j = j\Delta t, \qquad j = 0,1,2, \dots, N,$

where M and N denote the total number of spatial and temporal grid-spacings, respectively.

For illustrative motive, in the above discretizations of the computational domain, the FDM uses a topologically square network of lines to construct the discretization of the computational domain as shown in Figure 1. If we denote the numerical solution of $u(x, t)$ such that $u(i, j) = u\left(x_i, t_j\right)$. Approximations $u(i, j)$ to $u(x_i, t_j)$ are calculated at the point of intersection of these lines which are called nodal points or grid points in step by step manner in $t -$ direction, and $(i\Delta x, j\Delta t)$ is referred to as the (i, j) grid-point. The numerical solution to the given equation is an approximation to the exact solution that is obtained using a discrete representation to the given equation at the grid point x_i in the discrete spatial mesh at every time level t_j.

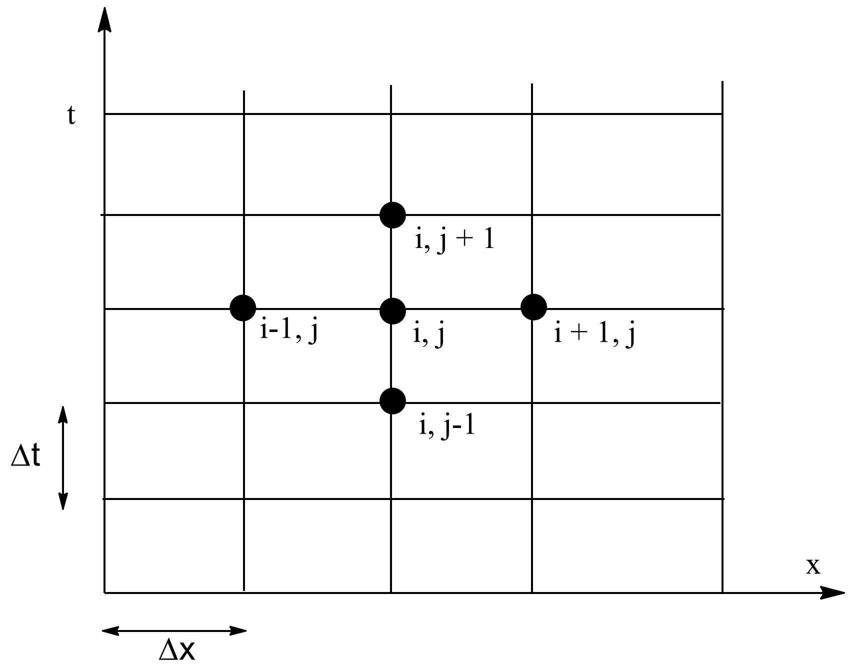


Figure 1: The finite difference grid in the solution region.

When a scheme uses the nodal values on two time levels t_j and t_{j+1}, then it is called a two-level formula. When a scheme uses the nodal values on three time levels t_{j-1}, t_j and t_{j+1}, as seen in Figure 1, then it is called a three-level formula. In this Thesis, all developed schemes that are used for solving the given problem are based on the two-level finite difference approximation.

Consider the following approximations of the derivatives in the ADE which incorporate time and space weights ϕ and θ as follows (Mohammadi *et al.*, 2011):

$$\frac{\partial u}{\partial t} = \frac{u(i,j+1) - u(i,j)}{\Delta t}, \tag{5.2.1}$$

$$\begin{aligned}\alpha\,\frac{\partial u}{\partial x} = (1-\phi)\left\{\frac{\alpha}{\Delta x}[(1-\theta)u(i,j) + \theta u(i+1,j) - (1-\theta)u(i-1,j) - \theta u(i,j)]\right\} \\ + \phi\left\{\frac{\alpha}{\Delta x}[(1-\theta)u(i,j+1) + \theta u(i+1,j+1) - (1-\theta)u(i-1,j+1) \right. \\ \left. - \theta u(i,j+1)]\right\},\end{aligned} \tag{5.2.2}$$

and

$$\begin{aligned}\beta\frac{\partial^2 u}{\partial x^2} = (1-\phi)\left\{\frac{\beta}{(\Delta x)^2}[u(i-1,j) - 2u(i,j) + u(i+1,j)]\right\} \\ + \phi\left\{\frac{\beta}{(\Delta x)^2}[u(i-1,j+1) - 2u(i,j+1) + u(i+1,j+1)]\right\},\end{aligned} \tag{5.2.3}$$

where ϕ is a time weighting factor and θ is the spatial weighting factor. Substituting equations (5.2.1) - (5.2.3) into equation (5.1.1) and after discretizing the source term explicitly and implicitly as

$$f = (1-\phi)f(i,j) + \phi f(i,j+1),$$

then we obtain

$$\begin{aligned}\frac{u(i,j+1)-u(i,j)}{\Delta t}&\\ &+ (1\\ &- \phi)\left\{\frac{\alpha}{\Delta x}[(1-\theta)u(i,j)+\theta u(i+1,j)-(1-\theta)u(i-1,j)-\theta u(i,j)]\right\}\\ &+ \phi\left\{\frac{\alpha}{\Delta x}[(1-\theta)u(i,j+1)+\theta u(i+1,j+1)-(1-\theta)u(i-1,j+1)\right.\\ &\left.- \theta u(i,j+1)]\right\}\\ &= (1-\phi)\left\{\frac{\beta}{(\Delta x)^2}[u(i-1,j)-2u(i,j)+u(i+1,j)]\right\}\\ &+ \phi\left\{\frac{\beta}{(\Delta x)^2}[u(i-1,j+1)-2u(i,j+1)+u(i+1,j+1)]\right\}\\ &+ (1-\phi)f(i,j)+\phi f(i,j+1),\end{aligned}$$

which is the same as saying

$$\begin{aligned}u(i,j+1) = -\Delta t(1&\\ &- \phi)\left\{\frac{\alpha}{\Delta x}[(1-\theta)u(i,j)+\theta u(i+1,j)-(1-\theta)u(i-1,j)-\theta u(i,j)]\right\}\\ &+ \phi\left\{\frac{\alpha}{\Delta x}[(1-\theta)u(i,j+1)+\theta u(i+1,j+1)-(1-\theta)u(i-1,j+1)\right.\\ &\left.- \theta u(i,j+1)]\right\} + \Delta t(1-\phi)\left\{\frac{\beta}{(\Delta x)^2}[u(i-1,j)-2u(i,j)+u(i+1,j)]\right\}\\ &+ \Delta t\phi\left\{\frac{\beta}{(\Delta x)^2}[u(i-1,j+1)-2u(i,j+1)+u(i+1,j+1)]\right\} + u(i,j)\\ &+ \Delta t(1-\phi)f(i,j)+\Delta t\phi f(i,j+1),\end{aligned}$$

which can be simplified as

$$\begin{aligned}
u(i,j+1) = {} & -\frac{\Delta t(1-\phi)\alpha(1-\theta)u(i,j)}{\Delta x} - \frac{\Delta t(1-\phi)\alpha\theta u(i+1,j)}{\Delta x} \\
& + \frac{\Delta t(1-\phi)\alpha(1-\theta)u(i-1,j)}{\Delta x} + \frac{\Delta t(1-\phi)\alpha\theta u(i,j)}{\Delta x} \\
& - \frac{\Delta t\phi\alpha(1-\theta)u(i,j+1)}{\Delta x} - \frac{\Delta t\phi\alpha\theta u(i+1,j+1)}{\Delta x} \\
& + \frac{\Delta t\phi\alpha(1-\theta)u(i-1,j+1)}{\Delta x} + \frac{\Delta t\phi\alpha\theta u(i,j+1)}{\Delta x} \\
& + \frac{\Delta t(1-\phi)\beta u(i-1,j)}{(\Delta x)^2} - \frac{2\Delta t(1-\phi)\beta u(i,j)}{(\Delta x)^2} + \frac{\Delta t(1-\phi)\beta u(i+1,j)}{(\Delta x)^2} \\
& + \frac{\Delta t\phi\beta u(i-1,j+1)}{(\Delta x)^2} - \frac{2\Delta t\phi\beta u(i,j+1)}{(\Delta x)^2} + \frac{\Delta t\phi\beta u(i+1,j+1)}{(\Delta x)^2} \\
& + u(i,j) + \Delta t(1-\phi)f(i,j) + \Delta t\phi f(i,j+1),
\end{aligned}$$

rearranging the above equation yields

$$\begin{aligned}
& \left(1 + \frac{\alpha\Delta t}{\Delta x}\phi(1-\theta) - \frac{\alpha\Delta t}{\Delta x}\phi\theta + 2\frac{\beta\Delta t}{(\Delta x)^2}\phi\right)u(i,j+1) \\
& \quad = \left(1 - \frac{\alpha\Delta t}{\Delta x}(1-\phi)(1-\theta) + \frac{\alpha\Delta t}{\Delta x}(1-\phi)\theta - 2\frac{\beta\Delta t}{(\Delta x)^2}(1-\phi)\right)u(i,j) \\
& \quad + \left(-\frac{\alpha\Delta t}{\Delta x}(1-\phi)\theta + \frac{\beta\Delta t}{(\Delta x)^2}(1-\phi)\right)u(i+1,j) \\
& \quad + \left(\frac{\alpha\Delta t}{\Delta x}(1-\phi)(1-\theta) + \frac{\beta\Delta t}{(\Delta x)^2}\phi\right)u(i+1,j+1) \\
& \quad + \left(\frac{\alpha\Delta t}{\Delta x}\phi(1-\theta) + \frac{\beta\Delta t}{(\Delta x)^2}\phi\right)u(i-1,j+1) + \Delta t(1-\phi)f(i,j) \\
& \quad + \Delta t\phi f(i,j+1).
\end{aligned} \qquad (5.2.4)$$

Letting $cr = \frac{\alpha\Delta t}{\Delta x}$ is the courant number and $pe = \frac{\alpha\Delta x}{\beta}$ is the peclet number, then equation (5.2.4) becomes

$$
\begin{aligned}
&\left(1 + cr\phi(1-\theta) - cr\phi\theta + 2\left(\frac{cr}{pe}\right)\phi\right)u(i,j+1) \\
&\qquad = \left(1 - cr(1-\phi)(1-\theta) + cr(1-\phi)\theta - 2\left(\frac{cr}{pe}\right)(1-\phi)\right)u(i,j) \\
&\qquad + \left(-cr(1-\phi)\theta + \left(\frac{cr}{pe}\right)(1-\phi)\right)u(i+1,j) \\
&\qquad + \left(cr(1-\phi)(1-\theta) + \left(\frac{cr}{pe}\right)\phi\right)u(i+1,j+1) \\
&\qquad + \left(cr\phi(1-\theta) + \left(\frac{cr}{pe}\right)\phi\right)u(i-1,j+1) + \Delta t(1-\phi)f(i,j) \\
&\qquad + \Delta t\phi f(i,j+1).
\end{aligned}
$$

After rearranging, we have that

$$
\begin{aligned}
&\left(1 + \phi\left(cr(1-2\theta) + 2\left(\frac{cr}{pe}\right)\right)\right)u(i,j+1) \\
&\qquad = \left(1 - (1-\phi)\left(cr(1-2\theta) + 2\left(\frac{cr}{pe}\right)\right)\right)u(i,j) \\
&\qquad - (1-\phi)\left(cr\theta - \left(\frac{cr}{pe}\right)\right)u(i+1,j) \\
&\qquad + (1-\phi)\left(cr(1-\theta) + \left(\frac{cr}{pe}\right)\right)u(i-1,j) \\
&\qquad - \phi\left(cr\theta - \left(\frac{cr}{pe}\right)\right)u(i+1,j+1) \\
&\qquad + \phi\left(cr(1-\theta) + \left(\frac{cr}{pe}\right)\right)u(i-1,j+1) + \Delta t(1-\phi)f(i,j) \\
&\qquad + \Delta t\phi f(i,j+1).
\end{aligned}
$$

Therefore, solving for the required time level the above equation can be abridged as

$$
\begin{aligned}
u(i,j+1) = \frac{1}{A}&[Bu(i,j) + Cu(i+1,j) + Du(i-1,j) + Eu(i+1,j+1) \\
&+ Fu(i-1,j+1) + \Delta t(1-\phi)f(i,j) + \Delta t\phi f(i,j+1)], \qquad (5.2.5)
\end{aligned}
$$

where

$$A = 1 + \phi\left(cr(1-2\theta) + 2\left(\frac{cr}{pe}\right)\right)$$

$$B = 1 - (1-\phi)\left(cr(1-2\theta) + 2\left(\frac{cr}{pe}\right)\right)$$

$$C = -(1-\phi)\left(cr\theta - \left(\frac{cr}{pe}\right)\right)$$

$$D = (1-\phi)\left(cr(1-\theta) + \left(\frac{cr}{pe}\right)\right)$$

$$E = -\phi\left(cr\theta - \left(\frac{cr}{pe}\right)\right)$$

$$F = \phi\left(cr(1-\theta) + \left(\frac{cr}{pe}\right)\right)$$

5.2.1. Explicit Finite Difference scheme

Assuming $\phi = 0$ in equation (5.2.5) yields the following explicit finite difference schemes

$$\begin{aligned} u(i,j+1) = {} & \frac{1}{A}[Bu(i,j) + Cu(i+1,j) + Du(i-1,j) + Eu(i+1,j+1) \\ & + Fu(i-1,j+1) + \Delta t f(i,j)], \end{aligned} \qquad (5.2.6)$$

where

$$A = 1$$

$$B = 1 - \left(cr(1-2\theta) + 2\left(\frac{cr}{pe}\right)\right)$$

$$C = -\left(cr\theta - \left(\frac{cr}{pe}\right)\right)$$

$$D = cr(1-\theta) + \left(\frac{cr}{pe}\right)$$

$$E = 0$$

$$F = 0$$

Substituting the values of A, B, C, D, E and F into equation (5.2.6) yields

$$\begin{aligned} u(i,j+1) = {} & \left(cr(1-\theta) + \left(\frac{cr}{pe}\right)\right)u(i-1,j) + \left(1 - \left(cr(1-2\theta) + 2\left(\frac{cr}{pe}\right)\right)\right)u(i,j) \\ & - \left(cr\theta - \left(\frac{cr}{pe}\right)\right)u(i+1,j) + \Delta t f(i,j). \end{aligned} \qquad (5.2.7)$$

Equation (5.2.7) is called explicit finite difference approximations to the mathematical model given in equation (5.1.1). Assume that $\theta = \frac{1}{2}$ in equation (5.2.7) and it could be written as the following forward in time centred in space (FTCS) explicit-type finite difference formula in solving the given equation:

$$u(i,j+1) = \left(\frac{cr}{2} + \left(\frac{cr}{pe}\right)\right)u(i-1,j) + \left(1 - 2\left(\frac{cr}{pe}\right)\right)u(i,j) \\ - \left(\frac{cr}{2} - \left(\frac{cr}{pe}\right)\right)u(i+1,j) + \Delta t f(i,j). \tag{5.2.8}$$

This scheme used the forward difference form for the time derivative and centred difference form for the spatial derivatives in both advection and diffusion terms. For the ease of writing tri-diagonal matrix form of FTCS scheme obtained in equation (5.2.8) and for solving Dirichlet boundary conditions this equation can be written as:

$$u(i,j+1) = (\lambda + \mu)u(i-1,j) + (1 - 2\mu)u(i,j) - (\lambda - \mu)u(i+1,j) \\ + \Delta t f(i,j), \tag{5.2.9}$$

where $i = 1, 2, 3, \dots, M-1, j = 0, 1, 2, \dots, N-1, \lambda = \frac{cr}{2}$ and $\mu = \frac{cr}{pe}$.

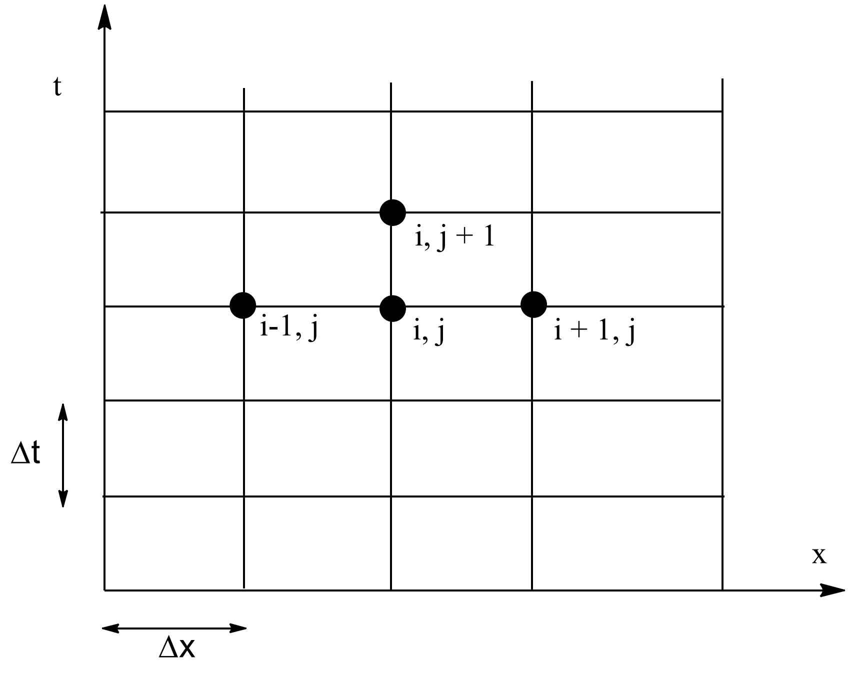


Figure 2: Represent point scheme for FTCS.

Note that the values $u(i,j+1)$ at the node $u(x_i, t_{j+1})$ is being obtained explicitly using the values on the previous time level t_j. The nodes that are used in the computations are given in figure 2. It is a two level scheme.

Moreover we are able to rewrite equation (5.2.9) in tri-diagonal matrix system form as follows which is to be solved iteratively;

$$\begin{bmatrix} u(1,j+1) \\ u(2,j+1) \\ \cdot \\ \vdots \\ u(M-2,j+1) \\ u(M-1,j+1) \end{bmatrix} = \begin{bmatrix} 1-2\mu & -\lambda+\mu & 0 & \cdot & \cdot & 0 \\ \lambda+\mu & 1-2\mu & -\lambda+\mu & & & \\ 0 & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & 0 \\ & & & \lambda+\mu & 1-2\mu & -\lambda+\mu \\ 0 & \cdot & \cdot & 0 & \lambda+\mu & 1-2\mu \end{bmatrix} \begin{bmatrix} u(1,j) \\ u(2,j) \\ \cdot \\ \vdots \\ u(M-2,j) \\ u(M-1,j) \end{bmatrix} + \begin{bmatrix} (\lambda+\mu)u(0,j)+\Delta t f(1,j) \\ \Delta t f(2,j) \\ \cdot \\ \vdots \\ \Delta t f(M-2,j) \\ -(\lambda-\mu)u(M,j)+\Delta t f(M-1,j) \end{bmatrix}. \tag{5.2.10}$$

Computational procedure of explicit FDS: In the explicit scheme the values of u at the level of time $j+1$ are given explicitly in terms of the values of u at the previous level j. The initial condition $u(x,0)=h(x)$ gives the solution at all the nodal points on the initial line (level 0). The boundary conditions $u(0,x)=g_1(t), u(L,x)=g_2(t), 0 \le t \le T$, give the solutions at all the nodal points on the boundary lines $x=0$ and $x=L$, for all time levels. We choose a value for Δx and Δt. The solutions at all nodal points that is at the interior points on level 1 are obtained using the explicit method. The computations are repeated for the required number of steps. If we perform n steps of computation, then we have computed the solutions up to time $t_n = n\Delta t$.

5.2.2. Implicit Finite Difference Scheme

Putting the value of $\phi = 1$ in equation (5.2.5) yields the following implicit FDS:

$$u(i,j+1) = \frac{1}{A}[Bu(i,j) + Cu(i+1,j) + Du(i-1,j) + Eu(i+1,j+1) + Fu(i-1,j+1) + \Delta t f(i,j+1)], \tag{5.2.11}$$

where

$$A = 1 + \left(cr(1-2\theta) + 2\left(\frac{cr}{pe}\right) \right)$$

$$B = 1$$

$$C = 0$$

$$D = 0$$

$$E = -\left(cr\theta - \left(\frac{cr}{pe}\right) \right)$$

$$F = \left(cr(1-\theta) + \left(\frac{cr}{pe}\right) \right).$$

Substituting the values of A, B, C, D, E and F into equation (5.2.11) yields

$$u(i,j+1) = \frac{1}{\left(1 + \left(cr(1-2\theta) + 2\left(\frac{cr}{pe}\right) \right)\right)} \left[u(i,j) - \left(cr\theta - \left(\frac{cr}{pe}\right) \right) u(i+1,j+1) \right.$$
$$\left. + \left(cr(1-\theta) + \left(\frac{cr}{pe}\right) \right) u(i-1,j+1) + \Delta t f(i,j+1) \right]. \qquad (5.2.12)$$

Equation (5.2.12) is called implicit finite difference approximations to the mathematical model given in equation (5.1.1). Assuming $\theta = \frac{1}{2}$ in equation (5.2.12) could be written as the following backward in time centred in space (BTCS)-type finite difference formula to solve the given equation:

$$u(i,j+1) = \frac{1}{\left(1 + 2\left(\frac{cr}{pe}\right)\right)} \left[\left(\frac{cr}{2} + \left(\frac{cr}{pe}\right) \right) u(i-1,j+1) + u(i,j) \right.$$
$$\left. - \left(\frac{cr}{2} - \left(\frac{cr}{pe}\right) \right) u(i+1,j+1) + \Delta t f(i,j+1) \right]. \qquad (5.2.13)$$

This scheme utilizes the backward difference form for the time derivatives and centred difference form for all spatial derivatives in both advection and diffusion terms. Rearranging equation (5.2.13) the unknowns are on the left hand side of the equation and they give us a tri-diagonal matrix system to solve equations (5.1.1) - (5.1.4). Putting the so far assigned values of λ and μ in the equation (5.2.13) for solving Dirichlet boundary conditions, it is known that

$$(\lambda - \mu)u(i+1,j+1) + (1+2\mu)u(i,j+1) - (\lambda + \mu)u(i-1,j+1)$$
$$= u(i,j) + \Delta t f(i,j+1) \qquad (5.2.14)$$

Equation (5.2.14) gives us a set of linear equations at every spatial point $u(i+1, j+1)$, where $i = 1, 2, 3, \dots, M-1, j = 0, 1, 2, \dots, N-1$ and $u(0, j+1)$and $u(M, j+1)$ are fixed because they are boundary conditions.

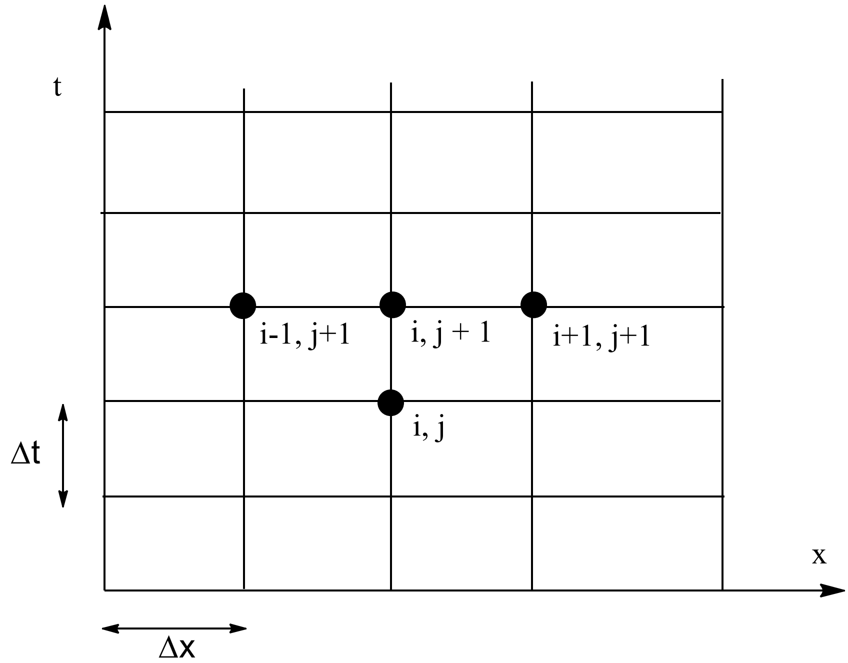


Figure 3: Represent point scheme for BTCS.

The tri-diagonal matrix system representation of equation (5.2.14) can be written in a matrix vector form as

$$\begin{bmatrix} 1+2\mu & \lambda-\mu & 0 & . & . & 0 \\ -(\lambda+\mu) & 1+2\mu & \lambda-\mu & . & . & . \\ 0 & . & . & . & . & . \\ . & & . & . & 0 & 0 \\ . & & & -(\lambda+\mu) & 1+2\mu & \lambda-\mu \\ 0 & . & . & 0 & -(\lambda+\mu) & 1+2\mu \end{bmatrix} \begin{bmatrix} u(1,j+1) \\ u(2,j+1) \\ . \\ \vdots \\ u(M-2,j+1) \\ u(M-1,j+1) \end{bmatrix}$$

$$= \begin{bmatrix} u(1,j) + (\lambda+\mu)u(0,j+1) + \Delta t f(1,j+1) \\ u(2,j) + \Delta t f(2,j+1) \\ . \\ \vdots \\ u(M-2,j) + \Delta t f(M-2,j+1) \\ u(M-1,j) - (\lambda-\mu)u(M,j+1) + \Delta t f(M-1,j+1) \end{bmatrix}. \qquad (5.2.15)$$

This system of equation is a tri-diagonal and strictly diagonally dominated which shows that it is non-singular. Non-singularity ensures that it is invertible and our equation will have a completely unique solution which we can obtain using Thomas algorithm

Computational procedure of implicit FDS: In implicit finite difference schemes, for evaluating one node in $j+1$ time level, we must know the value of grid-nodes that exist around it, in j and $j+1$ time levels. The initial condition $u(x,0)=h(x)$ gives the solution at all the nodal points on the initial line (level 0). The boundary conditions $u(0,x)=g_1(t), u(L,x)=g_2(t), 0\le t\le T$, give the solutions at all the nodal points on the boundary lines $x=0$ and $x=L$ for all time levels. We choose a value for Δx and Δt. The difference equations at all nodal points on the first time level are written. This system of equations is solved to obtain the values at all the nodal points on this time level. The computations are repeated for the required number of steps. If we perform n steps of computation, then we have computed the solutions up to time $t_n=n\Delta t$.

5.3. Neumann Boundary Conditions

In the previous section, we have presented numerical schemes used for solving the equation (5.1.1) with Dirichlet boundary conditions. Now we consider equation (5.1.1) with Neumann boundary conditions.

From equations (5.1.3) and (5.1.4), if $\delta_n=0$ and $\gamma_m\neq 0$, we have

$$u_x(0,t)=g_1(t), \qquad 0\le t\le T, \tag{5.3.1}$$
$$u_x(L,t)=g_2(t), \qquad 0\le t\le T, \tag{5.3.2}$$

which has Neumann conditions at $x=0$ and $x=L$.

It is possible to apply forward or backward difference to implement these boundary conditions at the left and right end of the domain, but it is generally most effective to apply the central difference formula via introducing the fictitious concentration $u(i-1,j)$ at the external grid point $x=(i-1)\Delta x$ and as shown in Figure 4. The boundary condition at $i-1$ is represented by;

$$u_x(0,j)=\frac{u(1,j)-u(-1,j)}{2\Delta x}.$$

Also, introduce $u(i+1,j)$ at the end of the external grid point $x=(i+1)\Delta x$. The boundary condition at $i+1$ can be represented by

$$u_x(i,j)=\frac{u(i+1,j)-u(i-1,j)}{2\Delta x}. \tag{5.3.3}$$

Figure 4: Introducing fictitious concentration.

The concentration $u(-1,j)$ and $u(M+1,j)$ are unknown and this leads to more equations. It is possible to eliminate $u(-1,j)$ and $u(M+1,j)$ between these equations. The above techniques are applied to find boundary conditions in the following schemes.

5.3.1. Imposing Neumann Boundary Conditions in the FTCS Scheme

Considering explicit FTCS scheme representation of equation (5.2.9), at $x = 0$ gives us

$$u(0,j+1) = (-\lambda+\mu)u(1,j) + (1-2\mu)u(0,j) + (\lambda+\mu)u(-1,j) + \Delta t f(0,j). \quad (5.3.4)$$

Applying central difference for the boundary at $x = 0$ at time level j, then we obtain

$$u_x(0,j) = \frac{u(1,j) - u(-1,j)}{2\Delta x}. \qquad (5.3.5)$$

Substituting equation (5.3.5) into equation (5.1.3), we get an approximation of the Neumann condition at $(0, j\Delta t)$ as

$$u(-1,j) = u(1,j) - 2\Delta x g_1(j\Delta t). \qquad (5.3.6)$$

Using equation (5.3.6), then equation (5.3.4) yields

$$u(0,j+1) = 2\mu u(1,j) + (1-2\mu)u(0,j) - 2(\lambda+\mu)\Delta x g_1(j\Delta t) + \Delta t f(0,j). \qquad (5.3.7)$$

Now consider explicit scheme at $x = L = M\Delta x$, then

$$u(M,j+1) = (-\lambda+\mu)u(M+1,j) + (1-2\mu)u(M,j) + (\lambda+\mu)u(M-1,j) \\ + \Delta t f(M,j). \qquad (5.3.8)$$

We apply the following central difference formula for right boundary condition at $x = L$, at time level j

$$u_x(M,j) = \frac{u(M+1,j) - u(M-1,j)}{2\Delta x}. \tag{5.3.9}$$

Substituting equation (5.3.9) into equation (5.3.2), we obtain an approximation of the Neumann condition at $((M+1)\,\Delta x, j\Delta t)$ as

$$u(M+1,j) = u(M-1,j) + 2\Delta x g_2(j\Delta t). \tag{5.3.10}$$

Using equation (5.3.10), then equation (5.3.8) yields

$$\begin{aligned} u(M,j+1) = (1-2\mu)u(M,j) + 2\mu u(M-1,j) + 2(-\lambda+\mu)\Delta x g_2(j\Delta t) \\ + \Delta t f(M,j). \end{aligned} \tag{5.3.11}$$

We can write in tri-diagonal matrix system form as

$$\begin{aligned} &\begin{bmatrix} u(0,j+1) \\ u(1,j+1) \\ \cdot \\ \vdots \\ u(M-1,j+1) \\ u(M,j+1) \end{bmatrix} \\ &= \begin{bmatrix} 1-2\mu & 2\mu & 0 & \cdot & \cdot & 0 \\ \lambda+\mu & 1-2\mu & -\lambda+\mu & & & \\ 0 & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & 0 \\ & & & \lambda+\mu & 1-2\mu & -\lambda+\mu \\ 0 & \cdot & \cdot & 0 & 2\mu & 1-2\mu \end{bmatrix} \begin{bmatrix} u(0,j) \\ u(1,j) \\ \cdot \\ \vdots \\ u(M-1,j) \\ u(M,j) \end{bmatrix} \\ &+ \begin{bmatrix} -2(\lambda+\mu)\Delta x g_1(j\Delta t) + \Delta t f(0,j) \\ \Delta t f(1,j) \\ \cdot \\ \vdots \\ \Delta t f(M-1,j) \\ 2(-\lambda+\mu)\Delta x g_2(j\Delta t) + \Delta t f(M,j) \end{bmatrix}. \end{aligned} \tag{5.3.12}$$

5.3.2. Imposing Neumann Boundary Conditions in the BTCS Scheme

Considering the implicit BTCS scheme representation of equation (5.2.14), at $x = 0$ gives us

$$(\lambda-\mu)u(1,j+1)+(1+2\mu)u(0,j+1)-(\lambda+\mu)u(-1,j+1) \\ =u(0,j)+\Delta t f(0,j+1). \qquad (5.3.13)$$

Applying central difference for the boundary at $x=0$ at time level $j+1$, then we get

$$u_x(0,j+1)=\frac{u(1,j+1)-u(-1,j+1)}{2\Delta x}. \qquad (5.3.14)$$

Substituting equation (5.3.14) into equation (5.1.3), we get an approximation of the Neumann condition at $(0,(j+1)\Delta t)$ as

$$u(-1,j+1)=u(1,j+1)-2\Delta x g_1\big((j+1)\Delta t\big). \qquad (5.3.15)$$

Using equation (5.3.15), then equation (5.3.13) yields

$$-2\mu u(1,j+1)+(1+2\mu)u(0,j+1) \\ =-2(\lambda+\mu)\Delta x g_1\big((j+1)\Delta t\big)+\Delta t f(0,j+1). \qquad (5.3.16)$$

Now, consider explicit scheme at $x=L$,

$$(\lambda-\mu)u(M+1,j+1)+(1+2\mu)u(M,j+1)-(\lambda+\mu)u(M-1,j+1) \\ =u(0,j)+\Delta t f(0,j+1). \qquad (5.3.17)$$

We apply the following central difference formula for right boundary condition at $x=L$ at time level $j+1$, then we get

$$u_x(M,j+1)=\frac{u(M+1,j+1)-u(M-1,j+1)}{2\Delta x}. \qquad (5.3.18)$$

Substituting equation (5.3.9) into equation (5.3.2), we obtain an approximation of the Neumann condition at $((M+1)\,\Delta x,(j+1)\Delta t)$ as

$$u(M+1,j+1)=u(M-1,j+1)+2\Delta x g_2(j\Delta t). \qquad (5.3.19)$$

Using equation (5.3.19), then equation (5.3.17) yields

$$(1+2\mu)u(M,j+1)-2\mu u(M-1,j+1) \\ =u(M,j)+2(\mu-\lambda)\Delta x g_2\big((j+1)\Delta t\big)+\Delta t f(M,j+1). \qquad (5.3.20)$$

We can write in tri-diagonal matrix system form as

$$\begin{bmatrix} 1+2\mu & -2\mu & 0 & . & . & 0 \\ -(\lambda+\mu) & 1+2\mu & \lambda-\mu & . & . & . \\ 0 & . & . & . & . & . \\ . & & . & . & 0 & 0 \\ . & & & -(\lambda+\mu) & 1+2\mu & \lambda-\mu \\ 0 & . & . & 0 & -2\mu & 1+2\mu \end{bmatrix} \begin{bmatrix} u(0,j+1) \\ u(1,j+1) \\ . \\ \vdots \\ u(M-1,j+1) \\ u(M,j+1) \end{bmatrix}$$

$$= \begin{bmatrix} u(0,j) - 2(\lambda+\mu)\Delta x g_1\big((j+1)\Delta t\big) + \Delta t f(0,j+1) \\ u(1,j) + \Delta t f(1,j+1) \\ . \\ \vdots \\ u(M-1,j) + \Delta t f(M-1,j+1) \\ u(M,j) + 2(\mu-\lambda)\Delta x g_2\big((j+1)\Delta t\big) + \Delta t f(M,j+1) \end{bmatrix}. \qquad (5.3.21)$$

5.4. Convergence Analysis of the Finite Difference Schemes

Lax Theorem allows the complicated approach required to prove convergence of the solution of the finite difference equation to the solution of PDE to be replaced by using the relatively easy approaches of proving the stability of a finite difference equation and its consistency within the PDE. Strictly speaking, Lax has proved that under appropriate conditions a consistent scheme is convergent if and only if it is stable. As a result, showing the consistency and stability of the FDS is sufficient for convergence. After surveying the relevant literature on the concern, we observed that no practical stability criterion or the rate of convergence exist for (5.2.6). This finding was unexpected, in view of the simplicity that this kind of scheme has due to the fact stability is usually independent of the source term (Li et al., 2017).

5.4.1. Consistency and Stability Analysis of Explicit FTCS Scheme

In this Thesis context, the idea of consistency illustrates how well the schemes approximate equation (5.1.1), usually in smooth regions of the solutions (i.e. the solution is continuously differentiable up to order four). In view that finite difference discretization converges at the rate of the Truncation errors if the exact solution is smooth enough, we expand the exact solution at the mesh points of the scheme with a Taylor series and insert the Taylor expansions in the scheme to calculate the Truncation error and determine its order in the approximation used. Then, we see that as the discrete step sizes approach to zero, their Truncation error also approaches to zero which suggests that the difference approximations are consistent or compatible with the given equation.

In the following, we illustrate how this will be carried out in terms of a Taylor series expansion. In the showing of the consistency of the provided schemes, we assume that the exact solution has continuous higher order partial derivatives in the computational domain. Substituting the previously assigned values of cr and pe into the scheme obtained in equation (5.2.8) and rearranging, we have the local truncation error which is given by:

$$T(x_i,t_j) = \frac{u(x_i,t_{j+1}) - u(x_i,t_j)}{\Delta t} + \alpha\left(\frac{u(x_{i+1},t_j) - u(x_{i-1},t_j)}{2\Delta x}\right) - \beta\left(\frac{u(x_{i+1},t_j) - 2u(x_i,t_j) + u(x_{i-1},t_j)}{(\Delta x)^2}\right) - f(x_i,t_j). \tag{5.4.1}$$

By using Taylor series expansion, we have:

$$u(x_i,t_{j+1}) = u(x_i,t_j) + \Delta t\frac{\partial u(x_i,t_j)}{\partial t} + \frac{(\Delta t)^2}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2} + \frac{(\Delta t)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial t^3} + \cdots \tag{5.4.2}$$

$$u(x_{i+1},t_j) = u(x_i,t_j) + \Delta x\frac{\partial u(x_i,t_j)}{\partial x} + \frac{(\Delta x)^2}{2}\frac{\partial^2 u(x_i,t_j)}{\partial x^2} + \frac{(\Delta x)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3} + \frac{(\Delta x)^4}{24}\frac{\partial^4 u(x_i,t_j)}{\partial x^4} + \cdots \tag{5.4.3}$$

$$u(x_{i-1},t_j) = u(x_i,t_j) - \Delta x\frac{\partial u(x_i,t_j)}{\partial x} + \frac{(\Delta x)^2}{2}\frac{\partial^2 u(x_i,t_j)}{\partial x^2} - \frac{(\Delta x)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3} + \frac{(\Delta x)^4}{24}\frac{\partial^4 u(x_i,t_j)}{\partial x^4} + \cdots \tag{5.4.4}$$

From equation (5.4.2), we have that

$$u(x_i,t_{j+1}) - u(x_i,t_j) = \Delta t\frac{\partial u(x_i,t_j)}{\partial t} + \frac{(\Delta t)^2}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2} + \cdots, \tag{5.4.5}$$

Subtracting equation (5.4.4) from equation (5.4.3), we have

$$u(x_{i+1},t_j) - u(x_{i-1},t_j) = 2\Delta x\frac{\partial u(x_i,t_j)}{\partial x} + 2\frac{(\Delta x)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3} + \cdots \tag{5.4.6}$$

Adding equation (5.4.3) and equation (5.4.4), we obtain

$$u(x_{i+1},t_j) + u(x_{i-1},t_j) = 2u(x_i,t_j) + (\Delta x)^2\frac{\partial^2 u(x_i,t_j)}{\partial x^2} + \frac{(\Delta x)^4}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4} + \cdots \tag{5.4.7}$$

Again from equation (5.4.7), we have

$$u(x_{i+1},t_j)-2u(x_i,t_j)+u(x_{i-1},t_j)$$
$$=(\Delta x)^2\frac{\partial^2 u(x_i,t_j)}{\partial x^2}+\frac{(\Delta x)^4}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4}+\cdots \tag{5.4.8}$$

Substituting equations (5.4.5), (5.4.6) and (5.4.8), yields

$$\begin{aligned}T(x_i,t_j)=&\left(\frac{\partial u(x_i,t_j)}{\partial t}+\frac{\Delta t}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2}+\cdots\right)\\&+\alpha\left(\frac{\partial u(x_i,t_j)}{\partial x}+\frac{(\Delta x)^2}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3}+\cdots\right)\\&-\beta\left(\frac{\partial^2 u(x_i,t_j)}{\partial x^2}+\frac{(\Delta x)^2}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4}+\cdots\right)-f(x_i,t_j),\\=&\frac{\partial u(x_i,t_j)}{\partial t}+\alpha\frac{\partial u(x_i,t_j)}{\partial x}-\beta\frac{\partial^2 u(x_i,t_j)}{\partial x^2}-f(x_i,t_j)+\frac{\Delta t}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2}\\&+\alpha\frac{(\Delta x)^2}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3}-\beta\frac{(\Delta x)^2}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4},\end{aligned} \tag{5.4.9}$$

where $u(x_i,t_j)$ is the solution of the ADE with source term. Then equation (5.4.9) becomes

$$T(x_i,t_j)=\frac{\Delta t}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2}+\alpha\frac{(\Delta x)^2}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3}-\beta\frac{(\Delta x)^2}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4} \tag{5.4.10}$$

Thus, $T\to 0$ as Δt and $\Delta x\to 0$. So that, the scheme is consistent with the mathematical model given in equation (5.1.1) and the order of the scheme is $O(\Delta t+(\Delta x)^2)$. Thus, the FTCS solution to equation (5.1.1) is $O(\Delta t)$ accurate in time and $O((\Delta x)^2)$ accurate in space.

Consistency is only a necessary but not a sufficient condition for convergence. In the above, we have proved that the explicit FTCS scheme is consistent with equation (5.1.1), so, to complete proving convergence of the solution we have to prove that explicit FTCS scheme is also stable. As a result, the stability of the explicit FTCS scheme obtained in equation (5.2.8) has been investigated by using the Von-Neumann stability analysis. To accomplish that, applying the procedures of Von-Neumann stability analysis to the equation (5.2.8) and after using the assigned values and then setting $\eta=\frac{\Delta t}{(\Delta x)^2}$, we obtain (Note that the stability is usually independent of the source term thus set $f=0$.)

$$g(\xi)e^{ijh\xi}=e^{ijh\xi}-\frac{\alpha\Delta x\eta}{2}\left(e^{i(j+1)h\xi}-e^{i(j-1)h\xi}\right)+\beta\eta\left(e^{i(j-1)h\xi}-2e^{ijh\xi}+e^{i(j+1)h\xi}\right),$$

$$g(\xi)e^{ijh\xi}=e^{ijh\xi}-\frac{\alpha\Delta x\eta}{2}e^{ijh\xi}\left(e^{ih\xi}-e^{-ih\xi}\right)+\beta\eta e^{ijh\xi}\left(e^{-ih\xi}-2e^{ijh\xi}+e^{ih\xi}\right),$$

$$g(\xi)e^{ijh\xi} = e^{ijh\xi}\left(-\frac{\alpha\Delta x\eta}{2}\left(e^{ih\xi} - e^{-ih\xi}\right) + \beta\eta\left(e^{-ih\xi} - 2e^{ijh\xi} + e^{ih\xi}\right)\right). \qquad (5.4.11)$$

Removing the $e^{ijh\xi}$ factor on both sides, then equation (5.4.11) becomes

$$g(\xi) = 1 - \frac{\alpha\Delta x\eta}{2}\left(e^{ih\xi} - e^{-ih\xi}\right) + \beta\eta\left(e^{-ih\xi} - 2 + e^{ih\xi}\right). \qquad (5.4.12)$$

By Euler's formula

$$\left.\begin{aligned} e^{ih\xi} &= \cos(h\xi) + isin(h\xi) \\ e^{-ih\xi} &= \cos(h\xi) - isin(h\xi) \end{aligned}\right\}. \qquad (5.4.13)$$

Using equation (5.4.13) on the equation (5.4.12) yields

$$\begin{aligned} g(\xi) &= 1 - \frac{\alpha\Delta x\eta}{2}\left(2isin(h\xi)\right) + \beta\eta(2\cos(h\xi) - 2), \\ &= 1 - 2\beta\eta(1 - \cos(h\xi)) - \alpha\Delta x\eta isin(h\xi). \end{aligned}$$

Now, by taking the magnitude of $g(\xi)$ we obtain

$$|g(\xi)|^2 = 1 - 4\beta\eta(1 - \cos(h\xi)) + 4\beta^2\eta^2(1 - \cos(h\xi))^2 + \alpha^2h^2\eta^2sin^2(h\xi).$$

For $(1 - \cos(h\xi)) > 0$ the condition $|g(\xi)|^2 \leq 1$ is equivalent to

$$-4\beta\eta(1 - \cos(h\xi)) + 4\beta^2\eta^2(1 - \cos(h\xi))^2 + \alpha^2h^2\eta^2sin^2(h\xi) \leq 0,$$

$$-4\beta\eta(1 - \cos(h\xi)) + 4\beta^2\eta^2(1 - \cos(h\xi))^2 \leq -\alpha^2h^2\eta^2sin^2(h\xi),$$

$$4\beta\eta(1 - \cos(h\xi))\left(-1 + \beta\eta(1 - \cos(h\xi))\right) \leq -\alpha^2h^2\eta^2sin^2(h\xi). \qquad (5.4.14)$$

Dividing equation (5.4.14) by $4\beta\eta(1 - \cos(h\xi))$ and using trigonometric identity $cos^2(h\xi) + sin^2(h\xi) = 1$, we get

$$\beta\eta(1 - \cos(h\xi)) - 1 \leq \frac{-\alpha^2h^2\eta^2\left(1 - cos^2(h\xi)\right)}{4\beta\eta(1 - \cos(h\xi))},$$

$$2\beta\eta - 1 \leq 0,$$

$$\eta \leq \frac{1}{2\beta}.$$

From which it follows that if the Von-Neumann criterion holds, that is if $|g(\xi)| \leq 1$ for all h and ξ satisfying $1 - \cos(h\xi) > 0$, then $|g(\xi)|^2 \leq 1$. Therefore, the obtained stability of the scheme developed in equation (5.2.8) would guarantee that the scheme is stable when

$$\Delta t \leq \frac{(\Delta x)^2}{2\beta}. \qquad (5.4.15)$$

This implies that as we decrease the spatial interval Δx for better accuracy, we must also decrease the time step Δt at the cost of more computations in order not to lose the stability. Therefore, the explicit FTCS scheme is convergent by the result of Lax-Equivalence Theorem.

5.4.1. Consistency and Stability Analysis of Implicit BTCS Scheme

Local truncation error of the BTCS scheme is

$$\begin{aligned} T(x_i,t_j) &= \frac{u(x_i,t_{j+1})-u(x_i,t_j)}{\Delta t}+\alpha\left(\frac{u(x_{i+1},t_{j+1})-u(x_{i-1},t_{j+1})}{2\Delta x}\right) \\ &\quad -\beta\left(\frac{u(x_{i+1},t_{j+1})-2u(x_i,t_{j+1})+u(x_{i-1},t_{j+1})}{(\Delta x)^2}\right)-f(x_i,t_{j+1}). \qquad (5.4.16) \end{aligned}$$

Use Taylor's expansion for $u(x_{i-1},t_{j+1})$ and $u(x_{i+1},t_{j+1})$, we have the following

$$\begin{aligned} u(x_{i+1},t_{j+1}) &= u(x_i,t_j)+\Delta x\frac{\partial u(x_i,t_j)}{\partial x}+\Delta t\frac{\partial u(x_i,t_j)}{\partial t}+\frac{(\Delta x)^2}{2}\frac{\partial^2 u(x_i,t_j)}{\partial x^2} \\ &\quad +\frac{(\Delta t)^2}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2}+\Delta x\Delta t\frac{\partial^2 u(x_i,t_j)}{\partial x\partial t}+\frac{(\Delta x)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3} \\ &\quad +\frac{(\Delta t)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial t^3}+\frac{(\Delta x)^2}{2}\Delta t\frac{\partial^3 u(x_i,t_j)}{\partial x^2\partial t}+\Delta x\frac{(\Delta t)^2}{2}\frac{\partial^3 u(x_i,t_j)}{\partial x\partial t^2} \\ &\quad +\frac{(\Delta x)^4}{24}\frac{\partial^4 u(x_i,t_j)}{\partial x^4}+\frac{(\Delta t)^4}{24}\frac{\partial^4 u(x_i,t_j)}{\partial t^4}+\frac{(\Delta x)^2(\Delta t)^2}{4}\frac{\partial^4 u(x_i,t_j)}{\partial x^2\partial t^2} \\ &\quad +\frac{(\Delta x)^3}{6}\Delta t\frac{\partial^4 u(x_i,t_j)}{\partial x^3\partial t}+\Delta x\frac{(\Delta t)^3}{6}\frac{\partial^4 u(x_i,t_j)}{\partial x\partial t^3}+\cdots, \qquad (5.4.17) \end{aligned}$$

$$\begin{aligned} u(x_{i-1},t_{j+1}) &= u(x_i,t_j)-\Delta x\frac{\partial u(x_i,t_j)}{\partial x}+\Delta t\frac{\partial u(x_i,t_j)}{\partial t}+\frac{(\Delta x)^2}{2}\frac{\partial^2 u(x_i,t_j)}{\partial x^2} \\ &\quad +\frac{(\Delta t)^2}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2}-\Delta x\Delta t\frac{\partial^2 u(x_i,t_j)}{\partial x\partial t}-\frac{(\Delta x)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3} \\ &\quad +\frac{(\Delta t)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial t^3}+\frac{(\Delta x)^2}{2}\Delta t\frac{\partial^3 u(x_i,t_j)}{\partial x^2\partial t}-\Delta x\frac{(\Delta t)^2}{2}\frac{\partial^3 u(x_i,t_j)}{\partial x\partial t^2} \\ &\quad +\frac{(\Delta x)^4}{24}\frac{\partial^4 u(x_i,t_j)}{\partial x^4}+\frac{(\Delta t)^4}{24}\frac{\partial^4 u(x_i,t_j)}{\partial t^4}+\frac{(\Delta x)^2(\Delta t)^2}{4}\frac{\partial^4 u(x_i,t_j)}{\partial x^2\partial t^2} \\ &\quad -\frac{(\Delta x)^3}{6}\Delta t\frac{\partial^4 u(x_i,t_j)}{\partial x^3\partial t}-\Delta x\frac{(\Delta t)^3}{6}\frac{\partial^4 u(x_i,t_j)}{\partial x\partial t^3}+\cdots \qquad (5.4.18) \end{aligned}$$

Adding equation (5.4.17) and (5.4.18), we get

$$
\begin{aligned}
&u(x_{i+1},t_{j+1}) + u(x_{i-1},t_{j+1}) \\
&\qquad = 2u(x_i,t_j) + 2\Delta t\frac{\partial u(x_i,t_j)}{\partial t} + (\Delta x)^2\frac{\partial^2 u(x_i,t_j)}{\partial x^2} + (\Delta t)^2\frac{\partial^2 u(x_i,t_j)}{\partial t^2} \\
&\qquad + \frac{2(\Delta t)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial t^3} + (\Delta x)^2\Delta t\frac{\partial^3 u(x_i,t_j)}{\partial x^2\partial t} + \frac{(\Delta x)^4}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4} \\
&\qquad + \frac{(\Delta t)^4}{12}\frac{\partial^4 u(x_i,t_j)}{\partial t^4} + \frac{(\Delta x)^2(\Delta t)^2}{2}\frac{\partial^4 u(x_i,t_j)}{\partial x^2\partial t^2} + \cdots \qquad (5.4.19)
\end{aligned}
$$

Using equation (5.4.19) and equation (5.4.2), we get

$$
\begin{aligned}
&u(x_{i+1},t_{j+1}) - 2u(x_i,t_{j+1}) + u(x_{i-1},t_{j+1}) \\
&\qquad = (\Delta x)^2\frac{\partial^2 u(x_i,t_j)}{\partial x^2} + \frac{(\Delta x)^4}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4} + \cdots \qquad (5.4.20)
\end{aligned}
$$

Subtracting equation (5.4.18) from equation (5.4.17), we have

$$
\begin{aligned}
&u(x_{i+1},t_{j+1}) - u(x_{i-1},t_{j+1}) \\
&\qquad = 2\Delta x\frac{\partial u(x_i,t_j)}{\partial x} + 2\Delta x\Delta t\frac{\partial^2 u(x_i,t_j)}{\partial x\partial t} - \frac{2(\Delta x)^3}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3} \\
&\qquad + \Delta x(\Delta t)^2\frac{\partial^3 u(x_i,t_j)}{\partial x\partial t^2} + \frac{2(\Delta x)^3}{6}\Delta t\frac{\partial^4 u(x_i,t_j)}{\partial x^3\partial t} + 2\Delta x\frac{(\Delta t)^3}{6}\frac{\partial^4 u(x_i,t_j)}{\partial x\partial t^3} \\
&\qquad + \cdots \qquad (5.4.21)
\end{aligned}
$$

Substituting equations (5.4.2), (5.4.21) and (5.4.20), we obtain

$$
\begin{aligned}
T(x_i,t_j) &= \left(\frac{\partial u(x_i,t_j)}{\partial t} + \frac{\Delta t}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2} + \cdots\right) \\
&\quad + \alpha\left(\frac{\partial u(x_i,t_j)}{\partial x} + \Delta t\frac{\partial^2 u(x_i,t_j)}{\partial x\partial t} - \frac{(\Delta x)^2}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3} + \cdots\right) \\
&\quad - \beta\left(\frac{\partial^2 u(x_i,t_j)}{\partial x^2} + \frac{(\Delta x)^2}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4} + \cdots\right) - f(x_i,t_j),
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{\partial u(x_i,t_j)}{\partial t} + \alpha\frac{\partial u(x_i,t_j)}{\partial x} - \beta\frac{\partial^2 u(x_i,t_j)}{\partial x^2} - f(x_i,t_j) + \frac{\Delta t}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2} \\
&\qquad + \alpha\Delta t\frac{\partial^2 u(x_i,t_j)}{\partial x\partial t} - \alpha\frac{(\Delta x)^2}{6}\frac{\partial^3 u(x_i,t_j)}{\partial x^3} - \beta\frac{(\Delta x)^2}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4}, \qquad 5.4.22)
\end{aligned}
$$

where $u(x_i,t_j)$ is the solution of the ADE with source term. Then equation (5.4.22) becomes

$$
T(x_i,t_j) = \frac{\Delta t}{2}\frac{\partial^2 u(x_i,t_j)}{\partial t^2} + \alpha\Delta t\frac{\partial^2 u(x_i,t_j)}{\partial x\partial t} - \beta\frac{(\Delta x)^2}{12}\frac{\partial^4 u(x_i,t_j)}{\partial x^4} + \cdots \qquad 5.4.23)
$$

Thus, $T \to 0$ as Δt and $\Delta x \to 0$. So that, the scheme is consistent with the mathematical model given in equation (5.1.1) and the order of the scheme is $O(\Delta t + (\Delta x)^2)$. Thus, the BTCS solution to equation (5.1.1) is $O(\Delta t)$ accurate in time and $O((\Delta x)^2)$ accurate in space.

Applying the Von-Neumann stability analysis in the equation (5.2.13), we obtain

$$g(\xi)e^{ijh\xi} + \frac{cr}{2}\left(g(\xi)e^{i(j+1)h\xi} - g(\xi)e^{i(j-1)h\xi}\right) - \left(\frac{cr}{pe}\right)\left(g(\xi)e^{i(j+1)h\xi} - 2g(\xi)e^{ijh\xi} + g(\xi)e^{i(j-1)h\xi}\right) = e^{ijh\xi},$$

$$g(\xi)e^{ijh\xi}\left(1 + \frac{cr}{2}\left(e^{ih\xi} - e^{-ih\xi}\right) - \left(\frac{cr}{pe}\right)\left(e^{ih\xi} - 2 + e^{-ih\xi}\right)\right) = e^{ih\xi}.$$

Removing the $e^{ijh\xi}$ factor, we obtain

$$g(\xi)\left(1 + \frac{cr}{2}\left(e^{ih\xi} - e^{-ih\xi}\right) - \left(\frac{cr}{pe}\right)\left(e^{ih\xi} - 2 + e^{-ih\xi}\right)\right) = 1,$$

$$g(\xi) = \frac{1}{1 + \frac{cr}{2}\left(e^{ih\xi} - e^{-ih\xi}\right) - \left(\frac{cr}{pe}\right)\left(e^{ih\xi} - 2 + e^{-ih\xi}\right)}. \tag{5.4.24}$$

Substituting Euler's formula given in equation (5.4.13) into equation (5.4.24), we obtain

$$g(\xi) = \frac{1}{1 + cr2isin(h\xi) - 2\left(\frac{cr}{pe}\right)(\cos(h\xi) - 1)}, \tag{5.4.24}$$

and having the equality

$$\cos(h\xi) - 1 = -2sin^2\left(\frac{h\xi}{2}\right), \tag{5.4.25}$$

which is obtained from trigonometric identity. Using equation (5.4.25), then equation (5.4.24) becomes:

$$g(\xi) = \frac{1}{1 + 4\left(\frac{cr}{pe}\right)sin^2\left(\frac{h\xi}{2}\right) + 2crisin(h\xi)}.$$

Taking the modulus and using the property of complex numbers, it follows that:

$$|g(\xi)|^2 = \frac{1}{\left(1 + 4\left(\frac{cr}{pe}\right)sin^2\left(\frac{h\xi}{2}\right)\right)^2 + \left(2crsin(h\xi)\right)^2}.$$

Taking square root on both sides, we have that

$$|g(\xi)| = \sqrt{\frac{1}{\left(1 + 4\left(\frac{cr}{pe}\right)sin^2\left(\frac{h\xi}{2}\right)\right)^2 + 4cr^2sin^2(h\xi)}} \leq 1.$$

Hence, it is obvious that the denominator is greater than or equal to 1 and so that the fraction under the square root is less than or equal to 1 thus the inequality of the $|g(\xi)| \leq 1$ is exactly true. As a result, the Von-Neumann stability analysis implies that the scheme in equation (5.2.13) is unconditionally stable. Therefore, the implicit BTCS scheme is convergent.

Remark: Stability analysis for finite difference approximations to time dependent PDEs is often tricky, tedious and difficult. For this reason, Gershgorin's Theorem is used to check and confirm the accuracy of the above stability results.

5.5. Derivation of sixth order Finite Difference Approximation

In this section, we derive sixth order central finite difference approximations of first and second derivatives of advection and diffusion terms, respectively. However, deriving higher order FDM from Tayler series is cumbersome task. Now, we derive the first derivative by using a Taylor series expansion. We start the procedure by expressing the value of $u_{i-3}, u_{i-2}, u_{i-1}, u_{i+1}, u_{i+2}$ and u_{i+3} in terms of u_i as follows:

$$\begin{aligned} u_{i-3} = u_i - 3h\left(\frac{\partial u}{\partial x}\right)_i + \frac{(3h)^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i - \frac{(3h)^3}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i + \frac{(3h)^4}{24}\left(\frac{\partial^4 u}{\partial x^4}\right)_i \\ - \frac{(3h)^5}{120}\left(\frac{\partial^5 u}{\partial x^5}\right)_i + \frac{(3h)^6}{720}\left(\frac{\partial^6 u}{\partial x^6}\right)_i - \frac{(3h)^7}{5040}\left(\frac{\partial^7 u}{\partial x^7}\right)_i + \cdots, \end{aligned} \qquad (5.5.1)$$

$$\begin{aligned} u_{i-2} = u_i - 2h\left(\frac{\partial u}{\partial x}\right)_i + \frac{(2h)^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i - \frac{(2h)^3}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i + \frac{(2h)^4}{24}\left(\frac{\partial^4 u}{\partial x^4}\right)_i \\ - \frac{(2h)^5}{120}\left(\frac{\partial^5 u}{\partial x^5}\right)_i + \frac{(2h)^6}{720}\left(\frac{\partial^6 u}{\partial x^6}\right)_i - \frac{(2h)^7}{5040}\left(\frac{\partial^7 u}{\partial x^7}\right)_i + \cdots, \end{aligned} \qquad (5.5.2)$$

$$\begin{aligned} u_{i-1} = u_i - h\left(\frac{\partial u}{\partial x}\right)_i + \frac{h^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i - \frac{h^3}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i + \frac{h^4}{24}\left(\frac{\partial^4 u}{\partial x^4}\right)_i - \frac{h^5}{120}\left(\frac{\partial^5 u}{\partial x^5}\right)_i \\ + \frac{h^6}{720}\left(\frac{\partial^6 u}{\partial x^6}\right)_i - \frac{h^7}{5040}\left(\frac{\partial^7 u}{\partial x^7}\right)_i + \cdots, \end{aligned} \qquad (5.5.3)$$

$$\begin{aligned} u_{i+1} = u_i + h\left(\frac{\partial u}{\partial x}\right)_i + \frac{h^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i + \frac{h^3}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i + \frac{h^4}{24}\left(\frac{\partial^4 u}{\partial x^4}\right)_i + \frac{h^5}{120}\left(\frac{\partial^5 u}{\partial x^5}\right)_i \\ + \frac{h^6}{720}\left(\frac{\partial^6 u}{\partial x^6}\right)_i + \frac{h^7}{5040}\left(\frac{\partial^7 u}{\partial x^7}\right)_i + \cdots, \end{aligned} \qquad (5.5.4)$$

$$u_{i+2} = u_i + (2h)\left(\frac{\partial u}{\partial x}\right)_i + \frac{(2h)^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i + \frac{(2h)^3}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i + \frac{(2h)^4}{24}\left(\frac{\partial^4 u}{\partial x^4}\right)_i + \frac{(2h)^5}{120}\left(\frac{\partial^5 u}{\partial x^5}\right)_i + \frac{(2h)^6}{720}\left(\frac{\partial^6 u}{\partial x^6}\right)_i + \frac{(2h)^7}{5040}\left(\frac{\partial^7 u}{\partial x^7}\right)_i + \cdots, \tag{5.5.5}$$

$$u_{i+3} = u_i + (3h)\left(\frac{\partial u}{\partial x}\right)_i + \frac{(3h)^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i + \frac{(3h)^3}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i + \frac{(3h)^4}{24}\left(\frac{\partial^4 u}{\partial x^4}\right)_i + \frac{(3h)^5}{120}\left(\frac{\partial^5 u}{\partial x^5}\right)_i + \frac{(3h)^6}{720}\left(\frac{\partial^6 u}{\partial x^6}\right)_i + \frac{(3h)^7}{5040}\left(\frac{\partial^7 u}{\partial x^7}\right)_i \ldots, \tag{5.5.6}$$

As such, we can express the first derivative by multiplying equations (5.5.1), (5.5.2), (5.5.3), (5.5.4), (5.5.5) and (5.5.6) by the coefficients a_1, a_2, a_3, a_4, a_5 and a_6 respectively. Then taking the summation of these six equations (5.5.1) – (5.5.6) and upon rearrangement of obtained equation and collecting like terms of factors and pulling out $\left(\frac{\partial u}{\partial x}\right)_i, \left(\frac{\partial^2 u}{\partial x^2}\right)_i, \left(\frac{\partial^3 u}{\partial x^3}\right)_i, \left(\frac{\partial^4 u}{\partial x^4}\right)_i, \left(\frac{\partial^5 u}{\partial x^5}\right)_i, \left(\frac{\partial^6 u}{\partial x^6}\right)_i$ and $\left(\frac{\partial^7 u}{\partial x^7}\right)_i$, one obtains the following abridged expression:

$$\begin{aligned}
a_1 u_{i-3} + a_2 u_{i-2} + a_3 u_{i-1} + a_4 u_{i+1} + a_5 u_{i+2} + a_6 u_{i+3} \\
&= (a_1 + a_2 + a_3 + a_4 + a_5 + a_6) u_i \\
&+ h\left\{(-3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6)\left(\frac{\partial u}{\partial x}\right)_i\right. \\
&+ \frac{h}{2}(9a_1 + 4a_2 + a_3 + a_4 + 4a_5 + 9a_6)\left(\frac{\partial^2 u}{\partial x^2}\right)_i \\
&+ \frac{h^2}{6}(-27a_1 - 8a_2 - a_3 + a_4 + 8a_5 + 27a_6)\left(\frac{\partial^3 u}{\partial x^3}\right)_i \\
&+ \frac{h^3}{24}(81a_1 + 16a_2 + a_3 + a_4 + 16a_5 + 81a_6)\left(\frac{\partial^4 u}{\partial x^4}\right)_i \\
&+ \frac{h^4}{120}(-243a_1 - 32a_2 - a_3 + a_4 + 32a_5 + 243a_6)\left(\frac{\partial^5 u}{\partial x^5}\right)_i \\
&+ \frac{h^5}{720}(729a_1 + 64a_2 + a_3 + a_4 + 64a_5 + 729a_6)\left(\frac{\partial^6 u}{\partial x^6}\right)_i \\
&+ \frac{h^6}{5040}(-2187a_1 - 128a_2 - a_3 + a_4 + 128a_5 \\
&\left. + 2187a_6)\left(\frac{\partial^7 u}{\partial x^7}\right)_i\right\}.
\end{aligned} \tag{5.5.7}$$

At this stage; in order to satisfy the accurate sixth order in equation (5.5.7), we need to find the values of the coefficients a_1, a_2, a_3, a_4, a_5 and a_6 such that the coefficient of $\left(\frac{\partial u}{\partial x}\right)_i$ must equal 1 and the coefficients of $\left(\frac{\partial^2 u}{\partial x^2}\right)_i, \left(\frac{\partial^3 u}{\partial x^3}\right)_i, \left(\frac{\partial^4 u}{\partial x^4}\right)_i, \left(\frac{\partial^5 u}{\partial x^5}\right)_i$ and $\left(\frac{\partial^6 u}{\partial x^6}\right)_i$ must be zeroes and the coefficient of $\left(\frac{\partial^7 u}{\partial x^7}\right)_i$ should be not zero. We can call the coefficient of u_i to be $b = a_1 + a_2 + a_3 + a_4 + a_5 + a_6$, thus, one obtains the following linear equations to be solved:

$$\begin{aligned}
&A_1: && -b + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 0\\
&A_2: && -3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6 = 1\\
&A_3: && 9a_1 + 4a_2 + a_3 + a_4 + 4a_5 + 9a_6 = 0\\
&A_4: && -27a_1 - 8a_2 - a_3 + a_4 + 8a_5 + 27a_6 = 0\\
&A_5: && 81a_1 + 16a_2 + a_3 + a_4 + 16a_5 + 81a_6 = 0\\
&A_6: && -243a_1 - 32a_2 - a_3 + a_4 + 32a_5 + 243a_6 = 0\\
&A_7: && 729a_1 + 64a_2 + a_3 + a_4 + 64a_5 + 729a_6 = 0,
\end{aligned} \tag{5.5.8}$$

where equation (5.5.7) has truncation error as:

$$\tau_c' = \frac{h^6}{5040}(-2187a_1 - 128a_2 - a_3 + a_4 + 128a_5 + 2187a_6)\left(\frac{\partial^7 u}{\partial x^7}\right)_i. \tag{5.5.9}$$

The seven equation (5.5.8) are solved for the unknowns $b, a_1, a_2, a_3, a_4, a_5$ and a_6. The first step is to use equation A_2 to eliminate the unknown a_1 from A_3, A_4, A_5, A_6 and A_7 by performing:

$(A_3 + 3A_2) \rightarrow A_3,$

$(A_4 - 9A_2) \rightarrow A_4,$

$(A_5 + 27A_2) \rightarrow A_5,$

$(A_6 - 81A_2) \rightarrow A_6,$

$(A_7 + 243A_2) \rightarrow A_7.$

Resulting in the system

$$\begin{aligned}
&A_1: && -b + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 0\\
&A_2: && -3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6 = 1\\
&A_3: && -2a_2 - 2a_3 + 4a_4 + 10a_5 + 18a_6 = 3\\
&A_4: && 10a_2 + 8a_3 - 8a_4 - 10a_5 = -9\\
&A_5: && -38a_2 - 26a_3 + 28a_4 + 70a_5 + 162a_6 = 27\\
&A_6: && 130a_2 + 80a_3 - 80a_4 - 130a_5 = -81
\end{aligned} \tag{5.5.10}$$

$$A_7: \qquad -422a_2 - 242a_3 + 244a_4 + 550a_5 + 145a_6 = 243,$$

where the new equations (5.5.10) are, for simplicity, again labelled $A_1, A_2, A_3, A_4, A_5, A_6$ and A_7. In the new system (5.5.10), A_3 is used to eliminate a_2 from A_4, A_5, A_6 and A_7 by the operation:

$(A_4 + 5A_3) \rightarrow A_4,$
$(A_5 - 19A_3) \rightarrow A_5,$
$(A_6 + 65A_3) \rightarrow A_6,$
$(A_7 - 211A_3) \rightarrow A_7.$

Resulting in the system

$$\begin{array}{ll}
A_1: & -b + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 0 \\
A_2: & -3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6 = 1 \\
A_3: & -2a_2 - 2a_3 + 4a_4 + 10a_5 + 18a_6 = 3 \\
A_4: & -2a_3 + 12a_4 + 40a_5 + 90a_6 = 6 \\
A_5: & 12a_3 - 48a_4 - 120a_5 - 180a_6 = -30 \\
A_6: & -50a_3 + 180a_4 + 520a_5 + 1170a_6 = 114 \\
A_7: & 180a_3 - 600a_4 - 1560a_5 - 2340a_6 = -390,
\end{array} \tag{5.5.11}$$

where the new equations (5.5.11) are, for simplicity, again labelled A_1, A_2, A_3, A_4, A_5 and A_6. In the new system (5.5.11), A_4 is used to eliminate a_3 from A_5, A_6 and A_7 by the operation:

$(A_4 - 6A_3) \rightarrow A_4,$
$(A_5 - 25A_3) \rightarrow A_5,$
$(A_6 - 90A_3) \rightarrow A_6,$

Resulting in the system:

$$\begin{array}{ll}
A_1: & -b + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 0 \\
A_2: & -3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6 = 1 \\
A_3: & -2a_2 - 2a_3 + 4a_4 + 10a_5 + 18a_6 = 3 \\
A_4: & -2a_3 + 12a_4 + 40a_5 + 90a_6 = 6 \\
A_5: & 24a_4 + 120a_5 + 360a_6 = 6 \\
A_6: & -120a_4 - 480a_5 - 1080a_6 = -36 \\
A_7: & 480a_4 + 2040a_5 + 5760a_6 = 150,
\end{array} \tag{5.5.12}$$

where the new equations (5.5.12) are, for simplicity, again labelled $A_1, A_2, A_3, A_4, A_5, A_6$ and A_7. In the new system (5.5.12), A_5 is used to eliminate a_4 from A_6 and A_7 by the operation:

$(A_6 + 5A_5) \rightarrow A_6.$

$(A_7 - 20A_5) \rightarrow A_7.$

Resulting in the system:

$$\begin{array}{ll} A_1: & -b + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 0 \\ A_2: & -3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6 = 1 \\ A_3: & -2a_2 - 2a_3 + 4a_4 + 10a_5 + 18a_6 = 3 \\ A_4: & -2a_3 + 12a_4 + 40a_5 + 90a_6 = 6 \\ A_5: & 24a_4 + 120a_5 + 360a_6 = 6 \\ A_6: & 120a_5 + 720a_6 = -6 \\ A_7: & -360a_5 - 1440a_6 = 30, \end{array} \tag{5.5.13}$$

where the new equations (5.5.13) are, for simplicity, again labelled $A_1, A_2, A_3, A_4, A_5, A_6$ and A_7. In the new system (5.5.13), A_6 is used to eliminate a_5 from A_7 by the operation:

$(A_7 + 3A_6) \rightarrow A_7.$

The resulting system is

$$\begin{array}{ll} A_1: & -b + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 0 \\ A_2: & -3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6 = 1 \\ A_3: & -2a_2 - 2a_3 + 4a_4 + 10a_5 + 18a_6 = 3 \\ A_4: & -2a_3 + 12a_4 + 40a_5 + 90a_6 = 6 \\ A_5: & 24a_4 + 120a_5 + 360a_6 = 6 \\ A_6: & 120a_5 + 720a_6 = -6 \\ A_7: & 720a_6 = 12. \end{array} \tag{5.5.13}$$

The system of equations (5.5.13) is now in reduced form and can easily be solved for the unknown by a backward-substitution process:

$$a_6 = \frac{1}{60}, a_5 = \frac{-9}{60}, a_4 = \frac{45}{60}, a_3 = \frac{-45}{60}, a_2 = \frac{9}{60}, a_1 = \frac{-1}{60} \, and \; b = 0.$$

It can be easily be verified that these values also satisfy the equations in (5.5.8). Substituting solutions into equations (5.5.7) give the six-order finite difference approximation of $\frac{\partial u}{\partial x}$ as follows:

$$\left(\frac{\partial u}{\partial x}\right)_i = \frac{-u_{i-3} + 9u_{i-2} - 45u_{i-1} + 45u_{i+1} - 9u_{i+1} + u_{i+3}}{60h}, \tag{5.5.14}$$

and truncation error

$$\tau_c' = O(h^6) = \frac{h^6}{5040}(-2187a_1 - 128a_2 - a_3 + a_4 + 128a_5 + 2187a_6)\left(\frac{\partial^7 u}{\partial x^7}\right)_i,$$

$$\tau_c' = \frac{h^6}{5040}\left(-2187\left(\frac{-1}{60}\right) - 128\left(\frac{9}{60}\right) - \left(\frac{-45}{60}\right) + \left(\frac{45}{60}\right) + 128\left(\frac{-9}{60}\right) + 2187\left(\frac{1}{60}\right)\right)\left(\frac{\partial^7 u}{\partial x^7}\right)_i,$$

$$\tau_c' = \frac{h^6}{140}\left(\frac{\partial^7 u}{\partial x^7}\right)_i = 0.007143h^6\left(\frac{\partial^7 u}{\partial x^7}\right)_i.$$

We can also perform similar approach to obtain six order finite difference approximation for second derivative of diffusion term. To achieve the six order finite difference approximation for second derivative, again start by substituting the value of $u_{i-3}, u_{i-2}, u_{i-1}, u_{i+1},\ u_{i+2}$ and u_{i+3} interms of u_i and multiplying them by a_1, a_2, a_3, a_4, a_5 and a_6 respectively and collocated the summation of these equation we obtain the following abridged expression:

$$\begin{aligned} a_1u_{i-3} + a_2u_{i-2} + a_3u_{i-1} + a_4u_{i+1} + a_5u_{i+2} + a_6u_{i+3} &= (a_1 + a_2 + a_3 + a_4 + a_5 + a_6)u_i \\ &+ h^2\left\{\frac{1}{h}(-3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6)\left(\frac{\partial u}{\partial x}\right)_i\right. \\ &+ \frac{1}{2}(9a_1 + 4a_2 + a_3 + a_4 + 4a_5 + 9a_6)\left(\frac{\partial^2 u}{\partial x^2}\right)_i \\ &+ \frac{h}{6}(-27a_1 - 8a_2 - a_3 + a_4 + 8a_5 + 27a_6)\left(\frac{\partial^3 u}{\partial x^3}\right)_i \\ &+ \frac{h^2}{24}(81a_1 + 16a_2 + a_3 + a_4 + 16a_5 + 81a_6)\left(\frac{\partial^4 u}{\partial x^4}\right)_i \\ &+ \frac{h^3}{120}(-243a_1 - 32a_2 - a_3 + a_4 + 32a_5 + 243a_6)\left(\frac{\partial^5 u}{\partial x^5}\right)_i \\ &+ \frac{h^4}{720}(729a_1 + 64a_2 + a_3 + a_4 + 64a_5 + 729a_6)\left(\frac{\partial^6 u}{\partial x^6}\right)_i \\ &+ \frac{h^5}{5040}(-2187a_1 - 128a_2 - a_3 + a_4 + 128a_5 \\ &\left. + 2187a_6)\left(\frac{\partial^7 u}{\partial x^7}\right)_i\right\}. \end{aligned} \tag{5.5.15}$$

At this stage; in order to satisfy the accurate sixth order in equation (5.5.15), we need to find the values of the coefficients a_1, a_2, a_3, a_4, a_5 and a_6 such that the coefficient of $\left(\frac{\partial^2 u}{\partial x^2}\right)_i$ must equal 1 and the coefficients of $\left(\frac{\partial u}{\partial x}\right)_i, \left(\frac{\partial^3 u}{\partial x^3}\right)_i, \left(\frac{\partial^4 u}{\partial x^4}\right)_i, \left(\frac{\partial^5 u}{\partial x^5}\right)_i$ and $\left(\frac{\partial^6 u}{\partial x^6}\right)_i$ must be zeroes and the coefficient of $\left(\frac{\partial^7 u}{\partial x^7}\right)_i$ should be not zero. We can call the coefficient of u_i as b such that $c = a_1 + a_2 + a_3 + a_4 + a_5 + a_6$, thus we obtain the following linear equations to be solved:

$$\begin{aligned}
&A_1: && -c + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 0\\
&A_2: && -3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6 = 0\\
&A_3: && 9a_1 + 4a_2 + a_3 + a_4 + 4a_5 + 9a_6 = 2\\
&A_4: && -27a_1 - 8a_2 - a_3 + a_4 + 8a_5 + 27a_6 = 0\\
&A_5: && 81a_1 + 16a_2 + a_3 + a_4 + 16a_5 + 81a_6 = 0\\
&A_6: && -243a_1 - 32a_2 - a_3 + a_4 + 32a_5 + 243a_6 = 0\\
&A_7: && 729a_1 + 64a_2 + a_3 + a_4 + 64a_5 + 729a_6 = 0\,.
\end{aligned} \tag{5.5.16}$$

Yielding, the truncation error

$$\tau_c'' = \frac{h^5}{5040}(-2187a_1 - 128a_2 - a_3 + a_4 + 128a_5 + 2187a_6)\left(\frac{\partial^7 u}{\partial x^7}\right)_i + \frac{h^6}{40320}(6561a_1 + 256a_2 + a_3 + a_4 + 256a_5 + 6561a_6)\left(\frac{\partial^8 u}{\partial x^8}\right)_i.$$

We follow the same procedure as above to solve the system of linear equations (5.5.16), reduced form and can easily be solved for the unknown by a backward substitution process:

$$\begin{aligned}
&A_1: && -b + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 0\\
&A_2: && -3a_1 - 2a_2 - a_3 + a_4 + 2a_5 + 3a_6 = 0\\
&A_3: && -2a_2 - 2a_3 + 4a_4 + 10a_5 + 18a_6 = 2\\
&A_4: && -2a_3 + 12a_4 + 40a_5 + 90a_6 = 10\\
&A_5: && 24a_4 + 120a_5 + 360a_6 = 22\\
&A_6: && 120a_5 + 720a_6 = -10\\
&A_7: && 720a_6 = 8\,.
\end{aligned} \tag{5.5.17}$$

Yielding,

$$a_6 = \frac{1}{90}, a_5 = \frac{-3}{20}, a_4 = \frac{3}{2}, a_3 = \frac{3}{2}, a_2 = \frac{-3}{20}, a_1 = \frac{1}{90}\ and\ c = \frac{245}{90}.$$

When these values are substituted in equation (5.5.15) gives:

The sixth-order finite difference approximation of $\frac{\partial^2 u}{\partial x^2}$ as

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_i = \frac{2u_{i-3} - 27u_{i-2} + 270u_{i-1} - 490u_i + 270u_{i+1} - 27u_{i+2} + 2u_{i+3}}{180h^2}. \tag{5.5.18}$$

With the truncation error

$$\tau_c'' = \frac{h^5}{5040}(-2187a_1 - 128a_2 - a_3 + a_4 + 128a_5 + 2187a_6)\left(\frac{\partial^7 u}{\partial x^7}\right)_i$$
$$+ \frac{h^6}{40320}(6561a_1 + 256a_2 + a_3 + a_4 + 256a_5 + 6561a_6)\left(\frac{\partial^8 u}{\partial x^8}\right)_i,$$
$$\tau_c'' = \frac{h^6}{560}\left(\frac{\partial^8 u}{\partial x^8}\right)_i = 0.0017857h^6\left(\frac{\partial^8 u}{\partial x^8}\right)_i.$$

For illustrative purpose, in the above calculation, we presented the derivation of six order central finite difference approximation for first and second derivatives. The similar procedure can be carried out for the other approximation of any order in a similar fashion. A numerical approximation to ADE with a source term of equation (5.1.1) can be obtained by replacing the derivatives by the following approximations

$$\left(\frac{\partial u}{\partial t}\right)_i^j + \alpha\left(\frac{\partial u}{\partial x}\right)_i^j = \beta\left(\frac{\partial^2 u}{\partial x^2}\right)_i^j + f_i^j. \tag{5.5.19}$$

Using forward difference approximation for time discretization and substituting the approximations obtained in equations (5.5.14) and (5.5.18) into equation (5.5.19) gives:

$$\frac{u_i^{j+1} - u_i^j}{k} + \alpha\frac{-u_{i-3}^j + 9u_{i-2}^j - 45u_{i-1}^j + 45u_{i+1}^j - 9u_{i+2}^j + u_{i+3}^j}{60h}$$
$$= \beta\frac{2u_{i-3}^j - 27u_{i-2}^j + 270u_{i-1}^j - 490u_i^j + 270u_{i+1}^j - 27u_{i+2}^j + 2u_{i+3}^j}{180h^2}$$
$$+ f_i^j + O(k, h^6).$$

Solving for the new value and ignoring the error terms yields

$$u_i^{j+1} = u_i^j - \frac{\alpha k}{60h}\left(-u_{i-3}^j + 9u_{i-2}^j - 45u_{i-1}^j + 45u_{i+1}^j - 9u_{i+2}^j + u_{i+3}^j\right)$$
$$+ \frac{\beta k}{180h^2}\left(2u_{i-3}^j - 27u_{i-2}^j + 270u_{i-1}^j - 490u_i^j + 270u_{i+1}^j - 27u_{i+2}^j\right.$$
$$\left. + 2u_{i+3}^j\right) + kf_i^j.$$

Letting $Pe = \frac{\alpha h}{\beta}$ and $Cr = \frac{\alpha k}{h}$, then it becomes

$$u_i^{j+1} = u_i^j - \frac{Cr}{60}\left(-u_{i-3}^j + 9u_{i-2}^j - 45u_{i-1}^j + 45u_{i+1}^j - 9u_{i+2}^j + u_{i+3}^j\right) + \frac{Cr}{180Pe}\left(2u_{i-3}^j - 27u_{i-2}^j + 270u_{i-1}^j - 490u_i^j + 270u_{i+1}^j - 27u_{i+2}^j + 2u_{i+3}^j\right) + kf_i^j.$$

General difference approximation becomes in compact form as:

$$u_i^{j+1} = \left(\frac{Cr}{60} + \frac{2Cr}{180Pe}\right)u_{i-3}^j + \left(\frac{-9Cr}{60} - \frac{27Cr}{180Pe}\right)u_{i-2}^j + \left(\frac{-45Cr}{60} + \frac{270Cr}{180Pe}\right)u_{i-1}^j + \left(1 - \frac{490Cr}{180Pe}\right)u_i^j + \left(\frac{45Cr}{60} + \frac{270Cr}{180Pe}\right)u_{i+1}^j + \left(\frac{9Cr}{60} - \frac{27Cr}{180Pe}\right)u_{i+2}^j + \left(\frac{-Cr}{60} + \frac{2Cr}{180Pe}\right)u_{i+3}^j + kf_i^j. \quad (5.5.24)$$

5.6. Numerical Experiments

Indeed, it is a very arduous task to perform the computation of the presented schemes analytically. So that MATLAB software has been used to make different numerical illustrations with the help of tables and graphs for each considered scheme. As it has been introduced in section (1.2), the numerical solution of ADE with source term has not been done analytically. So that throughout this section, the analysis has been drawn by writing new MATLAB code for each developed schemes of FDM. In order to check the accuracy of the numerical schemes that are developed for solving equation (5.1.1), four test problems have been considered and comparison made has been between the proposed schemes with an exact solution. Problems 1 and 2 are solved by FTCS and BTCS schemes and problems 3 and 4 are solved by sixth order finite difference scheme. The problems of ADE are chosen such that their exact solutions are known. For Problems 1 and 2, maximum absolute errors are computed by the following formulas

$$\text{Absolute Error} = L_\infty = \max_{1 \le i \le M}\left|u_{ij} - \hat{u}_{ij}\right|,$$

where u_{ij} and $\hat{u}_{ij}$ denote the exact and numerical solution of the problem, respectively.

Problem 1 (with Dirichlet boundary conditions) (Dehghan, 2004): A problem for which the exact solution is known is considered to test the accuracy of the proposed methods described for solving the ADE. In this example, we have considered the advection-diffusion equation $(5.1.1) - (5.1.4)$ with initial condition,

$$u(x,0) = h(x) = \exp\left[-\frac{(x+0.5)^2}{0.00125}\right],$$

and boundary conditions

$$u(0,t) = g_1(t) = \frac{0.025}{\sqrt{0.000625+0.02t}}\exp\left[-\frac{(0.5-t)^2}{0.00125+0.04t}\right],$$

$$u(1,t) = g_2(t) = \frac{0.025}{\sqrt{0.000625+0.02t}}\exp\left[-\frac{(1.5-t)^2}{0.00125+0.04t}\right].$$

In addition, we have considered $\alpha = 1, \beta = 0.01$ and the right side function $f(x,t) = 0$. The analytical solution of the given problem in a region bounded by $0 \le x \le 1$ and $0 \le t \le 1$ is given by

$$u(x,t) = \frac{0.025}{\sqrt{0.000625+0.02t}}\exp\left[-\frac{(x+0.5-t)^2}{0.00125+0.04t}\right].$$

In this example the values of various parameters are chosen $\beta = 0.01\, m^2/s, \alpha = 1\, m/s, \Delta x = 0.01\, m, \Delta t = 0.0001\, s$. The modelled problem has been solved for both FTCS and BTCS schemes and the results are shown in Table 1 and Figures 5 and 6 below. Table 1 shows the maximum absolute errors (measured in L_∞ norm). Figures 5 and 6 compare the exact and numerical solutions for different values of Δx and Δt. As it can be seen from Table 1, the maximum absolute errors for the FTCS scheme are smaller than that of BTCS scheme which signify that FTCS scheme performs a little bit better than BTCS scheme. Moreover, it can be observed from the recorded results that the numerical solutions for both schemes improve as the value of Δx decreases.

Table 1: Maximum absolute error of problem 1 for FTCS and BTCS schemes at different Δx.

Δt	Δx	FTCS	BTCS
0.0001	0.02	0.003062	0.003232
0.0001	0.01	7.699605×10^{-4}	9.807912×10^{-4}
0.0001	0.005	2.833035×10^{-4}	4.643343×10^{-4}
0.0001	0.0025	2.249207×10^{-4}	4.088092×10^{-4}

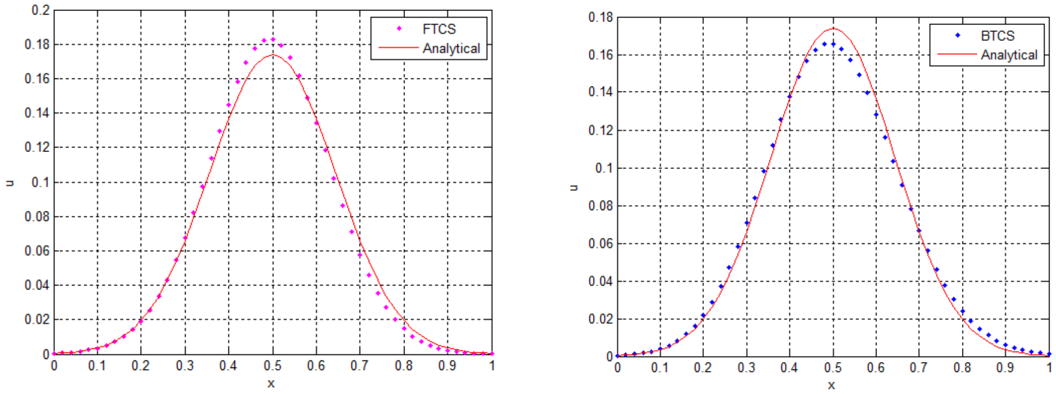


Figure 5: Comparison of analytical and numerical solutions of FTCS and BTCS schemes for problem 1 with $\Delta x = 0.02\ m, \Delta t = 0.004\ s$.

Comparison of the exact solution and the numerical solution obtained with both FTCS and BTCS schemes for $\Delta x = 0.02\ m, \Delta t = 0.004\ s$ are depicted in Figure 5. Figure 5 showed that both schemes give very good results and closer to the exact solution. On the other hand, Figure 6 shows the simulation results for problem 1 with $\Delta x = 0.01m, \Delta t = 0.0001\ s$. It is also obvious that a decrease in both Δx and Δt is accompanied by a decrease in the maximum absolute errors. In addition, the numerical solutions became very close to the exact solution which is indicated by the overlap between the numerical and exact solutions for both schemes.

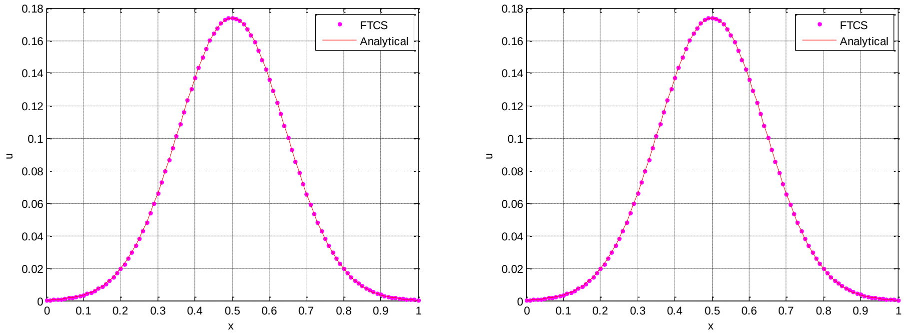


Figure 6: Comparison of analytical and numerical solutions of FTCS and BTCS schemes for problem 1 with $\Delta x = 0.01m, \Delta t = 0.0001\ s$.

Problem 2 (with Neumann boundary conditions) (Mittal and Jain 2012): In this problem, we have considered the following equation:

$$\frac{\partial u(x,t)}{\partial t} + \alpha \frac{\partial u(x,t)}{\partial x} = \beta \frac{\partial^2 (x,t)}{\partial x^2} + f(x,t), \quad 0 < x < 1, 0 < t \leq 1,$$

with $\alpha = 0.1, \beta = 0.02$, initial condition

$$h(x) = \exp(Dx),$$

and boundary conditions given by

$$\frac{\partial u}{\partial x}(0,t) = g_1(t) = Dexp(-0.09t), \ \frac{\partial u}{\partial x}(1,t) = g_2(t) = Dexp(D - 0.09t).$$

The right-hand side function is $f(x,t) = 0$ and the exact solution is given by $u(x,t) = exp(Dx - 0.09t)$.

In this problem we have considered the following parameters $\alpha = 0.1, \beta = 0.02, D = 1.1771243446770,\ \Delta x = 0.02\ m,$ and $\Delta t = 0.004\ s$. The numerical results are presented in Table 2 and Figures 7 and 8. Table 2 shows the maximum absolute errors measured in L_∞ norm. The figures depict the comparison between the exact and numerical solutions for different values of Δx and Δt.

Table 2: The Maximum absolute error of problem 2 for FTCS and BTCS schemes at different Δx.

Δt	Δx	FTCS	BTCS
0.0001	0.02	6.494970×10^{-5}	6.278485×10^{-5}
0.0001	0.01	1.707386×10^{-5}	1.491189×10^{-5}
0.0001	0.005	5.080476×10^{-6}	2.919231×10^{-6}
0.0001	0.0025	2.080605×10^{-6}	6.181249×10^{-7}

From Table 2 it can be seen that the maximum absolute error due to both schemes decrease with Δx. It is also worth noticing that BTCS scheme performs a little bit better than the FTCS scheme. From the recorded numerical results one can infer that the numerical solutions become highly accurate and approach the exact solution as the value of Δx tends to zero.

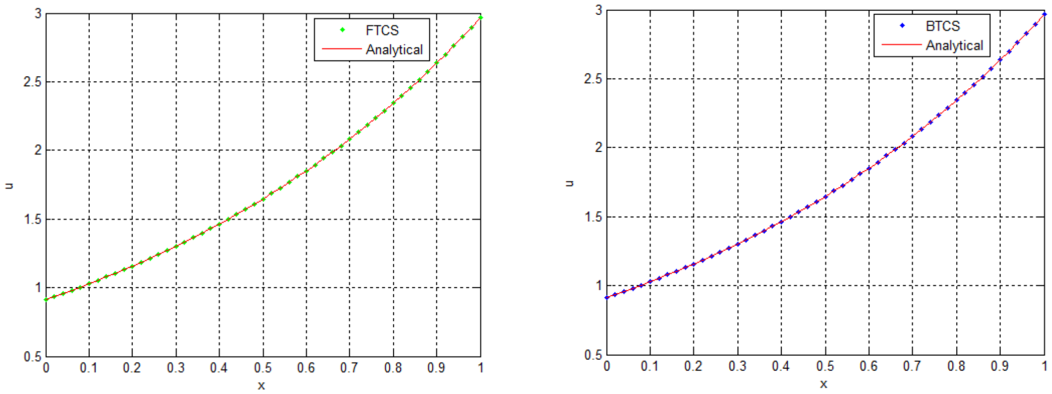


Figure 7: Comparison of analytical and numerical solutions of FTCS and BTCS schemes for problem 2 with $\Delta x = 0.02\ m, \Delta t = 0.004\ s$.

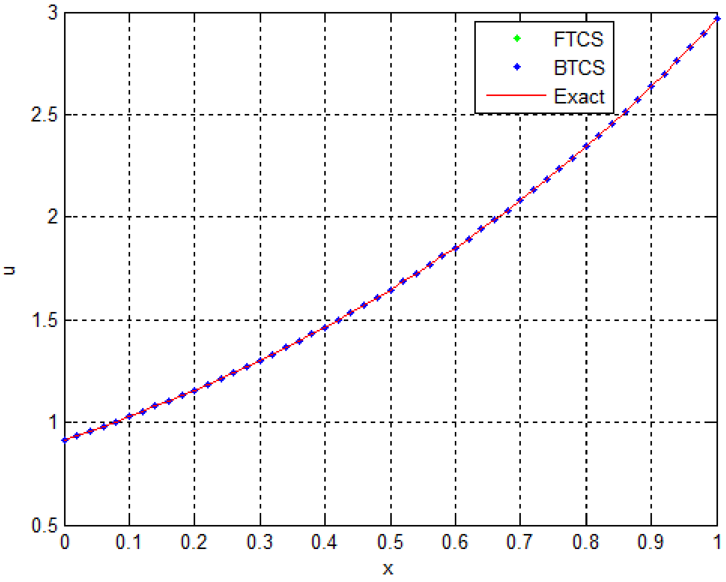


Figure 8: Comparison of exact with numerical solutions of FTCS and BTCS schemes for Problem 2 with $\Delta x = 0.02\ m, \Delta t = 0.004\ s$.

In general, from the numerical error analysis presented in the above Tables and Figures one can observe that the proposed schemes provide highly accurate results.

Rates of convergence of the schemes

Theoretically, FTCS and BTCS schemes are known to have a second order rate of convergence in space. In order to confirm this numerically, experiments have been conducted by varying the step sizes Δt and Δx. The maximum absolute error and the convergence rates are computed and shown for a selected nodal point $t_i = 1$. In the experiments, the grid sizes are refined for the sake of investigating the impact on the accuracy of the solutions and rate of convergence of the schemes.

The step size Δt is set extremely small to reduce the effect from the temporal dimension, so that the discretization error in time $\Delta t_n = \frac{(\Delta x_n)^2}{2}, n = 1,2,3,4$ is negligible, to have a proper resolution expecting that the resulting error of the solution will tend to zero. The order of accuracy, denoted by q, determines convergence rate between errors and it is obtained by $q_n = \frac{\log(e_n/e_{n+1})}{\log(\Delta x_n/\Delta x_{n+1})} = \frac{\log(e_n/e_{n+1})}{\log(2)}$, where e_n and e_{n+1} are errors with grid sizes Δx_n and $\Delta x_n/2$ respectively.

Table 3: Convergence rate of FTCS and BTCS schemes for solving problem 1.

Δx_n	Max. error (e_n) FTCS	Rate (q_n) FTCS	Max. error (e_n) BTCS	Rate (q_n) BTCS
0.02	0.00303995		0.00337843	
0.01	7.68951432×10^{-4}	1.98308270	8.77181757×10^{-4}	1.94539564
0.005	1.92573740×10^{-4}	1.99748149	2.28429883×10^{-4}	1.94112441
0.0025	4.81613603×10^{-5}	1.99946294	6.12743643×10^{-5}	1.89839588

Table 4: Convergence rate of FTCS and BTCS schemes for solving problem 2.

Δx_n	Max. error (e_n) FTCS	Rate (q_n) FTCS	Max. error (e_n) BTCS	Rate (q_n) BTCS
0.02	6.60321582×10^{-5}		6.17024517×10^{-5}	
0.01	1.65333636×10^{-5}	1.99778854	1.54523823×10^{-5}	1.99749854
0.005	4.13492412×10^{-6}	1.99944750	3.86476425×10^{-6}	1.99937696
0.0025	1.03384077×10^{-6}	1.99984685	9.66307288×10^{-7}	1.99982647

Tables 3 and 4 above present the maximum absolute errors and convergence rates for solving problem 1 and 2, respectively. From the above Tables, it can be observed that, both schemes are second order accurate in the spatial dimension as expected, since the obtained convergence rates are approximately 2.

Problem 3 (ADE with source term and Dirichlet Boundary conditions): In this example, we consider the ADE with source term given in equation $(5.1.1) - (5.1.4)$ in $[0,10]$ with $\alpha = -1, \beta = 1$ with boundary conditions:

$$u(0,t) = g_1(t) = 0, \qquad u(1,t) = g_2(t) = 0.$$

The right hand side function is $f(x,t) = [\sin(\pi x) - \pi t cos(\pi t)] \exp(-\pi^2 t)$. The exact solution of the problem is given by $u(x,t) = t sin(\pi x) \exp(-\pi^2 t)$ and the initial condition can be extracted from the exact solution which is given by

$$u(x,0) = h(x) = 0.$$

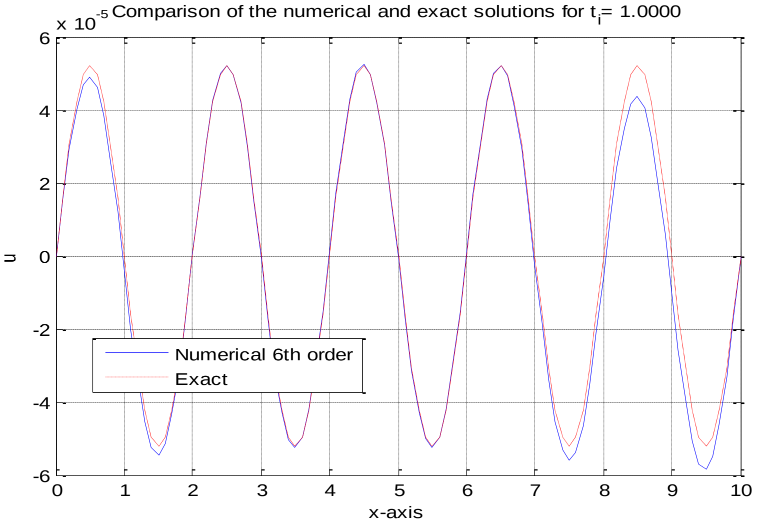


Figure 9: Comparison of analytical solution and numerical solution of sixth order FDS for problem 3 with $\Delta x = 0.1m, \Delta t = 0.001s$.

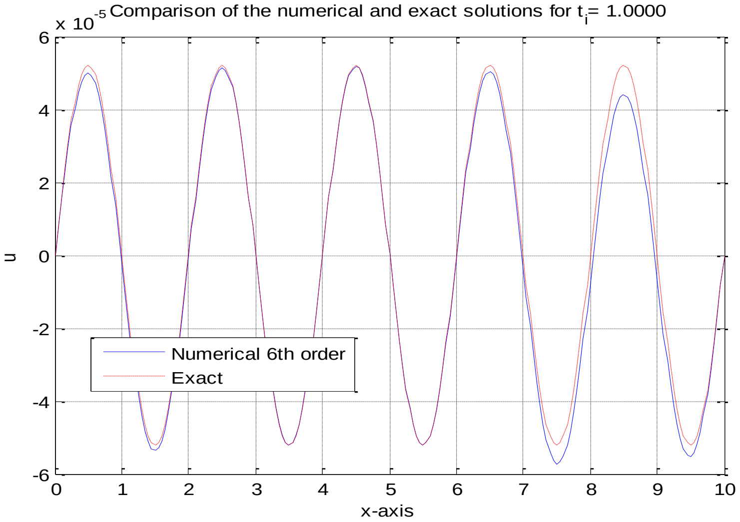


Figure 10: Comparison of analytical solution and numerical solution of sixth order FDS for problem 3 with $\Delta x = 0.05m, \Delta t = 0.001s$.

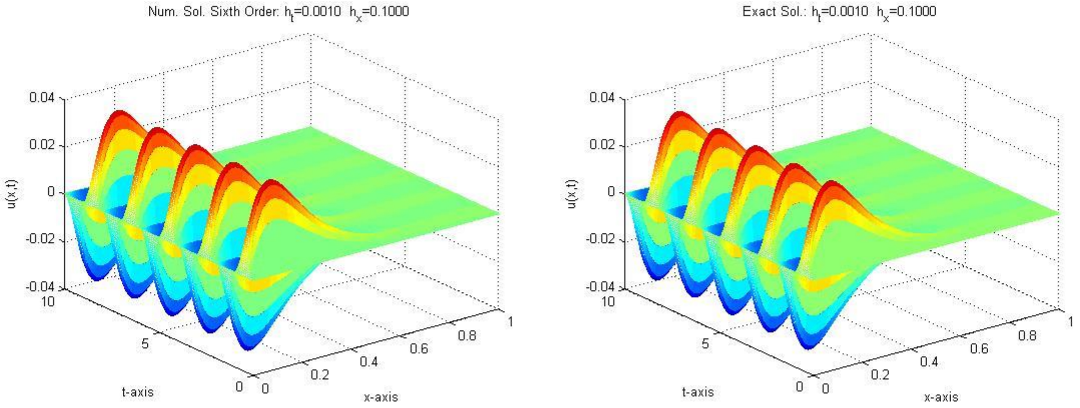


Figure 11: Space-time graph of numerical solution of sixth order FDS and exact solution for problem 3.

Problem 4 (ADE with source term and Dirichlet Boundary conditions): In this example, the considered ADE is the one which is given in equation $(5.1.1)-(5.1.4)$ in $[0,10]$ with $\alpha=\beta=1$ with the boundary conditions:

$g_1(t)=0, g_2(t)=\exp(-t), t>0.$

The right hand side functions $f(x,t)=(x^2+2x-2)\exp(-t)$. The exact solution of the problem is given by $u(x,t)=x^2\exp(-t)$ and the initial condition was obtained from the exact solution

$u(x,0)=h(x)=x^2.$

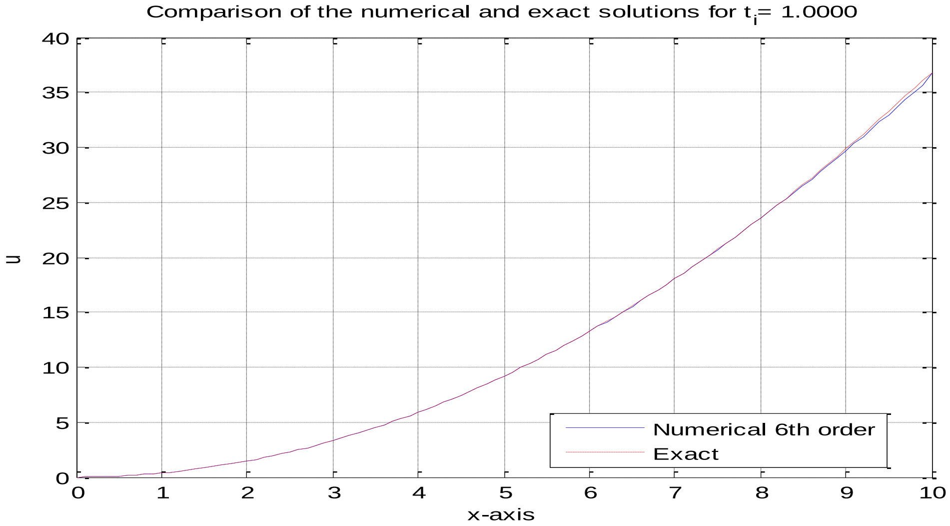


Figure 12: Comparison of analytical solution and numerical solution of sixth order FDS for problem 4 with $\Delta x=0.1m, \Delta t=0.001s$.

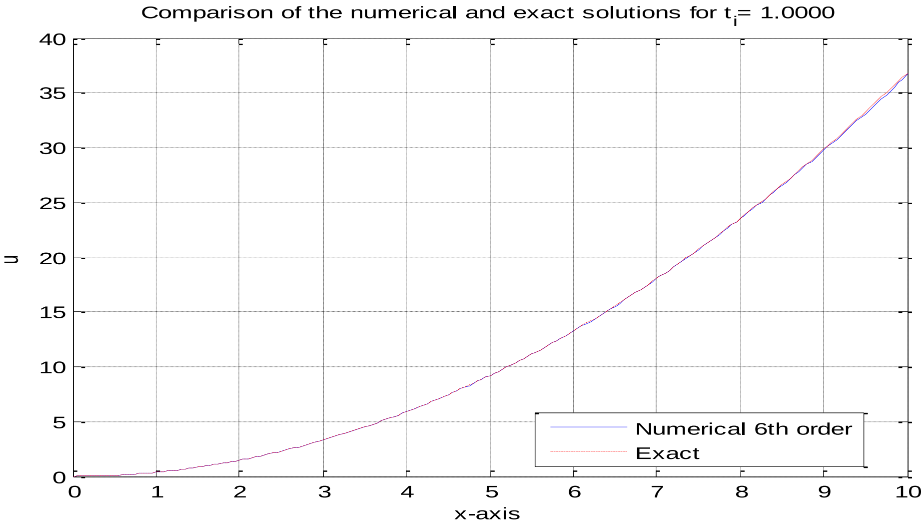
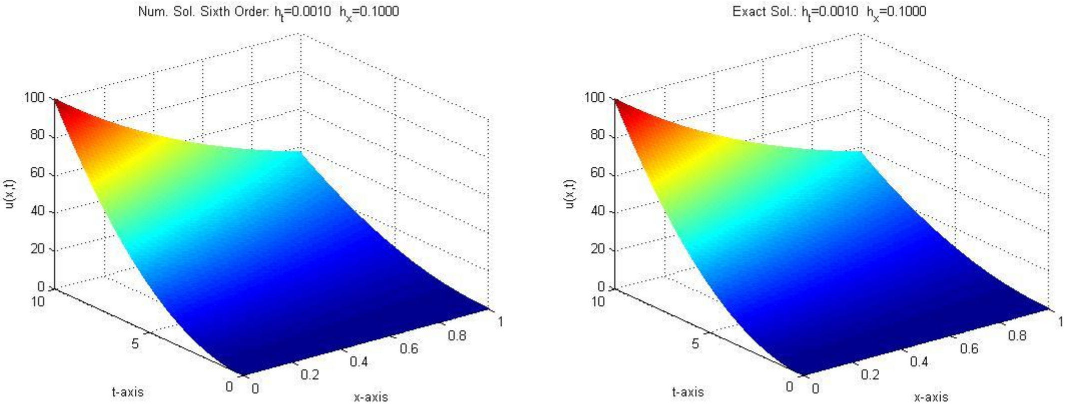

Comparison of the numerical and exact solutions for t_i= 1.0000

Figure 13: Comparison of analytical solution and numerical solution of sixth order FDS for problem 4 with $\Delta x = 0.05m, \Delta t = 0.001s$.


Num. Sol. Sixth Order: h_t=0.0010 h_x=0.1000

Exact Sol.: h_t=0.0010 h_x=0.1000

Figure 14: Space-time graph of numerical solution of sixth order FDS and exact solution for problem 4.

Figure 9, Figure 10, Figure 12 and Figure 13 present the numerical results at a selected nodal point $t_i = 1$. It can be observed from the numerical results depicted in these figures that the accuracy and efficiency of the proposed sixth order finite difference scheme is very high and the numerical solutions coincide with the exact solutions.

Figure 11 and Figure 14 illustrate numerical solution of sixth order FDS and exact solution for whole domain (3D version of the numerical results). It is observed that, the numerical solution of sixth order and exact solution are very close to each other to the extent that the two cannot be distinguished. An excellent agreement between the numerical and exact solutions indicates that the sixth order FDS is very promising and could be used for solving related problems.

6. SUMMARY, CONCLUSION AND RECOMMENDATIONS

6.1. Summary

This Thesis is devoted to developing new numerical schemes for the solution of one-dimensional ADEs with and without a source term which incorporate Dirichlet and Neumann boundary conditions. The schemes are derived based on weighted finite difference and higher order finite difference approximations. First, explicit FTCS and implicit BTCS schemes were developed using weighted finite difference approximation. Discretization of the ADEs resulted in tri-diagonal system of linear algebraic equations, which are solved by Thomas algorithm. Using Lax-Equivalence Theorem, the convergence of the schemes was analysed. The stability of the aforementioned schemes were analysed by using Von-Neumann stability analysis and shown that BTCS scheme is unconditionally stable. Whereas the FTCS scheme is conditionally stable, i.e., stable when $\frac{cr}{pe} \leq \frac{1}{2}$, and this relation restricts, the values of the increments of time and space.

The stabilities that have been made in the present work had also been verified by the Gershgorin’s Theorem which ensures the obtained stabilities are accurate. Both of the above schemes worked well according to the stability criteria and produced reasonable results when tested on ADEs without a source term. The performance of the schemes for the considered problems was measured by calculating the L_∞ error and rate of convergence.

Secondly, a new sixth order finite difference scheme was derived by using Taylor series expansion and applied to solve one-dimensional ADEs with a source term. In developing this scheme, a sixth order central difference in space and a first order forward difference in time were employed.

6.2. Conclusion

In order to investigate the performance of the proposed schemes four test problems which have exact solutions were considered. FTCS and BTCS schemes were tested on ADEs without source terms. As a result, it is concluded that the FTCS scheme is slightly better than BTCS on ADE with Dirichlet boundary conditions. On the other hand, BTCS scheme is a little bit better than FTCS on ADE with Neumann boundary conditions. The accuracy and stability of both schemes have been investigated and it is concluded that the numerical results coincide with the analytical ones. Most importantly the analyses presented in this Thesis show that, both schemes are second order accurate in the spatial dimension. The developed sixth order scheme was also tested on two ADEs with a source term. According to the numerical results, we came to the conclusion that the numerical solution of the said scheme is in excellent agreement with the exact solution.

6.3. Recommendation

In this Thesis new numerical schemes for solving ADEs with constant coefficients have been developed. For future work we recommend the following directions:

- Modify the present schemes for solving ADEs with general boundary conditions
- Change the parameters in the developed weighted finite difference schemes to obtain new schemes, for instance Crank-Nicolson
- Use higher order time discretization technique in the developed sixth order scheme
- Develop new schemes for solving ADEs with variable coefficients

7. REFERENCES

Agusto, F.B. and Bambigbola, O.M. 2007. Numerical treatment of the mathematical models for water pollution. *Journal of Mathematics and Statistics*, 3(4): 172-180.

Ahmed, S.G. 2012. A numerical algorithm for solving advection-diffusion equation with constant and variable coefficients. *Journal of Open Numerical Methods*, 4: 1-7.

Ahsan, M. 2012. Numerical solution of the advection-diffusion equation using Laplace transform finite analytical method. *International journal of river basin management*, 10 (2): 177-188.

Appadu, A.R. 2013. Numerical solution of the 1D advection-diffusion equation using standard and nonstandard finite difference schemes. *Journal of Applied Mathematics*.

Azad, T.M.A.K. and Andallah, L.S. 2016. Stability analysis of finite difference schemes for an advection diffusion equation. *Bangladesh Journal of Scientific Research*, 29(2): 143-151.

Bahar, E. and Gurarslan, G. 2017. Numerical Solution of Advection-Diffusion Equation Using Operator Splitting Method. *International Journal of Engineering and Applied Sciences*, 9 (4): 76-88.

Benedini, M. 2011. Water quality models for rivers and streams, state of the art and future perspectives. *European water*, 34 (1): 27-40.

Bulut, H., Akturk, T. and Ucar, Y. 2013. The solution of advection diffusion equation by the finite elements method. *International Journal of Basic and Applied Sciences IJBAS-IJENS*, 13 (1): 267-282.

Causon, D.M. and Mingham, C.G. 2010. Introductory finite difference methods for PDEs.

Fernandes, C. and Karney, B. 2001. Numerical solution of the Advection-Dispersion-Reaction Equation under transient hydraulic conditions. *In proceedings of the congress-international association for hydraulic research*, 58-63.

Gane, S. 2000. Solution of the advection equation using finite difference schemes and the method of characteristics. PhD thesis, University of Glasgow.

Gerald, C.F. and Wheatley, O.P. 2004. Applied numerical analysis. *Pearson Education India.*

Gerdt, V.P. and Robertz, D. 2010. Consistency of finite difference approximations for linear PDE systems and its algorithmic verification. *In Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation*, 53-59.

Gurarslan, G., Alkaya, D., Karahan, H., Sari, M. and Yasar, M. 2013. Numerical solution of advection-diffusion equation using a sixth-order compact finite difference method. *Hindawi Publishing Coportion Mathematical problems in Engineering*.

Igboekwe, M.U. and Achi, N.J. 2011. Finite difference method of modelling groundwater flow. *Journal of Water Resource and Protection*, 3 (3): 192-198.

Isenberg, J. and Gutfinger, C. 1973. Heat transfer to a draining film. *Int. J. Heat Mass Transf*, 16: 505–512.

Jayanti, S. 2018. Computational Fluid Dynamics for Engineers and Scientists. *Springer*.

Johari, H., Rusli, N. and Yahya, Z. 2018. Finite difference formulation for prediction of water pollution. *IOP conference series: Materials science and Engineering*, 318.

Kaur, H. and Kang, S.M. 2016. Haar Wavelets Based Time Discretization Technique For Solving Nonlinear Partial Differential Equations. *International Journal of Pure and Applied Mathematics*, 108(1): 63-78.

Lee, W.T. 2011. Tridiagonal matrices: Thomas algorithm. *MS6021, Scientific Computation, University of Limerick*.

Li, Z., Qiao, Z. and Tang, T. 2017. Numerical solution of differential equations: introduction to finite difference and finite element methods.

Love, E. and Rider, W.J. 2013. On the convergence of finite difference methods for PDE under temporal refinement. *Computers & Mathematics with Applications*, 66(1):33-40.

Marquis, D., De Moor, H., Porter, K. and Nathanson, M. 2016. Gershgorin's Circle Theorem for Estimating the Eigenvalues of a Matrix with Known Error Bounds.

Mitchell, S.L. and Vynnycky, M. 2009. Finite-difference methods with increased accuracy and correct initialization for one-dimensional Stefan problems. *Applied Mathematics and Computation*, 215(4): 1609-1621.

Mittal, R.C. and Jain, R.K. 2012. Numerical solution of convection-diffusion equation using cubic B-splines collocation methods with Neumann's boundary conditions. *International Journal of Applied Mathematics and Computation*, 4(2):115-127.

Miyaoka, T.Y., Meyer, J.F.D.C.A. and Souza, J. 2017. A General Boundary Condition with Linear Flux for Advection-Diffusion Models. *TEMA (São Carlos)*, 18(2): 253-272.

Mohammadi, A., Manteghian, M. and Mohammadi, A. 2011. Numerical solution of the one-dimensional advection-diffusion equation using simultaneously temporal and spatial weighted parameters. *Australian Journal of Basic and Applied Sciences*, 5(6): 1536-1543.

Mojtabi, A. and Deville, M.O. 2015. One-dimensional linear advection–diffusion equation: Analytical and finite element solutions. *Computers and Fluids*, 107: 189-195.

Neill, S.P. and Hashemi, M.R., 2018. Fundamentals of Ocean Renewable Energy: Generating Electricity from the Sea. *Academic Press*.

Parlange, J. Y. 1980. Water transport in soils. *Annu. Rev. Fluid Mech*, 12(12): 77– 102.

Peiró, J. and Sherwin, S. 2005. Finite difference, finite element and finite volume methods for partial differential equations. *In Handbook of materials modeling*, 2415-2446.

Prieto, F.U., Muñoz, J.J.B. and Corvinos, L.G. 2011. Application of the generalized finite difference method to solve the advection–diffusion equation. *Journal of Computational and Applied Mathematics*, 235 (7): 1849-1855.

Quarteroni, A., Sacco, R. and Saleri, F. 2006. Direct Methods for the Solution of Linear Systems. *In Numerical Mathematics,* 59-124.

Randall, J.L. 1998. Finite Difference Methods for Diffusion Equations. *Draft version for use in the course A Math*, 585 – 586.

Rigal, A. 1979. Stability analysis of explicit finite difference schemes for the Navier–Stokes equations. *International journal for numerical methods in engineering*, 14(4): 617-620.

Runkel, R.L. 1996. Solution of the advection-dispersion equation: continuous load of finite duration. *Journal of Environmental Engineering*, 122 (9): 830-832.

Sadiku, M.N. 2000. Numerical techniques in electromagnetics. *CRC press*.

Sanjaya, F. and Mungkasi, S. 2017. A simple but accurate explicit finite difference method for the advection-diffusion equation. *Journal of Physics: Conference Series*. *IOP Publishing*, 909.

Sari, M., Gürarslan, G., and Zeytinoğlu, A. 2010. High-order finite difference schemes for solving the advection-diffusion equation. *Mathematical and Computational Applications*, 15(3): 449-460.

Sheikholeslami, M., Ellahi, R., Hassan M. and S. Soleimani, S. 2014. A study of natural convection heat transfer in a Nano fluid filled enclosure with elliptic inner cylinder. *Int. J. Numer. Methods Heat Fluid Flow*, 24: 1906–1927.

Siddiqi, S.S. and Arshad, S. 2013. Numerical solution of convection-diffusion integro-differential equations with a weakly singular kernel. *J. Basic Appl. Sci. Res*, 3: 106-120.

Soomro, A.S., Tularam, G.A. and Shaikh, M.M. 2013. A Comparison of numerical methods for solving the unforced Van Der Pol's equation. *Mathematical Theory and Modeling*, 3 (2): 66-78.

Strikwerda, J.C. 2004. Finite difference schemes and partial differential equations. 88

Su, L.D., Jiang, Z.W. and Jiang, T.S. 2015. Numerical Method for One-Dimensional Convection-diffusion Equation Using Radical Basis Functions. *Journal of Physical Mathematics*, 6 (2).

Thomas, J.W. 2013. Numerical partial differential equations: finite difference methods. *Springer Science & Business Media*, 22.

Thomée, V. 2001. From finite differences to finite elements a short history of numerical analysis of partial differential equations. *Journal of Computational and Applied Mathematics*, 128: 1-54.

Thongmoon, M. and McKibbin, R. 2006. A comparison of some numerical methods for the advection-diffusion equation.

Veling, E.J.M. 2012. Radial transport in a porous medium with Dirichlet, Neumann and Robin-type inhomogeneous boundary values and general initial data: Analytical solution and evaluation. *J. Eng. Math*, 75: 173–189.

Wang, H., Zhang, Y., Ma, X., Qiu, J. and Liang, Y. 2016. An efficient implementation of fourth-order compact finite difference scheme for Poisson equation with Dirichlet boundary conditions. *Computers & Mathematics with Applications*, 71(9):1843-1860.

Yang, W.Y., Cao, W., Chung, T.S. and Morris, J. 2005. Applied numerical methods using MATLAB. *John Wiley and Sons.*

Young, T. and Mohlenkamp, M.J. 2012. Introduction to numerical methods and MATLAB programming for engineers. *Department of Mathematics, Ohio University*.