Numerical Solutions and Stability of Some Partial Differential Equations Using Finite Difference Methods

Thesis submitted in partial fulfillment of the requirements

for the award of the degree of

Masters of Science

in

Mathematics and Computing

submitted by

Ramandeep Kaur

Roll No: 301103015

under

the guidance of

Dr. Ram Jiwari

to the

School of Mathematics and Computer Applications,
Thapar University,
Patiala- 147004 (Punjab) INDIA

DEDICATED

TO

GOD, MY TEACHERS AND MY PARENTS

CERTIFICATE

*I hereby certify that the work which is being presented in the thesis entitled "***Numerical Solutions of Some Parabolic Partial Differential Equations Using Finite Difference Method***" which is being submitted for the award of degree of master of Science, School of Mathematics and Computer Applications, Thapar University, Patiala is an authentic record of my own work carried out under the supervision of* ***Dr. Ram Jiwari***

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

Ramandeep kaur

Ramandeep kaur

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

17/6/13

Dr. Ram Jiwari

(Assistant Professor)

SMCA, Thapar University

Patiala

R K Sharma 1.7.13

Countersigned by

Dr. Rajesh Kumar
(Associate Professor & Head)
School of Mathematics & Computer Applications
Thapar University, Patiala

Dr. S.K Mohapatra
Dean of Academic Affairs
Thapar Uniersity, Patiala

ACKNOWLEDGEMENT

It gives me immense pleasure to acknowledge my sincere gratitude to my academic supervisor, Dr. Ram Jiwari, School of Mathematics and Computer Applications, Thapar University, Patiala, for his constant help, encouragement and support throughout the course of this work.

I would like to extend my special thanks to Dr. Rajesh Kumar,Associate Professor and Head, School of Mathematics and Computer Applications, Thapar University, Patiala, for providing help and necessary facilities in the department and directly or indirectly encouraged me to work harder during the whole course.

I am very grateful to my close friend Jasjeet Singh for him kind support.

(Ramandeep Kaur)

Patiala
17, June , 2013

Contents

Chapter 1 is introductory in nature. Besides stating some numerical techniques like Finite Difference methods, Finite Element method, Finite Volume method and methods of weighted residuals it gives an introduction to Finite Difference method and existing literature review.

In chapter 2, we consider the one dimensional heat equation

$$\frac{\partial c}{\partial t} = \frac{\partial^2 c}{\partial x^2} \qquad , 0 \le x \le L \qquad t > 0$$

This problem is one of the well-known second order linear partial differential equation [33, 34]. It shows that heat equation describes irreversible process and makes a distance between the previous and next steps. Such equations arise very often in various applications of science and engineering describing the variation of temperature (or heat distribution) in a given region over some time [34]. It can be expressed as the heat flow in the rod with diffusion c_{xx} along the rod where the coefficient is the thermal diffusivity of the rod and L is the length of the rod. In this model, the flow of the heat in one-dimension that is insulated everywhere except at the two end points.

In this chapter, finite difference method is proposed for the numerical solutions of one dimensional heat equation. Two test examples are considered to test the accuracy and efficiency of the method.

In chapter 3, we consider the one dimensional advection-diffusion equation $\frac{\partial c}{\partial t}+u\frac{\partial c}{\partial x}=D\frac{\partial^2 c}{\partial x^2}$, $0<x<L$, $0<t\leq T$

The mathematical model describing the transport and diffusion processes is the one-dimensional advection-diffusion equation. Mathematical modeling of heat transport, pollutants and suspended matter in groundwater involves the solution of a convection–diffusion equation.

In this chapter, finite difference method is proposed for the numerical solutions of one dimensional advection-diffusion equation. Two test examples are considered to test the accuracy and efficiency of the method. The absolute errors are calculated for the both examples.

Chapter 1

Introduction

1.1 Classification of Second Order Partial Differential Equations

Consider general second order partial differential equation of the form

$$A\frac{\partial^2 U}{\partial x^2} + B\frac{\partial^2 U}{\partial x\partial y} + C\frac{\partial^2 U}{\partial y^2} + D\frac{\partial U}{\partial x} + E\frac{\partial U}{\partial y} + FU = G(x,y) \tag{1.1}$$

The equation (1.1) is classified in the following categories:

(i) Parabolic Equation if $B^2 - 4AC = 0$

(ii) Elliptic Equation if $B^2 - 4AC < 0$

(iii) Hyperbolic Equation if $B^2 - 4AC > 0$

Heat, Laplace and wave equations are the simple example of these equations which re given as follows

(a) Parabolic Equation: The equation $U_t = kU_{xx}$, is 1D diffusion equation and can be used to model the time-dependent temperature distribution along a heated 1D bar.

(b) Elliptic Equation: The equation $U_{xx} + U_{yy} = f(x,y)$ is called Poisson Equation when f(x,y)=0 then, the equation $U_{xx} + U_{yy} = 0$ is called Laplace equation which may be used to model the steady state temperature distribution in a plate or incompressible potential flow.

(c) Hyperbolic Equation: The equation $U_{tt} = c^2 U_{xx}$, where c is constant is wave equation and may be used to model string or 1D supersonic flow.

1.2 Initial and Boundary Conditions

PDEs require proper initial conditions (ICs) and boundary conditions (BCs) in order to define what is known as a well-posed problem. If too many conditions are specified then there will be no solution; if too few conditions are specified the solution will not be unique. If the ICs/BCs are specified in the wrong place or at the wrong time then the solution will not depend smoothly on the ICs/BCs and small errors in the ICs/BCs will bring about large changes in the solution. This is referred to as an ill-posed problem. However, we need to understand the properties of the solution to these simple model PDEs before attempting to solve more complicated PDEs.

1.3 Numerical Solutions of Partial Differential Equations

Partial differential equations (PDEs) 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 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. Currently, there are many numerical techniques available in the literature. Among them, the finite difference (FD), finite element (FE), and finite volume (FV) methods fall under the category of low order methods, whereas spectral and pseudo spectral methods are considered global methods. Sometimes the latter two methods are considered as subsets of the method of weighted.

1.3.1 Finite Element Method

Finite element method (FEM) represents a powerful and general class of techniques for the approximate solution of partial differential equations. The basic idea in the FEM is to find the solution of a complicated problem by replacing it by a simpler one. Since the actual problem is replaced by a simpler one in finding the solution, we will be able to find only an approximate solution rather than the exact solution. This method is mostly used for the accurate solution of complex engineering problems with abundant software available commercially. FEM was first developed in 1956 for the analysis of aircraft structural problems. Thereafter, within a decade, the potentialities of the method for the solution of different types of applied science and engineering problems were recognized. Over the years, the FEM technique has been so well established that today it is considered to be one of the best methods for solving a wide variety of practical problems efficiently. In fact, the method has become one of the active research areas for applied mathematicians. Based on the variational principle, basic procedures of the FEM include: obtaining functional (variational expressions) from corresponding differential equations, dividing interested region into small elements, constructing interpolation model for each element, assembling all elements' contributions to the global system, and finally solving the global-matrix problems. The systematic generality of FEM makes it possible to construct a general-purposed computer program for a wide range of problems. In this method, the region is divided into Subregions (elements), which could be different shapes i.e. triangular, rectangular, curvilinear, ring, or infinite.

Moreover, non uniform unstructured meshes and adaptive meshing procedures can be employed to significantly improve the accuracy and efficiency of FEM programs. Furthermore, FEM scheme can be established not only by the variational method but also by the Galerkin method or the least squares method, so FEM can still be used even though a variation principle does not exit or cannot be identified. Boundary conditions can be easily applied once the mesh generation is done. However, the pre-and post-processes of the computed set up always play an important role for a good FEM program. It has been applied to a number of physical problems, where the governing differential equations are available. In Finite Element method the domain is divided into a finite number of sub domains called elements and nodes are located at predetermined locations around the elements boundary. The elements, along with the nodes, form the mesh, which can be refined to provide any level of accuracy desired.

1.3.2 Finite Volume Method

Finite volume methods (FVMs) form a relatively general class of discretizations for certain types of partial differential equations. These methods start from balance equations over local control volumes, e.g., the conservation of mass in diffusion problems. When these conservation equations are integrated by parts over each control volume, certain terms yield integrals over the boundary of the control volume. For example, mass conservation can be written as a combination of source terms inside the control volume and fluxes across its boundary. Of course the fluxes between neighboring control volumes are coupled. If this natural coupling of boundary fluxes is included in the discretization, then the local conservation laws satisfied by the continuous problem are guaranteed to hold locally also for the discrete problem. This is an important aspect of FVMs that makes them suitable for the numerical treatment of, e.g., problems in fluid dynamics. Another valuable property is that when FVMs are applied to elliptic problems that satisfy a boundary maximum principle, they yield discretizations that satisfy a discrete boundary maximum principle even on fairly general grids. FVMs were proposed originally as a means of generating finite difference methods on general grids.

Today, however, while FVMs can be interpreted as finite difference schemes, their convergence analysis are usually facilitated by the construction of a related finite element method and a study of its convergence properties. The fundamental idea of the finite volume method can be implemented in various ways in the construction of the control volumes, in the localization of the degree

of freedom, and in the discretization of the fluxes through the boundaries of the control volumes. There are two basically classes of FVM. First, in cell-centred methods each control volume that surrounds a grid point has no vertices of the original triangulation lying on its boundary. The second approach, vertex-centred methods, uses vertices of the underlying triangulation as vertices of control volumes.

1.3.3 Method of Weighted Residuals

The methods of weighted residuals are the approximate methods which determine the solution of the differential equation in the form of functions which are closed in some sense to the exact solution. Consider a differential equation

$$l(u) = 0 \tag{1.2}$$

with initial condition, $I(u) = 0$ and boundary condition, $S(u) = 0$. The solution of differential equation $U(x)$ is approximated by a finite series of functions $\phi_k(x)$ as follows:

$$U(x) = U_0(x) + \sum_{k=1}^{N} a_k \phi_k(x) \tag{1.3}$$

where $\phi_k(x)$ are the basis or trial functions, a_k are the coefficients to be determined that satisfy the differential equation, and N are the number of functions. The form of $U_0(x)$ is chosen to satisfy the boundary and the initial conditions exactly. There is another approach in which exact solutions of the differential equation are known and these are added together to satisfy the boundary conditions approximately. It is also possible to formulate a method in which the differential equation and the boundary conditions are satisfied approximately. In general, the approximate solution does not satisfy the partial differential equation exactly, and substituting its value results in a residual, R as

$$R(x, a_1, a_2, ..., a_N) = l(U(x)) \tag{1.4}$$

which in turn is minimized in some sense. For a given N the a_k's are chosen by requiring that an integration of the weighted residual over the domain is zero. Thus

$$(W_k(x), R) = 0 \tag{1.5}$$

By letting $k = 1, 2, ..., N$ system of equations involving only as a_k's is obtained. For unsteady partial differential equation this would be a system of ordinary differential equations, for steady problems a system of algebraic equations obtained. Different choices of $W_k(x)$ give rise to the different methods within the class.

1.3.4 Differential Quadrature Method

The differential quadrature method (DQM) is a higher order numerical technique for solving partial differential equations. In the nineteen century, most of the numerical simulations of engineering problems can be carried out by the low order FD, FE, and FV methods using a large number of grid points. In some practical applications, however, numerical solutions of PDEs are required at only a few specified points in the physical domain. To achieve an acceptable degree of accuracy, low order methods still require the use of a large number of grid points to obtain accurate solutions at these specified points. In seeking an efficient discretization technique to obtain accurate numerical solution using a considerably small number of grid points, Richard Bellman and his associates [4] introduced the method of differential quadrature in the early 1970s. The DQM, akin to the conventional integral quadrature method, approximates the partial derivative of a function at any location by a linear summation of all the function values along a mesh line. The key procedure in the differential quadrature application lies in the determination of the weighting coefficients. Initially, Bellman and his associates proposed two methods to compute the weighting coefficients for the first order derivative. The first method is based on an ill-conditioned algebraic equation system. The second method uses a simple algebraic formulation, but the coordinates of the grid points are fixed by the roots of the shifted Legendre polynomial. In earlier applications of the DQM, Bellman's first method was usually used because it allows the use of an arbitrary grid point distribution. However, since the algebraic equation system of this method is ill-conditioned, the number of the grid points usually used is less than 13. This drawback limits the application of the DQM.

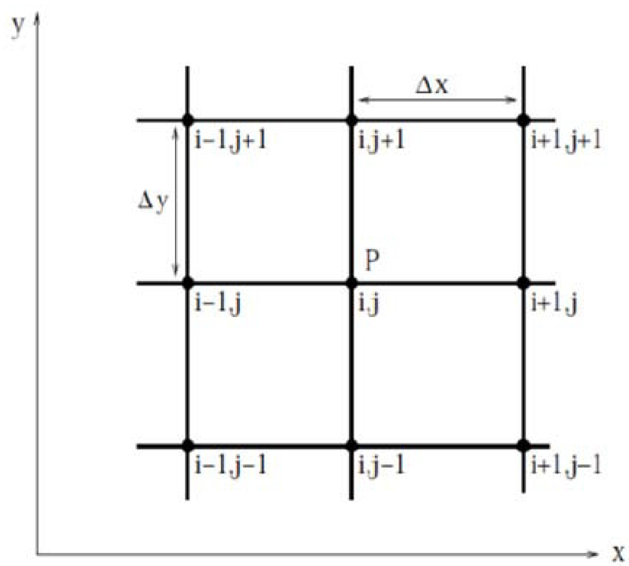
The DQM and its applications were rapidly developed after the late 1980s, thanks to the innovative work in the computation of the weighting coefficients by researchers [5, 7, 8, 20, 28]. As a result, the DQM has emerged as a powerful numerical discretization tool in the past decade. As compared to the conventional low order finite difference and finite element methods, the DQM can obtain very accurate numerical results using a considerably smaller number of grid points and hence requiring relatively little computational effort. Recently, Jiwari et al [35-51] have used differential quadrature methods for some parabolic and hyperbolic partial differential equations.

1.3.5 Finite Difference Methods

As one quickly learns, the differential equations that can be solved by explicit analytic formulae are few and far between. Consequently, the development of accurate numerical approximation schemes is essential for extracting quantitative information as well as achieving a qualitative understanding of the various behaviors of their solutions. Even in cases, such as the heat and wave equations, where explicit solution formulas (either closed form or infinite series) exist, numerical methods can still be profitably employed. Indeed, one can accurately test a proposed numerical algorithm by running it on a known solution, providing yet another motivation to search for explicit solutions. An alternative approach is to use a manufactured solution, in which one starts with a preselected function, which almost certainly is not a solution to the problem at hand. Nevertheless, substituting this function into the differential equation and the relevant initial and/or boundary conditions leads to an inhomogeneous problem of the same character as the original. After running the numerical algorithm on the modified problem, one can test for accuracy by comparing the numerical output with the preselected function. The lessons learned in the design and testing of numerical algorithms for simpler "solved" examples are of inestimable value when confronting more challenging problems. Many of the basic numerical solution schemes for partial differential equations can be fit into two broad themes. The first, to be developed in the present chapter, are the finite difference methods, obtained by replacing the derivatives in the equation by appropriate numerical differentiation formulae. We thus start with a brief discussion of some elementary finite difference formulae used to numerically approximate first and second order derivatives of functions. We then establish and analyze some of the most basic finite difference schemes for the heat equation, first order transport equations, the second order wave equation, and the Laplace and Poisson equations. As we will learn, not all finite difference approximations lead to accurate numerical schemes, and one must deal with the issues of stability and convergence in order to distinguish reliable from worthless methods. In fact, inspired by Fourier analysis, the crucial stability criterion follows from how the numerical scheme handles basic complex exponentials. The second category of numerical solution techniques are the finite element methods (1.3.1).

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:-

(i) Dividing the solution into grids of nodes



- Δx and Δy spacing in positive x and y direction
- Δx and Δy not necessarily uniform

- In some cases, numerical calculations performed on transformed computational plane having uniform spacing in transformed variables but non uniform spacing in physical plane
- Grid points identified by indices i and j in positive x and y direction

(ii) Approximating the given differential equation by finite difference equivalence that relates the solutions to grid points.

(iii) Solving the difference equations subject to the prescribed boundary conditions and/or initial conditions.

Our main aim of the present study is to discuss some finite difference schemes for the numerical solutions of some parabolic equations. So, we are giving some finite difference schemes for parabolic equations in the following next chapter.

1.4 Organization of Thesis

In this thesis an attempt has been made to solve some parabolic partial differential equations by using finite difference methods. The chapter wise summary of thesis is as follows.

In chapter 2, we consider the one dimensional heat equation

$$\frac{\partial c}{\partial t} = \frac{\partial^2 c}{\partial x^2} \qquad , 0 \le x \le L \qquad t > 0$$

This problem is one of the well-known second order linear partial differential equation [33,34]. It shows that heat equation describes irreversible process and makes a distance between the previous and next steps. Such equations arise very often in various applications of science and engineering describing the variation of temperature (or heat distribution) in a given region over some time [34]. It can be expressed as the heat flow in the rod with diffusion C_{xx} along the rod where the coefficient is the thermal diffusivity of the rod and L is the length of the rod . In this model, the flow of the heat in one-dimension that is insulated everywhere except at the two end points.

In this chapter, finite difference methods are proposed for the numerical solutions of one dimensional heat equation. Two test examples are considered to test the accuracy and efficiency of the methods.

In chapter 3, we consider the one dimensional advection-diffusion equation

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} = D\frac{\partial^2 c}{\partial x^2} \quad , \quad 0 < x < L \quad , \qquad 0 < t \leq T$$

The mathematical model describing the transport and diffusion processes is the one-dimensional advection-diffusion equation. Mathematical modeling of heat transport, pollutants and suspended matter in groundwater involves the solution of a convection–diffusion equation.

In this chapter, finite difference methods are proposed for the numerical solution of one dimensional advection-diffusion equation. Two test examples are considered to test the accuracy and efficiency of the method. The absolute errors are calculated for the both examples.

Chapter 2

Finite Difference Methods for Heat and Wave Equations

2. 1 Introduction

In this chapter, we will discuss finite difference methods and their stability for solving one-dimensional heat and wave equations. The equations have great importance in science and engineering. This problem is one of the well-known second order linear partial differential equation [33, 34]. It shows that heat equation describes irreversible process and makes a distance between the previous and next steps. Such equations arise very often in various applications of science and engineering describing the variation of temperature (or heat distribution) in a given region over some time [34]. It can be expressed as the heat flow in the rod with diffusion c_{xx} along the rod where the coefficient is the thermal diffusivity of the rod and L is the length of the rod. In this model, the flow of the heat in one-dimension that is insulated everywhere except at the two end points. Solutions of this equation are functions of the state along the rod and the time t. In the past, this problem has been widely worked over a number of years by numerous authors. But it is still an interesting problem since many physical phenomena can be formulated into PDEs with boundary conditions. The heat equation is of fundamental importance in diverse scientific fields. It is the prototypical parabolic partial differential equation in mathematics. In probability theory, the heat equation is connected with the study of Brownian motion via the Fokker–Planck equation. Numerical solutions of those equations are very useful to study physical phenomena. One of the linear evolution equations which we deal with the numerical solution is the heat equation [34].

2.2 Heat Equation

Consider the one-dimensional heat equations

$$\frac{\partial c}{\partial t}=\frac{\partial^2 c}{\partial x^2}, \qquad 0 \le x \le L \tag{2.2}$$

with initial and boundary conditions

$$c(0,t) = c_0 \qquad t > 0 \tag{2.2a}$$

$$c(L,t)=c_L \qquad t>0 \tag{2.2b}$$

$$c(x,0) = f(x) \qquad t = 0 \tag{2.2c}$$

The equation (2.2) is defined in the space domain $0 \le x \le L$ and the domain can also be normalized varying from 0 to 1 by change of variable, if required.

2.2.1 Explicit Finite Difference Method for Heat Equation

Consider one dimensional heat equation

$$\frac{\partial c}{\partial t}=\frac{\partial^2 c}{\partial x^2} \tag{2.2.1}$$

The equivalent finite difference approximation of the equation is

$$\frac{c_i^{n+1}-c_i^n}{\Delta t}=\left[\frac{c_{i+1}^n-2c_i^n+c_{i-1}^n}{\Delta x^2}\right] \tag{2.2.1a}$$

where $x_i = i\Delta x, i = 1,2,3..., N, t_n = n\Delta t,\ n = 1,2,3..., N-1,$ we use the forward difference formula for the derivative with respective to *t* and central difference formula for the with respect to x. If we let

$$r = \frac{\Delta t}{k(\Delta x)^2} \qquad (2.2.1b)$$

Equation (2.2.1a) can be written as

$$c_i^{n+1} = rc_{i+1}^n + (1-2r)c_i^n + rc_{i-1}^n \qquad (2.2.1c)$$

Explicit method uses the fact that we know the dependent variable, c at all *x* at time *t* from initial conditions. Since the equation contains only one unknown, c_i^{n+1} (i.e. c at time $t + \Delta t$), it can be obtained directly from known values of *c* at *t*.

Thus, the values of c along the first time row $t = \Delta t$ can be calculated in terms of the boundary and initial conditions, then the values of c along the second row, $t = 2\Delta t$ are calculated in terms of the first time row, and so on.

2.2.2 Crank-Nicolson Scheme

The Crank-Nicolson scheme is an implicit scheme and in this scheme the mesh points are discretized at the midpoint of nth and $(n+1)th$ levels, i.e. at $(x_i, t_{n+1/2})$.

Discretized the time derivative and space derivative in equation (2.2.1) by central differences, we have

$$\frac{c_i^{n+1} - c_i^n}{\Delta t} = \frac{1}{2}\left[\frac{c_{i+1}^n - 2c_i^n + c_{i-1}^n}{\Delta x^2} + \frac{c_{i-1}^{n+1} - 2c_i^{n+1} + c_{i+1}^{n+1}}{\Delta x^2}\right] + O(\Delta t^2) + O(\Delta x^2) \qquad (2.2.2a)$$

It may be noted that since the values of c are not available at the midpoint, the second derivative at the point $\left(i, n+\frac{1}{2}\right)$ is replaced by the average values at the (i,n) and $(i,n+1)$ points. The truncation error in the Crank-Nicolson scheme is $O(\Delta t^2) + O(\Delta x^2)$

Rearranging the equation (2.3a) and putting $r = \frac{\Delta t}{(\Delta x)^2}$ we get

$$-rc_{i+1}^{n+1} - rc_{i-1}^{n+1} + 2(1+r)c_i^{n+1} = rc_{i+1}^n + rc_{i-1}^n + 2(1-r)c_i^n + \Delta t\left[o(\Delta t^2) + o(\Delta x^2)\right]$$

Neglecting the error terms, the Crank-Nicolson scheme becomes

$$-rc_{i+1}^{n+1} - rc_{i-1}^{n+1} + 2(1+r)c_i^{n+1} = rc_{i+1}^n + rc_{i-1}^n + 2(1-r)c_i^n \qquad (2.2.2b)$$

for $i=1,2,...,N-1$

In order to find the value of c at the $(n+1)th$ level, we have to invoke it for $i = 1(1)N-1$ which will result in $N-1$ equation in $N-1$ unknowns and their solution will give the value of c at the $(n+1)th$ time level. It may also be mentioned that the resulting system of equations is tridiagonal and can be very easily solved by Gaussian elimination, Thomas algorithm and some other standard method.

2.3 Stability Analysis of Finite Difference Method

There are two methods to discuss the stability of finite difference methods

(a) Matrix method

(b) Fourier method

2.3.1. Stability of Explicit Scheme

(a) Matrix method

In the explicit scheme the values of c at the level of time $n+1$ are given explicitly in terms of the values of c at the previous level n, i.e.

$$c_{(i,n+1)} = rc_{(i+1,n)} + (1-2r)c_{(i,n)} + rc_{(i-1,n)}, \quad i = 1,2,...,N-1 \tag{2.3.1a}$$

Let us suppose that the values of c at the nth level are not the true values and have certain error in them. We have computed c at the $(n+1)th$ level using these values which will also have error in it. Let us denote the true value by c, approximate (actually computed) value by c^* and the associated error in c^* by e, so that the computed value can be expressed as,

$$c^*(i,n+1) = rc^*(i+1,n) + (1-2r)c^*(i,n) + rc^*(i-1,n) \tag{2.3.1b}$$

Subtracting (2.3.1b) from (2.3.1a), we get

$$e_{(i,n+1)} = re_{(i+1,n)} + (1-2r)e_{(i,n)} + re_{(i-1,n)} \qquad i = 1,2,......., N-1 \tag{2.3.1c}$$

where $e_{(i,n)}$ denotes the error in the exact value c and the computed value c^* at the (i,n) mesh points. It may be remembered that $c_{(0,n+1)} = c_0$ and $c_{(N,n+1)} = U_L$ on account of the boundary conditions and their value are not being computed at any time. Hence there is no question of error in the values of c at the endpoints so that $e_{(0,n)} = e_{(N,n)} = 0$. We can now write (2.3.1c) in matrix form as follows:

$$\begin{bmatrix} e_{1,n+1} \\ e_{2,n+1} \\ e_{3,n+1} \\ \cdots\cdots\cdots \\ \cdots\cdots \\ e_{N-1,n+1} \end{bmatrix} = \begin{bmatrix} 1-2r & r & 0 & 0 & \cdots\cdots & 0 \\ r & 1-2r & r & 0 & \cdots\cdots & 0 \\ 0 & r & 1-2r & r & \cdots\cdots & 0 \\ & & \cdots\cdots & & & . \\ & & & & & . \\ & & & \cdots\cdots & & \\ & & & & r & 1-2r \end{bmatrix} \begin{bmatrix} e_{1,n} \\ e_{2,n} \\ e_{3,n} \\ \cdots\cdots \\ \cdots\cdots \\ e_{N-1,n} \end{bmatrix} \qquad (2.3.1d)$$

$$or\ e_{j+1} = Ae_j \qquad (2.3.1e)$$

where $$e_{n+1}^T = \begin{bmatrix} e_{1,n+1} & e_{2,n+1} & e_{3,n+1} & \cdots\cdots & e_{N-1,n+1} \end{bmatrix} \qquad (2.3.1f)$$

$$e_n^T = \begin{bmatrix} e_{1,n} & e_{2,n} & e_{3,n} & \cdots\cdots & e_{N-1,n} \end{bmatrix} \qquad (2.3.1g)$$

$$A = \begin{bmatrix} 1-2r & r & 0 & 0 & \cdots\cdots & 0 \\ r & 1-2r & r & 0 & \cdots\cdots & 0 \\ 0 & r & 1-2r & r & \cdots\cdots & 0 \\ & & \cdots\cdots & & & . \\ & & & & & . \\ & & & \cdots\cdots & & \\ & & & & r & 1-2r \end{bmatrix}_{N-1\times N-1} \qquad (2.3.1h)$$

Then according to formula (2.3.1e) these error will become Ae_0 at the next step ignoring the computing error introduced at that step, so that $e_1 = Ae_0$ Then at the following step the contribution of e_1 will be $e_2 = Ae_1 = A^2e_0$. Proceeding in the same manner, after k step, the error e_0 will contribute to overall error

$$e_k = A^k e_0 \qquad k=1,2,..., \qquad (2.3.1k)$$

It should be noted that local computational error has been neglected at each time level. We observe from (2.3.1k) that e_k is dependent on A^k . If elements of matrix A become smaller and smaller tending to zero, or remain bounded, as k becomes infinitely larger than the scheme will be stable. On the other hand, if they increase, then the scheme will be unstable.

Let us suppose that the matrix $A\{(N-1)\times(N-1)\}$ possesses eigen value λ_s with Corresponding eigen vectors v_S, $s=1,2,...,N-1$ and that they are all distinct. They satisfy the equation,

$$Av=\lambda v \quad or\ Av_S=\lambda_S v_S, \qquad s=1,2,...,N-1 \tag{2.3.1l}$$

we can express the error vector e_0 by a linear combination of these vectors say

$$e_0=\sum_{s=1}^{N-1} c_S V_S \tag{2.3.1m}$$

where

$$v_s^T=\left[v_{1s}\ \ v_{2s},...,v_{N-1,s}\right] \tag{2.3.1n}$$

It may be noted that the $(N-1)$coefficient, c_s can be uniquely determined from the system of equation (2.3.1m).

we can express the error vector e_0 by a linear combination of these vectors, say

$$A^k e_0 = A^{k-1}\sum_{s=1}^{N-1} c_S A v_S, \qquad \text{form (2.3.1m)}$$

$$= A^{k-1}\sum c_S \lambda_S V_S \qquad \text{from(2.3.1l)}$$

$$=\sum_{s=1}^{N-1} c_S \lambda_S^k V_S \tag{2.3.1o}$$

The terms $A^k e_0$ on the left side (2.3.1o) denotes the propagation of error e_0 at the kth step as shown in (2.3.1k). if this error is not to increase indefinitely with k, then we must have

$$|\lambda_s| \le 1,\ s = 1,2,\ldots,N-1$$

$$or \quad |\lambda_{\max}| \le 1$$

Thus for the Explicit scheme to be stable, the modulus of the largest eigen value of matrix A, given by (2.3.1h), should not exceed unity.

(b) Fourier method:

The Explicit Method is

$$c_{(i,n+1)} = rc_{(i+1,n)} + (1-2r)c_{(i,n)} + rc_{(i-1,n)}, \quad i = 1,2,\ldots,N-1$$

$$let \ \ c_i^n = \sum_{m=-\infty}^{m=-\infty} A_m(t_n)e^{j\beta_m x_i} \qquad where \ \ j = \sqrt{-1}$$

$$\sum_{m=-\infty}^{m=\infty} A_m(t_{n+1})e^{j\beta_m x_i} = \lambda\left[\sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_{i+1}} + \sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_{i-1}}\right]$$

$$+(1-2\lambda)\sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_i}$$

$$A_m(t_{n+1})e^{j\beta_m x_i} = A_m(t_n)\left[\lambda(e^{j\beta_m x_{i+1}} + e^{j\beta_m x_{i-1}}) + (1-2\lambda)e^{j\beta_m x_i}\right]$$

Dividing both sides by $e^{j\beta_m x_i}$

$$\frac{A_m(t_{n+1})}{A_m(t_n)} = \lambda\left[e^{j\beta_m h} + e^{-j\beta_m h}\right] + (1-2\lambda)$$

$$\frac{A_m(t_{n+1})}{A_m(t_n)} = 2\lambda\cos\beta_m h + (1-2\lambda)$$

$$=2\lambda(\cos\beta_m h-1)+1$$

$$=-4\lambda\sin\frac{\beta_m h}{2}+1 \qquad \Rightarrow 1-4\lambda\sin\frac{\beta_m h}{2}$$

Magnification factor $\dfrac{A_m(t_{n+1})}{A_m(t_n)}$

For stability

$$\left|\frac{A_m(t_{n+1})}{A_m(t_n)}\right|\le 1 \qquad \Rightarrow \left|1-4\lambda\sin^2\frac{\beta_m h}{2}\right|\le 1$$

$\Rightarrow \lambda \le \dfrac{1}{2\sin^2\dfrac{\beta_m h}{2}}$ since the value of $\sin^2\dfrac{\beta_m h}{2}$ is always less than or equal to one ,the

minimum value of λ should be $\frac{1}{2}$.The condition for Stability of the Explicit Scheme is $\lambda \le \frac{1}{2}$

2.3.2 Stability of Crank Nicolson's (C-N) Method

The method of Crank Nicolson is

$$-rc_{i+1}^{n+1}-rc_{i-1}^{n+1}+2(1+r)c_i^{n+1}=rc_{i+1}^{n}+rc_{i-1}^{n}+2(1-r)c_i^{n}$$

now we check the stability by fourier method

$$let\ \ c_i^n=\sum_{m=-\infty}^{m=\infty}A_m(t_n)e^{j\beta_m x_i} \qquad where\ \ j=\sqrt{-1}$$

$$-\lambda \sum_{m=-\infty}^{m=\infty} A_m\left(t_{n+1}\right)e^{j\beta_m x_{i+1}} - \lambda \sum_{m=-\infty}^{m=\infty} A_m\left(t_{n+1}\right)e^{j\beta_m x_{i-1}}$$

$$+ 2(1+\lambda) \sum_{m=-\infty}^{m=\infty} A_m\left(t_{n+1}\right)e^{j\beta_m x_i}$$

$$= \lambda \sum_{m=-\infty}^{m=\infty} A_m\left(t_n\right)e^{j\beta_m x_{i+1}} + \lambda \sum_{m=-\infty}^{m=\infty} A_m\left(t_n\right)e^{j\beta_m x_{i-1}}$$

$$+ 2(1-\lambda) \sum_{m=-\infty}^{m=\infty} A_m\left(t_n\right)e^{j\beta_m x_i}$$

$$A_m\left(t_{n+1}\right)\left(-\lambda\left(e^{-j\beta_m h} + e^{j\beta_m h}\right) + 2(1+\lambda)\right)$$

$$= A_m\left(t_n\right)\left(\lambda\left(e^{-j\beta_m h} + e^{j\beta_m h}\right) + 2(1-\lambda)\right)$$

$$A_m\left(t_{n+1}\right)\left(-2\lambda\cos\beta_m h + 2(1+\lambda)\right) = A_m\left(t_n\right)\left(2\lambda\cos\beta_m h + 2(1-\lambda)\right)$$

$$\frac{A_m(t_{n+1})}{A_m(t_n)} = \frac{2\lambda\cos\beta_m h + 2(1-\lambda)}{-2\lambda\cos\beta_m h + 2(1+\lambda)}$$

$$\frac{\lambda\left(\cos\beta_m h - 1\right) + 1}{-\lambda\left(\cos\beta_m h - 1\right) + 1} = \frac{1 - 2\lambda\sin^2\frac{\beta_m h}{2}}{1 + 2\lambda\sin^2\frac{\beta_m h}{2}}$$

Magnification factor $\dfrac{A_m(t_{n+1})}{A_m(t_n)}$

For stability

$$\left|\frac{A_m(t_{n+1})}{A_m(t_n)}\right| \le 1 \qquad \Rightarrow \left|\frac{1-2\lambda\sin^2\frac{\beta_m h}{2}}{1+2\lambda\sin^2\frac{\beta_m h}{2}}\right| \le 1$$

$$\left|1-2\lambda\sin^2\frac{\beta_m h}{2}\right| \le \left|1+2\lambda\sin^2\frac{\beta_m h}{2}\right|$$

which is true for all values of $\lambda \ge 0$ since $0 \le \sin^2\frac{\beta_m h}{2} \le 1$.

2.4 Wave Equation

Consider a one-space dimensional wave equation

$$\frac{\partial^2 c}{\partial t^2} = u^2\frac{\partial^2 c}{\partial x^2}, \qquad a \le x \le b \tag{2.4a}$$

with initial and boundary conditions

$$c(x,0) = f(x), \tag{2.4b}$$

$$\frac{\partial c(x,0)}{\partial t} = g(x), \tag{2.4c}$$

$$c(a,t) = f_1(t), \qquad c(b,t) = f_2(t) \tag{2.4d}$$

Equation (2.4a) represents the notation of vibrating string stretched between two points. where c is denotes the displacement of a point on the string at a distance x, at any instant t while the string vibrates in the $c-x$ plane. The string is assumed to be uniform and elastic and that $u^2 = \frac{T}{m}$, where T is the tension in the string and m is its mass per unit length. The equation may also represent the displacement of a longitudinally vibrating bar or of sound waves in a pipe. In two-space dimension it may represent deflection of a membrane.

2.4.1 Explicit Method for Wave Equation

Discretizing the equation (2.4a) at the mesh point (i,n), we have

$$\frac{c_i^{n-1} - 2c_i^n + c_i^{n+1}}{\Delta t^2} + O(\Delta t^2) = \frac{c_{i-1}^n - 2c_i^n + c_{i+1}^n}{\Delta x^2} + O(\Delta x^2) \qquad (2.4.1a)$$

Neglecting the truncation error, we can write from (2.4.1a)

$$c_i^{n+1} = r^2\left(c_{i-1}^n + c_{i+1}^n\right) + 2\left(1 - r^2\right)c_i^n - c_i^{n-1} \qquad i = 1(1)N-1 \qquad (2.4.1b)$$

where $\quad r = \frac{\Delta t}{\Delta x}$.

2.4.2. Implicit Method for Wave Equation

Discretizing partial differential equation (2.4a) at (i,n) mesh point such time derivatives is replaced by central difference and space derivative by the average of central difference approximations at $(i,n+1)$ and $(i,n-1)$, i.e.

$$\frac{c_i^{n-1}-2c_i^n+c_i^{n+1}}{\Delta t^2}=\frac{1}{2}\left[\frac{c_{i-1}^{n+1}-2c_i^{n+1}+c_{i+1}^{n+1}}{\Delta x^2}+\frac{c_{i-1}^{n-1}-2c_i^{n-1}+c_{i+1}^{n-1}}{\Delta x^2}\right]$$

After transposing terms we get,

$$-r^2c_{i-1}^{n+1}+2\left(1+r^2\right)c_i^{n+1}-r^2c_{i+1}^{n+1}=r^2c_{i-1}^{n-1}+2\left(1+r^2\right)c_i^{n-1}+r^2c_{i+1}^{n-1}+4c_i^n \qquad i=1(1)N-1 \qquad (2.4.2a)$$

2.5 Numerical Experiments

In this section, we have considered some numerical examples for heat and wave equations to check the accuracy and efficiency of the finite differences methods proposed in the above sections.

Example 2.1: Consider the one dimensional heat equation

$$\frac{\partial c}{\partial t}=\frac{1}{\pi^2}\frac{\partial^2 c}{\partial x^2} \quad , \quad 0<x<1 \quad , \quad t>0 \qquad (2.5a)$$

with initial and boundary conditions

$$c(x,0)=\sin(\pi x) \quad , \quad 0<x<1 \quad , \quad c(0,t)=c(1,t)=0 \quad , \quad t\geq 0 \qquad (2.5b)$$

The exact solution is given by

$$c(x,t) = e^{-t} \sin(\pi x) \tag{2.5c}$$

The numerical results of the example are shown in Table 2.1 and Figures 1-7. Table 2.1 shows absolute error at different time and time steps length *t*. Figure 1 -7 show the comparison of numerical and exact solutions at different times.

Example 2.2: Consider the heat conduction equation

$$\frac{\partial c}{\partial t} = \frac{\partial^2 c}{\partial x^2} \qquad 0 \le x \le 1, \quad t > 0 \tag{2.5d}$$

with initial and boundary conditions

$$c(x,0) = \begin{cases} 2x & 0 \le x \le \frac{1}{2} \\ 2(1-x) & \frac{1}{2} \le x \le 1 \end{cases} \tag{2.5e}$$

$$c(0,t) = c(1,t) = 0 \quad t > 0 \tag{2.5f}$$

The numerical results of the example are shown in Table 2.2 and Figures 8-14. Table 2.2 shows absolute error at different time and time steps length *t*. Figure 8 -14 show the comparison of numerical and exact solutions at different times.

Example 2.3: Consider a one dimensional wave equation

$$\frac{\partial^2 c}{\partial t^2} = \frac{\partial^2 c}{\partial x^2}, \qquad 0 \le x \le 1, \quad t \ge 0 \tag{2.5g}$$

with initial and boundary conditions

$$c(x,0) = 2x(1-x) \tag{2.5h}$$

$$\frac{\partial c(x,0)}{\partial t} = 0 \tag{2.5i}$$

$$c(0,t) = c(1,t) = 0 \tag{2.5j}$$

The numerical results at different value for *t and x* of the example are shown in Table 2.3.

Table 2.1: Maximum absolute error of example 1 for different method at different time t.

Δt	t	Explicit Method	Crank-Nicholson Method
0.0001	0.48	7.5653×10^{-5}	3.8689×10^{-6}
0.0001	0.58	3.3863×10^{-5}	1.7441×10^{-6}
0.0001	0.78	6.3147×10^{-6}	3.2623×10^{-7}
0.0001	1.0	9.2134×10^{-7}	4.7731×10^{-8}
0.0001	5.0	3.8148×10^{-23}	1.7165×10^{-24}

Table 2.2: Maximum absolute error of example 2.1 for different method at different time t.

Δt	t	Crank-Nicholson Method	Explicit Method
0.0001	0.48	3.7283×10^{-6}	6.1652×10^{-5}
0.0001	0.58	1.6431×10^{-6}	3.2347×10^{-5}
0.0001	0.78	3.2333×10^{-7}	6.4178×10^{-6}
0.0001	1.0	4.8721×10^{-8}	8.3145×10^{-7}
0.0001	5.0	1.5162×10^{-23}	3.4248×10^{-22}

Table 2.3: Numerical solution of example 2.3 for different value of t and x.

t	x=0	x=0.1	x=0.2	x=0.3	x=0.4	x=0.5	x=0.6
0	0	0.18	0.32	0.42	0.48	0.50	0.48
0.1	0	0.16	0.30	0.40	0.46	0.48	0.46
0.2	0	0.12	0.24	0.34	0.40	0.42	0.40
0.3	0	0.08	0.16	0.24	0.30	0.32	0.30
0.4	0	0.04	0.08	0.12	0.16	0.18	0.16
0.5	0	0	0	0	0	0	0

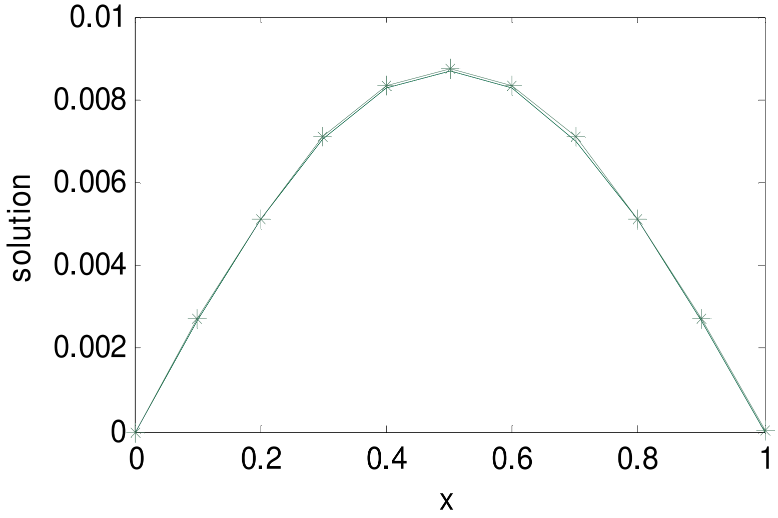


Figure 1: Comparison of exact and numerical solutions of Example 2.1 by explicit method for $\Delta t = 0.0001$ *at* $t = 0.48$.

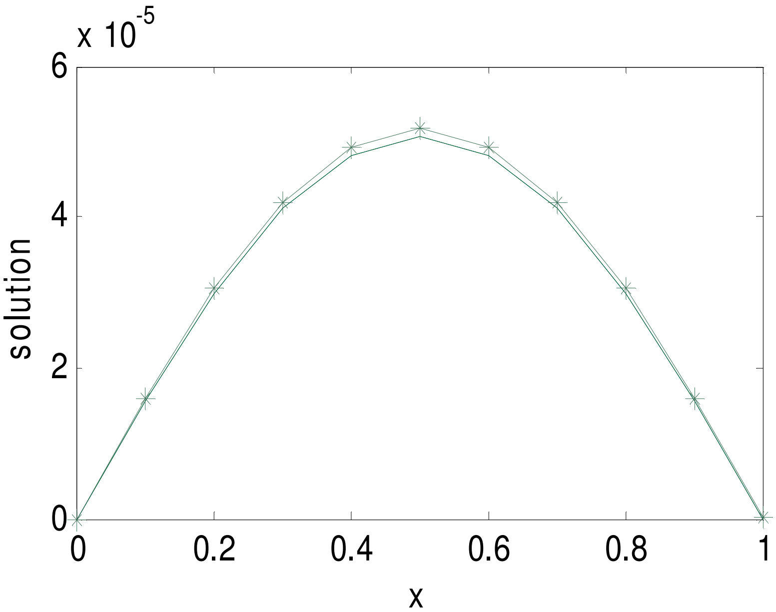


Figure 2: Comparison of exact and numerical solutions of Example 2.1 by explicit method for $\Delta t = 0.0001, t = 1$.

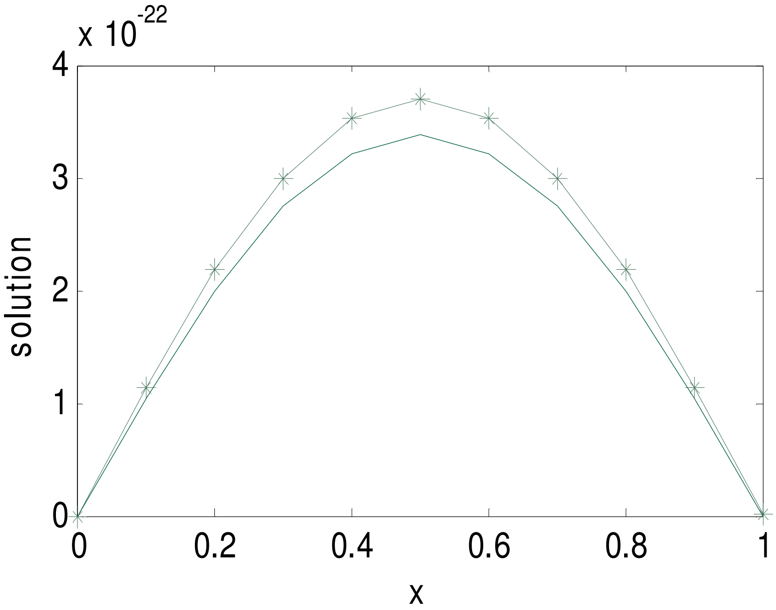


Figure 3: Comparison of exact and numerical solutions of Example 2.1 by explicit method for $\Delta t = 0.0001,\ t = 5$.

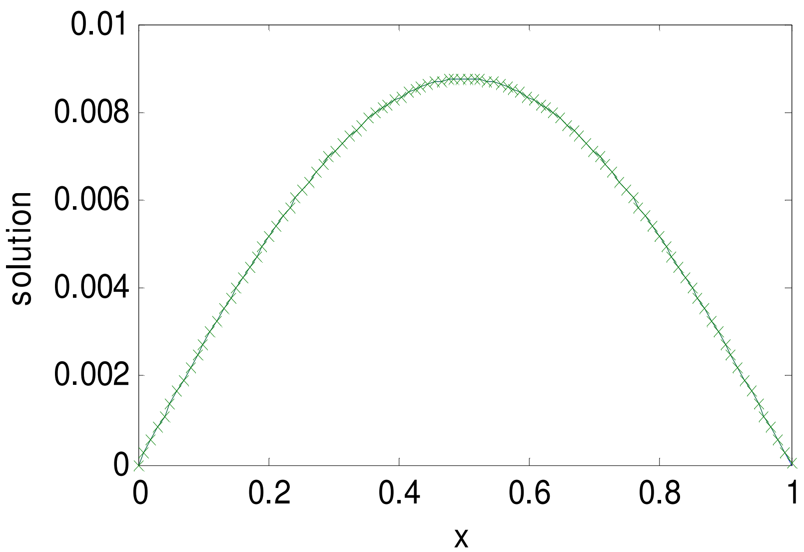


Figure 4: Comparison of exact and numerical solutions of Example 2.1 by Crank-Nicholson method for $\Delta t = 0.0001\ at\ t = 0.48$.

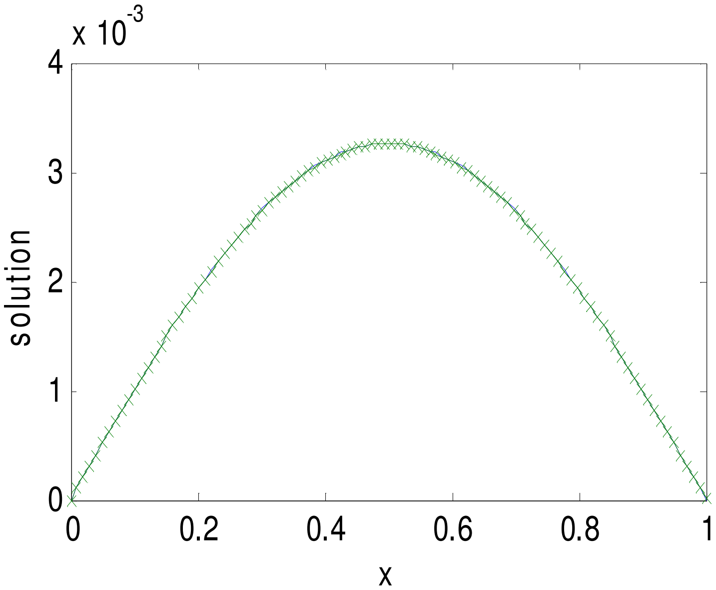


Figure 5: Comparison of exact and numerical solutions of Example 2.1 by Crank-Nicholson method for $\Delta t = 0.0001 \; at \; t = 1$.

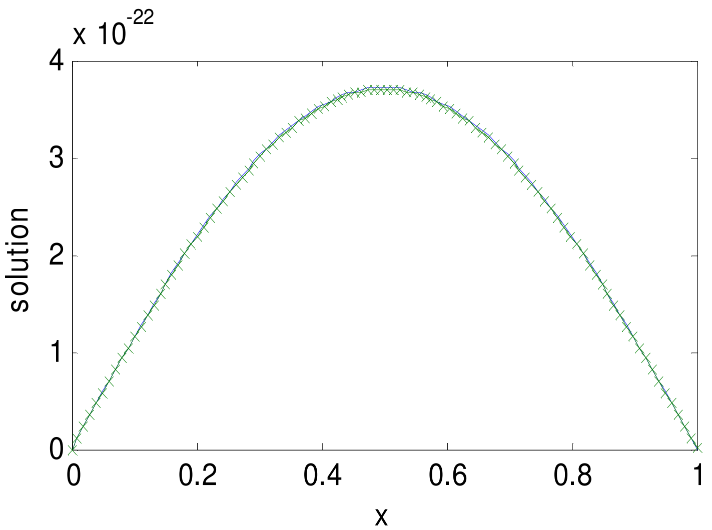


Figure 6: Comparison of exact and numerical solutions of Example 2.1 by Crank-Nicholson method for $\Delta t = 0.0001, t = 1$.

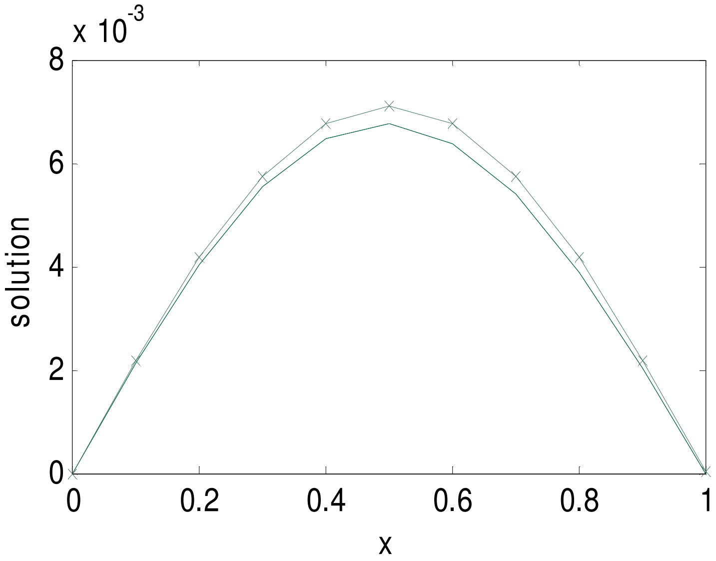


Figure 7: Comparison of exact and numerical solutions of Example 2.2 by explicit method for $\Delta t = 0.0001\ at\ t = 0.48$

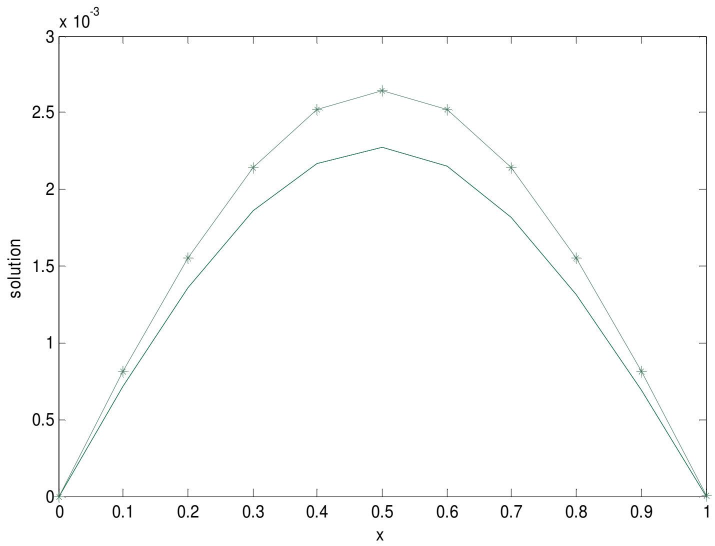


Figure 8: Comparison of exact and numerical solutions of Example 2.2 by explicit method for $\Delta t = 0.0001\ at\ t = 0.58$.

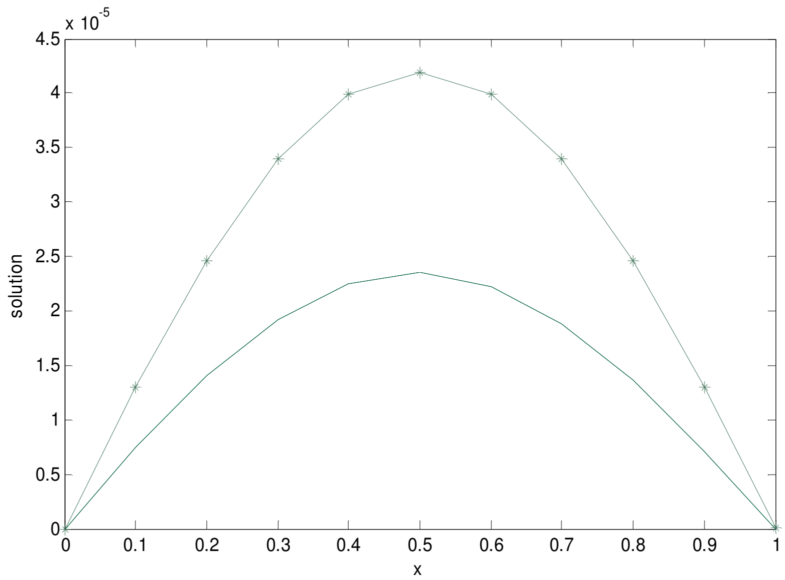


Figure 9: Comparison of exact and numerical solutions of Example 2.2 by explicit method for $\Delta t = 0.0001, t = 1$.

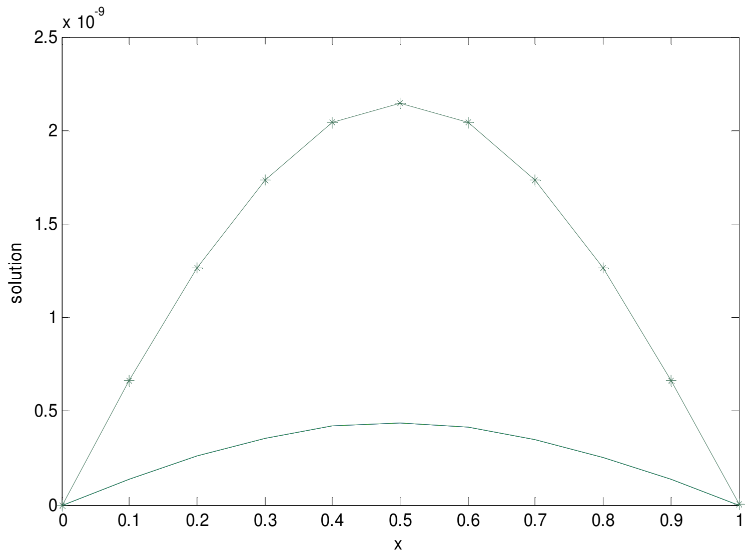


Figure 10: Comparison of exact and numerical solutions of Example 2.2 by explicit method for $\Delta t = 0.0001, t = 2$.

Chapter 3

Finite difference techniques for the Advection-diffusion equation

3.1. Introduction

The mathematical model describing the transport and diffusion processes is the one-dimensional advection-diffusion equation. Mathematical modeling of heat transport, pollutants and suspended matter in groundwater involves the solution of a convection–diffusion equation. When the velocity field is complex and changing in time, transport processes can't be analytically calculated, numerical approximations are necessary. The finite difference method is a well-established and solution techniques are covered in textbooks [3, 14, 25, 32, 33]. Many popular finite difference methods, such as Noye and Tan [22], have used a weighted discretisation with the modified equivalent partial differential equation for solving one-dimensional advection–diffusion equations (ADE). Later, the authors extended this scheme to solve two-dimensional ADE [23]. The upwind scheme of Spal-ding [31] and the flux-corrected scheme are available for the solution of the depth-averaged form of the ADE. Also, widely used is the split-operator approach [19, 29], in which the advection and diffusion terms are solved simultaneously by two different numerical methods.

Lam [16] points out that the central difference approximation will overestimate the advective flux so much that it often causes a negative concentration to appear in the neighboring cells. Some researchers have suggested flux corrected methods, which take into account the mass flow rates and flow directions on the grid cell boundaries by interpolation [26].

To prevent this situation Leonard [17] introduced an upstream interpolation method, namely QUICK (Quadratic Upstream Interpolation Convective Kinematics) for one-dimensional unsteady flow to prevent this situation. Later, Leonard improved this scheme, eliminating the wiggles completely by introducing exponential integration into regions with sharp fronts.

Sommeijer and Kok [30] used various time-integration techniques to improve a finite differences model for the numerical solution of three-dimensional ADE. Their model was validated by comparing with analytical solutions of transport of a Gaussian pulse in unsteady non-uniform flow. Sankaranarayanan et al.[27] used a third-order upwind difference scheme as given by Kowalik and Murty for the advective terms of the ADE.

However, previously mentioned techniques require extensive matrix algebra at each time step. One of the tools for solving a PDE is a spreadsheet like Excel. There are many advantages of spreadsheets such as numerical, visual feedback and a graphical interface. Solutions obtained through the spreadsheet are readily plotted in the same worksheet. Any changes in the input parameters of the solution domain are then reflected graphically.

Spreadsheets are applied in different fields of engineering to solve partial differential equations such as free-surface, steady-state groundwater flow, transient groundwater problems and groundwater parameter estimation.

The main objective of this study is to develop a user friendly, flexible advection–diffusion modeling simulation algorithm for the ADE with FDM. Thus ADE Explicit Spreadsheet Simulation (ADEESS) model is proposed. ADEESS model uses the Saulyev 's FD schemes. The main advantage of these schemes is that they are unconditionally stable and are explicit in nature. ADEESS model is not required to use one of the most significant features of spreadsheets: that is ''iterative calculation''. Only the " copy & paste '' feature of spreadsheets has been automatized and solved with sequential steps with the help of a basic macro. The results of the model are validated with the analytical and numerical solutions in literature.

By changing only the weighting parameter in the proposed model is solves ADE and the results can be examined simultaneously with the spreadsheet interface. Thus, the effects of the model parameters (such as $u, \Delta t, \Delta x, D$) to the results can easily be examined graphically. In addition, model parameters $(such\ \ as\ \Delta t, \Delta x)$ may be adjusted easily in order to overcome the problem of overshooting and negative concentrations.

3.2. Mathematical model

The mathematical model describing the transport and diffusion processes is the one-dimensional advection–diffusion equation

$$\frac{\partial c}{\partial t}+u\frac{\partial c}{\partial x}=D\frac{\partial^2 c}{\partial x^2}, \quad 0<x<L, \quad 0<t\leq T \tag{3.1}$$

with initial conditions

$$c(x,0) = f(x) \quad 0 \leq x \leq L \tag{3.2}$$

and boundary conditions

$$\begin{aligned} c(0,t) &= g(t), \quad 0 < t \leq T \\ c(L,t) &= h(t), \quad 0 < t \leq T \end{aligned} \tag{3.3}$$

where f, g and h are known functions, while the function c is unknown, u is the velocity in x direction and D is the dispersion coeffcient. Note that u and D are considered to be positive constant values. In general, explicit finite difference methods are restricted to necessitate extremely small values for time step to maintain stability.

On the other hand, stable implicit finite difference schemes require the solution of simultaneous linear equations at each time step.

3.3. Numerical Solution of Advection-Diffusion Equation by Finite Difference Method

The solution domain of the problem is covered by a mesh of grid-lines

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

$$t_n = n\Delta t,...,n = 0,1,2,...,N$$

Where x_i and t_n are parallel to the space and time coordinate axes. The constant spatial and temporal grid-spacing are $\Delta x = L/M$ *and* $\Delta t = T/N$. Consider the following approximations of the derivatives in the advection–diffusion equation which incorporate time weights θ as follows:

$$\frac{\partial c}{\partial t} = \frac{c(i,n+1) - c(i,n)}{\Delta t} \tag{3.5}$$

$$u\frac{\partial c}{\partial x} = u\left\{\left\{\theta\left[\frac{c(i,n)-c(i-1,n)}{\Delta x}\right]\right\}+(1-\theta)\left[\frac{c(i+1,n)-c(i-1,n)}{2\Delta x}\right]\right\} \tag{3.6}$$

$$D\frac{\partial^2 c}{\partial x^2} = \frac{D[c(i-1,n)-2c(i,n)+c(i+1,n)]}{(\Delta x)^2} \tag{3.7}$$

Put these value in equation (3.1)

$$\frac{c(i,n+1)-c(i,n)}{\Delta t} + u\left\{\theta\left[\frac{c(i,n)-c(i-1,n)}{\Delta x}\right] + (1-\theta)\left[\frac{c(i+1,n)-c(i-1,n)}{2\Delta x}\right]\right\}$$

$$= D\left[\frac{c(i-1,n)-2c(i,n)+c(i+1,n)}{(\Delta x)^2}\right]$$

$$c(i,n+1) = -u\Delta t\left\{\theta\left[\frac{c(i,n)-c(i-1,n)}{\Delta x}\right] + (1-\theta)\left[\frac{c(i+1,n)-c(i-1,n)}{2\Delta x}\right]\right\}$$

$$+D\Delta t\left[\frac{c(i-1,n)-2c(i,n)+c(i+1,n)}{(\Delta x)^2}\right]+c(i,n)$$

$$c(i,n+1)=\frac{u\Delta tc(i-1,n)}{\Delta x}+\frac{u(1-\theta)\Delta tc(i-1,n)}{2\Delta x}+\frac{D\Delta tc(i-1,n)}{(\Delta x)^2}-\frac{u\Delta t\theta c(i,n)}{\Delta x}-2\frac{D\Delta tc(i,n)}{(\Delta x)^2}+c(i,n)$$

$$-\frac{u(1-\theta)\Delta tc(i+1,n)}{2\Delta x}+\frac{D\Delta tc(i+1,n)}{(\Delta x)^2}$$

$$c(i,n+1)=\left[S+(1+\theta)R\right]c(i-1,j)+\left[1-2S-2R\theta\right]c(i,n)+\left[S-R(1-\theta)\right]c(i+1,n) \tag{3.8}$$

$$where\quad R=\frac{u\Delta t}{2\Delta x}\quad and\ S=\frac{\Delta tD}{(\Delta x)^2}$$

3.3.1. Explicit Method

Assume that $\theta = 0$ in Equation (3.8) and it may be written as the following upwind explicit-type Finite Difference Formula is solving the ADE

$$c(i,n+1)=[S+R]c(i-1,n)+[1-2S]c(i,n)+[S-R]c(i+1,n)$$

3.3.2. Implicit Method

If assuming $\theta = 1$ in Equation (3.8) may be written the following Implicit-type finite difference formula to solve the ADE

$$c(i,n+1)=[S+2R]c(i-1,n)+[1-2(S)-2R]c(i,n)+[S]c(i+1,n)$$

3.3.3. Crank Nicholson Method

Assuming $\theta = \frac{1}{2}$ in Equation yields the following

$$c(i,n+1)=\left[S+\frac{3}{2}R\right]c(i-1,n)+\left[1-2(S)-R\right]c(i,n)+\left[S-\frac{1}{2}R\right]c(i+1,n)$$

3.3.4. Lax-Wendroff Method

If assuming $\theta = 2R$ in Equation (3.8) may be written the following Lax-Wendroff type finite difference formula to solve the ADE

$$c(i,n+1) = \left[S + R + 2R^2\right]c(i-1,n) + \left[1 - 2(S) - 4R^2\right]c(i,n) + \left[S - R + 2R^2\right]c(i+1,n)$$

3.4. Stability Analysis of Finite Difference Methods

In this section, we have analyzed the stability of finite difference methods discussed in the above section by using Fourier method.

3.4.1. Stability of Explicit Method

Let $c_i^n = \sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_i} \quad where \quad j = \sqrt{-1}$

$$\sum_{m=-\infty}^{m=\infty} A_m(t_{n+1})e^{j\beta_m x_i} = (\lambda_1 + \lambda_2)\sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_{i-1}}$$

$$+ (\lambda_1 - \lambda_2)\sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_{i+1}} + (1-2\lambda_1)\sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_i}$$

$$\frac{A_m(t_{n+1})}{A_m(t_n)} = (\lambda_1 + \lambda_2)e^{-j\beta_m h} + (\lambda_1 - \lambda_2)e^{j\beta_m h} + (1 - 2\lambda_1)$$

$$= \lambda_1\left(e^{-j\beta_m h} + e^{j\beta_m h}\right) + \lambda_2\left(e^{-j\beta_m h} - e^{j\beta_m h}\right) + 1 - 2\lambda_1$$

$$\frac{A_m(t_{n+1})}{A_m(t_n)} = 2\lambda_1(\cos\beta_m h - 1) + 2\lambda_2 j\sin\beta_m h + 1$$

$$= -4\lambda_1 \sin^2 \frac{\beta_m h}{2} + 1 + 2\lambda_2 j \sin \beta_m h$$

Magnification factor $\frac{A_m(t_{n+1})}{A_m(t_n)}$

for stability

$\left| \frac{A_m(t_{n+1})}{A_m(t_n)} \right| \le 1 \quad \Rightarrow \left| -4\lambda_1 \sin^2 \frac{\beta_m h}{2} + 1 + 2\lambda_2 j \sin \beta_m h \right| \le 1$ after equating real and imaginary part

$$\left| 1 - 4\lambda_1 \sin^2 \frac{\beta_m h}{2} \right| \le 1 \Rightarrow -1 \le 1 - 4\lambda_1 \sin^2 \frac{\beta_m h}{2} \le 1$$

$$\Rightarrow -2 \le -4\lambda_1 \sin^2 \frac{\beta_m h}{2} \le 0$$

$\Rightarrow \lambda_1 \le \frac{1}{2\sin^2 \frac{\beta_m h}{2}}$ since the value of $\sin^2 \frac{\beta_m h}{2}$ is always less than or equal to one ,the minimum value of S should be $\frac{1}{2}$.The condition for stability of explicit scheme is say $\lambda \le \frac{1}{2}$

3.4.2. Stability of the Implicit Scheme

The scheme is

$$c(i, n+1) = \lfloor \lambda_1 + 2\lambda_2 \rfloor c(i-1, n) + \lfloor 1 - 2\lambda_1 - 2\lambda_2 \rfloor c(i, n) + \lfloor \lambda_1 \rfloor c(i+1, n)$$

Similarly, assume as above

Let $c_i^n = \sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_i} \quad where \quad j = \sqrt{-1}$

$$\sum_{m=-\infty}^{m=\infty} A_m(t_{n+1})e^{j\beta_m x_i} = (\lambda_1 + 2\lambda_2)\sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_{i-1}}$$

$$+(\lambda_1)\sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_{i+1}} + (1-2\lambda_1-2\lambda_2)\sum_{m=-\infty}^{m=\infty} A_m(t_n)e^{j\beta_m x_i}$$

$$\frac{A_m(t_{n+1})}{A_m(t_n)} = (\lambda_1+2\lambda_2)e^{-j\beta_m h} + (\lambda_1)e^{j\beta_m h} + (1-2\lambda_1-2\lambda_2)$$

$$= \lambda_1\left(e^{-j\beta_m h} + e^{j\beta_m h}\right) + \lambda_2\left(e^{-j\beta_m h} - e^{j\beta_m h}\right) + 1 - 2\lambda_1$$

$$\frac{A_m(t_{n+1})}{A_m(t_n)} = 2\lambda_1(\cos\beta_m h - 1) + 2\lambda_2 j\sin\beta_m h + 1$$

$$= -4\lambda_1 \sin^2\frac{\beta_m h}{2} + 1 + 2\lambda_2 j\sin\beta_m h$$

Magnification factor $\frac{A_m(t_{n+1})}{A_m(t_n)}$

For stability

$$\left|\frac{A_m(t_{n+1})}{A_m(t_n)}\right| \le 1 \quad \Rightarrow \left|-4\lambda_1 \sin^2\frac{\beta_m h}{2} + 1 + 2\lambda_2 j\sin\beta_m h\right| \le 1$$

after equating real and imaginary part, we have

$$\left|1-4\lambda_1 \sin^2 \frac{\beta_m h}{2}\right| \le 1 \Rightarrow -1 \le 1-4\lambda_1 \sin^2 \frac{\beta_m h}{2} \le 1$$

$$\Rightarrow -2 \le -4\lambda_1 \sin^2 \frac{\beta_m h}{2} \le 0$$

$\Rightarrow \lambda_1 \le \dfrac{1}{2\sin^2 \frac{\beta_m h}{2}}$ since the value of $\sin^2 \frac{\beta_m h}{2}$ is always less than or equal to one ,the

minimum value of S should be $\frac{1}{2}$.The condition for stability of explicit scheme is say $\lambda \le \frac{1}{2}$.

3.4.3. Stability of Crank Nicholson Method:

The Crank-Nicholson Method is

$$c(i,n+1) = \left[S + \frac{3}{2}R\right]c(i-1,n) + \left[1-2(S)-R\right]c(i,n) + \left[S - \frac{1}{2}R\right]c(i+1,n) \qquad (3.4.3a)$$

Now we check the stability by Fourier method

Let $c_i^n = \sum_{m=-\infty}^{m=\infty} A_m(t_n) e^{j\beta_m x_i}$ *where* $j = \sqrt{-1}$

Substituting the value in equation (3.4.3a), we have

$$\sum_{m=-\infty}^{m=\infty} A_m(t_{n+1}) e^{j\beta_m x_i} = \left(\lambda_1 + \frac{3}{2}\lambda_2\right) \sum_{m=-\infty}^{m=\infty} A_m(t_n) e^{j\beta_m x_{i-1}}$$

$$+\left(1-2\lambda_1-\lambda_2\right) \sum_{m=-\infty}^{m=\infty} A_m(t_n) e^{j\beta_m x_i} + \left(\lambda_1 - \frac{1}{2}\lambda_2\right) \sum_{m=-\infty}^{m=\infty} A_m(t_n) e^{j\beta_m x_{i+1}}$$

After solving these, we have

$$\frac{A_m(t_{n+1})}{A_m(t_n)} = (\lambda_1 + \frac{3}{2}\lambda_2)e^{-j\beta_m h} + \left(1 - 2\lambda_1 - \lambda_2\right) + \left(\lambda_1 - \frac{1}{2}\lambda_2\right)e^{j\beta_m h}$$

$$= \lambda_1\left(e^{-j\beta_m h} + e^{-j\beta_m h} - 2\lambda_1\right) + \frac{1}{2}\lambda_2\left(3e^{-j\beta_m h} - e^{j\beta_m h} - 2\right)$$

$$= 2\lambda_1\left(\cos\beta_m h - 1\right) + \lambda_2\left(j\sin\beta_m h + e^{-j\beta_m h} - 1\right)$$

Magnification factor $\frac{A_m(t_{n+1})}{A_m(t_n)}$ should be less than one

$$\left|\frac{A_m(t_{n+1})}{A_m(t_n)}\right| \le 1 \quad \Rightarrow \left|2\left(\lambda_1 + \lambda_2\right)\sin^2\frac{\beta_m h}{2} + 1\right| \le 1 \qquad \left\{\because \lambda_1 + \lambda_2 = \lambda(say)\right\}$$

which is true for all values of $\lambda \ge 0$ since $0 \le \sin^2\frac{\beta_m h}{2} \le 1$.

3.5 Numerical Experiments

In this section, we have considered two test examples to check the accuracy and efficiency of the schemes developed for advection-diffusion equation.

Example 3.1: Considered the advection-diffusion equation (3.1) over the region bounded by $0 \le x \le 1$ [1] and the analytical solution is given by

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

The initial and boundary conditions are given by

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

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

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

In this example, the values of various parameters are used $D = 0.01m^2 / s, \ u = 1m / s.$ The numerical results are calculated with space step and time step $\Delta x = 0.01m \ and \ \Delta t = 0.0001$ s, respectively. This problem model has been solved for show the maximum absolute error of the problem shown in Table 3.1 at different value of t. The figure 1-12 compare the exact and numerical solutions at the different time t for different value of θ

Example 3.2: A problem for which the exact solution is known is considered to test the accuracy of the proposed methods described for solving the advection-diffusion equation. These techniques are applied to solve (3.1)-(3.4) with $f(x), g(t)$ *and* $h(t)$ known and c unknown.

The initial conditions

$$c(x,0) = \exp\left[-\frac{(x-x_0)^2}{D}\right]$$

The boundary conditions

$$c(0,t) = \frac{1}{\sqrt{4t+1}}\exp\left[-\frac{(-x_0-ut)^2}{D(4t+1)}\right]$$

$$c(9,t) = \frac{1}{\sqrt{4t+1}}\exp\left[-\frac{(9-x_0-ut)^2}{D(4t+1)}\right]$$

The analytical solution to the one-dimensional advection-diffusion of Gaussian pulse of unit height at x=1 in a region bounded by $0 \leq x \leq 9$ as given by Ref [1] is

$$c(x,t) = \frac{1}{\sqrt{4t+1}}\exp\left[-\frac{(x-x_0-ut)^2}{D(4t+1)}\right]$$

where u is the velocity in the x direction, x_0 is the centre of the initial Gaussian pulse. D is the diffusion coefficient in the x direction and t is the time coordinate. The values of the various parameters used are $D = 0.005 m^2 / s$ and $u = 0.8 m / s$.

This problem model has been solved for show the maximum absolute error of the problem shown in Table 3.2 at different value of *t*. The Figure 13-20 compare the exact and numerical solutions at the different time t for different value of θ.

Table 3.1: Maximum absolute error of example 3.1 for different method at different time t.

Maximum Absolute error					
Δt	t	Implicit Method	Explicit Method	Crank Nicholson	Lax-Wendroff
0.00001	0.5	4.4027×10^{-4}	8.1000×10^{-3}	4.4000×10^{-3}	4.333×10^{-4}
0.00001	0.75	5.7169×10^{-4}	1.3100×10^{-3}	7.1000×10^{-3}	5.6719×10^{-3}
0.00001	1.0	6.0759×10^{-4}	1.7000×10^{-3}	9.3000×10^{-3}	6.0400×10^{-3}
0.00001	2.0	1.5173×10^{-4}	2.7000×10^{-3}	1.4000×10^{-3}	1.4920×10^{-3}

Table 3.2: Maximum absolute error of Example 3.2 for different method at different time t.

Maximum Absolute error				
Δt	t	Implicit Method	Explicit Method	Crank Nicholson
0.0001	0.5	8.5100×10^{-2}	10.8100×10^{-2}	5.1200×10^{-2}
0.0001	1.0	7.8200×10^{-2}	7.7200×10^{-2}	4.9600×10^{-2}
0.0001	2.0	6.6500×10^{-2}	4.8000×10^{-2}	4.5000×10^{-2}
0.0001	2.5	6.2600×10^{-2}	4.5100×10^{-2}	4.3500×10^{-2}

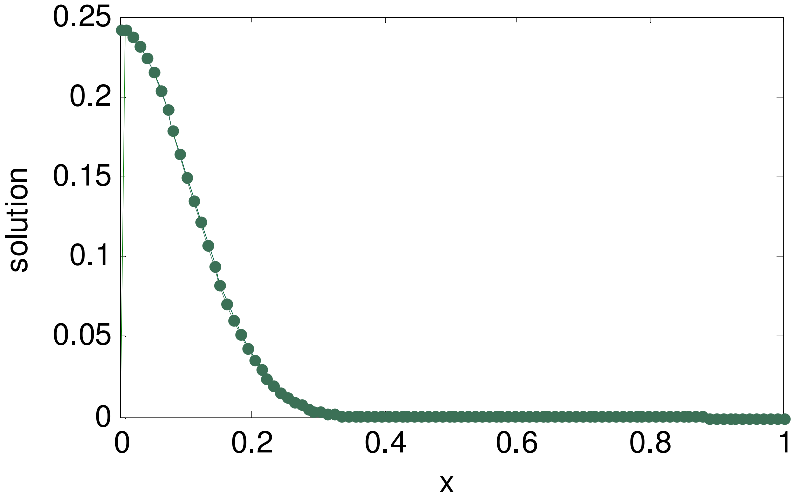


Figure 1: Comparision of exact and numerical solutions of Example 3.1 by explicit method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 0.5$.

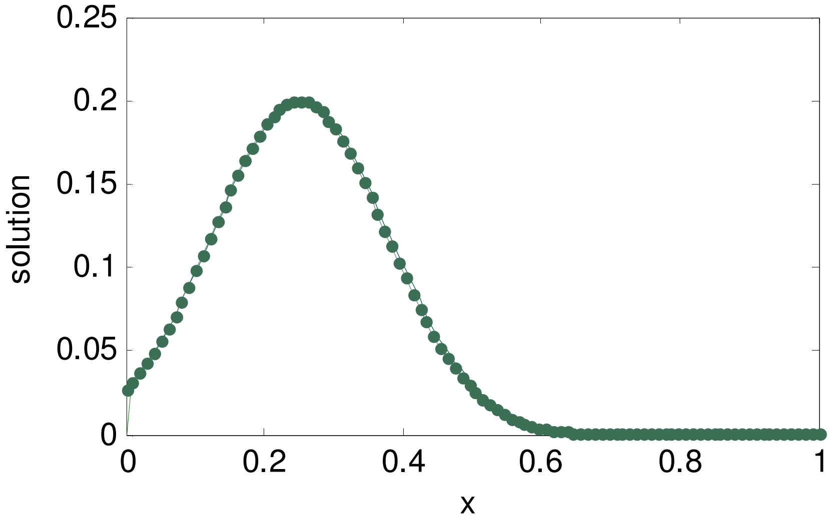


Figure 2: Comparision of exact and numerical solutions of Example 3.1 by explicit method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 0.75$.

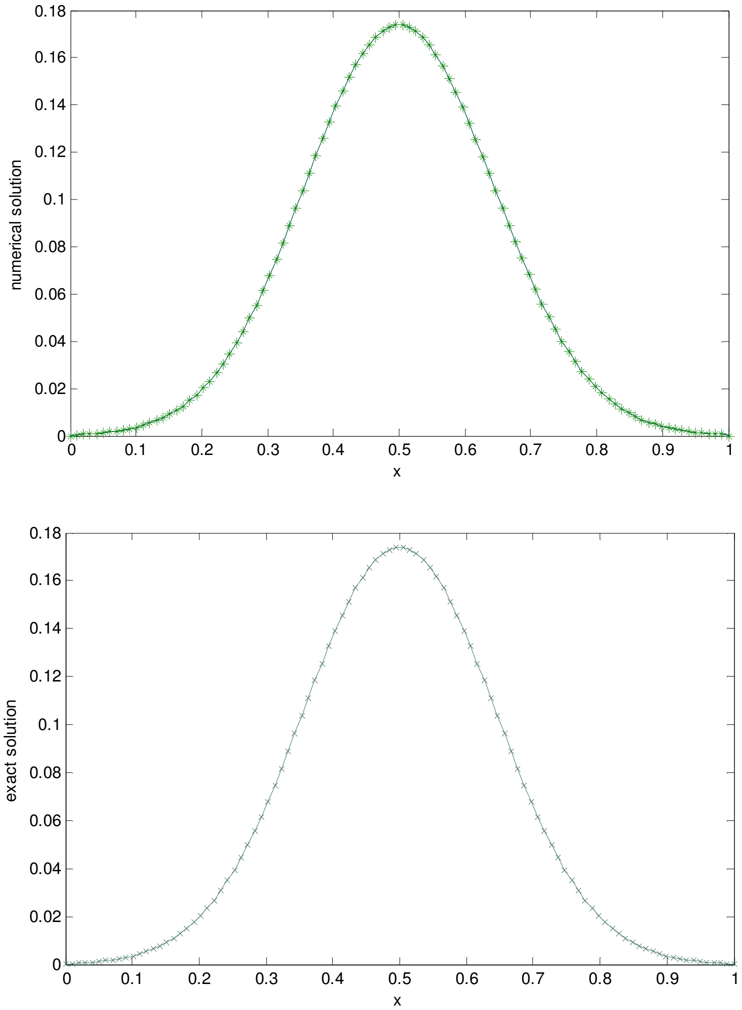


Figure 3: Comparision of exact and numerical solutions of Example 3.1 by Crank Nicolson method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 1.0$.

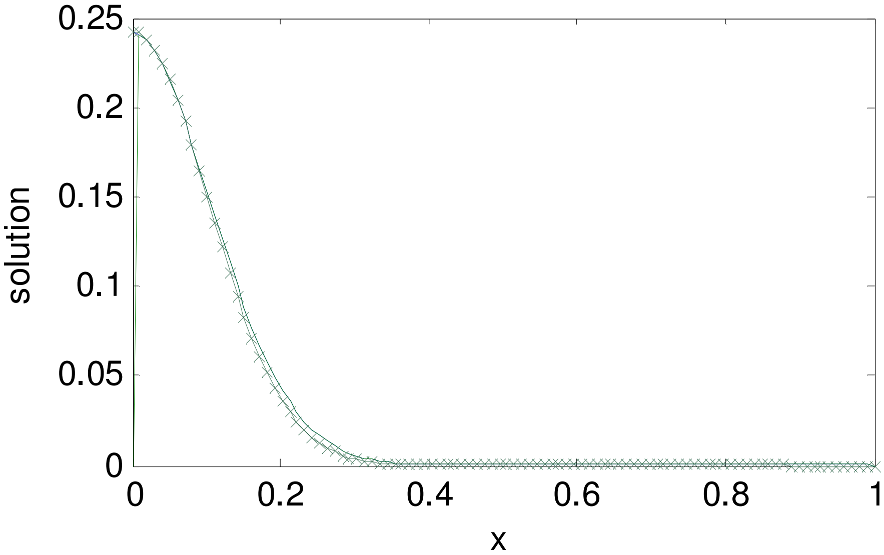


Figure 4: Comparison of exact and numerical solutions of Example 3.1 by Crank-Nicholson method for $N=100,\ D=0.01,\ u=1,\ \Delta t=0.0001\ at\ t=0.5$.

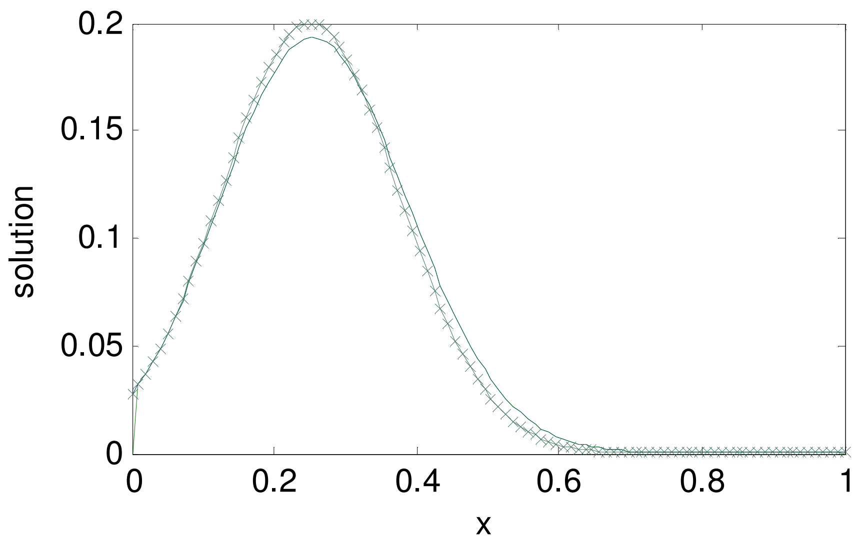


Figure 5: Comparison of exact and numerical solutions of Example 3.1 by Crank-Nicholson method for $N=100,\ D=0.01,\ u=1,\ \Delta t=0.0001\ at\ t=0.75$.

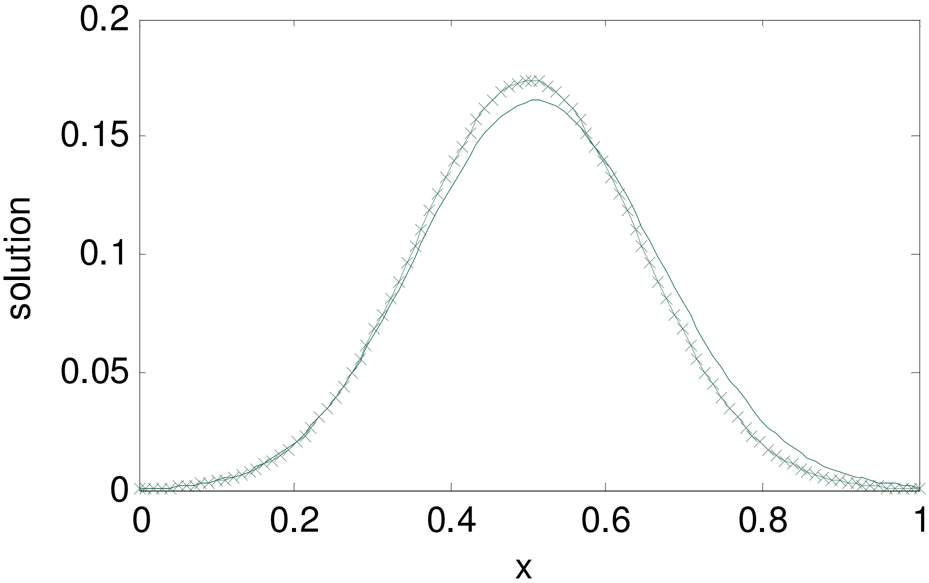


Figure 6: Comparison of exact and numerical solutions of Example 3.1 by Crank-Nicholson method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 1.0$.

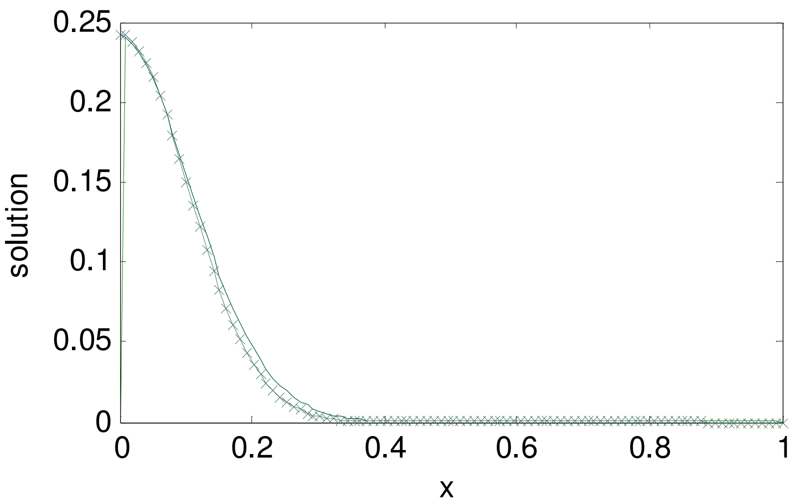


Figure 7: Comparison of exact and numerical solutions of Example 3.1 for implicit method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 0.5$.

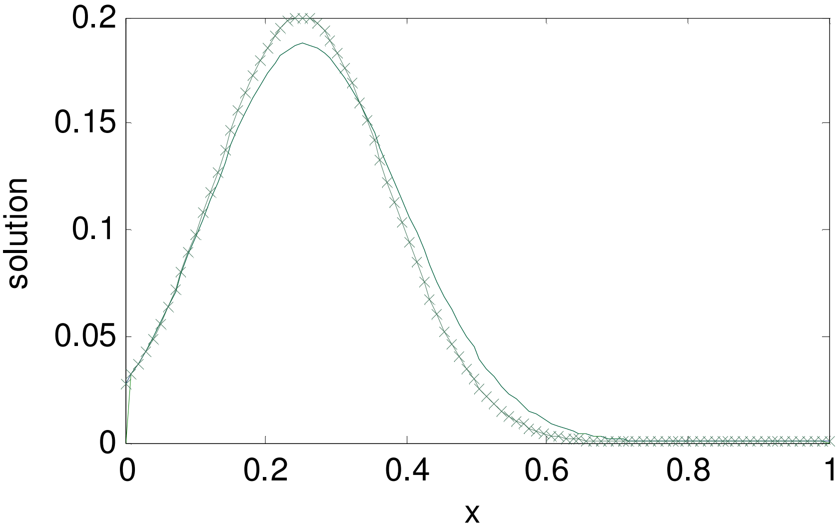


Figure 8: Comparison of exact and numerical solutions of Example 3.1 by implicit method, for

$N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 0.75$.

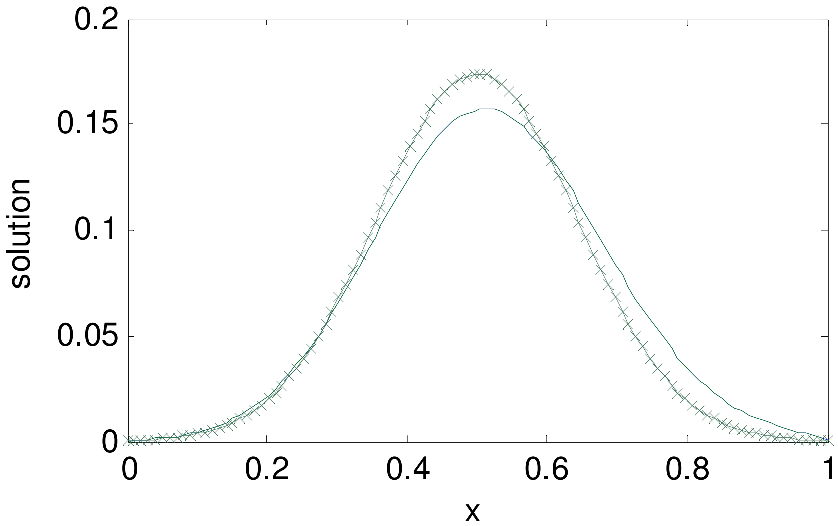


Figure 9: Comparison of exact and numerical solutions of Example 3.1 by implicit method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 1.0$.

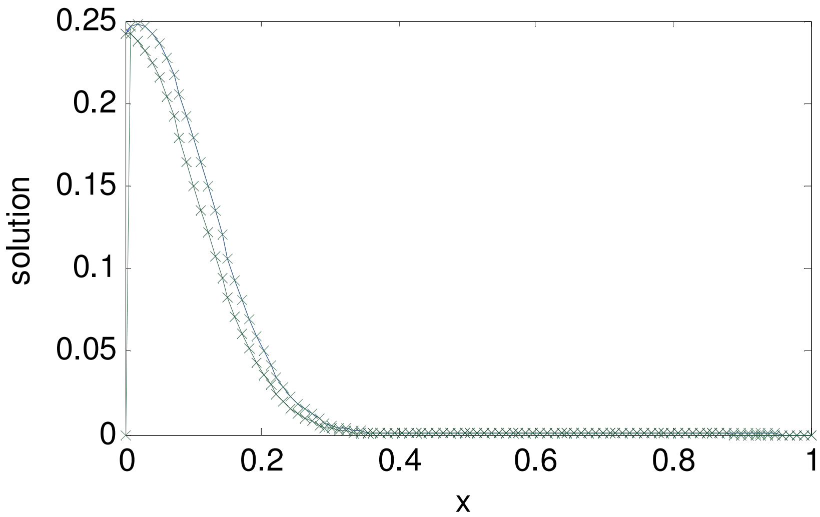


Figure 10: Comparison of exact and numerical solutions of Example 3.1 by Lax-Wendroff method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 1.0$.

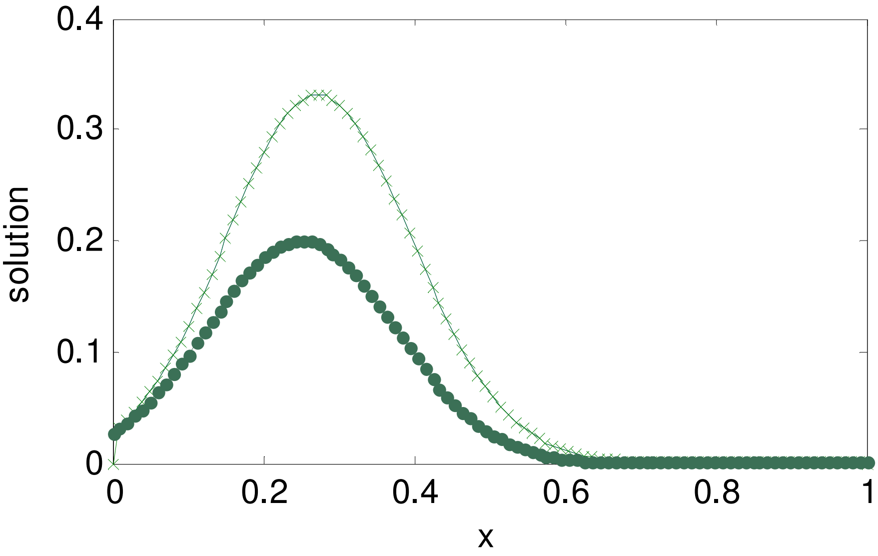


Figure 11: Comparison of exact and numerical solutions of Example 3.1 by Lax-Wendroff method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 0.75$.

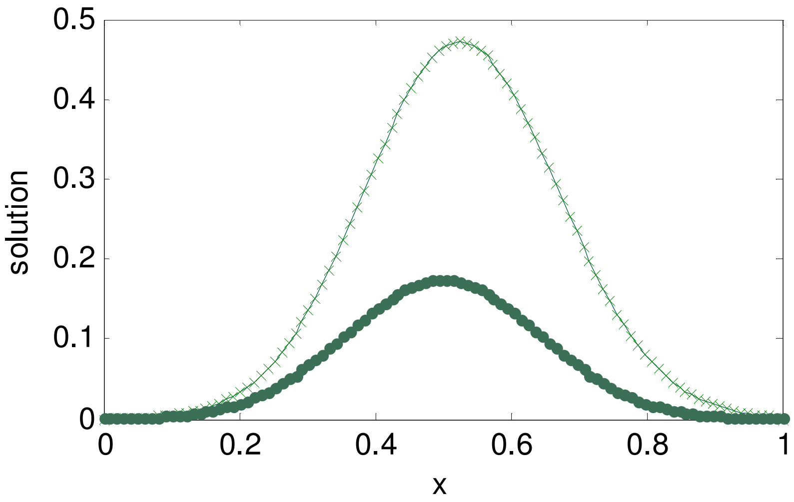


Figure 12: Comparison of exact and numerical solutions of Example 3.1 by Lax-Wendroff method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 1.0$.

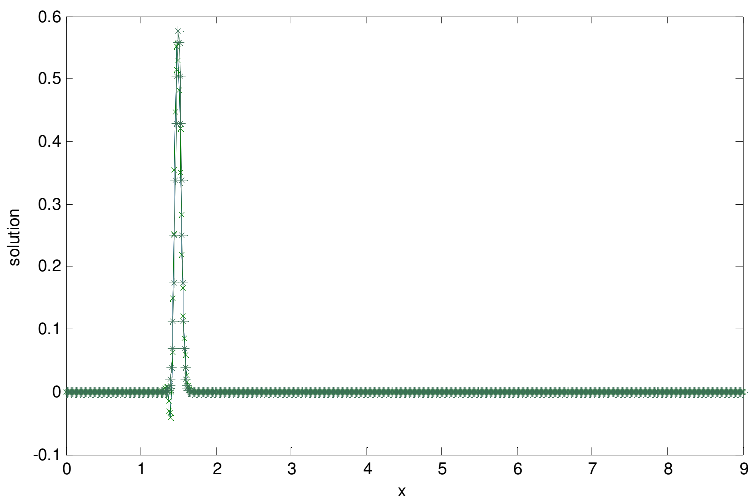


Figure 13: Comparision of exact and numerical solutions of Example 3.2 by explicit method for $N{=}100,\ D{=}0.01,\ u{=}1,\ \Delta t{=}0.0001\ at\ t{=}0.5$.

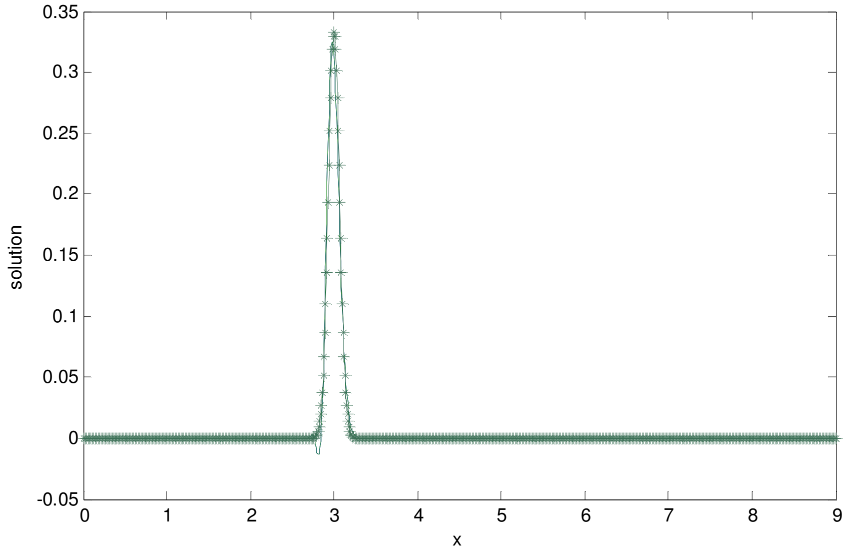


Figure 14: Comparision of exact and numerical solutions of Example 3.2 by explicit method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 2.0$.

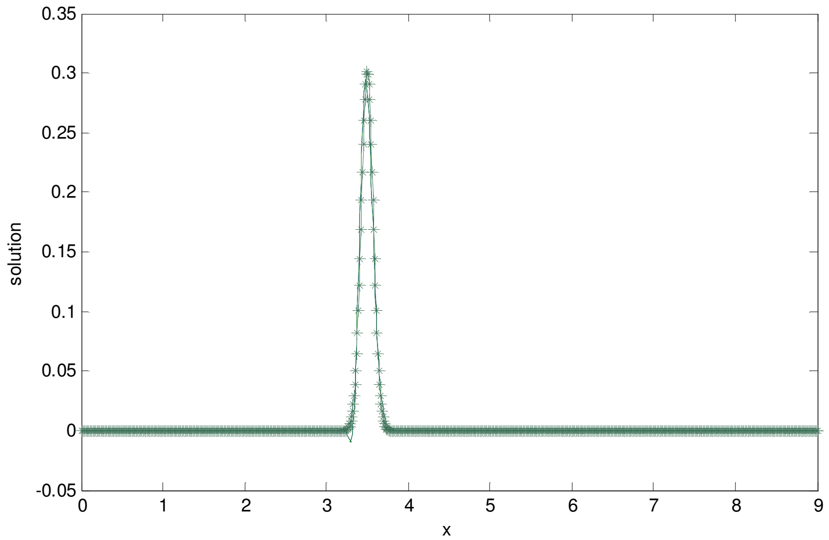


Figure 15: Comparision of exact and numerical solutions of Example 3.2 by explicit method for

$N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 2.5$.

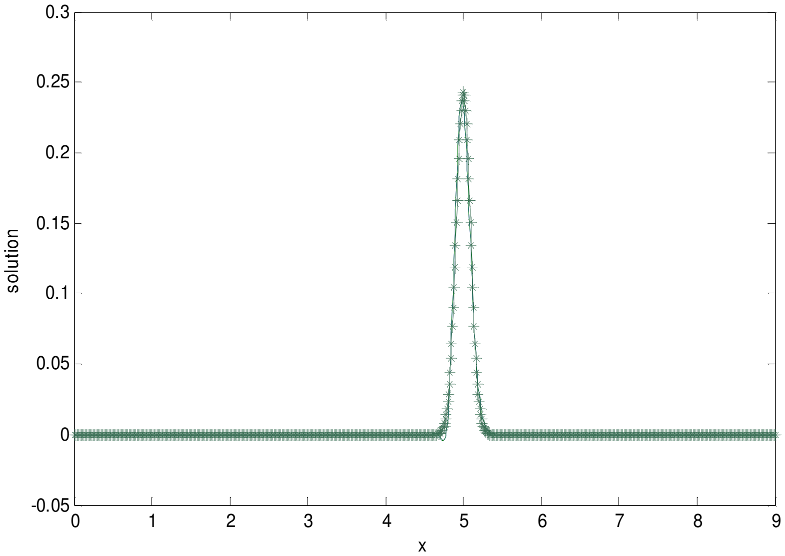


Figure 16: Comparision of exact and numerical solutions of Example 3.2 for explicit method for

$N{=}100,\ D{=}0.01,\ u{=}1,\ \Delta t{=}0.0001\ at\ t{=}4.0$.

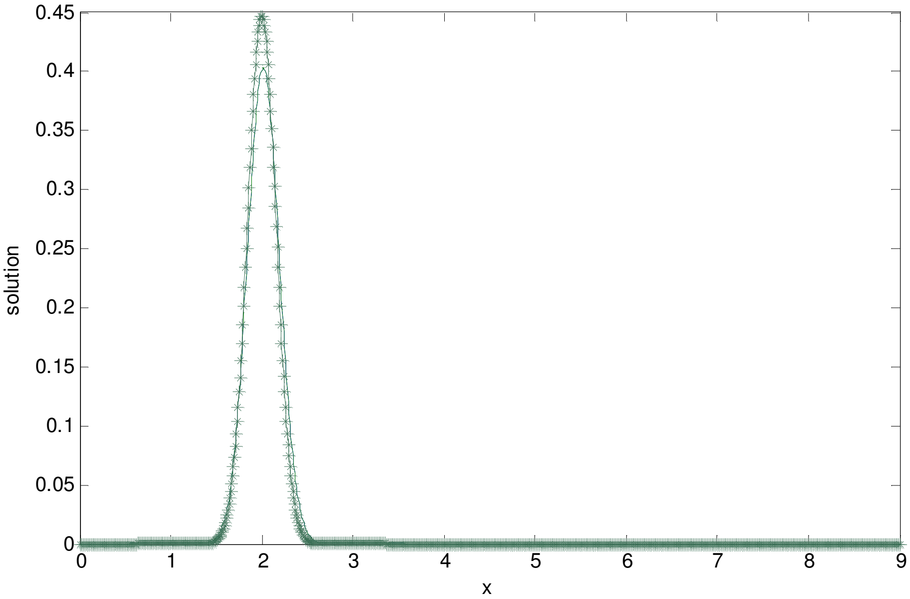


Figure17: Comparison of exact and numerical solutions of Example 3.2 by Crank-Nicholson method for $N = 100,\ D = 0.01,\ u = 1,\ \Delta t = 0.0001\ at\ t = 1.0$.

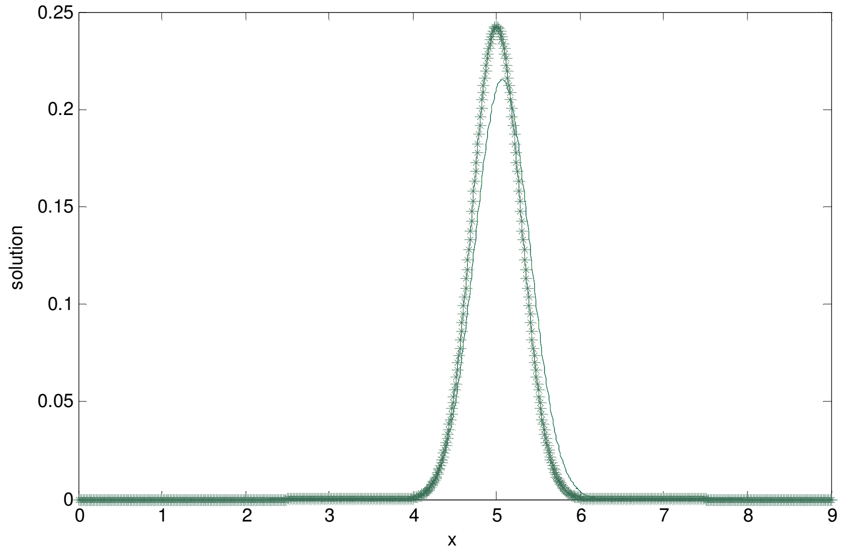


Figure 18: Comparison of exact and numerical solutions of Example 3.2 by Crank-Nicholson method for $N=100,\ D=0.01,\ u=1,\ \Delta t=0.0001\ at\ t=4.0$.

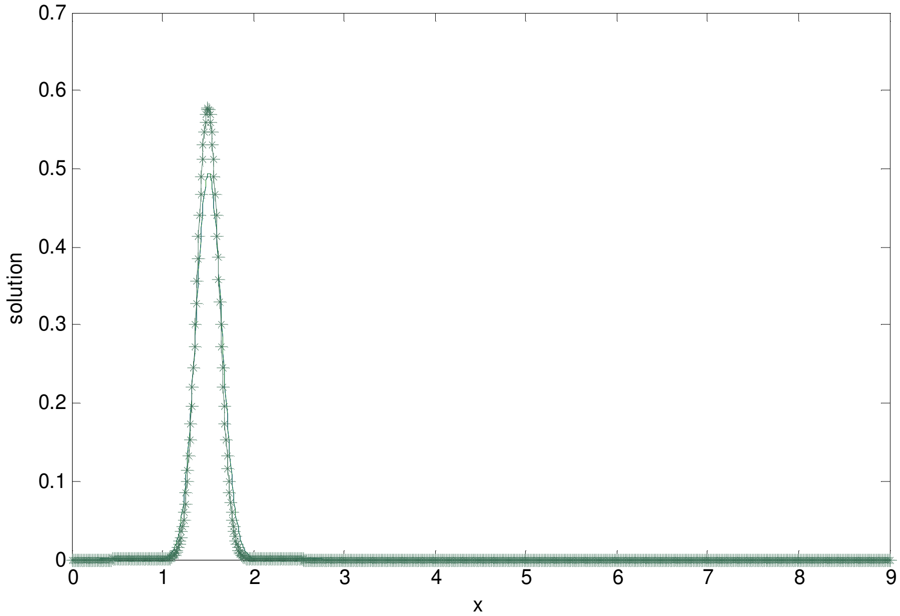


Figure 19: Comparison of exact and numerical solutions of Example 3.2 by implicit method for $N=100,\ D=0.01,\ u=1,\ \Delta t=0.0001\ at\ t=0.5$.

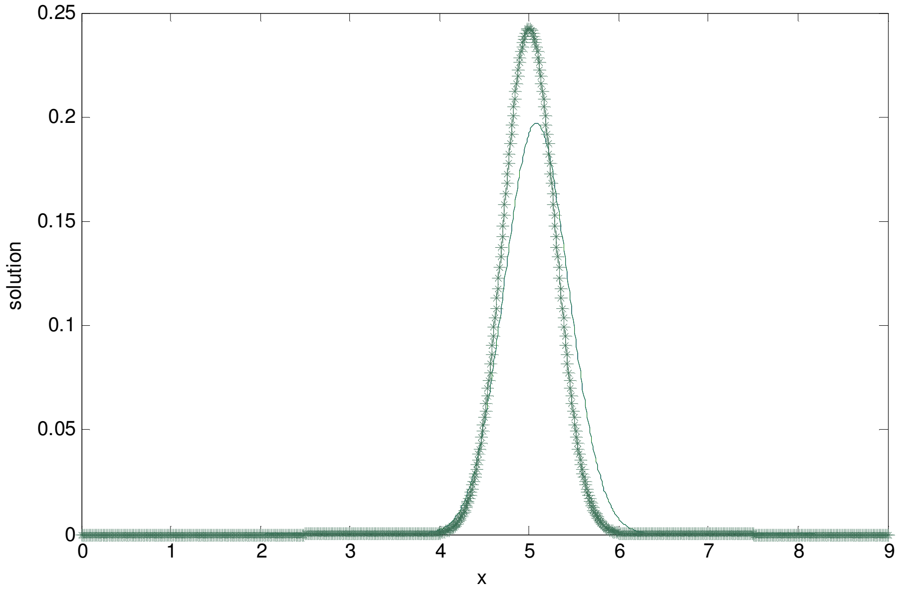


Figure 20: Comparison of exact and numerical solutions of Example 3.2 by implicit method for $N=100,\ D=0.01,\ u=1,\ \Delta t=0.0001\ at\ t=4.0$.

References

1 Anderson M. P., Bair E. S., The power of spreadsheet models. In: MODFLOW 2001 and other modeling odysseys proceedings, international ground water modeling center, Colorado school of mines, 2001 815–22.

2 Ayvaz M. T., Tuncan M., Karahan H., Tuncan A. An extended pressure application for transient seepage problems with a free surface, J Porous Media, 8(6) 2005 613–25.

3 Bear J., Bachmat Y., Introduction to modeling of transport phenomena. Dordrecht: Kluwer; 1990.

4 Bellman R., Kashef B.G., and J. Casti., Differential quadrature: A technique for the rapid solution of nonlinear partial differential equations, J. Comput. Phys., 10 1972 40-52.

5 Bert C.W. and Malik M., Differential quadrature method in computational mechanics, Appl. Mech. Rev., 49 (1) 1996 1-28.

6 Boris JB., Book DL., Flux corrected for transport algorithm that works, J Comput Phys., 1973 38–69.

7 Chen W., Zhong T.X., and Shu C., A Lyapunov formulation for efficient solution of the Poisson and convection-diffusion equations by differential quadrature method, J.Comput Phys., 141 (1) 1998 78-84.

8 Civan F., and Sliepcevich C.M., Differential quadrature for multi-dimensional problems,J. Math. Anal. Appl., 101 1984b 423-443.

9 Gentry RA., Martin RE, Daly BJ., An Eulerian differencing method for unsteady compressible flow problems, J Comput Phys., 1966 55–76.

10 Harten H., ENO schemes with subcell resolution, J Comp Phys., 1988 148–84.

11 Harten H., High resolution schemes for hyperbolic conservation laws, J Comp Phys., 1983 357–93.

12 Karahan H., Ayvaz MT., Transient groundwater modeling using spreadsheets, Adv Eng Software., 2005 374–84.

13 Karahan H., Ayvaz MT., Groundwater parameter estimation by optimization and dual reciprocity finite differences method, J Porous Media., 2005 211–23.

14 Kinzelbach W., Groundwater modeling: An introduction with sample programs in BASIC, Amsterdam: Elsevier; 1986.

15 Kowalik Z., Murty TS., Numerical modeling of ocean dynamics, Singapore: World Scientific.

16 Lam DCL., Computer modeling of pollutant transport in Lake Erie, Water Pollut., 1975 75–86.

17 -Leonard BP., A stable and accurate convective modeling procedure based on upstream formulation, Comput Meth Appl Mech Eng., 1979 19:59–98.

18 Leonard BP., Simple high accuracy resolution program for convective modeling of discontinuities, Int J Numer Meth Fluids., 1988 1291–318.

19 Li YS., Chen CP., An efficient split operator scheme for 2D advection diffusion equation using finite elements and characteristics, Appl Math Mod., 1989 248–53.

20 Malekzadeh P., and Karami G., Polynomial and harmonic differential quadrature methods for free vibration of variable thickness thick skew plates, Eng. Struc., 27 (2005) 1563-1574.

21 Michael GE., Rubin B., Total variation diminishing schemes and the two point upstream mobility weighting schemes, Soc Petr Eng Dallas, Techn Publ., 23947 1990.

22 Noye BJ., Tan HH., A third-order semi-implicit finite difference method for solving the one-dimensional convection–diffusion equation, Int J Numer Meth Eng., 1988 26:1615–29.

23 Noye BJ., Tan HH., Finite difference methods for the two dimensional advection diffusion equation, Int J Numer Meth Fluids., 1989 9:75–98.

24 Olsthoorn TN., The power of electronic worksheet: Modeling without special programs, Ground Water., 1985 23(3):381–90.

25 Remson I., Hornberger GM., Molz FJ. Numerical methods in subsurface hydrology, New York: Wiley Interscience., 1971.

26 Rubin B., Blunt MJ., High order implicit flux limiting schemes for Black oil simulation, Soc Petr Eng Dallas., Techn Publ., 21222 1990.

27 Sankaranarayanan S., Shankar NJ., Cheong HF., Three-dimensional finite difference model for transport of conservative pollutants, Ocean Eng., 1998 25(6):425–42.

28 Shu.C .,and Y. T. Chew., Fourier expansion-based differential quadrature and its application to Helmholtz eigenvalue problems, Commun., Numer., Methods Eng., 13 (8)(1997) 643-653.

29 Sobey RJ., Fractional step algorithm for estuarine mass transport, Int J Numer Meth Fluids., 1983 3:567–81.

30 Sommeijer BP., Kok J., Implementation and performance of the time integration of a 3D numerical transport model, Int J Numer Meth Fluids., 1995 21:349–67.

31 Spalding DB., A novel finite difference formulation for differential expression involving both first and second derivatives, Int J Numer Meth Fluids., 1972; 4:551–9.

32 Wang H. F., Anderson MP., Introduction to groundwater modeling: Finite Difference and Finite Element, London: Academic Press., 1982.

33 D. V. Widder., The Heat Equation., Academic Press., 1976.

34 J. R. Cannon., The One-Dimensional Heat Equation., Cambridge University Pres., 1984.

35 Ram Jiwari., R.C Mittal.,K K Sharma., A numerical scheme based on weighted average differential quadrature method for the numerical solution of Burgers' equation, Applied Mathematics and Computation., 219 (2013) 6680–6691.

36 Mittal R C., Ram Jiwari., Sharma K K., A numerical scheme based on differential quadrature method to solve time dependent Burgers' equation, Engineering Computations., 30 (1) (2013) 117-131.

37 Ram Jiwari., Haar wavelet quasilinearization approach for numerical simulation of Burgers' equation, Computer Physics Communications., 183 (2012) 2413-2423.

38 Mittal R C., Ram Jiwari., A differential quadrature method for solving Burgers'-type equation, International Journal of Numerical Methods for Heat and Fluid Flow., 22 (7), (2012), 880-895.

39 Ram Jiwari., Pandit S., Mittal R C., Numerical simulation of two-dimensional sine-Gordon solitons by differential quadrature method, Computer Physics Communications., 183 (2012) 600-616.

40 Ram Jiwari., Pandit S., Mittal R C., A Differential Quadrature Algorithm to Solve the Two Dimensional Linear Hyperbolic Telegraph Equation with Diriclet and Neumann Boun dary Conditions, Applied Mathematics and Computation., 218 (2012) 7279–7294.

41 Mittal R C., Ram Jiwari., Differential Quadrature Method for Numerical Solution of coupled viscous Burgers' equations, Int. J. for Comput. Methods in Eng. Science and Mech., 13 (2012), 1-5.

42 Sharma D., Ram Jiwari., Sheo Kumar., A comparative study of Modal matrix and finite elements methods for two point boundary value problems, Int. J. of Appl. Math., Mech., 8 (13) (2012), 29-45.

43 Ram Jiwari., Sapna Pandit., Mittal R C., A differential quadrature algorithm for the Numerical Solution of the Second-Order One dimensional Hyperbolic Telegraph Equation, Int J of Nonlinear Sciences, 13 (3) (2012), 259-266.

44 Ram Jiwari., Dinkar Shrma., Sheo Kumar., Numerical solutions of two point boundary value problems using Galerkin-Finite element method, Int J of Nonlinear Sciences., 13 (2)(2012), 204-210.

45 Mittal R C., Ram Jiwari., A Numerical Scheme for singularly perturbed Burger-Huxley Equation, J. Appl. Math. & Informatics., 29 (2011), No. 3-4, 813-829.

46 Mittal R C., Ram Jiwari., A Numerical Scheme for Some Nonlinear Differential Equations Models in Biology, Int. J. for Comput. Methods in Eng. Science and Mech., 12 (3), (2011), 134-140.

47 Mittal R C., Ram Jiwari., Numerical Study of Two-Dimensional Reaction-Diffusion Brusselator System, Appl. Math. Comput., 217 (12) (2011), 5404-5415.

48 Ram Jiwari., Dinkar Shrma., Sheo Kumar., Galerkin-Finite Element Method for the numerical solution of Advection-Diffusion equation, IJPAM., 70 (3) (2011), 389-399.

49 Mittal R C., Ram Jiwari., Numerical Study of Burger-Huxley Equation by Differential Quadrature Method, Int. J. of Appl. Math. and Mech., 5(8) (2009), 1-9.

50 Mittal R C., Ram Jiwari, Differential Quadrature Method for Two Dimensional Burgers' Equations, Int. J. for Comput. Methods in Eng. Science and Mech., 10 (2009), 450-459.

51 Mittal R C., and Ram Jiwari., Numerical Study of Fisher's Equation by using Differential Quadrature Method, Int. J. Information and systems Sciences., 5(1) (2008), 143-160.