Numerical and Dynamical Studies of Some Reaction-Diffusion Models

by

Zhen Chen

A thesis submitted to the
Faculty of the Graduate Studies of the
University of Manitoba in partial fulfillment
of the requirements for the degree of
Master of Science
Department of Mathematics
2002

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION PAGE

NUMERICAL AND DYNAMICAL STUDIES OF SOME REACTION-DIFFUSION MODELS

BY

Zhen Chen

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

Contents

Acknowledgements

I would like to express my appreciation to my supervisor, Professor A. B. Gumel, for his guidance and encouragement throughout the course of this study. I enjoyed the two years working with him.

I also thank the Department of Mathematics, University of Manitoba, for giving me the opportunity to further my study here by way of financial support.

I am thankful to all the professors, graduate students and support staff of the Mathematics Department for their support and encouragement.

Finally, I am indebted to my parents for their continuous support, love and kindness.

Abstract

This thesis focuses on the dynamical and numerical analyses of some quasi-linear partial differential equations (classified as reaction-diffusion equations) arising from the mathematical modelling of real-life phenomena in engineering, science, economics and medicine.

Local stability theory is used to investigate the qualitative behaviour of the associated ordinary differential systems (in the absence of diffusion and convection terms).

In order to circumvent the contrived numerical instabilities (such as oscillations, bifurcation, chaos or convergence to spurious solutions) associated with the use of standard explicit numerical methods (such as forward Euler and Runge-Kutta methods) for solving the aformentioned models, new and competitive finite-difference methods are constructed. These robust schemes, mostly "non-standard" in nature, have the same stability properties as the corresponding original ODE systems.

Chapter 1

Introduction

Non-linear partial differential equations (PDEs) of reaction-diffusion type arise in many application areas including developmental biology, ecology, physiology, finance and so on. With such a wide variety of applications, the choice of what to include in this thesis is a difficult one. Topics have been included both as typical models and as illustration of the robust numerical techniques which may be used to solve such systems.

Because of the non-linear and (at times) multi-dimensional nature of these models, their closed form solutions are usually difficult or impossible to find. Consequently, numerical schemes are employed to approximate these solutions.

The efficient numerical integration of systems of non-linear differential equations over long time intervals necessitates the use of time steps which are the largest possible bearing in mind accuracy and stability. Standard numerical methods (mostly explicit in nature), such as Euler and Runge-kutta (RK) methods, are widely used to solve systems of ordinary differential equations (ODEs) [19, 24, 34]. These methods, however, may not always be suited to solve the aforementioned non-linear problems. The reason being that they tend to induce contrived chaos and oscillations in the numerical results when certain parameter values and step sizes are used in the discretization (since these scheme-dependent instabilities are not a feature of the differential equations nor of the physical processes being modelled) [19, 24, 25].

To avoid contrived chaos, while retaining accuracy and stability, implicitly-designed methods should be preferred to explicit numerical methods. This is due to the well-known fact that implicit methods have superior stability

properties in comparison to standard explicit schemes [2, 19, 20, 24, 29]. The methods to be developed in this thesis, although implicit by construction, are applied explicitly. Therefore, while they enjoy superior stability properties in comparison to explicit schemes (because of their implicit nature), the new schemes also exploit the ease of implementation of explicit methods. These methods are adapted and extended for use to solve quasi-linear partial differential equations.

In this thesis, some robust finite-difference methods, mostly "non-standard" by construction, will be designed for the solution of some real-life models. Although our main purpose is to consider reaction-diffusion and reaction-diffusion-convection PDE systems, cases where there is no diffusion and convection (ODEs) will be considered first of all. Linearized dynamics analysis will be carried out to gain deeper insights into the qualitative behaviour of these associated ODE models. Competitive finite-difference methods, having the same qualitative features as the model, will then be constructed. These methods are then adapted and used to solve the corresponding reaction-diffusion and reaction-diffusion-convection cases. This idea will be employed in subsequent chapters (except Chapter 4) of this thesis in ways which permit the numerical solutions of the associated ODEs to be determined explicitly. In Chapter 4, the Black-Scholes reaction-diffusion-convection model, arising in option pricing, is solved *via* the *Method of Lines* semi-discretization technique involving the use of a sub-diagonal Padé approximant.

The numerical methods developed in this thesis are implemented using the C programming language on a UNIX platform at the University of Manitoba. Graphical and symbolic softwares such as Gnuplot and Maple are used to plot the figures and carry out symbolic computations. The thesis is written in Latex.

The outline of the thesis is as follows: some basic concepts about differ-

ential equations and finite difference methods will be reviewed in Chapter 2. This introductory material provides useful background needed for the more specific topics in the later chapters. In Chapter 3, a reaction-diffusion model with a cubic reaction term is considered. Its diffusion-free equivalent is studied first of all, where a competitive finite-difference method is designed and analysed.

In Chapter 4, the Black-Scholes reaction-diffusion-convection model is considered. An L_0-stable scheme, based on Padé approximation, is designed for its solution. A two-dimensional predator-prey reaction-convection-diffusion model with non-monotonic functional response is considered in Chapter 5. An SEIR model for disease transmission with a non-constant population size is studied in Chapter 6. In Chapters 3-6, the robust numerical schemes constructed are compared with standard numerical methods to verify their competitiveness. Conclusions are given in Chapter 7.

Chapter 2

Mathematical Tools

In this chapter, a brief review of some of the important mathematical concepts discussed or used in the thesis is given. The main components of the thesis are centred on the theories of partial differential equations, finite-difference methods, and dynamical systems. Detailed discussion on these topics can be found in the given references.

2.1 Partial Differential Equation

A differential equation in which there is only one independent variable is an ordinary differential equation. While a differential equation that contains at least two independent variables together with one or more partial derivatives of the dependent variable is called a partial differential equation. In general, a partial differential equation may be written in the form

$$f(x, y, \cdots, u, u_x, u_y, \cdots, u_{xx}, u_{yy}, \cdots) = 0, \tag{2.1}$$

involving several independent variables $x, y, \cdots$, an unknown function u of these variables, and the partial derivatives $u_x, u_y, \cdots, u_{xx}, u_{xy}, \cdots$ of this function. Equation (2.1) is considered in a suitable domain D of the n-dimensional space $\Re^n$ in the independent variables $x, y, \cdots$. We seek one or more functions $u = u(x, y, \cdots)$ which satisfy (2.1) identically in D. Such functions, if they exist, are called solutions of equation (2.1). From these many possible solutions, we attempt to select a particular one by introducing suitable additional conditions, called initial and/or boundary conditions.

The order of a partial differential equation is the order of the highest-ordered partial derivative appearing in the equation.

A partial differential equation is said to be linear if it is linear in the unknown function and all its derivatives with coefficients depending only on the independent variables; it is said to be quasilinear if it is linear in the highest-order derivative of the unknown function.

The general quasi-linear second-order partial differential equation in one dependent variable u, and two independent variables x, y, may be written as

$$Au_{xx} + Bu_{xy} + Cu_{yy} + Du_x + Eu_y + Fu = G, \tag{2.2}$$

where the coefficients are functions of x and y; and A, B and C do not vanish simultaneously. We shall assume that the function u and the coefficients are twice continuously differentiable in an open set in $\Re^2$. A PDE is said to be hyperbolic, parabolic, or elliptic at a point (x_0, y_0) accordingly as

$$B^2(x_0, y_0) - 4A(x_0, y_0)C(x_0, y_0) \tag{2.3}$$

is positive, zero or negative respectively. If this is true at all points in some domain, then the equation is said to be hyperbolic, parabolic, or elliptic in that domain.

This thesis will focus on a specific family of second-order quasi-linear parabolic partial differential equations that involve reaction and convection terms.

2.2 Reaction-Diffusion Equations

Reaction-diffusion systems are usually coupled systems (multiple number) of parabolic partial differential equations. They have a wide application in many areas. Some of them are models for pattern formation in morphogenesis, for predator-prey and other ecological systems, for conduction in nerves, for epidemics, for carbon monoxide poisoning, and for oscillating chemical reactions [4, 7, 8, 11, 12, 27, 28, 34, 36]. The equations in their simplest form

are written as

$$\mathbf{u}_t = \frac{\partial \mathbf{u}}{\partial t} = f(\mathbf{u}) + D\frac{\partial^2 \mathbf{u}}{\partial x^2} \tag{2.4}$$

where $\mathbf{u} = \mathbf{u}(x,t)$ is the vector of dependent variables, $f(\mathbf{u})$ is a non-linear vector-valued function of $\mathbf{u}$ (the reaction term), and D is the diffusion coefficient. The reaction term arise from any interaction between the components of $\mathbf{u}$. For example, $\mathbf{u}$ may be a vector of predator-prey models, when $f(\mathbf{u})$ represents the effect of predator-prey interactions, competition or symbiosis. The diffusion terms may represent molecular diffusion or some "random" movement of individuals in a population.

The reaction-diffusion system may be extended to reaction-diffusion-convection type given by

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{u}_t = f(\mathbf{u}) + D\frac{\partial^2 \mathbf{u}}{\partial x^2} + C\frac{\partial \mathbf{u}}{\partial x}, \tag{2.5}$$

where C is the convection coefficient [13, 32].

2.3 Reaction-Diffusion Equations and Dynamics

It is known that for reaction-diffusion systems (2.4) (involving reaction terms), the numerical treatment of the reaction terms is influential on the numerical results [29]. It is therefore important to consider the behaviour of the reaction term (given by $D = 0$ in (2.4)). The resulting ODE system, $\frac{d\mathbf{u}}{dt} = f(\mathbf{u})$, is often autonomous, non-linear, and therefore not always easy (or feasible) to solve analytically. Thus, in addition to carrying out dynamical analysis for such systems (to understand their qualitative behaviour), we need to design robust numerical procedures for their solutions.

Generally, real-life models contain system parameters. For different values of these parameters, the behaviour of the solutions can be qualitatively very different. Consequently, studying the changes of the nature of equilibrium solutions (equilibrium points) and branching of solutions when a parameter

passes through a certain value, called bifurcation, is crucial.

2.3.1 Stabiltiy of Linear Systems

Consider the linear differential equation, written as

$$\frac{d\mathbf{x}}{dt} = \dot{\mathbf{x}} = A\mathbf{x},\ \mathbf{x} \in R^n, \tag{2.6}$$

where A is a $n \times n$ matrix with constant coefficients. A general solution of (2.6), with the initial condition $\mathbf{x}(0) = \mathbf{x}_0$, is

$$\mathbf{x}(t) = e^{At}\mathbf{x}_0. \tag{2.7}$$

The system (2.6) has n eigenvalues $\lambda_1, \cdots, \lambda_n$ associated with the matrix A, and n eigenvectors $e_1, \cdots, e_n$, which may either be real or complex. More specifically, a general solution to (2.6) can be obtained by taking a linear superposition of the linear independent solutions determined by an exponential motion along the real eigenvectors or a spiralling motion in a plane spanned by the real and imaginary parts of the complex eigenvectors. The stabitilty properties of the system are then determined by the signs of the real parts of the eigenvalues as follows [38]:

(1) if all the real parts are negative, then the solution is asymptotically stable, and the motion will converge towards the origin.

(2) if at least one real part is positive, then the solution is unstable.

(3) if the eigenvalues are distinct and their real parts are all zero, then the solution is not asymptotically stable.

2.3.2 Stabiltiy of Non-linear Systems

In the case of the diffusion-free model, we typically need to investigate a first order autonomous non-linear ODE system of the form

$$\frac{d\mathbf{x}}{dt} = f(\mathbf{x}),\ \mathbf{x} \in R^n, \tag{2.8}$$

where $f(\mathbf{x})$ are non-linear functions.

For non-linear differential equations such as (2.8), one can only make very general statements concerning the behaviour of their solutions. The best way to start is the determination of possible equilibrium points, which are stationary solutions of (2.8), determined by setting its right hand side to zero. That is,

$$f(\mathbf{x}) = 0. \tag{2.9}$$

It should be pointed out that, even this (calculation of equilibrium solutions) can sometimes be a complicated task. However, once the equilibrium solutions are obtained, we can then use the principle of linear stability to characterize the behaviour of the system near such (equilibrium) point. The time evolution of a small perturbation ξ around an equilibrium point $\mathbf{x}_0$ is $\xi = \mathbf{x} - \mathbf{x}_0$. Thus,

$$\begin{aligned} \frac{d\xi}{dx} &= \dot{\mathbf{x}}, \\ &= f(\mathbf{x}), \\ &\approx f(\mathbf{x}_0) + J(\mathbf{x} - \mathbf{x}_0), \\ &= J\xi, \end{aligned}$$

where a first order Taylor expansion of $f(\mathbf{x})$ about $\mathbf{x}_0$ is used to obtain a linear approximation of $\dot{\xi}$. The matrix J has entries

$$J_{ij} = \frac{\partial f_i}{\partial c_j}, \tag{2.10}$$

and is called the Jacobian of the given system. This matrix determines the local behaviour of the solutions around the equilibrium point.

The advantage of the linearized system is that its behaviour turns out, in most cases, to provide very significant information about the behaviour of the original non-linear system. The stability of the linearized model around the equilibrium points are analogous to that described in subsection 2.3.1. In particular, for the cases (1) and (2), there exists a topological equivalence between the linearized system (2.6) and the full non-linear system (2.8). That is, if J has no eigenvalues with zero real part, then the stability is determined by the linearization [38]. On the other hand, if any of the eigenvalues has zero real part, then the stability cannot be determined by the linearization [38] and, consequently, the equilibrium point is structurally unstable (since its properties are likely to change qualitatively on the action of an arbitrarily small disturbance).

An equilibrium solution is said to be globally-asymptotically stable if all solution trajectories are attracted to it [38]. The Lyapunov theory is widely used to establish global stability of equilibrium solutions by constructing suitable Lyapunov functions [38].

Lyapunov Function

Definition. Let $\dot{\mathbf{x}} = f(\mathbf{x})$ be a differential equation on R^n and a unique equilibrium point p. A real valued C^1 function L is called a weak Lyapunov function on an open neighbourhood U of p provided $L(\mathbf{x}) > L(p)$ and $\dot{L}(\mathbf{x}) \leq 0$ for all $\mathbf{x} \in U \backslash \{p\}$. The function L is called a Lyapunov function on an open neighbourhood U of p (or strong, strict Lyapunov function) provided it is a weak Lyapunov function which also satisfies $\dot{L}(\mathbf{x}) < 0$ for all $\mathbf{x} \in U \backslash \{p\}$.

Theorem If L is a (strict) Lyapunov function on the neighborhood U of p for the differential equation, then p is globally asymptotically stable [41].

2.3.3 Bifurcation Theory

If the non-linear ODE depends on a parameter μ, then we have

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mu),\ \mu \in R. \tag{2.11}$$

One can say that as long as all the eigenvalues λ_i of J satisfy $Re(\lambda_i) \neq 0$, the equilibrium point $\mathbf{x}_0$ will be a smooth function of μ (Implicit Function Theorem [41]). But as soon as at least one eigenvalues λ_i satisfies $Re(\lambda_i) = 0$ for some value of μ_c of μ, there exists a bifurcation or singular point [38]. In addition, the exchange of stability of x_0 requires the criteria

$$\frac{d\lambda_i}{d\mu} \mid_{\mu_c} \neq 0. \tag{2.12}$$

It is important to note that bifurcations are associated with non-linear equations only.

There are several types of bifurcations (with qualitatively different features) namely: saddle-node bifurcation, pitchfork bifurcation, period doubling bifurcation, Hopf-bifurcation etc. The reader is referred to [38] for detailed discussions on non-linear dynamics.

2.4 Finite-Difference Methods

As stated earlier, analytical methods for differential equations are limited to certain special forms of equations. It is difficult to find analytical solutions for most non-linear differential equations. Numerical methods, on the other hand, have (in general) no limitations to only "special" forms.

The fundamental principle of numerical methods is the reduction of a differential equation to an approximation in terms of algebraic equations [2, 19, 24, 29, 35]. The finite-difference method proceeds by first identifying a finite number of discrete points within the domain of interest. These points are called nodes, and it is at these locations that approximations to the true

solution are computed. This is the discretization step. Next, the differential operator at the nodal points are replaced by discrete difference approximations. This is the approximation step which produces a set of algebraic equations with discrete nodal values as unknowns. The final step involves solving the algebraic system of equations. In this manner, a discrete approximation to the solution of the original differential equation is obtained.

Different approximations to derivatives lead to different finite-difference methods. In general, finite-difference methods are categorized into explicit and implicit methods. Although explicit methods are easier to be implemented (because they do not involve the solution of algebraic linear equations at each step size), they (standard explicit methods) are known to have very poor stability properties [2, 24].

2.5 Finite-Difference Methods for ODEs

To explore all the finite difference methods used in the literature is a daunting task. Instead, I will concentrate on the standard Euler and Runge-Kutta methods applied to first-order ODE systems. These two traditional methods are used extensively in this thesis for comparison with the new robust methods I have developed. These novel methods, standard or "non-standard", can be extended to systems of higher-order differential equations.

2.5.1 Forward Euler Method

Consider the first-order equation of the form

$$\frac{dy}{dx} = y' = f(x, y),\ y(x_0) = y_0. \tag{2.13}$$

The method is based on expanding the solution $y(x)$ of equation (2.13) in Taylor series about the point x_0. Suppose we choose a step size h (the interval

beyond x_0 where we evaluate the series) small enough that we may truncate the series after the first derivative term. Then,

$$y(x_0 + h) = y(x_0) + hy'(x_0) + \frac{y''(\xi)}{2}h^2,\ x_0 < \xi < x_0 + h. \tag{2.14}$$

In using this equation, the value of $y(x_0)$ is given by the initial condition and $y'(x_0)$ is evaluated from $f(x_0, y_0)$, given by the differential equation $dy/dx = f(x, y)$. It is of course necessary to use this method iteratively, advancing the solution to $x = x_0+2h$ after $y(x_0+h)$ has been found, then to $x = x_0+3h$, etc. Adopting a subscript notation for the successive y-values, $x_n = x_0+nh,\ y_n = y(x_n)$, and representing the error by the order relation, we may write the algorithm for the Euler method as:

$$\begin{aligned} y_{n+1} &= y_n + hy'_n + O(h^2), \\ x_n &= x_0 + nh,\ y_n = y(x_n). \end{aligned} \tag{2.15}$$

2.5.2 Runge-Kutta Methods

A further advance in accuracy can be achieved by using a group of methods due to the German mathematicians Runge and Kutta (see [19]). In particular, the fourth-order Runge-Kutta method (RK4) is widely used to integrate differential equations. Using the notations of the last subsection, the fourth-order Runge-Kutta method is given by,

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), \tag{2.16}$$

where

$$\begin{aligned} k_1 &= hf(x_n, y_n), \\ k_2 &= hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1), \\ k_3 &= hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2), \\ k_4 &= hf(x_n + h, y_n + k_3). \end{aligned}$$

2.6 Finite-Difference Methods for Parabolic PDEs

2.6.1 *Method of Lines* Semi-discretization

The *Method of Lines* (MOL) semi-discretization approach is often employed to solve parabolic and hyperbolic partial differential equations [15, 17, 20]. Consider the one dimensional heat equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \quad ; \; X_0 < x < X_1 \; , \; t > 0, \tag{2.17}$$

together with the boundary conditions

$$u(X_0, t) = u(X_1, t) = 0 \; , \; t > 0, \tag{2.18}$$

and initial conditions

$$u(x, 0) = f(x) \; ; \; X_0 \leq x \leq X_1 \; , \tag{2.19}$$

where $f(x)$ is a given continuous function of x and it is not necessarily specified that $f(X_0) = 0$ nor that $f(X_1) = 0$. Thus, discontinuities between initial conditions and boundary conditions may occur. The interval $X_0 \leq x \leq X_1$ is divided into $N+1$ subintervals each of width h so that $(N+1)h = X_1 - X_0$ and the time variable t is incremented in steps of length ℓ. The open region Ω and its boundary $\partial\Omega$ consisting of the lines $x = X_0$, $x = X_1$ and the axis $t = 0$ have thus been covered by a rectangular mesh, the mesh points having co-ordinates $x_m = X_0 + mh$ and $t_n = n\ell$ where $m = 0, 1, ..., N+1$ and $n = 0, 1, 2...$. The space derivative in (2.17) is replaced by its second-order central-difference approximation given by

$$\frac{\partial^2 u}{\partial x^2} = h^{-2}[u(x-h, t) - 2u(x, t) + u(x+h, t)] + O(h^2). \tag{2.20}$$

Equation (2.17) together with the boundary condition (2.18) are applied to

all N interior mesh points at time level $t = n\ell$ $(n = 0, 1, 2, ...)$ to give

$$\begin{aligned}
\frac{dU_1}{dt} &= h^{-2}[U_0(t) - 2U_1(t) + U_2(t)] \\
\frac{dU_2}{dt} &= h^{-2}[U_1(t) - 2U_2(t) + U_3(t)] \\
\frac{dU_3}{dt} &= h^{-2}[U_2(t) - 2U_3(t) + U_4(t)] \\
&\vdots \\
\frac{dU_m}{dt} &= h^{-2}[U_{m-1}(t) - 2U_m(t) + U_{m+1}(t)] \\
&\vdots \\
\frac{dU_{N-1}}{dt} &= h^{-2}[U_{N-2}(t) - 2U_{N-1}(t) + U_N(t)] \\
\frac{dU_N}{dt} &= h^{-2}[U_{N-1}(t) - 2U_N(t) + U_{N+1}(t)]
\end{aligned} \tag{2.21}$$

where U_m(t) is the (semi-discrete) solution of (2.17) when $x = x_m$ for any $t > 0$. The equations in (2.21) may be written in the form

$$\frac{d\mathbf{U}(t)}{dt} = A\mathbf{U}(t) \;\; ; \;\; \mathbf{U}(0) = \mathbf{f}, \tag{2.22}$$

where $\mathbf{U}(t) = [U_1(t), U_2(t), ..., U_N(t)]^T$, T denoting transpose, is the vector of approximations to the solution of the PDE (2.17) at $t = t_n = n\ell$ and $\mathbf{f}$ is the vector of initial conditions. The Nth-order tridiagonal matrix A, given by,

$$A = h^{-2}\begin{bmatrix}
-2 & 1 & & & & \mathbf{0} \\
1 & -2 & 1 & & & \\
 & 1 & -2 & 1 & & \\
 & & \ddots & \ddots & \ddots & \\
 & & & 1 & -2 & 1 \\
\mathbf{0} & & & & 1 & -2
\end{bmatrix}, \tag{2.23}$$

has real negative eigenvalues [17, 19]

$$\lambda_s = -4h^{-2}\sin^2[s\pi/\{2(N+1)\}],\ s = 1, 2, ..., N. \tag{2.24}$$

This method of transforming the initial/boundary-value problem (2.17) into the initial-value problem (2.22) is known as the *Method of Lines* (see [19, 20]).

The solution of the system of first-order, linear, ordinary differential equations (2.22) is

$$\mathbf{U}(t) = exp(tA)\mathbf{f}, \tag{2.25}$$

which satisfies the recurrence relation

$$\mathbf{U}(t+\ell) = exp(\ell A)\mathbf{U}(t) \quad ; \quad t = 0, \ell, 2\ell, 3\ell, ... \tag{2.26}$$

2.6.2 A Family of Numerical Methods

The development of a family of numerical methods will be based on making rational approximations to the matrix exponential term in (2.26). A commonly used family of approximants is [33]

$$exp(\ell A) \approx [I - \theta\ell A]^{-1}[I + (1-\theta)\ell A], \tag{2.27}$$

in which $0 \leq \theta \leq 1$ is a parameter and I is the identity matrix of order N. Putting (2.27) in (2.26) leads to the family of methods

$$(I - \theta\ell A)\mathbf{U}(t+\ell) = [I + (1-\theta)\ell A]\mathbf{U}(t)\ . \tag{2.28}$$

The three well known methods for solving (2.17) are obtained by setting $\theta = 0, 1$ and $\frac{1}{2}$.

2.6.2.1 An Explicit Method

Writing $\theta = 0$ in (2.28) gives

$$\mathbf{U}(t+\ell) = (I + \ell A)\mathbf{U}(t), \tag{2.29}$$

which, when applied to the general mesh point (x_m, t_n), gives the four-point, two-time-level explicit scheme

$$U_m^{n+1} = pU_{m-1}^n + (1-2p)U_m^n + pU_{m+1}^n \tag{2.30}$$

where $p = \dfrac{\ell}{h^2}$, $m = 1, 2, ..., N$ and $n = 0, 1, ...$, and U_m^n is the fully discrete solution of $\{(2.17), (2.18), (2.19)\}$ at the general mesh point. Clearly, U_m^n is an approximation to $u_m^n = u(x_m, t_n)$, the theoretical solution of $\{(2.17), (2.18), (2.19)\}$. This method is $O(h^2 + \ell)$ and requires $p = \ell/h^2 \leq 1/2$ for stability [33].

2.6.2.2 A Fully-Implicit Method

Writing, next, $\theta = 1$ in (2.28) gives

$$(I - \ell A)\mathbf{U}(t+\ell) = \mathbf{U}(t), \tag{2.31}$$

which, when applied to the general mesh point (x_m, t_n), gives the four-point, two-time-level, fully implicit scheme

$$-pU_{m-1}^{n+1} + (1+2p)U_m^{n+1} - pU_{m+1}^{n+1} = U_m^n. \tag{2.32}$$

This $O(h^2+\ell)$ method requires the application of a tridiagonal solver at each time step. Furthermore, it is L_0-stable (and, therefore, does not suffer any stability restriction) [33].

2.6.2.3 The Crank-Nicolson Method

Putting $\theta = \frac{1}{2}$ in (2.28) gives the well-known Crank-Nicolson method [19, 33]

$$\left(I - \frac{1}{2}\ell A\right)\mathbf{U}(t+\ell) = \left(I + \frac{1}{2}\ell A\right)\mathbf{U}(t) \tag{2.33}$$

which, on application to the general mesh point (x_m, t_n), leads to the six-point, two-time-level, implicit scheme

$$-\frac{p}{2}U_{m-1}^{n+1} + (1+p)U_m^{n+1} - \frac{p}{2}U_{m+1}^{n+1} = \frac{p}{2}U_{m-1}^{n} + (1-p)U_m^{n} + \frac{p}{2}U_{m+1}^{n}. \tag{2.34}$$

This $O(h^2+\ell^2)$ method requires $r = \dfrac{\ell}{h} < \dfrac{X_1 - X_0}{\pi}$ for stability for problems with discontinuities between initial and boundary conditions.

In summary, the best method (in terms of stability) associated with the use of (2.27) in (2.26) is the fully-implicit method (arising from using $\theta = 1$ in (2.27). Although the fully implicit method has superb stability property (due to its L_0-stable nature), it is only first-order accurate in time. Thus, there is a clear need to design L_0-stable numerical schemes for solving PDEs to higher-order of accuracy.

2.6.3 Padé Approximation

In order to achieve higher-order accuracy in time and desirable stability properties (L_0-stability), Padé approximants [15, 30, 37, 43] are often used to approximate the exponential in (2.26).

A Padé approximation, $R_{m,k}(Z)$, to an analytic funtion $f(Z)$ in a region of the complex plane containing the origin $Z = 0$ is given by

$$R_{m,k}(Z) = \frac{P_k(Z)}{Q_m(Z)}, \tag{2.35}$$

where $P_k(Z)$ and $Q_m(Z)$ are polynomials in Z of degrees k and m respectively having leading coefficients unity, and k, m are non-negative integers [37]. For

each pair of m and k, $P_k(Z)$ and $Q_m(Z)$ are those polynomials for which the Taylor series expansions of $R_{m,k}(Z)$ about the origin agrees with the Taylor series of $f(Z)$ for as many terms as possible [37].

Because the higher order Padés have complex poles, their use in approximating matrix exponentials enable the solution of the associated models to be computed in parallel (using complex arithmetic) *via* a partial-fraction splitting technique [15, 17].

In particular, the explicit method ($\theta = 0$ in (2.28)), fully-implicit method ($\theta = 1$ in (2.28)) and Crank-Nicolson method ($\theta = 1/2$ in (2.28)) correspond to the $(0,1)$, $(1,0)$ and $(1,1)$ Padé approximations to $exp(\ell A)$ in (2.26) respectively.

2.6.4 A_0- and L_0-Stability

The methods discussed earlier using MOL are of the form

$$\mathbf{U}(t+\ell) = R(\ell A)\mathbf{U}(t), \tag{2.36}$$

where $R(\ell A)$ is known as the *amplification matrix.* For example, the amplification matrices for the explicit, fully-implicit and Crank-Nicolson methods are $(I + \ell A)$, $(I - \ell A)^{-1}$ and $(I - \frac{1}{2}\ell A)^{-1}(I + \frac{1}{2}\ell A)$ respectively.

Definition: If $\mathbf{U}(t+\ell) = R(\ell A)\mathbf{U}(t)$, where A is a symmetric matrix of order N with negative, real eigenvalues (λ_s, $s = 1, 2, \ldots$), then the resulting numerical method is said to be A_0-stable if

$$|R(\ell A)| \leq 1 \text{ or } |R(\ell\lambda_s)| \leq 1 \text{ or } |R(-z_s)| \leq 1, \tag{2.37}$$

where $R(\ell A)$ or $R(-z_s)$ is the amplification symbol of the numerical method and $-z_s = -\ell\lambda_s > 0$ [19, 33].

Definition: An A_0-stable method for which, additionally,

$$\lim_{z\to\infty} R(-z) = 0, \tag{2.38}$$

is called an L_0-stable method.

An L_0-stable method has the advantage of damping all components of the numerical solution to zero as $t \to \infty$. For the Crank-Nicolson method, for instance,

$$R(-z) = \frac{1 - \frac{1}{2}z}{1 + \frac{1}{2}z}, \tag{2.39}$$

so that,

$$|R(-z)| = \left|\frac{1 - \frac{1}{2}z}{1 + \frac{1}{2}z}\right| < 1. \tag{2.40}$$

Thus, the Crank-Nicolson method is A_0-stable. However, because

$$\lim_{z\to\infty} R(-z) = \lim_{z\to\infty} \frac{1 - \frac{1}{2}z}{1 + \frac{1}{2}z} = -1 \neq 0, \tag{2.41}$$

the Crank-Nicolson method is not L_0-stable.

It is easy to show that the fully-implicit scheme (2.32) is L_0-stable. Generally, an important property of a sub-diagonal (m, k) Padé approximation, where $m > k$, is that it leads to L_0-stability. Thus, all frequency components and instabilities are not propagated from one step to the next. When $m = k$ (diagonal Padé approximation), on the other hand, the corresponding numerical method is only A_0-stable.

2.6.5 Local Truncation Errors

The *local truncation error* of a finite-difference scheme at a mesh point $(mh, n\ell)$ is the difference between the finite-difference scheme and the differential equation it is replacing. The difference between the theoretical

solution of the differential and difference equations is called the local discretization error and is represented at the mesh point $(mh, n\ell)$ by $u_m^n - U_m^n$ where $u_m^n = u(mh, n\ell)$ is the theoretical solution of the partial differential equation.

A difference equation is said to be consistent if the local truncation error tends to zero as the mesh is refined. That is, as $h, \ell \to 0$ simultaneously. Consider, by way of example, the general finite-difference method arising from the family (2.28). It can be seen that the method takes the form

$$\frac{U_m^{n+1} - U_m^n}{\ell} = h^{-2}[\theta(U_{m-1}^{n+1} - 2U_m^{n+1} + U_{m+1}^{n+1}) + (1-\theta)(U_{m-1}^n - 2U_m^n + U_{m+1}^n)] \tag{2.42}$$

from which it follows that the left hand side approximates $\dfrac{\partial u}{\partial t}$ while the right hand side gives a weighted approximation to $\dfrac{\partial^2 u}{\partial x^2}$. The local truncation error of (2.42) is given by

$$\begin{aligned} L(x,t) = & \frac{u(x,t+\ell) - u(x,t)}{\ell} \\ & - \frac{\theta\{u(x-h,t+\ell) - 2u(x,t+\ell) + u(x+h,t+\ell)\}}{h^2} \\ & - \frac{(1-\theta)\{u(x-h,t) - 2u(x,t) + u(x+h,t)\}}{h^2} \\ & - \left(\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2}\right) . \end{aligned} \tag{2.43}$$

The Taylor series expansion of (2.43) about $u(x,t)$ leads to

$$L(x,t) = \left(-\frac{1}{12}h^2\frac{\partial^4 u}{\partial x^4} + C_q \ell \frac{\partial^2 u}{\partial t^2}\right) + C_{q+1}\ell^2 \frac{\partial^3 u}{\partial t^3} + ... \tag{2.44}$$

where $C_q = C_2 = \dfrac{1}{2} - \theta$ and $C_{q+1} = C_3 = \dfrac{1}{6} - \dfrac{\theta}{2}$. Note that, $\dfrac{\partial u}{\partial t} = \dfrac{\partial^2 u}{\partial x^2} \Rightarrow$ $\dfrac{\partial^2 u}{\partial t^2} = \dfrac{\partial^3 u}{\partial x^2 \partial t}$ and $\dfrac{\partial^3 u}{\partial t^3} = \dfrac{\partial^4 u}{\partial x^2 \partial t^2}$. The term in the parentheses in (2.44) is called the *principal part* of the local truncation error. The component $-\dfrac{1}{12}h^2\dfrac{\partial^4 u}{\partial x^4}$ is due to the space discretization and the use of (2.20) in (2.17). This term will always appear in the principal part of the local truncation error of every method arising from (2.28) in which $0 \le \theta \le 1$.

The term C_q in (2.44) is the *time error constant* of the method. For the explicit scheme (2.30), θ is set to zero giving the time error constant $C_2 = \frac{1}{2}$ while for the fully implicit scheme (2.32) $\theta = 1$ giving $C_2 = -\frac{1}{2}$; thus the two methods are both first-order accurate in time. As for the Crank-Nicolson method (2.34), $\theta = \frac{1}{2}$ causing C_2 to vanish and the time error constant is consequently given by $C_3 = \frac{1}{12}$. Thus, the Crank-Nicolson method is second-order accurate in time and therefore more accurate. From (2.44), it can easily be seen that $L(x,t) \longrightarrow 0$ as $h, \ell \longrightarrow 0$ simultaneously. Therefore, every finite difference method arising from (2.28) with $0 \leq \theta \leq 1$ is consistent with the partial differential equation (2.17).

Chapter 3

Generalized Nagumo reaction-diffusion equation

Reaction-diffusion equations are known to feature in the mathematical modeling of many real-life phenomena such as celestial mechanics, electric impulses along nerve axon, population biology etc [7]. For example, the well-known Nagumo reaction-diffusion equation, which has applications in nerve conduction, is given by

$$u_t = u(u-a)(1-u) + u_{xx}. \tag{3.1}$$

The Nagumo equation is a simplification of the Hodgkin-Huxley equation for nerve conduction [7]. Reaction-diffusion models generally involve non-linear reaction terms, making them difficult to be solved analytically. Thus, there is a need to construct competitive numerical methods for their solutions.

We consider a generalized version of the model in (3.1) involving control parameters α, β and D given by

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2} + \alpha u - \beta u^3. \tag{3.2}$$

Since it is well-known that for the reaction-diffusion problems such as (3.2), the numerical treatment of the reaction terms is crucial in the long term dynamics of the solution (see [34]), the diffusion-free equivalent of (3.2) (obtained by setting $D = 0$ in (3.2)) is studied in detail below.

3.1 The Diffusion-free Case

3.1.1 Dynamical Analysis

Consider, first of all, the diffusion-free case

$$\frac{du}{dt} = \alpha u - \beta u^3. \tag{3.3}$$

The aim here is to carry out a detailed qualitative analysis of the equilibrium solutions of (3.3). The equilibria of (3.3) are obtained by setting its right hand side (RHS) to zero. That is, the equilibria of system (3.3) satisfy

$$\alpha u - \beta u^3 = 0. \tag{3.4}$$

This yields one trivial equilibrium $u_0 = 0$ and two non-trivial equilibria $u\pm = \pm\sqrt{\alpha/\beta}$ (for $\alpha/\beta > 0$). The Jacobian of the RHS of (3.3) is given by

$$J(u) = \alpha - 3\beta u^2. \tag{3.5}$$

Bifurcation occurs when $J(u^*) = 0$, where u^* denotes any of the equilibrium point. Substituting $u^* = 0$ into the Jacobian gives

$$J(u^*) = \alpha. \tag{3.6}$$

Thus, the only eigenvalue associated with $u^* = 0$ is $\lambda = \alpha$. It equals zero only when $\alpha = 0$. Note that when $\alpha = 0$,

$$\frac{d\lambda}{d\alpha} = 1 \neq 0. \tag{3.7}$$

Based on the bifurcation theory discussed in subsection 2.3.3, it follows that $\alpha = 0$ is the only bifurcation point for the equilibrium point $u^* = 0$. Similarly, substituting the equilibrium points $u\pm = \pm\sqrt{\alpha/\beta}$ into the Jacobian (3.5) gives $J(u\pm) = -2\alpha$. The eigenvalue $\lambda = -2\alpha$ is zero if and only if $\alpha = 0$. Here,

$$\frac{d\lambda}{d\alpha} |_{\alpha=0} = -2 \neq 0. \tag{3.8}$$

Consequently, bifurcation only occurs when $\alpha = 0$. Clearly, $J(0) = \alpha$ indicates that the trivial solution $u_0 = 0$ is locally-asymptotically stable (unstable) when $\alpha < 0 (> 0)$, and $J(u\pm) = -2\alpha$ indicates that $u\pm$ are locally-asymptotically stable (unstable) when $\alpha > 0 (< 0)$. The bifurcation diagram is depicted in Figure 3.1, where a solid line represents a stable solution while the dotted line indicates an unstable solution. It is clear from Figure 3.1 that for values of $\alpha \in (-\infty, 0)$, the trivial equilibrium point $u_0 = 0$ is stable. However, the moment α crosses zero (the bifurcation point), two new stable equilibrium points $u\pm = \pm\sqrt{\alpha/\beta}$ are formed (and u_0 loses its stability). This type of bifurcation is called supercritical pitchfork bifurcation (see [1, 38]). In addition to establishing the local stability of the equilibria of (3.3), I offer

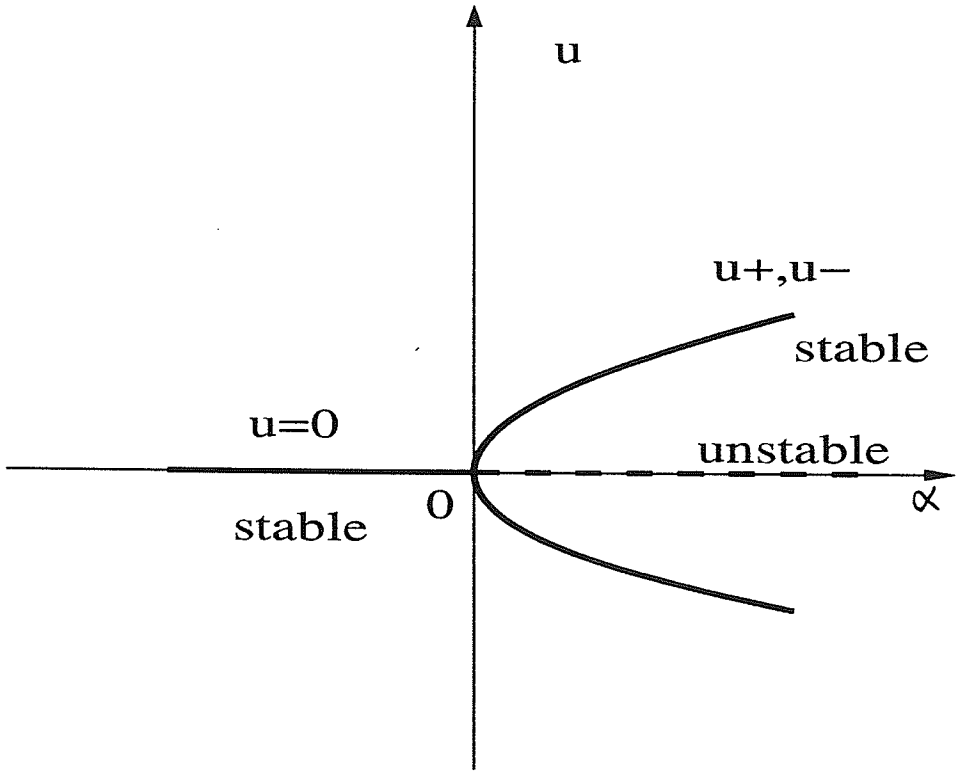

Figure. 3.1 The Diagram of Bifurcation

the following results on the global dynamics of the equilibria of (3.3).

Theorem 3.1: *The equilibrium $u^* = 0$ is globally-asymptotically stable if*

$\alpha < 0, \beta > 0$.

Proof: We have shown above that the equilibrium solutions $u\pm = \pm\sqrt{\alpha/\beta}$ are locally unstable for $\alpha < 0$. For the equilibrium solution $u^* = 0$, we construct the following Lyapunov function

$$L(u) = u^2. \tag{3.9}$$

It is easy to see that $L(u) > L(u^*) = 0,\ u \neq 0$, and

$$\frac{dL(u)}{dt} = \frac{dL(u)}{du}\frac{du}{dt} = 2(\alpha u^2 - \beta u^4) < 0, \text{ if } \alpha < 0, \beta > 0, \text{ and } u \neq 0. \tag{3.10}$$

Thus, $L(u)$ is a Lyapunov function for the equilibrium solution $u^* = 0$. This means that $u^* = 0$ is globally asympotically stable when $\alpha < 0, \beta > 0$. Consequently, all solution trajectories will be attracted to $u^* = 0$ when $\alpha < 0$ and $\beta > 0$. For the case $\alpha < 0$ and $\beta < 0$, I offer the following theorem.

Theorem 3.2: *The equilibrium $u^* = 0$ is globally-asymptotically stable on the interval $u^0 \in \left(-\sqrt{\alpha/\beta}, \sqrt{\alpha/\beta}\right)$ if $\alpha < 0, \beta < 0$.*

Proof: Using the same Lyapunov function for the equilibrium point $u = 0$, it is easy to verify that when $u^2 < \alpha/\beta$

$$\frac{dL(u)}{dt} = \frac{dL(u)}{du}\frac{du}{dt} = 2u(\alpha u - \beta u^3) = 2u^2(\alpha - \beta u^2) < 0,\ u \neq 0, \tag{3.11}$$

whenever $\alpha < 0$ and $\beta < 0$. These theoretical results are summarized in Table 3.1.

Table 3.1 : Stability of the equilibria of the Diffusion-free Model

α	β	Equilibrium points	Stability
< 0	> 0	0	globally-asymptotically stable
> 0	< 0	0	unstable
< 0	< 0	$0, \pm\sqrt{\alpha/\beta}$	$\pm\sqrt{\alpha/\beta}$ unstable, 0 stable when $u^2 < \frac{\alpha}{\beta}$
> 0	> 0	$0, \pm\sqrt{\alpha/\beta}$	0 unstable, $\pm\sqrt{\alpha/\beta}$ stable

3.1.2 Numerical Methods

Having determined the qualitative features of the diffusion-free model (Figure 3.1 and Table 3.1), the task ahead is to construct discrete models, for solving (3.3), having the same stability properties (as the model (3.3)). An important fact to consider whilst designing these discrete schemes is that they must satisfy the "positivity" property of the original (nerve conduction) model. Clearly, such a model requires the dependent variable (u) and the parameters (α, β, D) to be non-negative. Schemes that fail to preserve the positivity properties of the model generally tend to induce scheme-dependent (contrived) numerical instabilities (such as chaos, bifurcations, spurious zeros etc) [4,5,6,7,8,9].

The development of numerical methods is based on approximating the derivative in (3.3) by its first-order forward-difference approximation given by

$$\frac{du}{dt} = \frac{u(t+\ell) - u(t)}{\ell} + O(\ell), \tag{3.12}$$

where $\ell > 0$ is an increment in time (time-step). The interval $t \geq 0$ is discretized in the usual way at the points $t_n = nl$ $(n = 0, 1, 2, ...)$. The solution obtained by a numerical method at the point t_n will be denoted by U_n. Different numerical schemes are constructed depending on the approximations for the reaction terms on the right hand side of equation (3.3).

Method 1

For comparison purposes, the traditional explicit forward Euler method for solving (3.3), obtained by approximating the reaction term in (3.3) at the base time level t_n, is presented. It is given by:

$$\frac{U_{n+1} - U_n}{\ell} = \alpha U_n - \beta U_n^3, \tag{3.13}$$

so that,

$$U_{n+1} = U_n + \ell(\alpha U_n - \beta {U_n}^3). \tag{3.14}$$

The associated local truncation error of *Method 1*, denoted by $L_u^{(1)}[u(t), \ell]$, is obtained from (3.14). It is given by:

$$L_u^{(1)}[u(t), \ell] = u(t+\ell) - u(t) - \alpha \ell u(t) + \alpha \ell \beta [u(t)]^3, \tag{3.15}$$

from which it follows (using Taylor series expansion of $u(t+\ell)$ about ℓ),

$$L_u^{(1)}[u(t), \ell] = \left(\frac{1}{2}u''\right)\ell^2 + O(\ell^3). \tag{3.16}$$

Since the principal part of the local truncation error in (3.16) is of order ℓ^2, *Method 1* is first-order accurate (note that prime represents differentiation with respect to t).

Note that *Method 1* (given by (3.14)) has a negative term $(-\beta(U_n)^3)$ on its RHS. This clearly imply that *Method 1* fails to preserve the "positivity" property of the model (3.3). Because of such failure to satisfy the "positivity" property of the model, standard schemes (such as *Method 1* and Runge-Kutta methods) generally tend to induce scheme-dependent instability when certain parameter values and step-sizes are used. Such instabilities include contrived oscillations, bifurcations, chaos or convergence to spurious (false) steady-state (see [19]). If a numerical scheme is unstable, a small change in the initial condition may lead to chaos or overflow. To circumvent such contrived numerical instabilities, the following new robust scheme is proposed.

Method 2: Non-standard finite-difference method

The aim is to construct a numerical scheme that satisfies the "positivity property" of the model. To do so, the derivative in (3.3) is also approximated

using forward-difference and the reaction terms are approximated as follows:

$$\frac{U_{n+1} - U_n}{\ell} = \alpha(2U_n - U_{n+1}) - \beta {U_n}^2 U_{n+1}, \tag{3.17}$$

so that

$$U_{n+1} = \frac{(1 + 2\alpha\ell)U_n}{1 + \alpha\ell + \beta\ell {U_n}^2}. \tag{3.18}$$

Note that in order to preserve the positivity property of the model, the following "tricks" have been used:

(i) the linear term, αu, in (3.3) is approximated using a non-local implicit discretization namely: $\alpha u \rightarrow \alpha(2U_n - U_{n+1})$.

(ii) the cubic term in (3.3) is approximated as follows: $-\beta u^3 \rightarrow -\beta (U_n)^2 U_{n+1}$.

These two approximations ensure that all the U_{n+1} terms in the RHS of (3.17) are negative; and consequently, become positive when taken to the left to solve for U_{n+1} explicitly (3.18). Clearly, equation (3.18) shows that the novel non-standard method (*Method 2*) satisfies the positivity requirement of the model if $\alpha > 0$ and $\beta > 0$ (since no negative term exists on the RHS of (3.18)). That is to say, using any initial condition $U(0) > 0$, the solution generated using *Method 2* will remain positive. For many real-life systems, such as the model of nerve conduction (3.3), population dynamics or concentration densities, the dependent variables must be non-negative. Consequently, non-negative initial and boundary data evolve into non-negative solutions at later times (using *Method 2*). Mickens [25] showed that any discrete model that admits negative terms will exhibit scheme-dependent numerical instabilities.

Similary, the associated local truncation error of this method is obtained from (3.18), and is given by

$$L_u^{(2)}[u(t), \ell] = \left(\frac{1}{2}u'' + \alpha u' + \beta u^2 u'\right)\ell^2 + O(\ell^3). \tag{3.19}$$

Thus, *Method 2* is also first-order accurate.

Method 3: Second-order finite-difference method

The aim here is to design a finite-difference scheme with similar stability properties as *Method 2*, but of second-order accuracy. A new technique, based on using linear combinations of first-order method, is proposed as follows. Consider the following numerical schemes for solving (3.3):

$$\begin{aligned} M_A: &\quad U_{n+1} = U_n + \ell[\alpha U_{n+1} - \beta(U_n)^3], \\ M_B: &\quad U_{n+1} = U_n + \ell[\alpha U_n - \beta(U_n)^2 U_{n+1}], \\ M_C: &\quad U_{n+1} = U_n + \ell[\alpha U_n - \beta(U_n)^3]. \end{aligned} \tag{3.20}$$

The associated local truncation errors of these schemes are, respectively,

$$\begin{aligned} L_A: &\quad \left(\tfrac{1}{2}u'' - \alpha u'\right)\ell^2 + O(\ell^3), \\ L_B: &\quad \left(\tfrac{1}{2}u'' + \beta u^2 u'\right)\ell^2 + O(\ell^3), \\ L_C: &\quad \left(\tfrac{1}{2}u''\right)\ell^2 + O(\ell^3). \end{aligned} \tag{3.21}$$

Thus, these three methods are all first-order accurate. It can be seen that the linear combination

$$L_{comb} = \frac{1}{2}[L_A + 3L_B - 2L_C], \tag{3.22}$$

gives

$$L_{comb} = \frac{1}{2}\left[u'' - \alpha u' + 3\beta u^2 u'\right]\ell^2 + O(\ell^3). \tag{3.23}$$

Differentiating the differential equation in (3.3) reveals that the coefficient of ℓ^2 in (3.23) vanishes. Thus,

$$L_{comb} = O(\ell^3). \tag{3.24}$$

This implies that a novel second-order finite-difference method for solving (3.3) can be constructed by taking the linear combination $\frac{1}{2}(M_A + 3M_B -$

$2M_C$) and is given by

$$U_{n+1} = \frac{[1 + \frac{1}{2}\alpha\ell + \frac{1}{2}\beta\ell(U_n)^2]U^n}{1 + \frac{1}{2}\ell[3\beta(U_n)^2 - \alpha]}. \tag{3.25}$$

This method is denoted as *Method 3*. Note that, like the Euler method (*Method 1*), this method has a negative term ($-\frac{1}{2}\alpha\ell$) on its RHS.

3.1.3 Stability Analyses of the Numerical Methods

The aim here is to investigate the stability properties of the three numerical methods constructed in subsection 3.1.2. In particular, we want to determine which of the methods has exactly the same stability properties as the original continuous model (3.3) (depicted in Figure 3.1 and Table 3.1). The numerical methods constructed in subsection 3.1.2 are of the form

$$U_{n+1} = g(U_n);\ n = 0, 1, 2, \ldots, \tag{3.26}$$

where $g(U_n)$ represents the RHS of the numerical method. To analyse the fixed points of a numerical method, the associated equation $U = g(U)$ is now considered. In line with the Fixed Point Theorem [10], a fixed point U^* of a numerical method is said to be stable if $|g'(U^*)| < 1$ and to be unstable otherwise.

Stability of Method 1

It can be seem from (3.14) that the fixed points of *Method 1* are $U^* = 0$ and $U\pm = \pm\sqrt{\frac{\alpha}{\beta}}$. For this method, $g(U) = U + \ell(\alpha U - \beta U^3)$. Thus,

$$g'(U) = 1 + \alpha\ell - 3\ell\beta U^2, \tag{3.27}$$

so that $g'(0) = 1 + \alpha\ell$. Using the stability criterion above, it is clear that the fixed point $U^* = 0$ is locally-asymptotically stable if and only if $-2 <$

$\alpha\ell < 0$. Similarly, $g'(U\pm) = 1 - 2\alpha\ell$. Thus, the fixed points $U\pm$ are stable if $0 < \alpha\ell < 1$. Clearly, it follows that for values of α, ℓ outside the interval $-2 < \alpha\ell < 1$, this method (*Method 1*) will not converge to any of the fixed points ($U^* = 0$ or $U\pm = \pm\sqrt{\frac{\alpha}{\beta}}$). Thus, the method will fail (diverge or overflow) in these cases.

Stability of Method 2

This method has the same fixed points as *Method 1*. From (3.18)

$$g(U) = \frac{(1 + 2\alpha\ell)U}{1 + \alpha\ell + \beta\ell U^2}, \tag{3.28}$$

so that

$$g'(U) = \frac{(1 + 2\alpha\ell)}{1 + \alpha\ell + \beta\ell U^2} - \frac{2(1 + 2\alpha\ell)\beta\ell U^2}{(1 + \alpha\ell + \beta\ell U^2)^2}. \tag{3.29}$$

At the fixed point $U^* = 0$,

$$|g'(0)| = \left|\frac{1 + 2\alpha\ell}{1 + \alpha\ell}\right| > 1 \text{ for } \alpha\ell > 0. \tag{3.30}$$

Thus, $U^* = 0$ is locally-asymptotically stable for $\alpha\ell < 0$ and unstable for $\alpha\ell > 0$. At the fixed points $U\pm = \pm\sqrt{(\alpha/\beta)}$,

$$\left|g'\left(\pm\sqrt{(\alpha/\beta)}\right)\right| = \left|\frac{1}{1 + 2\alpha\ell}\right| < 1 \text{ for } \alpha\ell > 0. \tag{3.31}$$

This implies that the fixed points $U\pm = \pm\sqrt{\frac{\alpha}{\beta}}$ are always locally-asymptotically stable provided $\alpha\ell > 0$. Since the time step, ℓ, is always postive, it follows that *Method 2* is unconditonally stable for $\alpha > 0$.

Stability of Method 3

Here, too, the fixed points are the same as the original diffusion-free Nagumo equation. And, from (3.25),

$$g(U) = \frac{(1 + \frac{1}{2}\alpha\ell + \frac{1}{2}\beta\ell U^2)U}{1 + \frac{1}{2}\ell[3\beta U^2 - \alpha]}. \tag{3.32}$$

Thus,

$$g'(U) = \frac{\alpha\ell + 2 + 3\beta\ell U^2}{2 + \ell(3\beta U^2 - \alpha)} - \frac{6\beta U^2 \ell(2 + \alpha\ell + \beta\ell U^2)}{[2 + \ell(3\beta U^2 - \alpha)]^2}. \tag{3.33}$$

At the fixed point $U^* = 0$,

$$|g'(0)| = \left|\frac{2 + \alpha\ell}{2 - \alpha\ell}\right| > 1 \text{ for } \alpha\ell > 0, \tag{3.34}$$

so that $U^* = 0$ is locally-asymptotically stable for $\alpha\ell < 0$ and unstable for $\alpha\ell > 0$. At the fixed points $U\pm = \pm\sqrt{\alpha/\beta}$,

$$\left|g'\left(\pm\sqrt{\alpha/\beta}\right)\right| = \left|\frac{1 - \alpha\ell}{1 + \alpha\ell}\right| < 1 \text{ for } \alpha\ell > 0. \tag{3.35}$$

Thus, *Method 3* is also unconditionally stable (to $U\pm$) for $\alpha\ell > 0$. In conclusion, the two new finite-difference methods (*Method 2* and *Method 3*) I have constructed are both unconditionally-convergent for all values of the model parameters and step size.

3.1.4 Numerical Experiments

To verify the convergence properties of the methods constructed in section 3.1.2, numerous simulations are carried out to solve the model using various initial and parameter values as follows:

3.1.4.1 Effect of Time-step ℓ

The effect of time-step was monitored by carrying out numerous simulations for each method using $\alpha = 1$, $\beta = 1$ and different values of ℓ with $U_0 = 0.5$. RK4 simulations were also carried out for further comparisons. The result obtained, tabulated in Table 2, clearly confirm the unconditional-stability property of *Method 2* and *Method 3*. Figures 3.2, 3.3, 3.4 and 3.5 depict the solution profile generated using *Method 1*, Rk4, *Method 2* and *Method 3* respectively for $\alpha = \beta = 1$, $U_0 = 0.5$. For these values of α and β,

the correct steady-state solutions are $U\pm = \pm\sqrt{\alpha/\beta} = \pm 1$. It was observed that while *Method 1* gave profiles that show contrived oscillations (Figure 3.2), the RK4 method converged to false steady state $U^* = 0.82$ (Figure 3.3). Since values of $U < 0$ are biologically meaningless for the real-life model being considered, it follows from Figure 3.5 and Table 3.2 that the second-order unconditionally-stable scheme (*Method 3*) converges to a biologically unrealistic (albeit theoretically correct) steady state $U^* = -\sqrt{\alpha/\beta} = -1 < 0$ for certain initial conditions and step sizes. *Method 2*, on the other hand, always gave solution profiles that converged to the correct and biologically realistic steady state $U^* = 1.0$ (see Figure 3.4 and Table 3.2). The Euler method (*Method 1*), like the RK4 method, also converged to spurious solutions for certain values of α and β (see Figure 3.6). Figure 3.7 shows the case where the RK4 method exhibits chaotic dynamics.

Further additional simulations with various combinations of α, β, ℓ and U^0 were carried out. In all these simulations, unlike the other three methods, *Method 2* gave solution profiles that always converged to the correct (biologically meaningful) steady state $U+ = \sqrt{\alpha/\beta}$ for $\alpha > 0$. In other words, *Method 2* (although only first-order accurate) has the same stability properties as the original population model (3.3) (in the biologically-feasible range).

Table 3.2 : Effect of time step,ℓ, using $\alpha = 1, \beta = 1$ and $U_0 = 0.5$

ℓ	*Method 1*	*Method 2*	*Method 3*	*RK4*
0.05	c $\rightarrow 1$	c $\rightarrow 1$	c $\rightarrow 1$	c $\rightarrow 1$
0.5	c $\rightarrow 1$	c $\rightarrow 1$	c $\rightarrow 1$	c $\rightarrow 1$
1.0	chaos	c $\rightarrow 1$	c $\rightarrow 1$	c $\rightarrow 1$
1.5	chaos	c $\rightarrow 1$	osc c $\rightarrow 1$	c $\rightarrow 0.82$
10	overflow	c $\rightarrow 1$	c $\rightarrow -1$	overflow
1000	overflow	c $\rightarrow 1$	c $\rightarrow -1$	overflow

Notation:

(i) "c $\rightarrow \pm 1$" means monotonic convergence to $U^* = \pm 1$.

(ii) "osc c $\rightarrow \pm 1$" means oscillatory convergence to $U^* = \pm 1$.

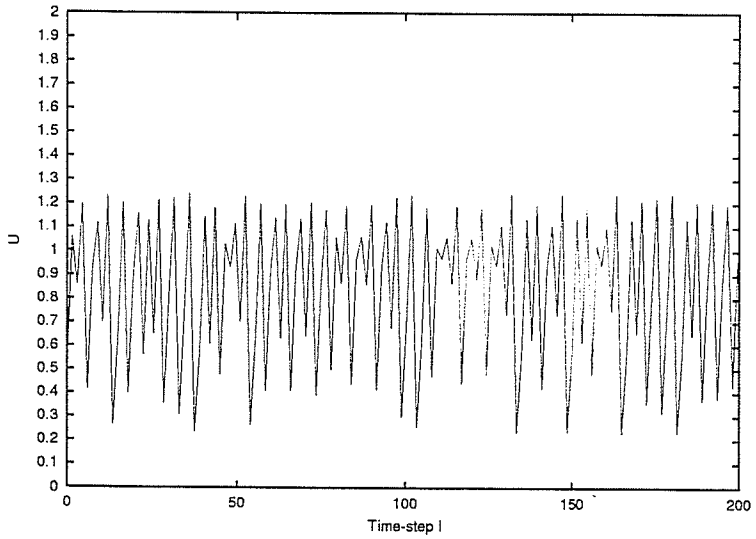

Figure 3.2 Profile of U using *Method 1* with $\alpha = 1, \beta = 1, U_0 = 0.5$ and $\ell = 1.5$

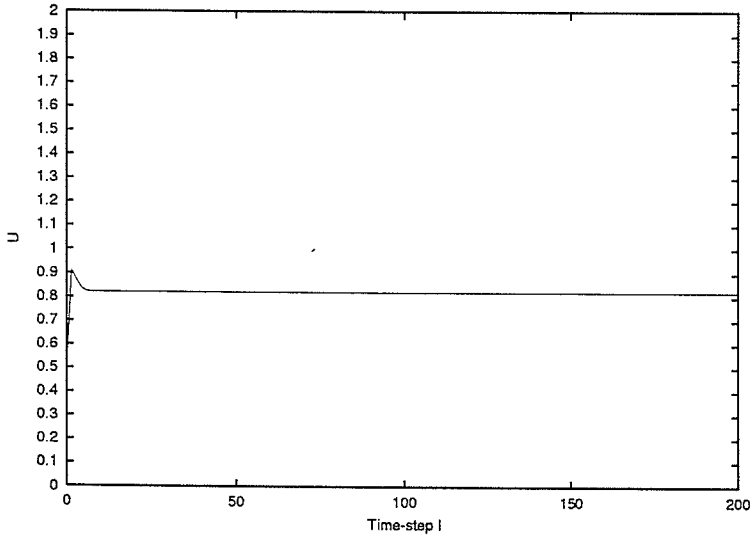

Figure 3.3 Profile of U using RK4 method with $\alpha = 1, \beta = 1, U_0 = 0.5$ and $\ell = 1.5$

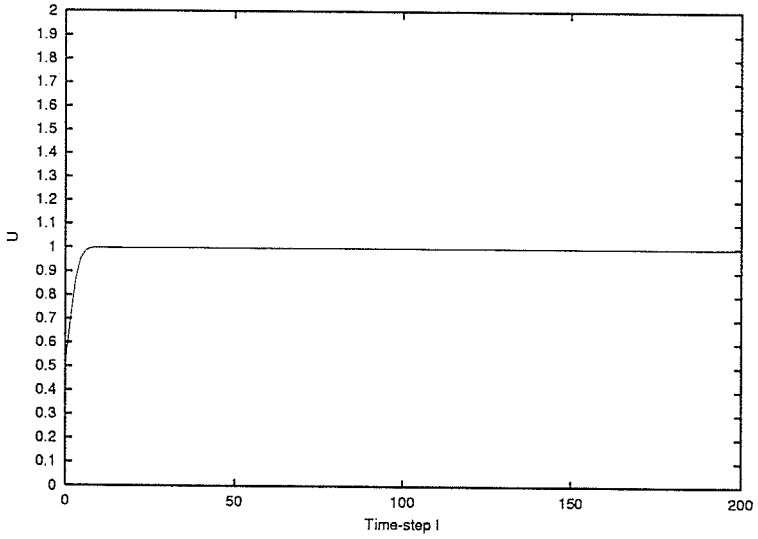

Figure 3.4 Profile of U using *Method 2* with $\alpha = 1, \beta = 1, U_0 = 0.5$ and $\ell = 1.5$

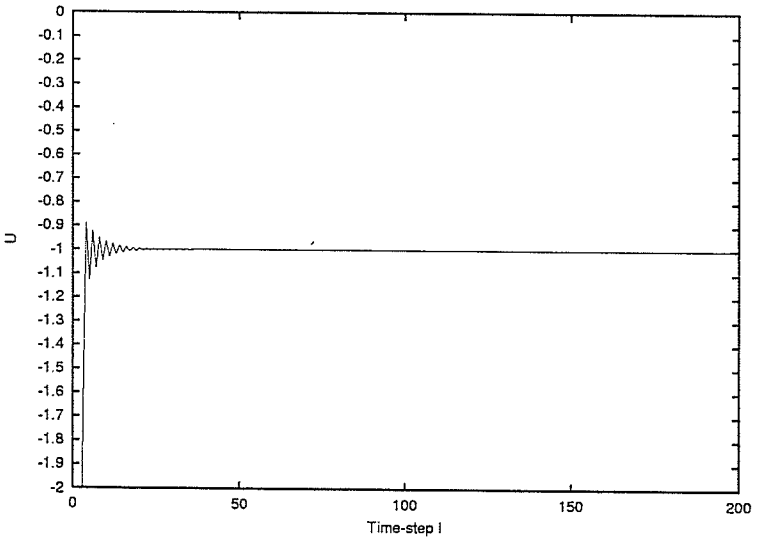

Figure 3.5 Profile of U using *Method 3* with $\alpha = 1, \beta = 1, U_0 = 0.5$ and $\ell = 10$

3.1.4.2 Effect of Initial-condition (U_0)

The effect of initial value U_0 was monitored by carrying out numerous simulations using different set of parameter values namely $\alpha = 2$ and $\beta = 1/2$.

Various values of U_0 are then used in the simulations with $\ell = 1.0$. These simulations are important because methods (or models) that exhibit great sensitivity to initial data are often associated with chaotic dynamics [1]. The results obtained are depicted in Table 3.3. It is clear from Table 3.3, that for negative initial values (although biologically unrealistic), *Method 2* and *Method 3* gave solution profiles that converged to $U- = -\sqrt{\alpha/\beta} = -2$. Similarly, for all positive initial values, the two methods converged to $U+ = \sqrt{\alpha/\beta} = 2$. This confirms that the fixed points, $U\pm = \pm\sqrt{\alpha/\beta}$, are locally-asymptotically stable. For large initial conditions (in magnitude), Table 3.3 shows that the RK4 and Euler methods fail (diverge). It is also worth mentioning that when ℓ is large (e.g. $\ell = 10$), with positive initial conditions, *Method 3* converges to the biologically unrealistic fixed point $U- = -2$. *Method 2*, on the other hand, always gave profiles that converge to $U+ = 2$ for all $\ell > 0$ and $U_0 > 0$. These results confirm the superior stability property of *Method 2* over the other methods.

Table 3.3 : Effect of U_0 using $\alpha = 2, \beta = 1/2$ and $\ell = 1.0$

U_0	*Method 1*	*Method 2*	*Method 3*	*RK4*
-10	overflow	c $\rightarrow -2$	c $\rightarrow -2$	overflow
-2.0	c $\rightarrow 2$	c $\rightarrow -2$	c $\rightarrow -2$	c $\rightarrow -2$
-0.5	c $\rightarrow -2$	c $\rightarrow -2$	c $\rightarrow -2$	c $\rightarrow -2$
0.5	c $\rightarrow 2$	c $\rightarrow 2$	c $\rightarrow 2$	c $\rightarrow 2$
2.0	c $\rightarrow -2$	c $\rightarrow 2$	c $\rightarrow 2$	c $\rightarrow 2$
10	overflow	c $\rightarrow 2$	c $\rightarrow 2$	overflow

3.1.4.3 Effect of α

To monitor the effect of the parameter α on the various numerical meth-

ods, we simulated the four methods using $\ell = 0.5$, $U_0 = 0.5$ and $\beta = 1.0$. The results generated by the methods are compared in Table 3.4 below. These results also confirm the robustness of *Method 2* over all the other methods.

Table 3.4 : Effect of α using $U_0 = 0.5, \beta = 1$ and $\ell = 0.5$

α	*Method 1*	*Method 2*	*Method 3*	*RK4*
1	c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow \sqrt{\alpha/\beta}$
1.5	c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow \sqrt{\alpha/\beta}$
2	osc c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow \sqrt{\alpha/\beta}$
4	overflow	c $\rightarrow \sqrt{\alpha/\beta}$	osc c $\rightarrow \sqrt{\alpha/\beta}$	oscillation
5	overflow	c $\rightarrow \sqrt{\alpha/\beta}$	c $\rightarrow -\sqrt{\alpha/\beta}$	chaos
10	overflow	c $\rightarrow \sqrt{\alpha/\beta}$	osc c $\rightarrow \sqrt{\alpha/\beta}$	overflow

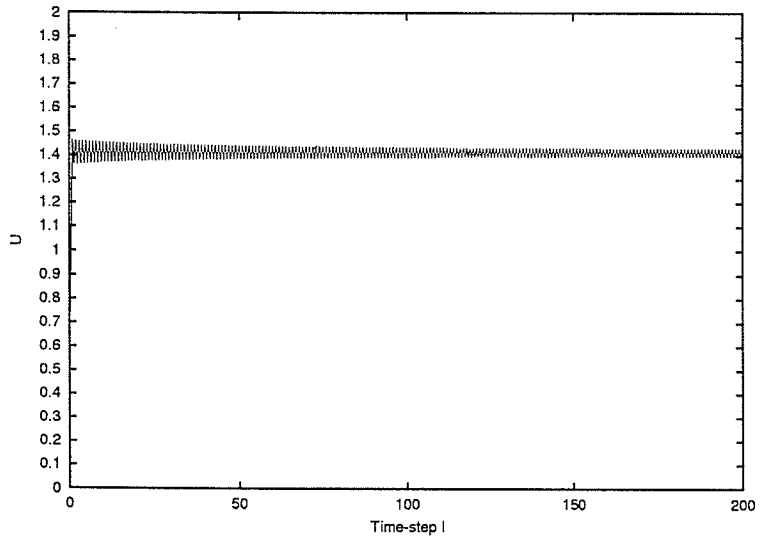

Figure 3.6 Profile of U using *Method 1* with $\alpha = 2, \beta = 1, U_0 = 0.5$ and $\ell = 0.5$

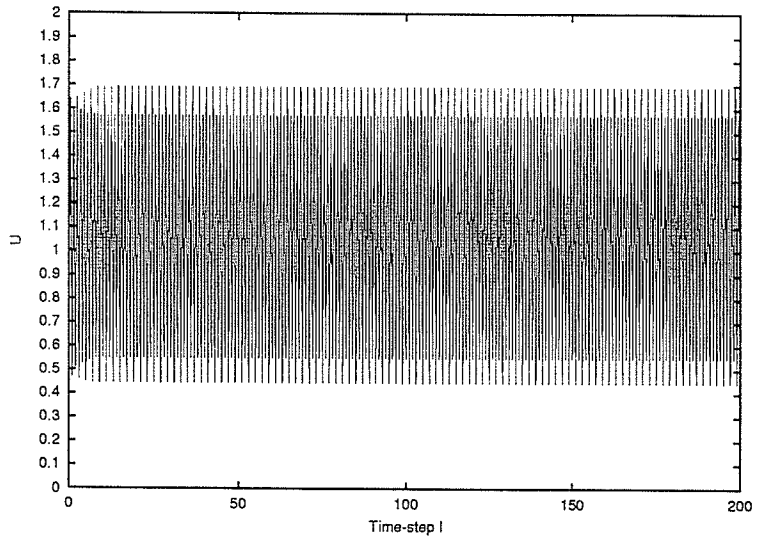

Figure 3.7 Profile of U using RK4 with $\alpha = 4, \beta = 1, U_0 = 0.5$ and $\ell = 0.5$

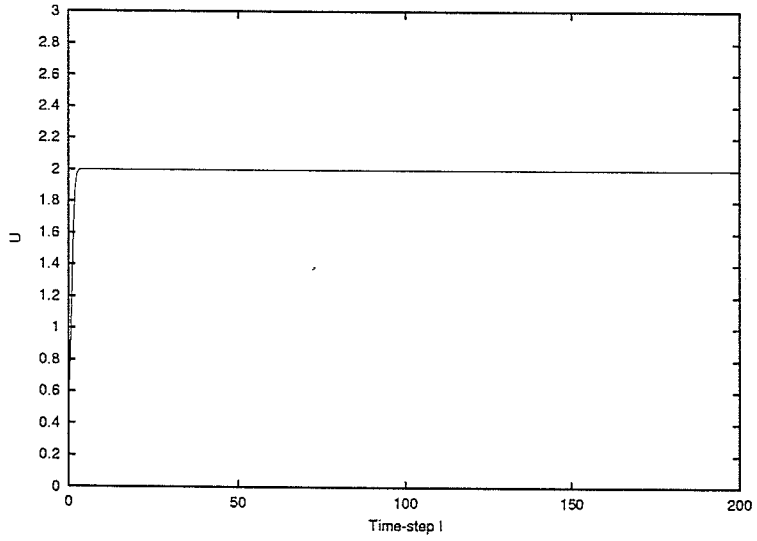

Figure 3.8 Profile of U using *Method 2* with $\alpha = 4, \beta = 1, U_0 = 0.5$ and $\ell = 0.5$

3.2 One-Dimensional Reaction-Diffusion Model

Consider, now, the generalized one-dimensional Nagumo reaction-diffusion system $(D > 0)$

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2} + \alpha u - \beta u^3. \tag{3.36}$$

3.2.1 Numerical Methods

In this section, a new finite-difference scheme is constructed for the full Nagumo reaction-diffusion model (3.36) ($\alpha > 0, \beta > 0, D > 0$). The interval $0 \leq x \leq L$ is divided into $N+1$ subintervals each of width h, so that $(N+1)h = L$. The notation U_m^n is used to express the solutions of the numerical method at the point (t_n, x_m).

3.2.1.1 Standard Explicit Method (*Method A*)

A standard explicit scheme for solving (3.36) can be constructed by approximating the diffusion term in (3.36) with its explicit central-difference approximation (CDA) given by

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{U_{m-1}^n - 2U_m^n + U_{m+1}^n}{h^2}, \tag{3.37}$$

and the diffusion-free part by the forward-Euler method (*Method 1*). This gives

$$\frac{U_m^{n+1} - U_m^n}{\ell} = D\frac{U_{m-1}^n - 2U_m^n + U_{m+1}^n}{h^2} + \alpha U_m^n - \beta (U_m^n)^3. \tag{3.38}$$

Thus,

$$U_m^{n+1} = (1 + \alpha\ell - \beta\ell(U_m^n)^2 - 2DP)U_m^n + DPU_{m+1}^n + DPU_{m-1}^n, \tag{3.39}$$

with $P = \ell/h^2$. As will be shown in the simulations section, this standard scheme fails when certain discretization or model parameters are used in the simulation. An alternative (more robust) scheme is constructed below.

3.2.1.2 Implicit Method (*Method B*)

Here, $\dfrac{\partial^2 u}{\partial x^2}$ is approximated using the implicit CDA given by

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{U_{m-1}^{n+1} - 2U_m^{n+1} + U_{m+1}^{n+1}}{h^2}, \tag{3.40}$$

and the diffusion-free part is approximated by *Method 2*. This gives

$$\begin{aligned}\frac{U_m^{n+1}-U_m^n}{\ell} &= D\frac{U_{m-1}^{n+1}-2U_m^{n+1}+U_{m+1}^{n+1}}{h^2} \\ &+ \alpha(2U_m^n - U_m^{n+1}) - \beta(U_m^n)^2 U_m^{n+1}, \end{aligned} \tag{3.41}$$

so that

$$-DPU_{m+1}^{n+1} - DPU_{m-1}^{n+1} + [1+\alpha\ell+\beta\ell(U_m^n)^2+2DP]U_m^{n+1} = (1+2\alpha\ell)U_m^n. \tag{3.42}$$

Clearly, *Method B* requires the application of tri-diagonal solvers at every time-step.

3.2.2 Stability Analysis

In order to gain some insights into the stability of *Method B* (*Method A* can be analyzed in a similar manner), the matrix method of stability analysis (see [19, 34, 37]) is employed. Note, first of all, that the von Neumann stability analysis is not feasible here since it only applies to linear, constant coefficient, finite-difference approximations.

Method B can be written in matrix-vector form as:

$$A^n\mathbf{U}^{n+1} = \mathbf{B}\mathbf{U}^n, \tag{3.43}$$

where A^n is a tri-diagonal matrix given by,

$$A^n = \begin{bmatrix} d_m^n & -DP & 0 & \cdots & 0 \\ -Dp & d_m^n & -DP & & \vdots \\ 0 & -DP & d_m^n & -DP & \\ & \ddots & \ddots & \ddots & 0 \\ \vdots & & -DP & d_m^n & -DP \\ 0 & \cdots & 0 & -DP & d_m^n \end{bmatrix}, \tag{3.44}$$

with

$$d_m^n = 1 + \alpha\ell + \beta\ell(U_m^n)^2 + 2DP;\ m = 1, 2, \ldots, N. \tag{3.45}$$

In (3.43), $\mathbf{B}$ is an N-dimensional vector given by

$$\mathbf{B} = [1 + 2\alpha\ell, 1 + 2\alpha\ell, \ldots, 1 + 2\alpha\ell, 1 + 2\alpha\ell]^T, \tag{3.46}$$

and

$$\mathbf{U}^n = [U_1^n, \ldots, U_{N-1}^n, U_N^n]^T. \tag{3.47}$$

The matrix method requires (see [36])

$$||(A^n)^{-1}\mathbf{B}||_S \leq 1 \tag{3.48}$$

for stability (with S denoting the spectral norm). And this localized condition must be satisfied at every time step. The inverse matrix $(A^n)^{-1}$ can be determined using, for instance, NAG (Numerical Algorithm Group) routine $F01ACF$, following which its spectral norm may be calcuated using NAG routine $F02AFF$.

3.2.3 Numerical Simulations

For simulation purposes, the numerical methods (*Method A* and *Method B*) are used to solve (3.36) subject to the boundary conditions $u(-1,t) = u(1,t) = 0$ and initial conditions $u(x,0) = f(x) = sin(\pi x)$.

3.2.3.1 Effect of Step-size ℓ

In order to monitor the effect of ℓ on the convergence properties of the two methods, the methods were simulated using $\alpha = \beta = D = 1$, $h = 0.05$, with various values of ℓ. The results, tabulated in Table 5.5, show that whilst *Method A* suffers from scheme-dependent instability (it fails for certain values of ℓ), the novel *Method B* always gave solution profiles that monotonically converge to the correct steady state solution ($U_0 = 0$).

3.2.3.2 Effect of Diffusion Coefficient D

In these simulations, the parameters α and β were assigned the value unity. The results generated by the two methods, using $\ell = 0.001$, $h = 0.05$, and various values of D, are shown in Table 5.6. This table, like Table 5.5, confirms the roubustness of *Method B* over *Method A*. Figures 3.9, 3.10, 3.11 and 3.12 depict the solution profiles generated using *Method B* at $t = 0.5$ for various values of D. The profiles of U at various times are depicted in Figure 3.13.

Additional numerical experiments, with numerous combinations of parameter values, further confirm the superior stability property of *Method B*. These findings are consistent with the fact that implicit methods, such as *Method B*, are more suitable for solving non-linear problems as against explicit ones (e.g. *Method A*).

It should be mentioned that when $D = 0$ (the diffusion-free case), *Method A* and *Method B* behave like *Method 1* and *Method 2* as expected.

Table 5.5 : Effect of diffusion coefficient D with $\ell = 0.001$

D	*Method A*	*Method B*
1	convergence	convergence
2	overflow	convergence
5	overflow	convergence
10	overflow	convergence
100	overflow	convergence

Table 5.6 : Effect of time-step ℓ with $D = 1$

ℓ	*Method A*	*Method B*
0.001	convergence	convergence
0.01	overflow	convergence
0.1	overflow	convergence
1	overflow	convergence
10	overflow	convergence

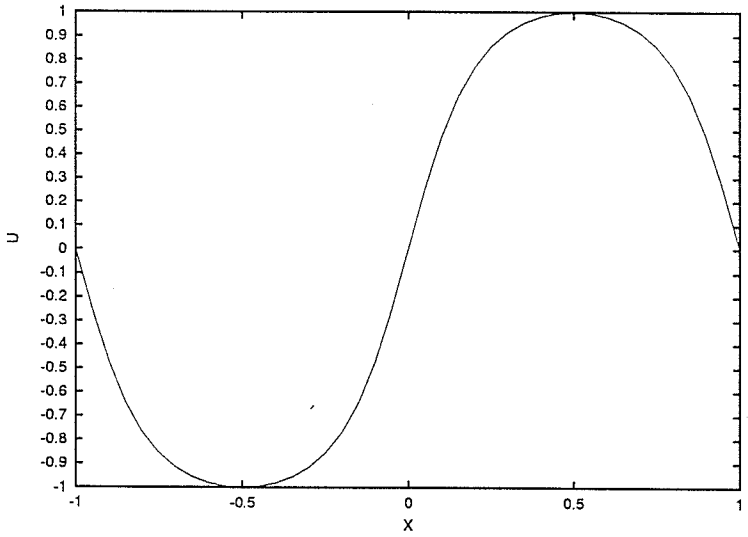

Figure 3.9 Profile of U at $t = 0.5$ using *Method B* with $\alpha = 1, \beta = 1$ and $D = 0$.

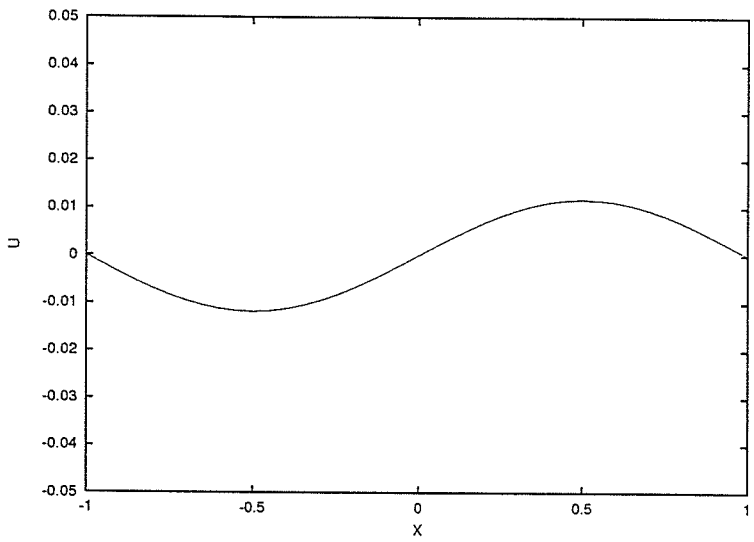

Figure 3.10 Profile of U at $t = 0.5$ using *Method B* with $\alpha = 1, \beta = 1$ and $D = 1$.

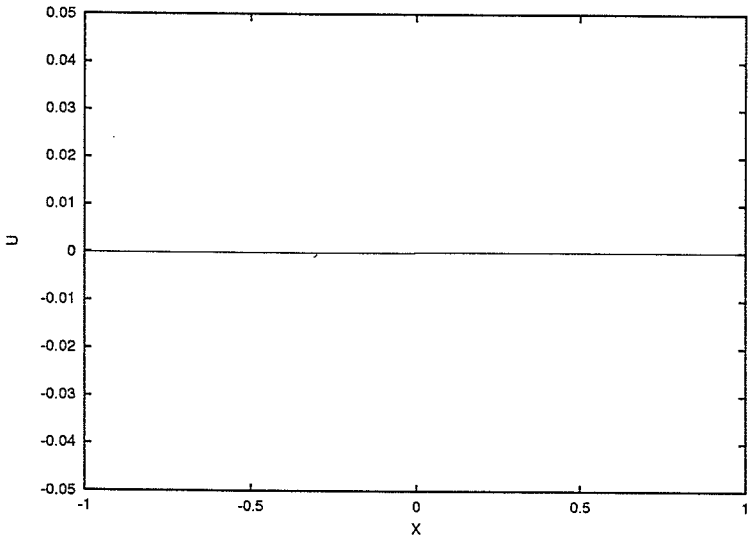

Figure 3.11 Profile of U at $t = 0.5$ using *Method B* with $\alpha = 1, \beta = 1$ and $D = 2$.

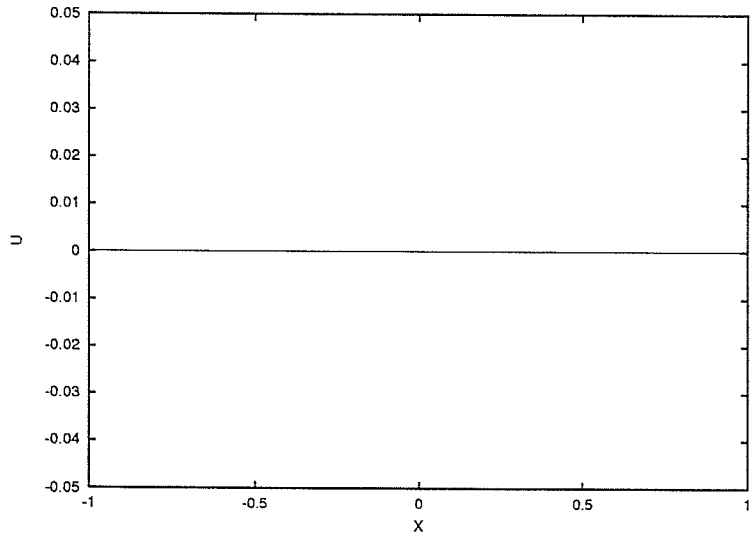

Figure 3.12 Profile of U at $t = 0.5$ using *Method B* with $\alpha = 1, \beta = 1$ and $D = 5$.

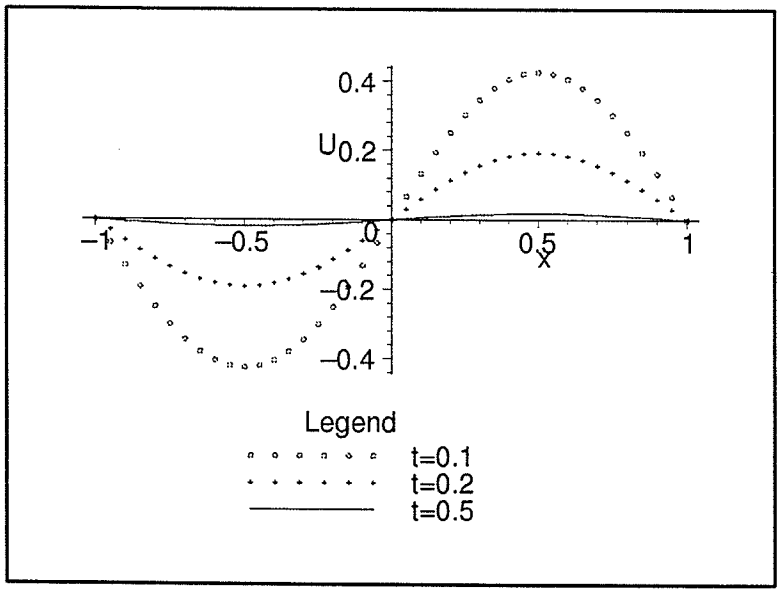

Figure 3.13 Profiles of U using *Method B* with $\alpha = 2, \beta = 2$ and $D = 1$.

Chapter 4

European Option Pricing Model

4.1 Background

As financial markets have become more sophisticated, more complex contracts than simple buy/sell trades have been introduced. Financial derivatives (such as options) can give investors of all kinds a great range of opportunities to tailor their dealings to their investment needs [39]. Option prices generally respond in an exaggerated way to changes in the underlying asset price and they depend on many factors, such as asset price, 'randomness' of the asset price (volatility) and interest rate [18]. Option pricing is considered to be among the most complex problems in mathematical finance because they need intensive application of stochastic calculus.

Since the introduction of the Black-Scholes model in 1973 [5], option pricing has received increasing attention from financial institutions. The Black-Scholes model, which deals with option valuation, can generally be transformed to the simple heat equation (for which well developed solution schemes are available) for simple option problems [18]. It should be mentioned, however, that it is often difficult to reduce a complex finance model to simpler forms.

The need for accurate and efficient numerical schemes for use to solve option pricing problems (so that the option price can be accurately predicted at a reduced computation cost) cannot be over emphasized. The three most widely-used numerical techniques in option pricing are Finite-difference, Lattice and Monte Carlo methods. The first two methods usually need more computations for complicated models [39]. A drawback of the Monte-Carlo

method is that it cannot be used to price options which are continuously early-exercisable (the option can be exercised before maturity) [18]. That is, this method is not suitable for pricing certain types of options such as American options. Although Lattice methods are perhaps the most popular numerical methods in computational finance, they are (in fact) simple explicit finite difference schemes. It has been demonstrated that standard Lattice methods are not suitable for a number of option pricing problems [39].

Various explicit and implicit schemes were used to solve models of option pricing [6, 16]. Courtedon [9] developed a second order accurate finite difference method for valuing options. Recently, Mayo [23] developed a fourth order method in the log of asset prices to evaluate American options. The focus of this chapter is, therefore, to develop and implement a new, stable, second-order, computationally-efficient, finite difference method for solving the Black-Scholes equation.

4.2. Discretization

In this section, a method based on the use of Padé approximation will be constructed and used to solve the vanilla European Put option, which is modelled through the use of the Black-Scholes equation. A put option is the right to sell the underlying assets (such as stocks) with a fixed price at a certain time (maturity). Suppose the maturity of the option occurs at time $t = T$. Then, the partial differential equation for a Put option is given by [18]

$$\frac{\partial u}{\partial t} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 u}{\partial S^2} + rS\frac{\partial u}{\partial S} - ru = 0, \tag{4.1}$$

subject to the boundary conditions

$$\begin{aligned} u(0,t) &= Ee^{-r(T-t)}, \\ \lim_{S\to\infty} u(S,t) &= 0, \end{aligned} \tag{4.2}$$

and the terminal condition

$$u(S,T) = \max(E - S, 0), \tag{4.3}$$

where u is the option price, t is time, σ is volatility, S is the asset price, r is the interest rate and E is the strike price. The B-S IBVP (4.1)-(4.3) will be discretized as follows:

A finite number of different, equally spaced times between the current time ($t = 0$) and the maturity of the option, ($t = T$) are chosen. That is, $\Delta t = T/N$ and the total of $N + 1$ times are considered. Similarly, a finite number of different, equally spaced stock prices are also chosen. We suppose that $S_{\max}$ is a stock price which is sufficiently high that, when it is reached, the put has virtually no value. We define $\Delta S = S_{\max}/M$ and consider a total of $M + 1$ stock prices. One of these is assumed to be the current stock price. Then, by the above discretization, a grid consisting of a total of $(M + 1)(N + 1)$ points is constructed. The mesh point (i, j) on the grid is the point that corresponds to time $i\Delta t$ and stock price $j\Delta S$. The variable u_{ij} is used to denote the value of the option at the (i, j) mesh point.

Using central difference approximation for $\dfrac{\partial u}{\partial S}$ and $\dfrac{\partial^2 u}{\partial S^2}$, given respectively by

$$\frac{\partial u}{\partial S} = \frac{u_{j+1}^{i+1} - u_{j-1}^{i+1}}{2\Delta S}, \tag{4.4}$$

$$\frac{\partial^2 u}{\partial S^2} = \frac{u_{j+1}^{i+1} + u_{j-1}^{i+1} - 2u_j^{i+1}}{\Delta S^2}, \tag{4.5}$$

the Black-Scholes equation is written in discrete form (using the *Method of Lines*) as

$$\frac{d\mathbf{U}}{dt} = A\mathbf{U} + \mathbf{C}(t). \tag{4.6}$$

In (4.6), the solution vector $\mathbf{U}$ is given by

$$\mathbf{U}^n = \{u_1^n, u_2^n, \ldots, u_M^n\} \text{ for } n = 0, 1, \ldots, N. \tag{4.7}$$

And $\mathbf{C}(t)$ arises from the use of the non-homogenous boundary conditions in (4.2), it is given by

$$\mathbf{C}(t) = [a_1 u(0,t), 0, \cdots, 0]^T; \tag{4.8}$$

A is a tridiagonal matrix given by

$$A = \begin{bmatrix} b_1 & c_1 & 0 & \cdots & 0 \\ a_2 & b_2 & c_2 & & \vdots \\ & \ddots & \ddots & \ddots & 0 \\ \vdots & & a_{M-1} & b_{M-1} & c_{M-1} \\ 0 & \cdots & 0 & b_M & c_M \end{bmatrix}, \tag{4.9}$$

where

$$a_j = \frac{rj}{2} - \frac{\sigma^2 j^2}{2}, \tag{4.10}$$

$$b_j = \sigma^2 j^2 + r, \tag{4.11}$$

and

$$c_j = \frac{-rj}{2} - \frac{\sigma^2 j^2}{2}. \tag{4.12}$$

Letting $\ell = \Delta t$ and $h = \Delta S$, it can be shown that the solution $\mathbf{U}(t)$ of (4.6) satisfies the recurrence relation [33]

$$\mathbf{U}(t+\ell) = e^{\ell A}\mathbf{U}(t) + \int_t^{t+\ell} e^{(t+\ell-s)A}\mathbf{C}(s)ds, \; t = 0, \ell, 2\ell, \ldots \tag{4.13}$$

It should be mentioned that the option pricing model is always associated with the terminal condition (condition at the maturity ($t = T$)) instead of the initial condition ($t = 0$). Therefore, in solving the difference equation, we should do a backward iteration instead of a forward iteration. Thus, by replacing ℓ with $-\ell$, equation (4.13) is rewritten as

$$\mathbf{U}(t-\ell) = e^{-\ell A}\mathbf{U}(t) + \int_t^{t-\ell} e^{(t-\ell-s)A}\mathbf{C}(s)ds, \; t = 0, \ell, 2\ell, \ldots \tag{4.14}$$

The integral term in (4.14) can be approximated, to second-order, using the trapezodial rule to give

$$\int_t^{t-\ell} e^{(t-\ell-s)A}\mathbf{C}(s)ds \approx -\frac{\ell}{2}\left[\mathbf{C}(t-\ell)+e^{-\ell A}\mathbf{C}(t)\right],\ \ t=0,\ell,2\ell,\ldots \quad (4.15)$$

Using (4.15) in (4.14) gives

$$\mathbf{U}(t-\ell)=e^{-\ell A}\mathbf{U}(t)-\frac{\ell}{2}\left[\mathbf{C}(t-\ell)+e^{-\ell A}\mathbf{C}(t)\right],\ t=0,\ell,2\ell,\ldots \quad (4.16)$$

The development of numerical methods for solving the Black-Scholes IBVP, consisting of equations (4.1)-(4.3), is now based on using appropriate approximation to the exponentials in (4.16). To achieve $O(h^2+\ell^2)$ and L_0-stability, the $(2,0)$ Padé approximant, given by,

$$e^{-\ell A}\approx\left(I+\ell A+\frac{1}{2}\ell^2A^2\right)^{-1}+O(\ell^3), \quad (4.17)$$

will be used to approximate $e^{-\ell A}$ in (4.16). This gives

$$B\mathbf{U}(t-\ell)=\mathbf{U}(t)-\frac{\ell}{2}\left[B\mathbf{C}(t-\ell)+\mathbf{C}(t)\right]\ \ t=0,\ell,2\ell,\ldots, \quad (4.18)$$

where $B=(I+\ell A+\frac{1}{2}\ell^2A^2)$ is a quin-diagonal matrix. Thus, the solution of the Black-Scholes equation can be obtained using an $O(h^2+\ell^2)$ scheme that involves the application of a quin-diagonal solver (to solve a linear algebraic system) at each time step. Although the L_0-stable algorithm (4.18) is of high-order accuracy in comparison to the traditional first-order explicit and implicit methods (based on $(0,1)$ and $(1,0)$ Padé approximants respectively), its sequential implementation may incur substantial computation costs (require the use of quin-diagonal solvers at each time step) and storage. A technique for circumventing this drawback is discussed in section 4.4.

4.3 Transformation to Simple Heat Equation

The PDE in (4.1) is a reaction-diffusion-convection equation with non-constant coefficients. Furthermore, the equation is clearly in "backward" form, with final data given at $t = T$.

The first thing to do is to get rid of S and S^2 in $\partial u/\partial S$ and $\partial^2 u/\partial S^2$ respectively. At the same time, we seek to convert the equation to a "forward" form. Using the transformation [39]

$$S = Ee^x,\ t = T - \frac{\tau}{\frac{\sigma^2}{2}} \text{ and } u(s,t) = Eg(x,\tau), \tag{4.19}$$

it can be shown that (4.1) becomes

$$\frac{\partial g}{\partial \tau} = \frac{\partial^2 g}{\partial x^2} + (k_1 - 1)\frac{\partial g}{\partial x} - k_1 g \tag{4.20}$$

where $k_1 = r/\frac{1}{2}\sigma^2$. Furthermore, the initial conditions become

$$g(x,0) = \max(1 - e^x, 0). \tag{4.21}$$

For some constants α and β (to be found), let

$$g = e^{\alpha x + \beta \tau} v(x,\tau). \tag{4.22}$$

Substituting (4.22) into (4.20) gives

$$\beta v + \frac{\partial v}{\partial \tau} = \alpha^2 v + 2\alpha \frac{\partial v}{\partial x} + \frac{\partial^2 v}{\partial x^2} + (k_1 - 1)(\alpha v + \frac{\partial v}{\partial x} - k_1 v). \tag{4.23}$$

The v term in (4.23) is eliminated by choosing

$$\beta = \alpha^2 + (k_1 - 1)\alpha - k_1. \tag{4.24}$$

It can be seen that the choice

$$0 = 2\alpha + (k_1 - 1) \tag{4.25}$$

eliminates the $\partial v/\partial x$ term as well. These equations for α and β give

$$\alpha = -\frac{1}{2}(k_1 - 1),\ \beta = -\frac{1}{4}(k_1 + 1)^2. \tag{4.26}$$

We then have

$$g(x,\tau) = e^{-\frac{1}{2}(k_1-1)x-\frac{1}{4}(k_1+1)^2\tau}v(x,\tau), \tag{4.27}$$

where,

$$\frac{\partial v}{\partial \tau} = \frac{\partial^2 v}{\partial x^2} \text{ for } -\infty < x < \infty, \tau > 0, \tag{4.28}$$

with,

$$v(x,0) = v_0(x) = \max\left[e^{\frac{1}{2}(k_1-1)x} - e^{\frac{1}{2}(k_1+1)x}, 0\right]. \tag{4.29}$$

Thus, the change of variables

$$S = Ee^x,\ t = T - \frac{\tau}{\frac{\sigma^2}{2}},\ u(S,t) = Eg(x,\tau), \tag{4.30}$$

with

$$g(x,\tau) = e^{-\frac{1}{2}(k_1-1)x-\frac{1}{4}(k_1+1)^2\tau}v(x,\tau),\ k_1 = \frac{2r}{\sigma^2}, \tag{4.31}$$

transforms the Black-Scholes model into the following IBVP for the European Put option

$$\frac{\partial v}{\partial \tau} = \frac{\partial^2 v}{\partial x^2} \tag{4.32}$$

subject to the initial and boundary conditions

$$\begin{aligned} v(x,0) &= \max(e^{\frac{1}{2}(k_1-1)x} - e^{\frac{1}{2}(k_1+1)x}, 0) \\ \lim_{x\to-\infty} v(x,\tau) &= e^{\frac{1}{2}(k_1-1)x+\frac{1}{4}(k_1-1)^2\tau} \\ \lim_{x\to\infty} v(x,\tau) &= 0. \end{aligned} \tag{4.33}$$

It is, of course, easier to solve the diffusion equation (4.32) than the original Black-Scholes equation (4.1). In other words, it is easier to numerically solve the simple heat equation and then, by change of variables, convert the solutions into those of the Black-Scholes model. It should be mentioned, however, that for complicated finance models, particularly the multi-factor models, it is not always feasible to reduce the problem to that of solving a

constant coefficient diffusion equation. In such cases, there is little choice but to design schemes for the generalized (full) Black-Scholes model. This is the main reason for our elaboration on the more general non-transformed case in Section 4.2.

A popular finite-difference method for solving PDEs such as (4.1) is the Crank-Nicholson method [2] which is based on the use of the diagonal $(1,1)$ Padé approximant in (4.16). This scheme, although second-order in space and time, is only A_0-stable. Thus, it is unsuited for use to solve problems with time-dependent boundary conditions or problems in which discontinuities exist between initial and boundary conditions (such as in the Black-Scholes model).

To circumvent these drawbacks, L_0-stable methods are used (see [19, 33] on the superiority of L_0-stable methods over A_0-stable ones). Therefore, a balance of accuracy, stability and available computing resources dictated my selection of the subdiagonal $(2,0)$ Padé approximant for use in (4.16) to solve the option pricing problem. It is easy to show that the method I constructed, resulting from the use of the $(2,0)$ Padé approximant in (4.16), is $O(h^2+\ell^2)$ and L_0-stable. Thus, the new method will be expected to have better stability properties in comparison to the A_0-stable Crank-Nicholson method.

As discussed earlier, the *Method of Lines* reduces the partial differential equation (4.32) with the non-homogenous boundary conditions (4.33) into a system of the form

$$\frac{d\mathbf{V}}{d\tau} = A_1\mathbf{V} + \mathbf{C_1}(\tau), \tag{4.34}$$

where, now,

$$A_1 = \begin{bmatrix} -2\rho & \rho & 0 & \cdots & 0 \\ \rho & -2\rho & \rho & & \vdots \\ & \ddots & \ddots & \ddots & 0 \\ \vdots & & \rho & -2\rho & \rho \\ 0 & \cdots & 0 & \rho & -2\rho \end{bmatrix}, \tag{4.35}$$

with $\rho = 1/h^2$, $\mathbf{C_1}(\tau) = [\rho v(0,\tau), 0, \cdots, 0]^T$ and $\mathbf{V}^n = \{v_1^n, v_2^n, \ldots, v_M^n\}$. Similarly, a grid consisting of a total of $(M+1)(N+1)$ points is constructed. If we denote ℓ, h as the time step in τ and the space step in x respectively, where $0 < \tau < T$ and $x_{\min} < x < x_{\max}$, then $M \times \ell = T$ and $N \times h = x_{\max} - x_{\min}$. Note that the terminal condition $t = T$ is now transformed into the initial condition $\tau = 0$. Thus, the solution vector $\mathbf{V}(\tau)$ satisfies the recurrence relation

$$\mathbf{V}(\tau + \ell) = e^{\ell A_1}\mathbf{V}(\tau) + \int_{\tau}^{\tau+\ell} e^{(\tau+\ell-k)A_1}\mathbf{C_1}(k)dk. \tag{4.36}$$

The integral in (4.36) is approximated using the trapezoidal rule to get

$$\mathbf{V}(\tau + \ell) = e^{\ell A_1}\mathbf{V}(\tau) + \frac{\ell}{2}\left[\mathbf{C_1}(\tau+\ell) + e^{\ell A_1}\mathbf{C_1}(\tau)\right]. \tag{4.37}$$

Using the $(2,0)$ Padé approximation to $e^{\ell A_1}$ gives

$$B_1\mathbf{V}(\tau + \ell) = \mathbf{V}(\tau) + \frac{\ell}{2}\left[B_1\mathbf{C_1}(\tau+\ell) + \mathbf{C_1}(\tau)\right] \tag{4.38}$$

where $B_1 = I - \ell A_1 + \frac{1}{2}\ell^2 A_1^2$. Like the method in Section 4.2, a quin-diagonal solver is needed to implement (4.38) at every time step.

4.4. Parallel Implementation

As mentioned above, the sequential implementation of the methods based on the $(2,0)$ Padé approximation require the use of quin-diagonal solvers. This is, of course, computationally demanding. The situation becomes worse if we use higher order Padé approximations (such as the $(3,0)$ or the $(4,0)$

Padé approximants). An alternative method is to employ parallel compution. With the rapid development of computer technology, parallel computing provides a mechanism for solving computational-intensive problems such as those associated with computational finance. Note that (4.38) can be re-written as

$$\begin{aligned} \mathbf{V}(\tau+\ell) &= \left(I-\ell A_1+\frac{1}{2}\ell^2 A_1^2\right)^{-1}\mathbf{V}(\tau) \\ &+ \frac{\ell}{2}\left[\mathbf{C}(\tau+\ell)+\left(I-\ell A_1+\frac{1}{2}\ell^2 A_1^2\right)^{-1}\mathbf{C}(\tau)\right]. \end{aligned} \tag{4.39}$$

The above equation can be expressed in partial-fraction splitting form as:

$$\begin{aligned} \mathbf{V}(\tau+\ell) &= \left[k_1(I-r_1\ell A_1)^{-1}+k_2(I-r_2\ell A_1)^{-1}\right]\mathbf{V}(\tau) \\ &+ \frac{\ell}{2}\left[k_1(I-r_1\ell A_1)^{-1}+k_2(I-r_2\ell A_1)^{-1}\right]\mathbf{C}(\tau)+\frac{\ell}{2}\mathbf{C}(\tau+\ell) \end{aligned} \tag{4.40}$$

where $r_1=\bar{r_2}=\frac{1}{2}(1+i)$ and $k_1=\bar{k_2}=\frac{1}{2}(1-i)$ are complex constants. Then, $\mathbf{V}(\tau+\ell)$ may now be obtained in parallel as follows:

$$\begin{aligned} \text{Processor 1}: \quad (I-r_1\ell A_1)\mathbf{w_1} &= k_1\mathbf{V}(\tau) \\ (I-r_2\ell A_1)\mathbf{w_2} &= k_2\mathbf{V}(\tau) \end{aligned} \tag{4.41}$$

$$\begin{aligned} \text{Processor 2}: \quad (I-r_1\ell A_1)\mathbf{z_1} &= k_1\mathbf{C}(\tau) \\ (I-r_2\ell A_1)\mathbf{z_2} &= k_2\mathbf{C}(\tau) \\ \mathbf{V}(\tau+\ell) = \quad & \mathbf{w_1}+\mathbf{w_2}+\frac{\ell}{2}\left[\mathbf{z_1}+\mathbf{z_2}+\mathbf{C}(\tau+\ell)\right] \end{aligned}$$

Note that, since $r_1=\bar{r_2}$ and $k_1=\bar{k_2}$, it follows that (see [15])

$$\begin{aligned} \text{Processor 1}: (I-r_1\ell A_1)\mathbf{w_1} &= k_1\mathbf{V}(\tau) \\ \mathbf{q_1} &= 2Re(\mathbf{w_1}) \end{aligned} \tag{4.42}$$

$$\text{Processor 2}: (I-r_1\ell A_1)\mathbf{z_1} = k_1\mathbf{C}(\tau)$$

$$\begin{aligned} \mathbf{q_2} &= 2Re(\mathbf{z_1}) \\ \mathbf{V}(\tau+\ell) &= \mathbf{q_1} + \frac{\ell}{2}[\mathbf{q_2} + \mathbf{C}(\tau+\ell)] \end{aligned}$$

In both (4.41) and (4.42), the solution vector $\mathbf{V}(\tau+\ell)$ is computed using two processors running currently, each of which solves a linear algebraic system using a single tri-diagonal solver at every time step (as against quin-diagonal solvers). Clearly, although not reported here, such parallel implementations lead to substantial savings in computation time and computer storage (in comparison to the corresponding sequential implementations). In summary, the main contribution in this chapter is the design of an $O(h^2+\ell^2)$, L_0-stable algorithm that enables the option price to be computed (predicted) efficiently (using parallel computing).

4.5. Numerical Verification

4.5.1 Full Black-Scholes Model

To test the behaviour of the numerical scheme (4.18), based on the use of the $(2,0)$ Padé approximant in (4.16), numerous simulations were carried out using the following parameter values: $T = 1$ year, $r = 0.1$, $\sigma = 0.3$ and $E = 100$. The true value of this Put at the stock price $S = 100$ is determined using the binomial tree method with up to 2000 steps, and is given as 7.216386.

The results generated using the method (4.18) is compared with those obtained using two well-known standard first-order methods namely: the $(0,1)$ Padé explicit method and the L_0-stable $(1,0)$ Padé implicit method. The results (absolute errors) are reported in Tables 4.1-4.3. In these tables, nt is the number of time steps, and nS is the number of intervals in the S direction. We choose $S_{\min} = 0$, $S_{\max} = 200$.

It can be seen from Table 4.1 that the explicit method is not robust

enough. It diverges for certain values of the discretization parameters.

Table 4.1 : Explicit method (based on $(0,1)$ Padé) for European Option without transformation

Exact	Put value	Absolute error	nt	nS
7.216386	6.472115	0.744271	10	20
7.216386	Divergence		10	40
7.216386	7.113129	0.103257	20	20
7.216386	Divergence		20	40
7.216386	7.083097	0.133289	40	20
7.216386	Divergence		40	40
7.216386	Divergence		40	80

Table 4.2 and Table 4.3 show that the $(1,0)$ Padé implicit method (which is L_0-stable) and the L_0-stable $(2,0)$ parallel method have much better stability properties than the explicit method arising from the use of $(0,1)$ Padé approximant in (4.16). This, of course, is consistent with the well-known fact that L_0-stable implicit schemes are more robust (in terms of stability) than explicit methods. Table 4.3 further confirm the superiority, in terms of computational accuracy, of the $O(h^2+\ell^2)$ $(2,0)$ Padé method over the other two first-order methods.

Table 4.2 : Implicit method (based on $(1,0)$ Padé) for European Option without transformation

Exact	Put value	Absolute error	nt	nS
7.216386	6.917384	0.299002	10	20
7.216386	7.047645	0.168741	10	40
7.216386	6.985141	0.231245	20	20
7.216386	7.112318	0.104068	20	40
7.216386	7.019024	0.197362	40	20
7.216386	7.144777	0.071609	40	40
7.216386	7.175542	0.040844	40	80
7.216386	7.183195	0.033191	40	160

Table 4.3 : Parallel $(2,0)$ Padé method for European Option without transformation

Exact	Put value	Absolute error	nt	nS
7.216386	7.046295	0.170091	10	20
7.216386	7.170949	0.045437	10	40
7.216386	7.051226	0.16516	20	20
7.216386	7.175597	0.040789	20	40
7.216386	7.052572	0.163814	40	20
7.216386	7.176864	0.039522	40	40
7.216386	7.207325	0.009061	40	80
7.216386	7.214904	0.001482	40	160

4.5.2 The Transformed Black-Scholes Model

The coordinates t, S are transformed to τ, x respectively (4.32),(4.33). We choose $x_{\min} = -9.5, x_{\max} = 0.5$, corresponding to $S_{\min} \approx 0, S_{\max} = 160$ with $0 < \tau < 0.045$, $\sigma = 0.3$. The new $O(h^2 + \ell^2)$, L_0-stable method designed in Section 4.3 is used to solve the transformed Black-Scholes model. The results obtained are also compared with those obtained using the standard first-order explicit and implicit methods. These are shown in Tables 4.4-4.6.

Table 4.4 : Explicit method (based on $(0,1)$ Padé) for transformed European Option model

Exact	Put value	Absolute error	$n\tau$	nx
7.216386	6.328616	0.88777	10	40
7.216386	8.664708(failed)		10	80
7.216386	7.050204	0.166182	20	80
7.216386	Divergence		20	160
7.216386	6.204828	1.011558	40	40
7.216386	7.019965	0.196421	40	80
7.216386	7.188730	0.027656	40	160
7.216386	7.208365	0.008021	40	200

Table 4.5 : Implicit method (based on $(1,0)$ Padé) for transformed European Option model

Exact	Put value	Absolute error	$n\tau$	nx
7.216386	6.007630	1.208756	10	40
7.216386	6.866610	0.349776	10	80
7.216386	6.928363	0.288023	20	80
7.216386	7.103772	0.112614	20	160
7.216386	6.124482	1.091904	40	40
7.216386	6.959040	0.257346	40	80
7.216386	7.132133	0.084253	40	160
7.216386	7.152209	0.064177	40	200

Table 4.6 : Parallel $(2,0)$ Padé method for transformed European Option model

Exact	Put value	Absolute error	$n\tau$	nx
7.216386	6.157258	1.059128	10	40
7.216386	6.983973	0.232413	10	80
7.216386	6.988059	0.228327	20	80
7.216386	7.159135	0.057251	20	160
7.216386	6.163922	1.052464	40	40
7.216386	6.989179	0.227207	40	80
7.216386	7.160113	0.056273	40	160
7.216386	7.179975	0.036411	40	200

Although the transformation to the simple heat equation greatly eases the computation process, it is evident from Tables 4.4-4.6 that more computation steps (in both spatial and temporal directions) are needed to get

the same accuracy obtained in the non-transformed Black-Scholes equation. In both cases (the transformed and the non-transformed), the above tables clearly show that the two implicit methods (based on $(1,0)$ and $(2,0)$ Padé approximations in (4.16) and (4.36)) are more robust in comparison to the explicit method (based on $(0,1)$ Padé). This finding confirms the fact that the implicit schemes (although generally more computationally involved) have superior stability properties than explicit methods. Furthermore, the novel $O(h^2+\ell^2)$ L_0-stable method I constructed proved to be more reliable in better (accurately) predicting the option price. Similar results were obtained when various (other) combinations of parameter values were used.

Chapter 5

A Predator-Prey System

5.1 Functional Responses

A functional response of the predator to the prey density refers to the change in the density of prey attacked *per* unit time *per* predator as the prey density changes. The simplest model of functional response is obtained by assuming that, in the time available for searching, the total change in the prey density/substrate concentration is proportional to the prey density/substrate concentration.

Hence, if $x(t)$ represents the prey density/substrate concentration at time t, the functional response is $ax(t)$, where $a > 0$ is a constant. Such a monotonic response function was used by Lotka in 1925 to study a hypothetical chemical reaction; and by Volterra in 1926 to model a predator-prey interaction (see [31] and the references therein).

Further studies on predator-prey systems subject to functional responses have been carried out by numerous authors [14, 26, 31]. In most of these studies, it is assumed that the functional response passes through the origin and has continuous second derivative. It is also assumed that the functional response is an increasing and bounded function. It is easy to check that standard functional responses such as Holling type II [14], given by $f(x) = \frac{x}{1+x}$, and Ivlev function [26], given by $f(x) = 1 - e^{-ax}$ $(a > 0)$, satisfy these conditions.

The Michaelis-Menten response function, which features prominently in enzymatic reactions, is given by

$$p(x) = \frac{mx}{a+x}, \tag{5.1}$$

where $m > 0$ is the maximal growth rate of the species and $a > 0$ is the half-saturation constant [31]. The sigmoidal response function,

$$p(x) = \frac{mx^2}{a + bx + x^2}, \tag{5.2}$$

is another widely-used response function. Setting $b = 0$ in (5.2) gives the well-known Holling type-III function [31]

$$p(x) = \frac{mx^2}{a + x^2}. \tag{5.3}$$

In general, a response function $p(x)$ satisfies the following hypothesis [31]:

- $p(x)$ is a continuously twice differentiable function defined on $[0, \infty)$.
- $p(0) = 0$. $p'(x) > 0$
- $\lim_{x \to \infty} p(x) = M < \infty$.

These assumptions imply that $p(x)$ is monotonic (which is true in many predator-prey systems). However, experimental observations indicate that this need not always be true. In microbial dynamics, for instance, experiments reveal that non-monotonic responses occur at the microbial level [14]. Such an inhibitory effect is modelled using the Monod-Haldane response function, given by [14]

$$p(x) = \frac{mx}{a + bx + x^2}. \tag{5.4}$$

In this chapter, we concentrate on the dynamical and numerical analyses of a predator-prey system of Gause-type [26] with a non-monotonic "group defence" functional response.

5.2 Group Defence Model

"Group defence" describes the phenomenon whereby predation is decreased, or even prevented altogether, due to the increased ability of the

prey to better defend or disguise themselves when their numbers are large enough [40]. For example, a single musk ox can be successfully attacked by wolves. Small herds of musk ox (2-6 animals) are attacked but with rare success. No successful attacks have been observed in larger herds [40].

To study the predator-prey interaction when the prey exhibits group defense, Freedman and Wolkowicz [14], Mischaikow and Wolkowicz [26], and Wolkowicz [40] proposed the following model

$$\begin{aligned} \dot{x} &= xg(x,K) - yp(x), \\ \dot{y} &= y(-D + q(x)). \end{aligned} \tag{5.5}$$

where x and y are functions of time representing population densities of prey and predator respectively; $K > 0$ is the carrying capacity of the prey and $D > 0$ is the death rate of the predator. The function $g(x,K)$ represents the specific growth rate of the prey in the absence of the predator and is assumed to satisfy the following conditions (for $x \geq 0, K > 0$) [40]:

$$\begin{gathered} g(K,K) = 0, g(0,K) > 0, \lim_{K\to\infty} g(0,K) < \infty, g_x(x,K) < 0, \\ g_K(x,K) \geq 0, g_{xK}(x,K) > 0, \lim_{K\to\infty} g_x(x,K) = 0. \end{gathered} \tag{5.6}$$

It is easy to see that the logistic function

$$g(x,K) = r(1 - \frac{x}{K}), \tag{5.7}$$

satisfies the conditions in (5.6). In (5.5), the function $p(x)$ denotes the predator response. It is assumed that $p(x)$ satisfies

$$\begin{aligned} p(0) = 0,\ p(x) > 0; &\qquad x > 0. \\ p'(x) > 0; &\qquad 0 \leq x < M, \\ p'(x) < 0; &\qquad x > M. \end{aligned} \tag{5.8}$$

The rate of conversion of prey to predator is described by $q(x)$. It is assumed that $q(x)$ has properties similar to $p(x)$. That is,

$$\begin{aligned} q(0) = 0,\ q(x) > 0; \qquad & x > 0. \\ q'(x) > 0; \qquad & 0 \leq x < M \\ q'(x) < 0; \qquad & x > M \end{aligned} \tag{5.9}$$

Furthermore, it is assumed that the constant M holds for both functions p and q (although this may not always hold in general). We shall now study the dynamics of the five-parameter model [31]:

$$\dot{x} = f_1(x,y) = rx\left(1 - \frac{x}{K}\right) - \frac{xy}{a+x^2}, \tag{5.10}$$

$$\dot{y} = f_2(x,y) = y\left(\frac{\mu x}{a+x^2} - D\right).$$

It is easy to see from (5.10) that the functions $p(x) = \dfrac{x}{a+x^2}$ and $q(x) = \mu p(x)$ satisfy the assumptions (5.8) and (5.9) respectively. Since all the model parameters and variables must be non-negative (otherwise, the model is not biologically realistic), we only restrict our attention to the dynamics of (5.10) in the first quadrant $\{(x,y) : x \geq 0,\ y \geq 0\}$ in the (x,y) plane. The aim here is to determine the qualitative features of the model (5.10) (using local stability theory) and then design a robust numerical scheme (with the same stability properties) for its solution. First of all, we discuss the existence of equilibrium solutions.

5.3 Equilibrium Solutions

The equilibrium solutions of (5.10) satisfy

$$rx\left(1 - \frac{x}{K}\right) - \frac{xy}{a+x^2} = 0, \tag{5.11}$$

$$y\left(\frac{\mu x}{a+x^2}-D\right) \;=\; 0.$$

Solving the second equation of (5.11) gives

$$y^* = 0 \;\text{ or }\; \frac{\mu x^*}{a+(x^*)^2} - D = 0. \tag{5.12}$$

Substituting $y^* = 0$ into the first equation gives

$$rx^*\left(1-\frac{x^*}{K}\right) = 0. \tag{5.13}$$

Thus, the two equilibrium solutions of the system (5.10) are $(0,0)$ and $(K,0)$. Furthermore, interior equilibrium solutions $(x \neq 0, y \neq 0)$ exist for (5.10) if and only if

$$\frac{\mu x^*}{a+(x^*)^2} - D = 0. \tag{5.14}$$

It follows from (5.14) that an interior equilibrium of (5.10) exists if

$$D(x^*)^2 - \mu x^* + aD = 0 \tag{5.15}$$

has a positive root. The condition for this to hold is

$$\mu^2 - 4aD^2 \geq 0. \tag{5.16}$$

Biologically, (5.16) is necessary for the persistence (no extinction) of the ecosystem [31]. It can be shown that the system (5.10) undergoes a saddle node bifurcation [31] whenever $\mu^2 - 4aD^2 = 0$. After computing the x-component of an interior equilibrium (corresponding to a positive root of (5.15)), the corresponding y-component is obtained by solving the following equation for y in terms of x^*. That is, from (5.11), we have

$$rx^*\left(1-\frac{x^*}{K}\right) - y\frac{x^*}{a+(x^*)^2} = 0, \tag{5.17}$$

so that

$$y^* = r\left(1-\frac{x^*}{K}\right)[a+(x^*)^2]. \tag{5.18}$$

Clearly, to ensure $y^* > 0$, a second condition must be satisfied namely:

$$x^* < K. \tag{5.19}$$

Consequently, the phase portraits of system (5.10) can be divided into the following three cases

- **Case 1:** $\mu^2 - 4aD^2 < 0$. In this case, the system (5.10) has no interior equilibria (since the roots of (5.15) are complex);
- **Case 2:** $\mu^2 - 4aD^2 = 0$ and $\frac{\mu}{2D} < K$. In this case, the system (5.10) has a unique interior equilibrium (x_0, y_0) where $x_0 = \frac{\mu}{2D}$, $y_0 = r(1 - \frac{x_0}{K})(a + x_0^2)$;
- **Case 3:** $\mu^2 - 4aD^2 > 0$. Here, the system (5.10) has at most two interior equilibria in addition to the equilibria $(0, 0)$ and $(K, 0)$. They are denoted by (x_1, y_1), (x_2, y_2), where

$$x_1 = \frac{\mu - \sqrt{\mu^2 - 4aD^2}}{2D}, \quad y_1 = r\left(1 - \frac{x_1}{K}\right)\left(a + x_1^2\right), \tag{5.20}$$

$$x_2 = \frac{\mu + \sqrt{\mu^2 - 4aD^2}}{2D}, \quad y_2 = r\left(1 - \frac{x_2}{K}\right)\left(a + x_2^2\right).$$

5.4 Stability Analysis

It is easy to show that the Jacobian of the model (5.10) is

$$J(x, y) = \begin{bmatrix} r\left[1 - \dfrac{2x}{K} - \dfrac{y(a - (x)^2)}{(a + x^2)^2}\right] & \dfrac{x}{a + x^2} \\ \\ \dfrac{y\mu\left[a - x^2\right]}{\left[a + x^2\right]^2} & \dfrac{\mu x}{a + x^2} - D \end{bmatrix}. \tag{5.21}$$

Substituting $(x, y) = (0, 0)$ into (5.21) gives

$$J(0, 0) = \begin{bmatrix} r & 0 \\ 0 & -D \end{bmatrix}. \tag{5.22}$$

The eigenvalues of (5.22) are $r > 0$ and $-D < 0$ (since all model parameters are non-negative). Thus, the equilibrium solution $(0, 0)$ is unstable. Similarly, substituting $(x, y) = (K, 0)$ into the Jacobian (5.21) gives

$$J(K,0) = \begin{bmatrix} -r & \dfrac{K}{a+K^2} \\ \\ 0 & \dfrac{\mu K}{a+K^2} - D \end{bmatrix}. \tag{5.23}$$

The eigenvalues of $J(K, 0)$ are $-r < 0$ and $\dfrac{\mu K}{a+K^2} - D$. Clearly, the equilibrium solution $(K, 0)$ is stable if and only if $\dfrac{\mu K}{a+K^2} - D < 0$. The dynamics analysis of the other associated (interior) equilibrium solutions (x_1, y_1) and (x_2, y_2) can also be examined in a similar way (see [31] for detailed discussion).

5.5 Numerical Methods

The aim here is to construct a competitive numerical scheme that has the same stability and convergence properties as the predator-prey system (5.10). As in subsection 3.1.2, the development of numerical methods is based on approximating the derivatives in (5.10) by their first-order forward-difference approximations given by

$$\frac{dX}{dt} = \frac{X(t+\ell) - X(t)}{\ell} + O(\ell), \tag{5.24}$$

$$\frac{dY}{dt} = \frac{Y(t+\ell) - Y(t)}{\ell} + O(\ell).$$

5.5.1 Forward Euler Method

The forward Euler method for solving (5.10) is given by

$$\frac{X_{n+1} - X_n}{\ell} = rX_n\left(1 - \frac{X_n}{K}\right) - \frac{X_n Y_n}{a + X_n^2}, \tag{5.25}$$

$$\frac{Y_{n+1}-Y_n}{\ell} = Y_n\left(\frac{\mu X_n}{a+X_n^2}-D\right).$$

That is,

$$X_{n+1} \equiv g_1(X_n,Y_n) = \left(1+r\ell-r\ell\frac{X_n}{K}-\frac{\ell Y_n}{a+X_n^2},\right)X_n \tag{5.26}$$

$$Y_{n+1} \equiv g_2(X_n,Y_n) = Y_n+\ell Y_n\left(\frac{\mu X_n}{a+X_n^2}-D\right)$$

5.5.2 Novel Implicit Method

A novel method for solving (5.10) is:

$$\frac{X_{n+1}-X_n}{\ell} = rX_n\left(1-\frac{X_{n+1}}{K}\right)-\frac{X_{n+1}Y_n}{a+X_n^2}, \tag{5.27}$$

$$\frac{Y_{n+1}-Y_n}{\ell} = Y_n\left(\frac{\mu X_{n+1}}{a+X_{n+1}^2}-D\right),$$

so that,

$$X_{n+1} \equiv h_1(X_n,Y_n) = \frac{(1+r\ell)X_n}{1+\ell\left(\frac{X_n r}{K}+\frac{Y_n}{a+X_n^2}\right)}, \tag{5.28}$$

$$Y_{n+1} \equiv h_2(X_n,Y_n) = \frac{\left(1+\frac{\mu\ell X_{n+1}}{a+X_{n+1}^2}\right)Y_n}{1+D\ell}.$$

Although the novel method (5.27) is constructed implicitly, equation (5.28) enables it to be implemented explicitly at every time-step. Since it does not have a negative term on its right hand side, the implicit method (5.28) satisfies the positivity requirement of the model (5.10). That is to say, with any positive initial condition ($X(0) > 0$ and $Y(0) > 0$), solving the

model (5.10) using (5.28) guarantees non-negative solutions (that is, $X(t) > 0$ and $Y(t) > 0$ for all $0 \leq t < \infty$). The standard forward Euler method (5.26), on the other hand, admits negative terms on its right hand side. Consequently, it will be expected to suffer from scheme-dependent instabilities.

5.6 Fixed Points Analyses

5.6.1 Forward Euler Method

It is easy to see that the fixed points of the Forward Euler method (5.26) are the same as the equilibrium points of (5.10). In line with the Fixed Point Theorem [38], a fixed point (X^*, Y^*) of a numerical method is said to be stable if all the associated eigenvalues (of the Jacobian at the fixed point) are less than one in magnitude. The Jacobian of (5.26) is

$$J(X,Y) = \begin{bmatrix} 1 + r\ell - \dfrac{2\ell r X}{K} - \dfrac{\ell Y}{a+X^2} + \dfrac{2\ell X^2 Y}{(a+X^2)^2} & -\dfrac{\ell X}{a+X^2} \\ \\ \ell Y\left(\dfrac{\mu}{a+X^2} - \dfrac{2\mu X^2}{(a+X^2)^2}\right) & 1 + \ell\left(\dfrac{\mu X}{a+X^2} - D\right) \end{bmatrix}. \tag{5.29}$$

Evaluating the Jacobian (5.29) at the fixed point $(0,0)$ gives

$$J(0,0) = \begin{bmatrix} 1 + r\ell & 0 \\ 0 & 1 - D\ell \end{bmatrix}. \tag{5.30}$$

The eigenvalues of (5.30) are $1 + r\ell > 1$ (since r and ℓ are positive) and $1 - D\ell$. Thus, the fixed point $(0,0)$ is unstable. Similarly, evaluating (5.29) at $(K,0)$ gives

$$J(K,0) = \begin{bmatrix} 1 - r\ell & -\dfrac{\ell K}{a+K^2} \\ \\ 0 & 1 + \ell\left(\dfrac{\mu K}{a+K^2} - D\right) \end{bmatrix}. \tag{5.31}$$

The eigenvalues of (5.31) are $1 - r\ell$ and $1 + \ell\left(\frac{\mu K}{a + K^2} - D\right)$. Thus, the fixed point $(K, 0)$ is stable if and only if

$$|1 - r\ell| < 1 \text{ and } \left|1 + \ell\left(\frac{\mu K}{a + K^2} - D\right)\right| < 1. \tag{5.32}$$

The inequalities in (5.32) imply that $r\ell < 2$ and $D - \frac{2}{\ell} < \frac{\mu K}{a + K^2} < D$. Thus, it is clear that for large step sizes, even when $\frac{\mu K}{a + K^2} < D$, the forward Euler method will fail to converge to the fixed point $(K, 0)$. This is not consistent with the dynamics of the original system (5.10); which has a stable fixed point $(K, 0)$ provided $\frac{\mu K}{a + K^2} < D$. Thus, the forward Euler scheme will suffer scheme-dependent numerical instability.

5.6.2 Novel Implicit Method

It is easy to verify that the fixed points of the new implicit method (5.28) are the same as the equilibrium points of the original system (5.10). Evaluating the associated Jacobian of (5.28) at the fixed point $(0, 0)$ gives

$$J(0,0) = \begin{bmatrix} 1 + r\ell & 0 \\ 0 & \frac{1}{1 + D\ell} \end{bmatrix}, \tag{5.33}$$

with eigenvalues $1 + r\ell > 1$ and $\frac{1}{1 + D\ell}$. Thus, the fixed point $(0, 0)$ is unstable. Similarly, the Jacobian at the fixed point $(K, 0)$ is

$$J(K,0) = \begin{bmatrix} \frac{1}{1 + r\ell} & -\frac{K\ell}{(1 + r\ell)(a + K^2)} \\ 0 & \frac{1 + \frac{\mu\ell K}{a + K^2}}{1 + D\ell} \end{bmatrix}. \tag{5.34}$$

The eigenvalues of (5.34) are $\frac{1}{1 + r\ell} < 1$ for $r, \ell > 0$ and $\frac{1 + \frac{\mu\ell K}{a + K^2}}{1 + D\ell}$. The

fixed point $(K, 0)$ is stable provided

$$\left| \frac{1 + \dfrac{\mu \ell K}{a + K^2}}{1 + D\ell} \right| < 1. \tag{5.35}$$

This implies that $(K, 0)$ is stable if

$$\frac{\mu K}{a + K^2} < D. \tag{5.36}$$

This requirement, for the stability of $(K, 0)$, is exactly the same as in the continuous case discussed in Section 5.4. Thus, the implicit method (5.28) has the same stability property as the original system (5.10).

5.7 Numerical Experiments

To test the stability and convergence properties of the forward Euler and the novel implicit methods constructed in Section 5.5, numerous numerical experiments were carried out using the two methods to solve (5.10) as follows:

5.7.1 Experiment 1: Effect of the Time-Step, ℓ

The effect of time-step, ℓ, on the stability of the two methods was monitored by simulating the methods with the following parameter and initial values: $r = 0.1, D = 0.1, a = 100, \mu = 1.5, k = 1000$ and $X(0) = 10, Y(0) = 8$ and various step sizes. The results are tabulated in Table 5.1.

It is evident from Table 5.1 that the new implicit method is more competitive in terms of numerical stability. These experiments reveal that, when the time step is large, the forward Euler method ceases to satisfy the positive requirement of the model. For instance, using $\ell = 100$ and positive initial condition $X(0) = 10, Y(0) = 8$, the forward Euler method gave negative solution after one iteration; and diverges shortly thereafter. The novel implicit

method, on the other hand, gives profiles that are always free of such numerical instabilities. Table 5.2 depicts the effect of the time-step with $K = 15$ and $\mu = 2.5$ (retaining the other parameters and initial values as above). In this case, the forward Euler method gave wrong (spurious) solutions for large step sizes.

The effect of initial conditions was also monitored using the parameters used to generate Table 5.1. Table 5.3 depicts the case where $X(0) = 10$ and $Y(0) = 100$. Expectedly, the forward Euler method failed; but the novel implicit method always converged to the correct steady-state solution.

Table 5.1 : Effect of time-step

ℓ	Forward Euler Method	Novel Implicit Method
0.1	convergence	convergence
1.0	convergence	convergence
10	convergence	convergence
100	overflow	convergence
1000	overflow	convergence

Table 5.2 : Effect of time-step with $K = 15, \mu = 2.5$

ℓ	Forward Euler Method	Novel Implicit Method
0.1	convergence	convergence
1.0	convergence	convergence
10	spurious solution	convergence
100	overflow	convergence
1000	overflow	convergence

Table 5.3 : Effect of time-step with $X(0) = 10$ and $Y(0) = 100$

ℓ	Forward Euler Method	Novel Implicit Method
0.1	convergence	convergence
1.0	convergence	convergence
10	overflow	convergence
100	overflow	convergence
1000	overflow	convergence

5.7.2 Experiment 2: Effect of μ

The effect of μ on the dynamics of (5.10) was monitored by simulating the two methods with $r = 0.1, D = 0.2, a = 300, K = 12$, $X(0) = 10$ and $Y(0) = 8$. Table 5.4 depicts the results obtained for various values of μ using $\ell = 10$.

Table 5.4 : Effect of μ

μ	$\mu^2 - 4aD^2$	K	Forward Euler Method	Novel Implicit Method
5	< 0		converge to $(K, 0)$	converge to $(K, 0)$
6	< 0		converge to $(K, 0)$	converge to $(K, 0)$
7	> 0	$K < x_1$	converge to $(K, 0)$	converge to $(K, 0)$
8	> 0	$x_1 < K < x_2$	converge to (x_1, y_1)	converge to (x_1, y_1)
10	> 0	$x_1 < K < x_2$	chaos	converge to (x_1, y_1)

It is evident from Table 5.4 that, when $\mu^2 - 4aD$ is small but positive, both methods converge to the right steady-state solution. Furthermore, while the forward Euler method exhibits chaotic dynamics when μ becomes larger, the novel implicit method always converges to the correct steady state solution. We also observed that when the time step ℓ becomes larger, or the initial conditon $Y(0)$ becomes larger, the forward Euler method fails (confirming its well-known sensitivity to step-sizes and initial conditions). Further numerical simulations, using various combinations of parameters, gave results that are consistent with the above conclusions.

5.8 Predator-prey Model with Diffusion

A number of authors have incorporated diffusion in modelling epidemiological systems to account for the mobility (in spatial direction) of the various populations of the model (see, for instance [7]). Consequently, in order to simulate mobility and spatial distribution in our predator-prey model (5.10), we consider the case where the ratio of each of the model populations (x and y) can change at random in spatial direction. Thus, we extend the

model (5.10) to a reaction-diffusion one given by

$$\begin{aligned} \frac{\partial x}{\partial t} &= rx\left(1-\frac{x}{K}\right)-\frac{xy}{a+x^2}+P\frac{\partial^2 x}{\partial z^2}, \\ \frac{\partial y}{\partial t} &= y\left(\frac{\mu x}{a+x^2}-D\right)+P\frac{\partial^2 y}{\partial z^2}, \end{aligned} \tag{5.37}$$

where P is the diffusion constant and z is the spatial variable.

Like in previous chapters, a competitive numerical scheme for (5.37) can be constructed by using the novel implicit method (5.28) to approximate the diffusion-free part; and the weighted central-difference approximant, given by

$$\begin{aligned} P\left[\frac{\partial^2 x}{\partial z^2}\right] &\approx Ph^{-2}\{\frac{1}{2}\left[x(z-h,t+\ell)-2x(z,t+\ell)+x(z+h,t+\ell)\right] \\ &+ \frac{1}{2}\left[x(z-h,t)-2x(z,t)+x(z+h,t)\right]\}, \end{aligned} \tag{5.38}$$

for the spatial derivative. In (5.38), h is the space step.

The interval $0 \le z \le L$ is divided into $N+1$ subintervals each of width h, so that $(N+1)h = L$ and the time variable t is discretized in steps of length ℓ. The solutions $x(t_n, z_m)$ and $y(t_n, z_m)$, are sought at each point $(n\ell, mz)$ where $m = 0, 1, 2, \ldots, N+1$ and $n = 0, 1, 2, \ldots$. The notations X_m^n and Y_m^n will be used to denote the solutions of the numerical schemes. It is convenient to define the vectors

$$\mathbf{X^n} = (X_0^n, X_1^n, X_2^n, \ldots, X_N^n, X_{N+1}^n)^T$$

$$\mathbf{Y^n} = (Y_0^n, Y_1^n, Y_2^n, \ldots, Y_N^n, Y_{N+1}^n)^T$$

with T denoting transpose. Using the novel implicit method (5.28) and equation (5.38) in (5.37), we get

$$M_X : \frac{1}{\ell}\left(X_m^{n+1}-X_m^n\right) = rX_m^n - rX_m^n\frac{X_m^{n+1}}{K} - \frac{X_m^{n+1}Y_m^n}{a+(X_m^n)^2}$$

$$+ \quad Ph^{-2}\{\frac{1}{2}\left[X_{m-1}^{n+1} - 2X_m^{n+1} + X_{m+1}^{n+1}\right]$$
$$+ \quad \frac{1}{2}\left[X_{m-1}^{n} - 2X_m^{n} + X_{m+1}^{n}\right]\}, \qquad (5.39)$$

and

$$M_Y : \frac{1}{\ell}\left(Y_m^{n+1} - Y_m^n\right) = Y_m^n \frac{\mu X_m^{n+1}}{a + (X_m^{n+1})^2} - DY_m^{n+1}$$
$$+ \quad Ph^{-2}\{\frac{1}{2}\left[Y_{m-1}^{n+1} - 2Y_m^{n+1} + Y_{m+1}^{n+1}\right]$$
$$+ \quad \frac{1}{2}\left[Y_{m-1}^{n} - 2Y_m^{n} + Y_{m+1}^{n}\right]\}. \qquad (5.40)$$

Equations (5.39) and (5.40) may be rearranged to give

$$M_X : \; -\frac{1}{2}P\rho X_{m-1}^{n+1} - \frac{1}{2}P\rho X_{m+1}^{n+1} + \left[1 + rX_m^n\frac{\ell}{K} + \frac{Y_m^n\ell}{a + (X_m^n)^2} + P\rho\right] X_m^{n+1}$$
$$= \frac{1}{2}P\rho X_{m-1}^n + \frac{1}{2}P\rho X_{m+1}^n + [1 + r\ell - P\rho]X_m^n \qquad (5.41)$$

and

$$M_Y : \quad -\frac{1}{2}P\rho Y_{m-1}^{n+1} - \frac{1}{2}P\rho Y_{m+1}^{n+1} + [1 + D\ell + P\rho]\, Y_m^{n+1}$$
$$= \frac{1}{2}P\rho Y_{m-1}^n + \frac{1}{2}P\rho Y_{m+1}^n + \left[1 - P\rho + \frac{\mu X_m^{n+1}}{a + (X_m^{n+1})^2}\right] Y_m^n \quad ,(5.42)$$

in which $\rho = \ell/h^2$. In order to implement the above method {(5.41)-(5.42)}, we need to solve two sets of linear algebraic equations at each time step using tridiagonal solvers.

5.9 Numerical Experiments

The method {(5.41)-(5.42)} was simulated using initial conditions $X_m^0 = 10\sin\pi z$, $Y_m^0 = 8\sin\pi z$ $(m = 0, 1, \ldots, N+1)$ and Neuman boundary conditions, given by

$$\frac{\partial X}{\partial t}(0,t) = \frac{\partial X}{\partial t}(1,t) = 0, \qquad (5.43)$$

$$\frac{\partial Y}{\partial t}(0,t) = \frac{\partial Y}{\partial t}(1,t) = 0.$$

For simulation purposes, we use $r = 0.1$, $d = 0.2$, $K = 12$, $a = 300$, $\mu = 5$, $\ell = 0.01$ and $h = 0.05$ with various values of the diffusion coefficient P (namely : $P = 0, 1, 2, 5, 100$). The numerous numerical experiments carried out confirm the robustness of the new method {(5.41),(5.42)}. Furthermore, it was observed that the solution converges slowly for small P and faster for larger P. That is, as diffusion plays a more dominant role in the model dynamics, the method converges (to the correct steady state) faster.

Chapter 6

SEIR Epidemic Model

6.1 Introduction

Most classical models of disease transmission are of the SIR type. That is, they monitor the dynamics of the population of susceptible individuals (S), infectious individuals (I) and individuals who have recovered (by acquiring permanent or temporary immunity) (R) [3, 22]. These models are based on the assumption that once infected, each susceptible individual becomes infectious instantaneously.

In many diseases, however, it is known that members do not pass directly from the susceptible to the infectious class on becoming infected. They undergo a latent (or exposed) period of infection instead (see [21]). For example, in measles dynamics there is an incubation (latent) period of 11-14 days after infection before symptoms of the disease appear. For fox rabies, the incubation period is 28-30 days. For some diseases, the incubation period may be short enough that neglecting it does not affect the model significantly. For example, the incubation period for influenza is 1-3 days. To model this, an additional compartment is included in the classical SIR model by letting $E(t)$ denote the number of members, at time t, who are exposed to the disease (but not yet infectious). This type of model is called SEIR model.

The total population size in an SEIR type model is $N(t) = S(t) + E(t) + I(t) + R(t)$.

6.2 Mathematical Model

The total population is divided into four subclasses namely: susceptible (S), exposed (E), infectious (I), and recovered (R). The total infected popu-

lation is $E(t) + I(t)$. A transfer diagram for the model is given in Figure 6.1 below.

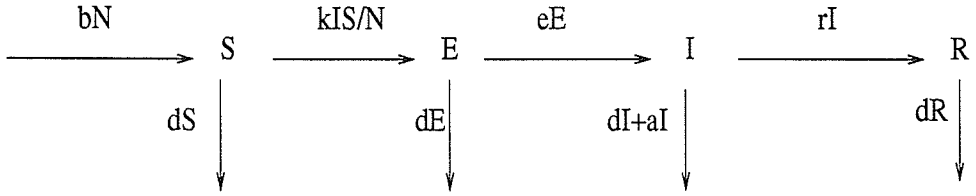

Figure 6.1 The Dynamical transfer of SEIR model

The model parameters b, d, e, k, r, a are interpreted in Table 6.1.

Table 6.1 : Description of model parameters

Parameter	Interpretation	Range
b	Natural birth rate	$b \in (0,1)$
d	Natural death rate	$d \in (0,1)$
e	Rate at which exposed individuals become infectious	$e \in (0,1)$
k	Effective *per capita* contact rate of infective individuals	$k \in (0,1)$
r	Recovery rate	$r \in (0,1)$
a	Disease-related death rate	$a \in (0,1)$

Using the transfer diagram above, we have the following SEIR model ([3]):

$$\begin{aligned} S' &= bN - dS - kIS/N, \\ E' &= kIS/N - (e+d)E, \\ I' &= eE - (r+a+d)I, \\ R' &= rI - dR, \end{aligned} \tag{6.1}$$

from which the total population size $N(t)$ can be determined by $N(t) = S(t) + E(t) + I(t) + R(t)$ or from the equation

$$N' = (b-d)N - aI, \tag{6.2}$$

which is obtained by adding the four equations in (6.1).

6.3 Dynamics of the SEIR Model

In order to understand the dynamics of the SEIR model (6.1), we make the following transformation. Let $\bar{S} = \frac{S}{N}, \bar{E} = \frac{E}{N}, \bar{I} = \frac{I}{N}$ and $\bar{R} = \frac{R}{N}$ denote the fractions of S, E, I and R in the total population N. Differentiating the new variables and substituting into (6.1) gives the normalized system:

$$\begin{aligned}
\bar{S}' &= b - b\bar{S} - k\bar{I}\bar{S} + a\bar{I}\bar{S}, \\
\bar{E}' &= k\bar{I}\bar{S} - (e+b)\bar{E} + a\bar{I}\bar{E}, \\
\bar{I}' &= e\bar{E} - (r+a+b)\bar{I} + a\bar{I}^2, \\
\bar{R}' &= r\bar{I} - b\bar{R} + a\bar{I}\bar{R}.
\end{aligned} \tag{6.3}$$

Since the variable $\bar{R}$ does not occur in the first three equations of the normalized system (6.3), we study the subsystem

$$\begin{aligned}
\bar{S}' &\equiv g_1 = b - b\bar{S} - k\bar{I}\bar{S} + a\bar{I}\bar{S}, \\
\bar{E}' &\equiv g_2 = k\bar{I}\bar{S} - (e+b)\bar{E} + a\bar{I}\bar{E}, \\
\bar{I}' &\equiv g_3 = e\bar{E} - (r+a+b)\bar{I} + a\bar{I}^2,
\end{aligned} \tag{6.4}$$

instead, and then determine $\bar{R}$ using $\bar{R} = 1 - \bar{S} - \bar{E} - \bar{I}$. My main aim in this chapter is to carry out a local stability analysis of the equilibrium solution(s) of (6.4) and then construct an appropriate numerical method for solving the model (quantitatively).

6.3.1 Equilibrium Solutions

It is easy to see that the equilibrium points of the normalized system are:

- $P_0 = (1, 0, 0)$: the disease-free equilibrium ($E^* = I^* = 0$),
- $P^* = (\bar{S}^*, \bar{E}^*, \bar{I}^*)$: the endemic equilibrium (where $\bar{E}^* > 0, \bar{I}^* > 0$).

The endemic equilibrium $P^* = (\bar{S}^*, \bar{E}^*, \bar{I}^*)$ could not be expressed in closed form (using standard mathematical softwares). However, the process by which it is obtained is shown below. Adding the equations in (6.4) gives

$$(b - a\bar{I}^*)(1 - \bar{S}^* - \bar{E}^* - \bar{I}^*) = r\bar{I}^*. \tag{6.5}$$

From the second equation of (6.4), we have

$$\bar{S}^* = \frac{b}{b + k\bar{I}^* - a\bar{I}^*}. \tag{6.6}$$

Similarly, $\bar{E}^*$ is determined from the third equation of (6.4)

$$\bar{E}^* = \frac{r + a + b - a\bar{I}^*}{e}\bar{I}^*. \tag{6.7}$$

Substituting (6.6) and (6.7) into (6.5) leads to

$$f(\bar{I}^*) = \sigma, \tag{6.8}$$

where

$$f(\bar{I}) = \left(1 - \frac{a}{e+b}\bar{I}\right)\left(1 - \frac{a}{r+a+b}\bar{I}\right)\left(1 + \frac{k-a}{b}\bar{I}\right), \tag{6.9}$$

and

$$\sigma = \frac{ke}{(e+b)(r+a+b)}. \tag{6.10}$$

Further details on the existence and uniqueness of P^* can be found in [21].

6.3.2 Local Stability Analysis

The local stability of the disease-free equilibrium are dertermined by investigating the eigenvalues of the Jacobian

$$J = \begin{bmatrix} \frac{\partial g_1}{\partial \bar{S}} & \frac{\partial g_1}{\partial \bar{E}} & \frac{\partial g_1}{\partial \bar{I}} \\ \frac{\partial g_2}{\partial \bar{S}} & \frac{\partial g_2}{\partial \bar{E}} & \frac{\partial g_2}{\partial \bar{I}} \\ \frac{\partial g_3}{\partial \bar{S}} & \frac{\partial g_3}{\partial \bar{E}} & \frac{\partial g_3}{\partial \bar{I}} \end{bmatrix} \tag{6.11}$$

where the functions g_1, g_2 and g_3 are defined in (6.4). It is easy to show that the Jacobian associated with g_1, g_2 and g_3 is the matrix

$$J = \begin{bmatrix} -b - k\bar{I} + a\bar{I} & 0 & -k\bar{S} + a\bar{S} \\ k\bar{I} & -e - b + a\bar{I} & k\bar{S} + a\bar{E} \\ 0 & e & -(r + a + b) + 2a\bar{I} \end{bmatrix}. \tag{6.12}$$

At the disease-free equilibrium point $P_0 = (1, 0, 0)$, J reduces to

$$J = \begin{bmatrix} -b & 0 & -k + a \\ 0 & -e - b & k \\ 0 & e & -(r + a + b) \end{bmatrix}, \tag{6.13}$$

which yields the characteristic equation

$$(\lambda + b)\left[\lambda^2 + (r + a + 2b + e)\lambda + (e + b)(r + a + b) - ke\right]. \tag{6.14}$$

Since all the model parameters are assumed positive (see Table 6.1), it is clear from (6.14) that $r + a + 2b + e > 0$ and $b > 0$. It follows, using the Routh-Hurwitz criterion (see [19, 42]), that the disease-free equilibrium point P_0 is stable provided

$$(e + b)(r + a + b) - ke > 0. \tag{6.15}$$

Thus, bifurcation occurs at the bifurcation boundary (obtained by setting (6.15) to zero)

$$\sigma = \frac{ke}{(e+b)(r+a+b)} = 1. \tag{6.16}$$

The quantity σ, known as the modified contact number [21], plays an important role in the dynamics of the SEIR model. This is because, when $\sigma \in (0,1)$, the disease-free equilibrium P_0 is stable (and the disease dies out from the community). For $\sigma \in (1,\infty)$, however, the disease-free equilibrium loses its stability, and a new stable equilibrium point (P^*) emerges. This equilibrium point (P^*) is associated with the persistence of the disease in the population. In other words, when the model parameters k, e, b, a, r are varied in such a way that σ increases from a positive value less than 1 to a positive value greater than 1, then a stable endemic equlibrium solution (P^*) bifurcates from the disease-free equilibrium point P_0. The reverse process (P^* to P_0) holds when $\sigma \in (1,\infty)$ decreases to $\sigma \in (0,1)$. Detailed global stability analysis of (6.4), summarized in Theorem 6.1 below, was reported by Li *et al.* [21].

Theorem 6.1 *If $\sigma \leq 1$, then the disease dies out, and the disease-free equilibrium point is globally asymptotically stable. If $\sigma > 1$, then the disease becomes endemic, and unique endemic equilibrium P^* is globally asymptotically stable provided that no non-constant periodic solutions exist.*

6.4 Novel (Non-standard) Numerical Method

In line with the numerical schemes constructed in Chapters 3 and 5, a non-standard numerical method that satisfies the positivity requirement of (6.4) will be designed. Using forward Euler approximations for the derivatives in (6.4), as before, and replacing the right hand side functions appropriately

leads to the following method for solving (6.4):

$$M_{\bar{S}} : \frac{1}{\ell}(\bar{S}_{n+1} - \bar{S}_n) = b - b\bar{S}_{n+1} - k\bar{I}_n\bar{S}_{n+1} + a\bar{I}_n\bar{S}_n \tag{6.17}$$

$$M_{\bar{E}} : \frac{1}{\ell}(\bar{E}_{n+1} - \bar{E}_n) = k\bar{I}_n\bar{S}_{n+1} - (e+b)\bar{E}_{n+1} + a\bar{I}_n\bar{E}_n$$

$$M_{\bar{I}} : \frac{1}{\ell}(\bar{I}_{n+1} - \bar{I}_n) = e\bar{E}_{n+1} - (r+a+b)\bar{I}_{n+1} + a(\bar{I}_n)^2$$

$$M_{\bar{R}} : \bar{R}_{n+1} = 1 - \bar{S}_{n+1} - \bar{E}_{n+1} - \bar{I}_{n+1}$$

The equations in (6.17) can be re-arranged to give

$$\bar{S}_{n+1} = \frac{b\ell + \bar{S}_n + a\ell\bar{I}_n\bar{S}_n}{1 + b\ell + k\ell\bar{I}_n},$$

$$\bar{E}_{n+1} = \frac{k\ell\bar{I}_n\bar{S}_{n+1} + \bar{E}_n + a\ell\bar{I}_n\bar{E}_n}{1 + (e+b)\ell},$$

$$\bar{I}_{n+1} = \frac{e\ell\bar{E}_{n+1} + \bar{I}_n + a\ell(\bar{I}_n)^2}{1 + (r+a+b)\ell},$$

$$\bar{R}_{n+1} = 1 - \bar{S}_{n+1} - \bar{E}_{n+1} - \bar{I}_{n+1}. \tag{6.18}$$

Although the Gauss-Seidel-like method (6.17) (see [2, 19]) is constructed implicitly, it can be implemented explicitly using (6.18). Thus, the solution of the SEIR model (6.4) can be computed explicitly.

It is clear from (6.18) that this method preserves the positivity of the SEIR model. The associated local truncation errors of this method is

$$\begin{aligned}
L_{\bar{S}} &= \left\{1 + \ell\left[b + k\bar{I}(t)\right]\right\}\bar{S}(t+\ell) - \bar{S}(t) - b\ell - a\ell\bar{I}_n\bar{S}_n, \qquad (6.19)\\
L_{\bar{E}} &= \{1 + \ell[(e+b)]\}\bar{E}(t+\ell) - \bar{E}(t) - k\ell\bar{I}(t)\bar{S}(t+\ell) - a\ell\bar{I}_n\bar{E}_n,\\
L_{\bar{I}} &= \{1 + \ell[(r+a+b)]\}\bar{I}(t+\ell) - \bar{I}(t) - e\ell\bar{E}(t+\ell) - a\ell(\bar{I}_n)^2,\\
L_{\bar{R}} &= \bar{R}(t+\ell) + \bar{S}(t+\ell) + \bar{E}(t+\ell) + \bar{I}(t+\ell) - 1,
\end{aligned}$$

in which $t = t_n$. Expanding the functions in (6.19) in Taylor series leads to

$$\begin{aligned} L_{\bar{S}} &= \left[\frac{1}{2}\bar{S}'' + b\bar{S}' + k\bar{I}\bar{S}'\right]\ell^2 + O(\ell^3)\ as\ \ell \to 0, \\ L_{\bar{E}} &= \left[\frac{1}{2}\bar{E}'' + (e+b)\bar{E}' - k\bar{I}\bar{S}'\right]\ell^2 + O(\ell^3)\ as\ \ell \to 0, \\ L_{\bar{I}} &= \left[\frac{1}{2}\bar{I}'' + (r+a+b)\bar{I}' - e\bar{E}'\right]\ell^2 + O(\ell^3)\ as\ \ell \to 0, \\ L_{\bar{R}} &= \left[\frac{1}{2}(\bar{R}'' + \bar{S}'' + \bar{E}'' + \bar{I}'')\right]\ell^2 + O(\ell^3)\ as\ \ell \to 0, \end{aligned} \tag{6.20}$$

confirming that the method (6.18) is first-order accurate.

6.5 Numerical Experiments

In order to test the behaviour of method (6.18) for solving the SEIR model, numerous numerical simulations were carried out using the following parameters and initial values: $b = 0.01$, $d = 0.02$, $k = 0.001$, $a = 0.002$, $e = 0.001$, $r = 0.001$, $S(0) = 0.5$, $E(0) = 0.2$, $I(0) = 0.1$ and $R(0) = 0.2$.

6.5.1 Experiment 1: Effect of Time-step, ℓ

To monitor the effect of the step size ($\ell > 0$) on the stability and convergence properties of method (6.18), the method was simulated using the aforementioned parameter values with various step sizes. The results, tabulated in Table 6.2, were compared with those obtained using the standard explicit RK4 and Euler methods. It is clear from Table 6.2 that the non-standard scheme (6.18) is more competitive in terms of numerical stability. Unlike the standard RK4 and Euler methods, method (6.18) did not suffer from any scheme-dependent numerical instability (e.g chaos, oscillations etc).

Although not proven analytically, Table 6.2 suggests that method (6.18) is unconditionally-stable (converging monotonically to the correct equilibrium states of the SEIR model). This is owing to the fact that the non-standard

scheme (6.18), unlike RK4 and Euler method, satisfies the positivity property of (6.4).

Table 6.2 : Effect of time step (ℓ)

ℓ	Euler	RK4	Non-standard method
0.01	convergence	convergence	convergence
1.0	overflow	convergence	convergence
10	overflow	convergence	convergence
100	overflow	oscillation	convergence
200	overflow	overflow	convergence

6.5.2 Experiment 2: Effect of the Modified Contact Number (σ)

As discussed analytically in Section 6.3, the modified contact number σ plays a crucial role in the dynamics of the SEIR model (6.4). Its effect was investigated by simulating method (6.18) with various values of σ. Based on the extensive numerical experiments carried out, we observed that for $\sigma \leq 1$, method (6.18) converges to the disease-free equilibra $P_0 = (1, 0, 0)$. On the other hand, when $\sigma > 1$, method (6.18) converges to the endemic equilibrium P^*. In both these cases, method (6.18) gave results that are consistent with the theoretical predictions of Section 6.3. These results are summarized in Table 6.3 below. Similar results were obtained with various combinations of model parameters.

Table 6.3 : Effect of modified contact number (σ)

σ	$\bar{S}^*$	$\bar{E}^*$	$\bar{I}^*$	$\bar{R}^*$	Comment
0.0069	1.0	0	0	0	P_0 is stable
0.5	1.0	0	0	0	P_0 is stable
1.0	1.0	0	0	0	P_0 is stable
5.59	0.175	0.760	0.059	0.181	P^* is stable
100	0.010	0.912	0.071	0.007	P^* is stable

6.6 Extention to the Reaction-Diffusion Model

As in Chapter 5, the original model (6.1) is extended to the following reaction-diffusion system:

$$\begin{aligned}
\frac{\partial \bar{S}}{\partial t} &= b - b\bar{S} - k\bar{I}\bar{S} + a\bar{I}\bar{S} + D\frac{\partial^2 \bar{S}}{\partial z^2}, \\
\frac{\partial \bar{E}}{\partial t} &= k\bar{I}\bar{S} - (e+d)\bar{E} + a\bar{I}\bar{E} + D\frac{\partial^2 \bar{E}}{\partial z^2}, \\
\frac{\partial \bar{I}}{\partial t} &= e\bar{E} - (r+a+d)\bar{I} + a\bar{I}^2 + D\frac{\partial^2 \bar{I}}{\partial z^2}, \\
\frac{\partial \bar{R}}{\partial t} &= r\bar{I} - b\bar{R} + a\bar{I}\bar{R} + D\frac{\partial^2 \bar{R}}{\partial z^2},
\end{aligned} \tag{6.21}$$

where D is a constant and z is the spatial variable. This system of PDEs is to be solved subject to appropriate boundary and initial conditions.

6.6.1 Numerical Method

In line with the strategy adopted in Chapter 5, method (6.18) will be used for the diffusion-free component of (6.21) and the weighted central-difference approximation is used for the spatial derivatives. This gives:

$$M_{\bar{S}} : \frac{1}{\ell}\left(\bar{S}_m^{n+1} - \bar{S}_m^n\right) = b - b\bar{S}_m^{n+1} - k\bar{I}_m^n\bar{S}_m^{n+1} + a\bar{I}_m^n\bar{S}_m^n$$

$$
\begin{aligned}
&+ \quad Dh^{-2}\{\frac{1}{2}\left[\bar{S}_{m-1}^{n+1} - 2\bar{S}_m^{n+1} + \bar{S}_{m+1}^{n+1}\right] \\
&+ \quad \frac{1}{2}\left[\bar{S}_{m-1}^{n} - 2\bar{S}_m^{n} + \bar{S}_{m+1}^{n}\right]\}, \qquad (6.22)
\end{aligned}
$$

$$
\begin{aligned}
M_{\bar{E}} : \frac{1}{\ell}\left(\bar{E}_m^{n+1} - \bar{E}_m^{n}\right) &= k\bar{I}_m^n\bar{S}_m^{n+1} - (e+b)\bar{E}_m^{n+1} + a\bar{I}_m^n\bar{E}_m^n \\
&+ \quad Dh^{-2}\{\frac{1}{2}\left[\bar{E}_{m-1}^{n+1} - 2\bar{E}_m^{n+1} + \bar{E}_{m+1}^{n+1}\right] \\
&+ \quad \frac{1}{2}\left[\bar{E}_{m-1}^{n} - 2\bar{E}_m^{n} + \bar{E}_{m+1}^{n}\right]\}, \qquad (6.23)
\end{aligned}
$$

$$
\begin{aligned}
M_{\bar{I}} : \frac{1}{\ell}\left(\bar{I}_m^{n+1} - \bar{I}_m^{n}\right) &= e\bar{E}_m^{n+1} - (r+a+b)\bar{I}_m^{n+1} + a(\bar{I}_m^n)^2 \\
&+ \quad Dh^{-2}\{\frac{1}{2}\left[\bar{I}_{m-1}^{n+1} - 2\bar{I}_m^{n+1} + \bar{I}_{m+1}^{n+1}\right] \\
&+ \quad \frac{1}{2}\left[\bar{I}_{m-1}^{n} - 2\bar{I}_m^{n} + \bar{I}_{m+1}^{n}\right]\}, \qquad (6.24)
\end{aligned}
$$

and

$$
\begin{aligned}
M_{\bar{R}} : \frac{1}{\ell}\left(\bar{R}_m^{n+1} - \bar{R}_m^{n}\right) &= r\bar{I}_m^{n+1} - b\bar{R}_m^{n+1} + a\bar{I}_m^{n+1}\bar{R}_m^n \\
&+ \quad Dh^{-2}\{\frac{1}{2}\left[\bar{R}_{m-1}^{n+1} - 2\bar{R}_m^{n+1} + \bar{R}_{m+1}^{n+1}\right] \\
&+ \quad \frac{1}{2}\left[\bar{R}_{m-1}^{n} - 2\bar{R}_m^{n} + \bar{R}_{m+1}^{n}\right]\}. \qquad (6.25)
\end{aligned}
$$

The method (6.22)-(6.25) can be rearranged, respectively, to give

$$
\begin{aligned}
M_{\bar{S}} : \quad & -\frac{1}{2}D\rho\bar{S}_{m-1}^{n+1} - \frac{1}{2}D\rho\bar{S}_{m+1}^{n+1} + \left[1 + b\ell + k\ell\bar{I}_m^n + D\rho\right]\bar{S}_m^{n+1} \\
= \quad & \frac{1}{2}D\rho\bar{S}_{m-1}^{n} + \frac{1}{2}D\rho\bar{S}_{m+1}^{n} + b\ell + [1 - D\rho + a\ell\bar{I}_m^n]\bar{S}_m^n, \qquad (6.26)
\end{aligned}
$$

$$
\begin{aligned}
M_{\bar{E}} : \quad & -\frac{1}{2}D\rho\bar{E}_{m-1}^{n+1} - \frac{1}{2}D\rho\bar{E}_{m+1}^{n+1} + \left[1 + (e+b)\ell - a\ell\bar{I}_m^n + D\rho\right]\bar{E}_m^{n+1} \\
= \quad & \frac{1}{2}D\rho\bar{E}_{m-1}^{n} + \frac{1}{2}D\rho\bar{E}_{m+1}^{n} + k\ell\bar{I}_m^n\bar{S}_m^{n+1} + [1 - D\rho + a\ell\bar{I}_m^n]\bar{E}_m^n \qquad (6.27)
\end{aligned}
$$

$$M_{\bar{I}}: \quad -\frac{1}{2}D\rho\bar{I}_{m-1}^{n+1} - \frac{1}{2}D\rho\bar{I}_{m+1}^{n+1} + \left[1 + (r+a+b)\ell - a\ell\bar{I}_m^n + D\rho\right]\bar{I}_m^{n+1}$$
$$= \frac{1}{2}D\rho\bar{I}_{m-1}^n + \frac{1}{2}D\rho\bar{I}_{m+1}^n + e\ell\bar{E}_m^{n+1} + [1 - D\rho + a\ell\bar{I}_m^n]\bar{I}_m^n, \qquad (6.28)$$

and

$$M_{\bar{R}}: \quad -\frac{1}{2}D\rho\bar{R}_{m-1}^{n+1} - \frac{1}{2}D\rho\bar{R}_{m+1}^{n+1} + \left[1 + b\ell - a\ell\bar{I}_m^{n+1} + D\rho\right]\bar{R}_m^{n+1}$$
$$= \frac{1}{2}D\rho\bar{R}_{m-1}^n + \frac{1}{2}D\rho\bar{R}_{m+1}^{n+1} + r\ell\bar{I}_m^{n+1} + [1 - D\rho + a\ell\bar{I}_m^{n+1}]\bar{R}_m^n \qquad (6.29)$$

6.6.2 Implementation

Method (6.26)-(6.29) may be applied with various initial and boundary condition. In this study, we use the following Neumann boundary conditions:

$$\frac{\partial\bar{S}}{\partial t}(0,t) = \frac{\partial\bar{S}}{\partial t}(1,t) = 0, \ \frac{\partial\bar{E}}{\partial t}(0,t) = \frac{\partial\bar{E}}{\partial t}(1,t) = 0, \qquad (6.30)$$

$$\frac{\partial\bar{I}}{\partial t}(0,t) = \frac{\partial\bar{I}}{\partial t}(1,t) = 0, \ \frac{\partial\bar{R}}{\partial t}(0,t) = \frac{\partial\bar{R}}{\partial t}(1,t) = 0.$$

It is easy to see that method (6.26)-(6.29) can be implemented for $n = 0, 1, 2, \ldots$ and $m = 0, 1, 2, \ldots, N+1$. However, in the cases $m = 0$ and $m = N+1$ (on the boundaries), this method introduces fictitious points outside the mesh. To circumvent this problem, central-difference approximations are used to approximate the zero-flux boundary conditions in (6.30). For instance, on the boundary $m = 0$, the equations in (6.30) can be approximated to second order (in h) using

$$\frac{\bar{S}_1^n - \bar{S}_{-1}^n}{2h} = 0, \ \frac{\bar{E}_1^n - \bar{E}_{-1}^n}{2h} = 0, \qquad (6.31)$$

$$\frac{\bar{I}_1^n - \bar{I}_{-1}^n}{2h} = 0, \ \frac{\bar{R}_1^n - \bar{R}_{-1}^n}{2h} = 0.$$

This leads to the approximations:

$$\bar{S}_1^n = \bar{S}_{-1}^n,\ \bar{E}_1^n = \bar{E}_{-1}^n,\ \bar{I}_1^n = \bar{I}_{-1}^n,\ \bar{R}_1^n = \bar{R}_{-1}^n. \tag{6.32}$$

Similarly, at the boundary $m = N+1$, we have:

$$\bar{S}_{N+2}^n = \bar{S}_N^n,\ \bar{E}_{N+2}^n = \bar{E}_N^n,\ \bar{I}_{N+2}^n = \bar{I}_N^n,\ \bar{R}_{N+2}^n = \bar{R}_N^n. \tag{6.33}$$

Equation (6.26) may be written in matrix-vector form as

$$\mathbf{A}_1^n \bar{\mathbf{S}}^{n+1} = \mathbf{B}_1^n \bar{\mathbf{S}}^n + \bar{\mathbf{b}}\ell;\ n = 0, 1, 2, \ldots, \tag{6.34}$$

where $\mathbf{A}_1^n$ and $\mathbf{B}_1^n$ are square matrices of order $(N+2) \times (N+2)$ given by

$$\mathbf{A}_1^n = \begin{bmatrix} a1_0^n & -D\rho & 0 & \cdots & 0 \\ -\frac{1}{2}D\rho & a1_1^n & -\frac{1}{2}D\rho & & \vdots \\ 0 & -\frac{1}{2}D\rho & a1_2^n & -\frac{1}{2}D\rho & \\ & \ddots & \ddots & \ddots & 0 \\ \vdots & & -\frac{1}{2}D\rho & a1_N^n & -\frac{1}{2}D\rho \\ 0 & \cdots & 0 & -D\rho & a1_{N+1}^n \end{bmatrix}, \tag{6.35}$$

$$\mathbf{B}_1^n = \begin{bmatrix} 1 - D\rho + a\ell\bar{I}_m^n & D\rho & 0 & \cdots & 0 \\ \frac{1}{2}D\rho & 1 - D\rho + a\ell\bar{I}_m^n & \frac{1}{2}D\rho & & \vdots \\ 0 & \frac{1}{2}D\rho & 1 - D\rho + a\ell\bar{I}_m^n & \frac{1}{2}D\rho & \\ & \ddots & \ddots & \ddots & 0 \\ \vdots & & \frac{1}{2}D\rho & 1 - D\rho + a\ell\bar{I}_m^n & \frac{1}{2}D\rho \\ 0 & \cdots & 0 & -D\rho & 1 - D\rho + a\ell\bar{I}_m^n \end{bmatrix}, \tag{6.36}$$

with

$$a1_m^n = 1 + b\ell + k\ell\bar{I}_m^n + D\rho;\ \ (m = 0, 1, 2, \ldots, N+1), \tag{6.37}$$

and $\bar{\mathbf{b}}$ is an $(N+2)$ vector given by

$$\bar{\mathbf{b}} = [b, b, \ldots, b, b]^T \tag{6.38}$$

Similarly, equation (6.27)-(6.29) may be written in matrix-vector form as

$$\begin{aligned}
\mathbf{A}_2^n\bar{\mathbf{E}}^{n+1} + \mathbf{C}_2^n\bar{\mathbf{S}}^{n+1} &= \mathbf{B}_1^n\bar{\mathbf{E}}^n;\ n = 0, 1, 2, \ldots, \\
\mathbf{A}_3^n\bar{\mathbf{I}}^{n+1} + \mathbf{C}_3^n\bar{\mathbf{E}}^{n+1} &= \mathbf{B}_1^n\bar{\mathbf{I}}^n;\ n = 0, 1, 2, \ldots, \\
\mathbf{A}_4^n\bar{\mathbf{R}}^{n+1} + \mathbf{C}_4^n\bar{\mathbf{I}}^{n+1} &= \mathbf{B}_2^n\bar{\mathbf{R}}^n;\ n = 0, 1, 2, \ldots.
\end{aligned} \tag{6.39}$$

where $\mathbf{B}_2^n$ is same as $\mathbf{B}_1^n$ except that its diagonal element is $1 - D\rho + a\ell\bar{I}_m^{n+1}$ instead of $1 - D\rho + a\ell\bar{I}_m^n$, and $\mathbf{A}_2^n, \mathbf{A}_3^n, \mathbf{A}_4^n$ and $\mathbf{C}_2^n, \mathbf{C}_3^n, \mathbf{C}_4^n$ are square matrices of order $(N+2) \times (N+2)$ given by

$$\mathbf{A}_2^n = \begin{bmatrix}
a2_0^n & -D\rho & 0 & \cdots & 0 \\
-\frac{1}{2}D\rho & a2_1^n & -\frac{1}{2}D\rho & & \vdots \\
0 & -\frac{1}{2}D\rho & a2_2^n & -\frac{1}{2}D\rho & \\
 & \ddots & \ddots & \ddots & 0 \\
\vdots & & -\frac{1}{2}D\rho & a2_N^n & -\frac{1}{2}D\rho \\
0 & \cdots & 0 & -D\rho & a2_{N+1}^n
\end{bmatrix}, \tag{6.40}$$

$$\mathbf{A}_3^n = \begin{bmatrix}
a3_0^n & -D\rho & 0 & \cdots & 0 \\
-\frac{1}{2}D\rho & a3_1^n & -\frac{1}{2}D\rho & & \vdots \\
0 & -\frac{1}{2}D\rho & a3_2^n & -\frac{1}{2}D\rho & \\
 & \ddots & \ddots & \ddots & 0 \\
\vdots & & -\frac{1}{2}D\rho & a3_N^n & -\frac{1}{2}D\rho \\
0 & \cdots & 0 & -D\rho & a3_{N+1}^n
\end{bmatrix}, \tag{6.41}$$

$$\mathbf{A}_4^n = \begin{bmatrix}
a4_0^n & -D\rho & 0 & \cdots & 0 \\
-\frac{1}{2}D\rho & a4_1^n & -\frac{1}{2}D\rho & & \vdots \\
0 & -\frac{1}{2}D\rho & a4_2^n & -\frac{1}{2}D\rho & \\
 & \ddots & \ddots & \ddots & 0 \\
\vdots & & -\frac{1}{2}D\rho & a4_N^n & -\frac{1}{2}D\rho \\
0 & \cdots & 0 & -D\rho & a4_{N+1}^n
\end{bmatrix}, \tag{6.42}$$

$$\mathbf{C}_2^n = \begin{bmatrix} -k\ell\bar{I}_m^n & & & \cdots & 0 \\ & -k\ell\bar{I}_m^n & & & \vdots \\ & & -k\ell\bar{I}_m^n & & \\ & \ddots & \ddots & \ddots & \\ \vdots & & & -k\ell\bar{I}_m^n & \\ 0 & \cdots & & & -k\ell\bar{I}_m^n \end{bmatrix}, \tag{6.43}$$

$$\mathbf{C}_3^n = \begin{bmatrix} -e\ell & & & \cdots & 0 \\ & -e\ell & & & \vdots \\ & & -e\ell & & \\ & \ddots & \ddots & \ddots & \\ \vdots & & & -e\ell & \\ 0 & \cdots & & & -e\ell \end{bmatrix}, \tag{6.44}$$

$$\mathbf{C}_4^n = \begin{bmatrix} -r\ell & & & \cdots & 0 \\ & -r\ell & & & \vdots \\ & & -r\ell & & \\ & \ddots & \ddots & \ddots & \\ \vdots & & & -r\ell & \\ 0 & \cdots & & & -r\ell \end{bmatrix}, \tag{6.45}$$

in which

$$\begin{aligned} a2_m^n &= 1 + (e+b)\ell + D\rho, \\ a3_m^n &= 1 + (r+a+b)\ell + D\rho, \\ a4_m^n &= 1 + b\ell + D\rho. \end{aligned} \tag{6.46}$$

At each time step, a tri-diagonal solver is employed to solve for $\bar{S}$, $\bar{E}$, $\bar{I}$, and $\bar{R}$ consecutively using (6.34) and (6.39).

6.6.3 Stability and Convergence

To analyse the stability of method (6.26)-(6.29), we consider its matrix form

$$\begin{aligned} \mathbf{A}_1^n \bar{\mathbf{S}}^{n+1} &= \mathbf{B}_1^n \bar{\mathbf{S}}^n + \bar{\mathbf{b}}\ell, \qquad (6.47) \\ \mathbf{A}_2^n \bar{\mathbf{E}}^{n+1} + \mathbf{C}_2^n \bar{\mathbf{S}}^{n+1} &= \mathbf{B}_1^n \bar{\mathbf{E}}^n, \\ \mathbf{A}_3^n \bar{\mathbf{I}}^{n+1} + \mathbf{C}_3^n \bar{\mathbf{E}}^{n+1} &= \mathbf{B}_1^n \bar{\mathbf{I}}^n, \\ \mathbf{A}_4^n \bar{\mathbf{R}}^{n+1} + \mathbf{C}_4^n \bar{\mathbf{I}}^{n+1} &= \mathbf{B}_2^n \bar{\mathbf{R}}^n. \end{aligned}$$

The system (6.47) may be written as a single system of order $4(N+2) \times 4(N+2)$ given by

$$\mathbf{K}^n \mathbf{Y}^{n+1} = \mathbf{M}^n \mathbf{Y}^n + \ell \mathbf{Z}, \qquad (6.48)$$

in which

$$\mathbf{K}^n = \begin{bmatrix} A_1^n & 0 & 0 & 0 \\ C_2^n & A_2^n & 0 & 0 \\ 0 & C_3^n & A_3^n & 0 \\ 0 & 0 & C_4^n & A_4^n \end{bmatrix}, \qquad (6.49)$$

$$\mathbf{M}^n = \begin{bmatrix} B_1^n & 0 & 0 & 0 \\ 0 & B_1^n & 0 & 0 \\ 0 & 0 & B_1^n & 0 \\ 0 & 0 & 0 & B_2^n \end{bmatrix}, \qquad (6.50)$$

with

$$\begin{aligned} \mathbf{Y}^n &= \left[(\bar{\mathbf{S}}^n)^T, (\bar{\mathbf{E}}^n)^T, (\bar{\mathbf{I}}^n)^T, (\bar{\mathbf{R}}^n)^T \right]^T, \\ \mathbf{Y}^{n+1} &= \left[(\bar{\mathbf{S}}^{n+1})^T, (\bar{\mathbf{E}}^{n+1})^T, (\bar{\mathbf{I}}^{n+1})^T, (\bar{\mathbf{R}}^{n+1})^T \right]^T, \\ \mathbf{Z} &= \left[(\bar{\mathbf{b}}^n)^T, \mathbf{0}^T \right]^T. \qquad (6.51) \end{aligned}$$

In (6.49,6.50), $\mathbf{0}$ is the zero matrix of order $N+2$, and in (6.51), 0 is the zero column vector of order $3(N+2)$. As discussed in Chapter 3, the matrix

method [19] requires

$$||(\mathbf{K}^n)^{-1}\mathbf{M}^n||_S \leq 1, \tag{6.52}$$

for stability.

6.6.4 Numerical Experiments

The implementation of method (6.26)-(6.29) to solve (6.21) requires the use of tri-diagonal solvers to solve four linear algebraic equations at every time step. Extensive numerical experiments were carried out. $0 \leq z \leq 1$, $\ell = 0.01$ and $h = 0.05$ were used. And we let $b = 0.01$, $d = 0.02$, $k = 0.001$, $a = 0.002$, $e = 0.001$, and $r = 0.001$. Initial conditions are given by $S_m^0 = 0.5\sin\pi z, E_m^0 = 0.2\sin\pi z, I_m^0 = 0.1\sin\pi z, R_m^0 = 0.2\sin\pi z$, $(m = 0, 1, 2, \ldots, N+1)$. The boundary conditions in (6.30) were used.

Various values of the modifiled contact number (σ) and diffusion coefficient (D) were used in order to monitor their effects on the reaction-diffusion model (6.21). Figures 6.2-6.5 depict the profiles of $\bar{S}$, $\bar{E}$, $\bar{I}$ and $\bar{R}$ respectively at $t = 5$ using $\sigma = 0.006993$ and $D = 0.001$. And, figures 6.6-6.9 depict the profiles of $\bar{S}$, $\bar{E}$, $\bar{I}$ and $\bar{R}$ respectively at $t = 5$ using $\sigma = 5.594406$ and $D = 0.001$. In all the simulations carried out, the method (6.26)-(6.29) gave results that are consistent with the theoretical analysis of the SEIR model.

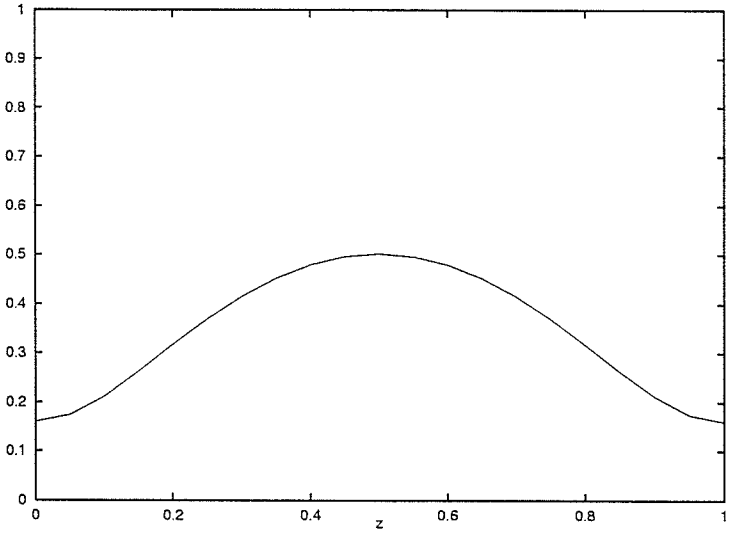

Figure 6.2 Profile of $\bar{S}$ at t=5 for σ=0.006993 and D=0.001 with h=0.05 and ℓ=0.01

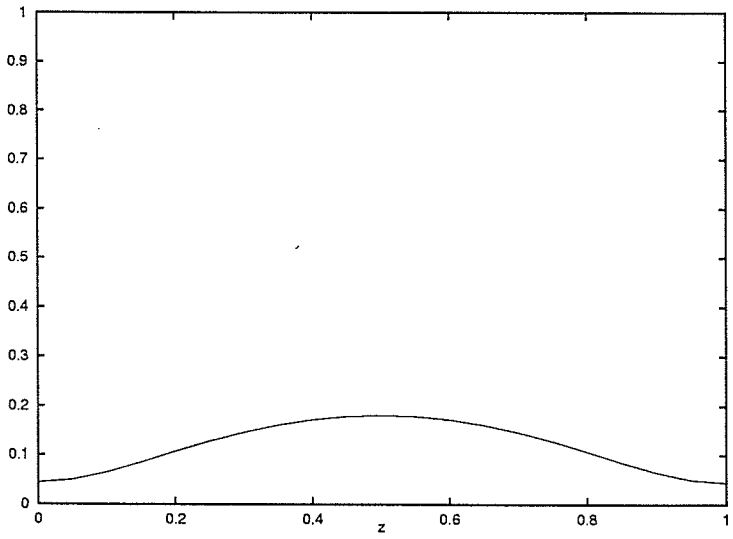

Figure 6.3 Profile of $\bar{E}$ at t=5 for σ=0.006993 and D=0.001 with h=0.05 and ℓ=0.01

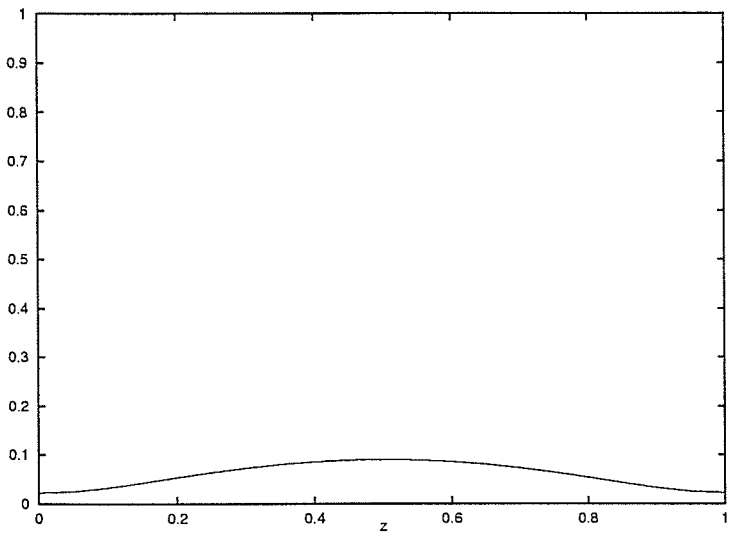

Figure 6.4 Profile of $\bar{I}$ at t=5 for σ=0.006993 and D=0.001 with h=0.05 and ℓ=0.01

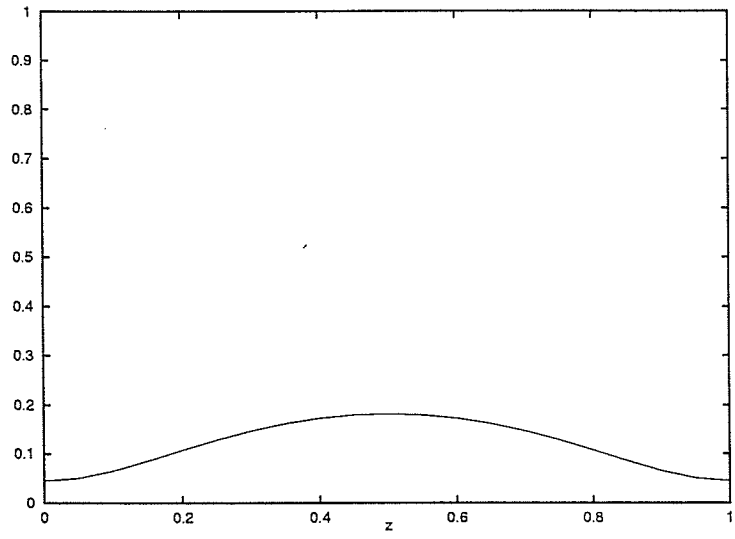

Figure 6.5 Profile of $\bar{R}$ at t=5 for σ=0.006993 and D=0.001 with h=0.05 and ℓ=0.01

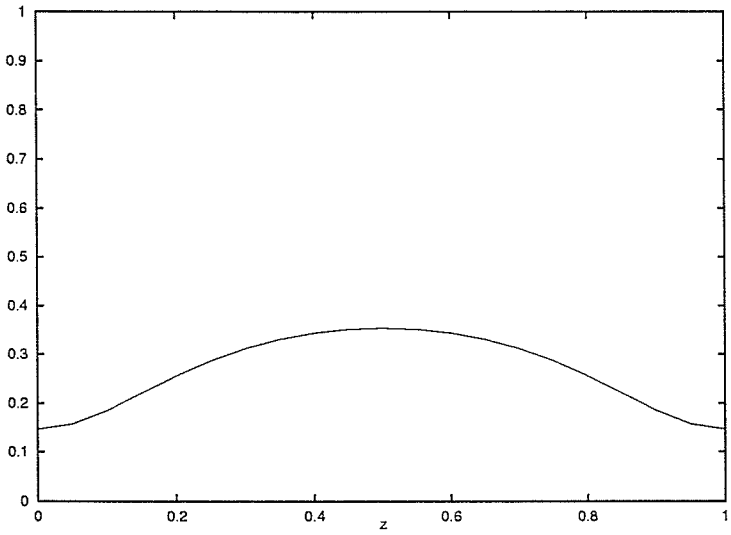

Figure 6.6 Profile of $\bar{S}$ at t=5 for σ=5.594406 and D=0.001 with h=0.05 and ℓ=0.01

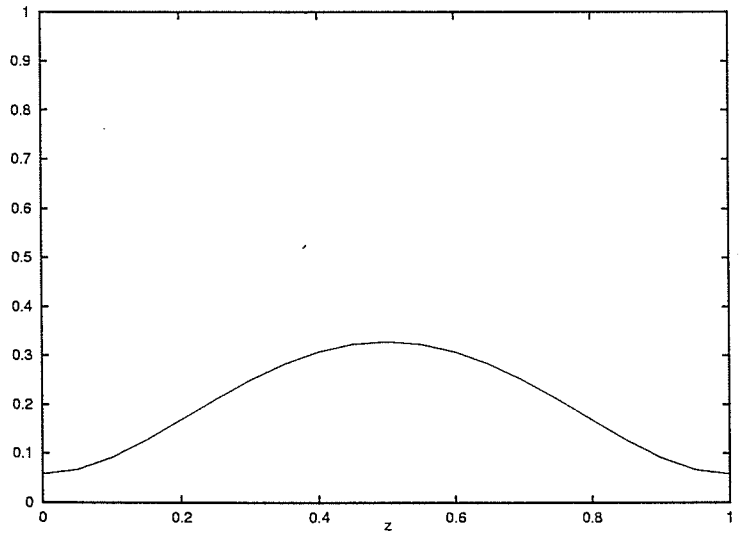

Figure 6.7 Profile of $\bar{E}$ at t=5 for σ=5.594406 and D=0.001 with h=0.05 and ℓ=0.01

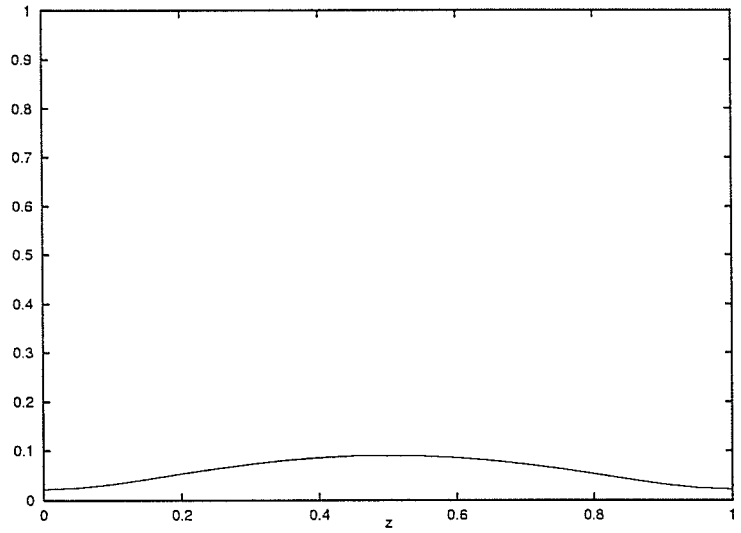

Figure 6.8 Profile of $\bar{I}$ at t=5 for σ=5.594406 and D=0.001 with h=0.05 and ℓ=0.01

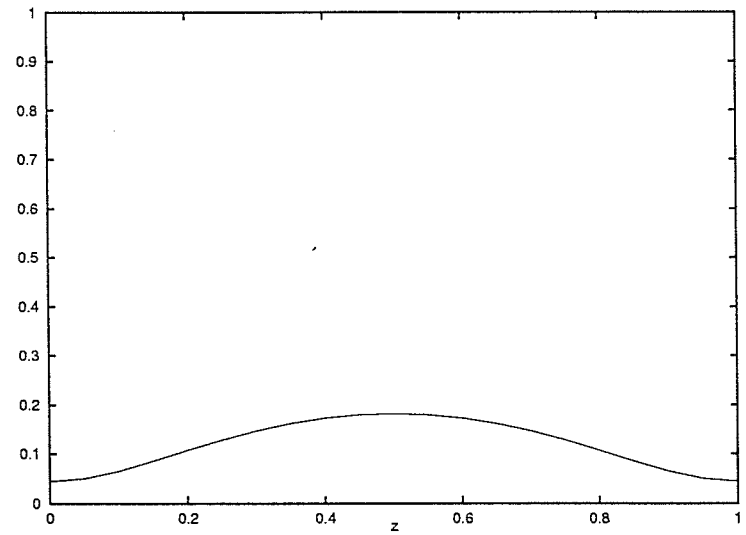

Figure 6.9 Profile of $\bar{R}$ at t=5 for σ=5.594406 and D=0.001 with h=0.05 and ℓ=0.01

6.7 Reaction-Diffusion-Convection Model

Assuming travelling wave type (disease) transmission, the SEIR model is extended to the following reaction-diffusion-convection system:

$$\begin{aligned}
\bar{S}' &= b - b\bar{S} - k\bar{I}\bar{S} + a\bar{I}\bar{S} + \alpha\frac{\partial \bar{S}}{\partial z} + D\frac{\partial^2 \bar{S}}{\partial z^2}, \\
\bar{E}' &= k\bar{I}\bar{S} - (e+d)\bar{E} + a\bar{I}\bar{E} + \alpha\frac{\partial \bar{E}}{\partial z} + D\frac{\partial^2 \bar{E}}{\partial z^2}, \\
\bar{I}' &= e\bar{E} - (r+a+d)\bar{I} + a\bar{I}^2 + \alpha\frac{\partial \bar{I}}{\partial z} + D\frac{\partial^2 \bar{I}}{\partial z^2}, \\
\bar{R}' &= r\bar{I} - b\bar{R} + a\bar{I}\bar{R} + \alpha\frac{\partial \bar{R}}{\partial z} + D\frac{\partial^2 \bar{R}}{\partial z^2}.
\end{aligned} \tag{6.53}$$

In (6.53), α is the convection constant. A robust scheme for solving (6.53) is constructed by using the above method for the reaction-diffusion component of (6.53) and then use the following weighted central-difference approximation for the convection term:

$$\begin{aligned}
\alpha\left[\frac{\partial \bar{S}}{\partial z}\right] &\approx \frac{\alpha}{2h}\{\frac{1}{2}\left[\bar{S}(z+h,t+\ell) - \bar{S}(z-h,t+\ell)\right] \\
&+ \frac{1}{2}\left[\bar{S}(z+h,t) - \bar{S}(z-h,t)\right]\}
\end{aligned} \tag{6.54}$$

The resulting method is given by:

$$\begin{aligned}
M_{\bar{S}}: \quad & \left(-\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{S}_{m-1}^{n+1} - \left(\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{S}_{m+1}^{n+1} \\
&+ \left[1 + b\ell + k\ell\bar{I}_m^n + D\rho\right]\bar{S}_m^{n+1} \\
&= \left(\frac{1}{2}D\rho - \frac{\alpha\ell}{4h}\right)\bar{S}_{m-1}^{n} + \left(\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{S}_{m+1}^{n} \\
&+ b\ell + [1 - D\rho + a\ell\bar{I}_m^n]\bar{S}_m^n,
\end{aligned} \tag{6.55}$$

$$\begin{aligned}
M_{\bar{E}}: \quad & \left(-\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{E}_{m-1}^{n+1} - \left(\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{E}_{m+1}^{n+1} \\
&+ \left[1 + (e+b)\ell + D\rho\right]\bar{E}_m^{n+1} \\
&= \left(\frac{1}{2}D\rho - \frac{\alpha\ell}{4h}\right)\bar{E}_{m-1}^{n} + \left(\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{E}_{m+1}^{n} \\
&+ k\ell\bar{I}_m^n\bar{S}_m^{n+1} + [1 - D\rho + a\ell\bar{I}_m^n]\bar{E}_m^n,
\end{aligned} \tag{6.56}$$

$$
\begin{aligned}
M_{\bar{I}}: \quad & \left(-\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{I}_{m-1}^{n+1} - \left(\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{I}_{m+1}^{n+1} \\
+ \quad & [1 + (r + a + b)\ell + D\rho]\,\bar{I}_m^{n+1} \\
= \quad & \left(\frac{1}{2}D - \frac{\alpha\ell}{4h}\right)\rho\bar{I}_{m-1}^{n} + \left(\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{I}_{m+1}^{n} \\
+ \quad & e\ell\bar{E}_m^{n+1} + [1 - D\rho + a\ell\bar{I}_m^n]\bar{I}_m^n, \qquad (6.57)
\end{aligned}
$$

and

$$
\begin{aligned}
M_{\bar{R}}: \quad & \left(-\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{R}_{m-1}^{n+1} - \left(\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{R}_{m+1}^{n+1} \\
+ \quad & [1 + b\ell + D\rho]\,\bar{R}_m^{n+1} \\
= \quad & \left(\frac{1}{2}D\rho - \frac{\alpha\ell}{4h}\right)\bar{R}_{m-1}^{n} + \left(\frac{1}{2}D\rho + \frac{\alpha\ell}{4h}\right)\bar{R}_{m+1}^{n} \\
+ \quad & r\ell\bar{I}_m^{n+1} + [1 - D\rho + a\ell\bar{I}_m^{n+1}]\bar{R}_m^n. \qquad (6.58)
\end{aligned}
$$

The above method, consisting of equations (6.55), (6.56), (6.57) and (6.58), proved to be robust when used to solve (6.53) subject to the boundary condtions and initial conditions (together with parameter values) used in section 6.2.

Chapter 7

Conclusion

Inevitably, the study of partial differential equations is a large undertaking and falls into several areas of mathematics. At one extreme, the main interest is in the existence and uniqueness of solutions. At the other extreme lies the search for useful (correct) solutions using analytical or numerical means. In this thesis, I presented some new numerical methods, mostly non-standard, for solving some non-linear differential equation models which frequently feature in the mathematical modelling of phenomena in engineering, epidemiology and finance. Assuming the original mathematical models are well-posed, my emphasis is on the design and analysis of robust numerical schemes for their solutions.

In Chapter 3, the well-known one-dimensional Nagumo reaction-diffusion equation was studied by, first of all, designing a competitive scheme for the diffusion-free case. This scheme has the same stability properties as the original (associated) ODE (resulting from rigorous local stability analysis). The scheme is then adapted and extended for use to solve the full Nagumo reaction-diffusion model. Comparisons with standard explicit schemes like forward Euler and Runge-Kutta methods were carried out.

L_0-stable methods, based on the use of the $(2,0)$ Padé approximant, were designed for the transformed and non-transformed Black-Scholes option pricing model in Chapter 4. In addition to having the desirable L_0-stability property, these methods can be implemented in a parallel architecture involving two processors with each processor solving a single algebraic linear system using tridiagonal solvers at each time-step (thereby minimizing CPU time).

The last two models considered in this thesis were systems of non-linear ODE and PDE models. For the predator prey model (with a non-monotonic

functional response) considered in Chapter 5, a new non-standard finite-difference method satisfying the positive conditions of the model was designed. This scheme, which is qualitatively equivalent to the original ODE model, was adapted to solve the predator-prey model with diffusion.

In Chapter 6, an SEIR model for disease transmission was studied. Here, a new Gauss-Seidel type implicitly-derived explicit method was constructed for the associated non-linear ordinary differential systems. Extensive numerical simulation confirms that this robust scheme is chaos-free, unconditionally stable, and converges to the correct steady-state solution of the model. In order to analyse the effect of mobility (random movement) of the populace, the SEIR model was extended to a reaction-diffusion model. Finally, a convection term was added to the reaction-diffusion model in order to model travelling wave-type transmission (modelling airborne spread of diseases). A robust scheme was then designed for the resulting reaction-diffusion-convection system.

The main contribution of this thesis is two-fold. The first is the dynamics (local) stability analysis of the associated ODE systems (necessary to understand the qualitative behaviour of the models). The second is the design and analyses of numerous robust numerical methods (mostly non-standard) for the solutions of both the ODE and the extended PDE systems studied in this thesis. The robust schemes, constructed in Chapters 3-6, proved to be more competitive than the standard numerical methods that are popularly used in the numerical analysis literature for the solution of differential equations (such as forward Euler and Runge-Kutta methods).

Considering the fact that most of the non-standard methods constructed in this thesis, although very robust in terms of stability and convergence, are only first-order accurate, a significant extension of this work will be to extend them (non-standard methods) to achieve higher-order accuracy.

References

[1] K. T. ALLIGOOD, T. D. SAUER AND J. A. YORKE, *Chaos, An Introduction to Dynamical Systems*, Springer, 1997.

[2] W. F. AMES, *Numerical Methods for Partial Differential Equations*, Academic Press, 1977.

[3] R.M. ANDERSON AND R. M. MAY, *Population Biology of Infectious Disease I*, Nature **180**(1979), pp. 361-367.

[4] R. ARIS, *The Mathematical Theory of Diffusion and Reaction in Permeable Catalysts*, Oxford Univ. Press, **Vols. I and II**, 1975.

[5] F. BLACK AND M. SCHOLES, *The Pricing of Options and Corporate Liabilities*, Journal of Political Economy **81**(1973), pp. 637-654.

[6] M. BRENNAN AND E. SCHWARTZ, *The Valution of American Put Options*, Journal of finance **32**(1977), pp. 323-338.

[7] N.F. BRITTON, *Reaction-Diffusion Equations and Their Applications to Biology*, Academic Press, 1986.

[8] M. A. CELIA AND W. G. GRAY, *Numerical Methods for Partial Differential Equations*, Prentice Hall, 1992.

[9] G. COURTEDON, *A More Accurate Finite Difference Approximation for the Valuation of Options*, Journal of Financial and Quantitative Analysis **17**(1982), pp. 697-705.

[10] R. L. DEVANEY, *An Introduction to Chaotic Dynamical Systems* Addison-Wesley Publishing Company **Second Edition**, 1989.

[11] G. EVANS, J. BLACKLEDGE AND P. YARDLEY, *Numerical Methods for Partial Differential Equations*, Springer, 1999.

[12] R. C. FIFE, *Mathematical Aspects of Reacting and Diffusing Systems. "Lecture Notes in Biomathematics"*, Springer-Verleg, **Vol 28**, 1979.

[13] W. E. FITZGIBBON, *Partial Differential Equations and Dynamics Systems*, Pitman Advanced Publishing Program, 1984.

[14] H. I. FREEDMAN AND G.S.K. WOLKOWICZ, *Predator-Prey System with Group Defense: The Paradox of Enrichment Revisited*, Bull. Math. Biol. **48**(1986), pp. 493-508.

[15] E. GALLOPULOS AND Y. SAAD, *On the Parallel Solution of Parabolic Equations*, **CSRD Report 854**, University of Illinois, Urbana-Champaign, 1989.

[16] R. GESKE AND K. SHASTRI, *Valuation by Approximation: A comparison of Alternative Option Valuation Techniques*, Journal of Financial and Quantitative Analysis **20**(1985), pp. 45-71.

[17] A. B. GUMEL, E. H. TWIZELL AND M. A. ARIGU, *Parallel L_0-stable Methods for the Multi-Dimensional Diffusion Equation*, Parallel Algorithms and Applications **11**(1997), pp. 13-25.

[18] J. HULL, *Options, Futures and Other Derivatives*, Prentice Hall, **Fouth Edition**, 1999.

[19] J. D. LAMBERT, *Numerical Methods for Ordinary Differential Systems, the Initial Value Problem*, John Wiley, 1991.

[20] J. D. LAWSON AND J. L. MORRIS, *The Extrapolation of First-order Methods for Parabolic Partial Differential Equations I*, SIAM J. Numer. Anal. **15-6**(1978), pp. 1212-1224.

[21] M. Y. Li, J. R. Graff, L. C. Wang and J. Karsai, *Global Dynamics of a SEIR Model With Varying Total Population Size*, Math. Biosci. **160**(1999), pp. 191-213.

[22] R. M. May and R. M. Anderson, *Population Biology of Infectious Disease I*, Nature **280**(1979), pp. 455-461.

[23] A. Mayo, *Fourth order Accurate Implicit Finite Difference Method for Evaluating American Options*, J. on Scientific and Statistical Computing (2000).

[24] R. E. Mickens, *Nonstandard Finite Difference Schemes for Reaction-Diffusion Equation*, Numerical Methods for Partial Differential Equations **15**(1999), pp. 201-224.

[25] R. E. Mickens, *Applications of Non-standard Finite-difference Schemes*, World Scientific, New Jersey, 2000.

[26] K. Mischaikow and G.S.K. Wolkowicz, *A Predator-Prey System Involving Group Defense: A Connection Matrix Approach*, Nonlinear Analysis **14**(1990), pp. 955-969.

[27] A. R. Mitchell and J. C. Brunch, *A Numerical Study of Chaos in a Reaction-Diffusion Equation*, Numer. Methods Partial Different. Equ. **1**(1985), pp. 13-23.

[28] V. S. Monoranjan, *Bifurcation Studies in Reaction-Diffusion II*, J. Comput. Appl. Math **11**(1984), pp. 27-37.

[29] W. G. Price, Yigong Wang and E. H. Twizell, *A Second-order, Chaos-free, Explicit Method for the Numerical Solution of a Cubic Reation Problem in Neurophysiology*, Numer. Methods Partial Differential Equ. **9**(1993), pp. 213-224.

[30] M. F. REUSCH, L. RATZAN, N. POMPHREY AND W. PARK, *Diagonal Padé Approximations for Initial Value Problems*, SIAM J. Sci. Stat. Comput. **9**(1988), pp. 829-838.

[31] SHIGUI RUAN AND DONGMEI XIAO, *Global Analysis in a Predator-Prey System with Nonmonotonic Functional Response*, SIAM J. Appl. Math. **61**(2001), pp. 1445-1472.

[32] B. D. SLEEMAN, *Analysis of Diffusion Equations in Biology*, Bull. IMA **17**(1981), pp. 7-13.

[33] E. H. TWIZELL, *Numerical Methods, With Applications in the Biomedical Sciences*, John Wiley and Sons, New York, 1988.

[34] E. H. TWIZELL, A. B. GUMEL AND Q. CAO, *A Second-order Scheme for the "Brusselator" Reaction-diffusion System*, Journal of Mathematical Chemistry **26**(1999), pp. 297-316.

[35] E. H. TWIZELL, YIGONG WANG AND W. G. PRICE, *Chaos-free Numerical Solutions of Reaction-diffusion Equations*, Proc. R. Soc London Ser. A **430**(1990), pp. 541-576.

[36] R. S. VARGA, *On Higher Order Stable Implicit Methods for Solving Parabolic Partial Differential Equations*, J. Math. Physics **40**(1961), pp. 220-231.

[37] R. S. VARGA, *Matrix Iterative Analysis*, Prentice Hall, 1962.

[38] F. VERHULST, *Nonlinear Differential Equations and Dynamical Systems*, Springer-Verlag, 1990.

[39] P. WILMOTT, S. HOWISON, J. DEWYNNE, *The Mathematics of Financial Derivatives, a Student Introduction*, Cambridge University Press, 1995.

[40] G. S. K. WOLKOWICZ, *Bifurcation Analysis of a Predator-Prey System Involving Group Defence*, SIAM J. Appl. Math. **48**(1988), pp. 592-606.

[41] P. YU, *Nonlinear Dynamics*, AM524 Notes, Dept. of Applied Math., UWO, 2000.

[42] P. YU AND Q. BI, *Analysis of Non-linear Dynamics and Bifurcations of Double Pendulum*, Journal of Sound and Vib. **217**(1998), pp. 691-736.

[43] V. ZAKIAN, *Comment on Rational Approximations to the Matrix Exponential*, Electron. lett. **7**(1971), pp. 261-262.