Numerical Treatment of Some Types of Nonlinear Partial Differential Equations

Zaki Mrzog H Alaofi

Thesis submitted in fulfilment of the requirement for the degree of Doctor of Philosophy

Victoria University, Australia

Institute for Sustainable Industries and Liveable Cities

May 2023

Abstract

Over the last decade, many quality research papers and monographs have been published focusing on numerical approximations of nonlinear partial differential equations (NPDEs). These equations are very important in mathematics and relevant to the study of various real-life phenomena from nature, physics, engineering and sciences. In this thesis, the cubic B-spline (CBS), non-polynomial spline, fractional calculus and Adomian decomposing methods are used to approximate solutions to the dissipative wave, the dispersive partial differential, coupled nonlinear non-homogeneous Klein–Gordon, linear space-fractional telegraph partial differential and generalised Burgers–Huxley equations. These approximate solutions have been proven to be stable and convergent in various studies. The numerical examples considered in this paper illustrate the efficiency of the method compared with those used in recent works published in this field.

This thesis investigates the treatment of some PDEs using numerical methods. One of the objectives of this thesis is to obtain accurate and constant numerical solutions to nonlinear integer and fractional order NPDEs.

The first chapter presents a general introduction, motivation for the study, research questions, contributions and objectives of the research and the research methodology, and outlines the thesis organisation.

Chapter 2 covers four main topics: PDEs, the B-spline method, the fractional calculus method and the Adomian method.

Chapter 3 focuses on numerical approximations to solve the dissipative wave equation based on the CBS method. The steps followed involve the governing equation and derivation of the proposed method; the initial state; stability analysis; and numerical examples.

Chapter 4 applies the non-polynomial spline method to identify an approximation solution for the third-order dispersive PDE. The steps followed involve analysis of the method; error analysis; stability analysis; and numerical examples.

Chapter 5 provides an approximate analysis for coupled nonlinear non-homogeneous Klein–Gordon equations using the CBS method. The steps followed involve the numerical method; stability analysis; and numerical examples.

Chapter 6 discusses the fractional calculus method for solving the linear space-fractional telegraph PDE. The steps followed involve derivation of the method; the spline relations; stability analysis; and numerical examples.

Chapter 7 investigates solution of the generalised Burgers–Huxley equation with high-order nonlinearity terms using the Adomian decomposition method. The steps followed involve global exponential stability; construction of the adaptive boundary control; the initial boundary value problem; and numerical examples.

The last chapter provides a summary of the thesis and makes some suggestions for further research.

Dedication

To the one who has taught me the tender without waiting; whose name I have carried with pride and respect, My Father.

To the meaning of love and the meaning of affection and dedication, the smile that solved all my life's mystery of existence, My Mother.

To those to whom I could never find enough words to express my appreciation, the ones whose support and value pushed me further in life, My Family.

To those who gave the best help and support, My Brothers.

To the very devoted, the candles who illuminated the darkness of my life, My Sisters.

To the distinguished, eminent, ever-caring friends, the ones who stood by me throughout my journey, the ones who kept on checking and asking and stayed with me on the path of success and goodness.

Declaration

"I, Zaki Mrzog Alaofi, declare that the PhD thesis entitled Numerical Treatment of Some Types of Nonlinear Partial Differential Equation is no more than 80,000 words in length including quotes and exclusive of tables, figures, appendices, bibliography, references and footnotes. This thesis contains no material that has been submitted previously, in whole or in part, for the award of any other academic degree or diploma. Except where otherwise indicated, this thesis is my own work".

"I have conducted my research in alignment with the Australian Code for the Responsible Conduct of Research and Victoria University's Higher Degree by Research Policy and Procedures.

Signature  Date 30/05/2023

Acknowledgements

In the name of Allah, The Most Gracious, The Most Merciful.

Praise be to Allah who gave me the strength, the inspiration and the ability to proceed successfully to bring this thesis to a close. Peace be upon His messenger Muhammad and his noble family.

Many people have helped me during preparation of my PhD thesis. I would like to thank them all. However, there were some with whom I interacted more closely and who I would like to point out here.

I wish to express my gratitude to my advisor Prof Sever Dragomir for supporting me academically; I'd like to thank him for his vast amount of knowledge and inspiration, and his steady help and needed criticism. It truly had a great impact on my choice to pursue graduate studies. I could never have done this without his patience and constant guidance. His emphasis on quality encouraged me to work diligently and thoroughly in my research.

Similarly, my deep gratitude genuinely goes to my advisor, Prof El-Danaf for his constant assistance. I will never be able to thank him enough for his patience and attentiveness, his constant support and encouragement. His wonderful guidance and insightful discussions with me were crucial to the successful completion of this research. I want to thank him again and again for his immense wisdom and endless patience, his constructive comments, guidance and support throughout my study. I would not have been able to complete this work without his friendly, generous and distinguished supervision.

I also wish to thank my collaborators Dr Adel Hadhoud and Dr Faisal Abd Alaal. They provided me with many great ideas and showed me many invaluable insights.

I would like to express my appreciation and thanks to Victoria University, particularly the College of Sport, Health and Engineering for granting me the opportunity to pursue higher studies.

I express my gratitude to King Khalid University for funding my work.

This thesis would never have been possible if not for the unconditional love of my family.

I am deeply indebted to my mother, for giving me inspiration at every stage of my life.

I am forever indebted to my father, who taught me the value of hard work; inspired me to pursue excellence; and truly believed in me before anyone else did.

My deepest feelings go to my family to whom I owe a great deal; they so tirelessly supported me throughout with thoughts and prayers. They were my source of strength in my darkest hours. I am deeply indebted for their love and understanding throughout my graduate years. They were always behind me and gave me their unconditional support even when it meant sacrificing the time we spent together. With them the completion of this thesis became much more joyful than ever.

I would also like to thank my brothers and sisters for their lifelong and unyielding support and encouragement.

Last but not least in recollection of many kindnesses, continuous encouragement, perseverance and patience, my grateful thanks go to all my friends and colleagues, wherever they may be, for bestowing blind faith on my capabilities; especially Ans, Abdulmalik, Faisal, Sultan, Musad and Mohammad. I thank all those who have ever conferred upon me their best wishes.

Contents

List of Publications

The results presented in this thesis have been published/submitted in the following papers:

[1] Z. M. Alaofi, T. S. El-Danaf, and S. S. Dragomir, “A numerical solution of the dissipative wave equation by means of cubic b-spline method,” *J. Phys. Commun*., vol. 5, no. 10, pp. 105014, 2021, https://doi.org/10.1088/2399-6528/ac2940.

[2] Z. M. Alaofi, T. S. El-Danaf, F. A. Alaal, and S. S. Dragomir, “Quartic non-polynomial spline for solving the third-order dispersive partial differential equation,” *American Journal of Computational Mathematics*, pp. 189–206, 2021, https://doi.org/10.4236/ajcm.2021.113013.

[3] Z. M. Alaofi, T. S. El-Danaf, F. E. I. A. Alaal, and S. S. Dragomir, “Numerical investigations of the coupled nonlinear non-homogeneous partial differential equations,” *J. Appl. Comput. Mech.*, vol. 8, no. 3, pp. 1054–1064, 2022, https://doi.org/10.22055/jacm.2022.39327.3388.

[4] Z. M. Alaofi, T. S. El-Danaf, and S. S. Dragomir, “Comparing solutions to the nonlinear dissipative wave equation,” *J. Appl. Math. Phys*., vol. 10, no. 4, pp. 1281–1296, 2022, https://doi.org/10.4236/jamp.2022.104040.

[5] Z. M. Alaofi, T. S. El-Danaf, A. Hadhoud, and S. S. Dragomir, “Computational analysis for solving the linear space-fractional telegraph equation,” *Open J. Model. Simul*., vol. 10, no. 3, pp. 267–282, 2022, https://doi.org/10.4236/ojmsi.2022.103014.

[6] Z. M. Alaofi, T. S. Ali, F. A. Alaal, and S. S. Dragomir, “Adaptive boundary control for the dynamics of the generalized Burgers–Huxley equation,” *Open J. Appl. Sci.*, vol. 12, no. 8, pp. 1416–1438, 2022, https://doi.org/10.4236/ojapps.2022.128098.

Presentations Made During the Author's Candidature

During my candidature I made a presentation to the prestigious Conference of Students at Victoria University in 2021. I was so blessed to be ranked first at the level of the College of Engineering and Sciences and second at the whole university level.

On a separate occasion, I won the Institute for Sustainable Industries and Liveable Cities (ISILC) Degree by Higher Research (HDR) Symposium Award 2022 Best Themed Presentation Engineering and Science.

List of Abbreviations

ADM	Adomian decomposition method
B-spline	Basis spline
CBS	Cubic B-spline
CFD	Caputo fractional derivative
DTM	Differential transform method
FEM	Finite element method
FTE	Fractional telegraph equation
GBHE	Generalised Burgers–Huxley equation
GRLW	Generalised regularised long wave
HAM	Homotropy analysis method
KdV	Korteweg–de Vries (equation)
GKdVB	Generalised Korteweg–de Vries–Burgers (equation)
KS	Kuramoto–Sivashinsky (equation)
MCBC-DQM	Modified cubic B-spline differential quadrature method
NKGE	Nonlinear Klein–Gordon equation
NLS	Nonlinear Schrödinger (equation)
NPDE	Nonlinear partial differential equation
NURBS	Non-uniform rational B-splines
ODE	Ordinary differential equation
PDE	Partial differential equation
RLW	Regularised long wave

List of Symbols

=	equality
≡	definition
≈	approximately equal
≠	inequality
<	less than
>	greater than
≪	much less than
≤	less than or equal
∝	proportional
+	addition
−	subtraction
× or ·	multiplication
÷ or /	division
±	plus or minus
√	square root
∑	summation
!	factorial
⇒	implies
|…|	absolute value
≅	congruence
∞	infinity
∈	an element
{,}	the set
$\mathbb{R}$	real numbers
a, b, …	coefficients of functions and equations
x, y, z	unknowns in functions and equations
Δ	discriminant
i, j, k	index variables in summations and products
t	time
e^x	natural exponential function

Sin(x)	sine function
Cos(x)	cosine function
Tan (x)	tangent function
Max (A)	maximum of set A
|| v ||	norm of vector v
Span(S)	span of set of vectors S
A^T	transpose of matrix A
A^{-1}	multiplicative inverse of matrix A
det (A)	determinant of square matrix A
$\binom{n}{k}$	combination (n choose k)
$\lim_{x \to c} f(x)$	limit of function f as x tends to c
$f', f'', f''', f^{(n)}$	first, second, third and nth derivative of function f
$\int_a^b f(x)$	definite integral of function f from a to b
Δx	change in variable x
dx	differential of variable x
∂x	partial differential of variable x
[a,b]	closed interval
(a,b)	open interval
Dx y	derivative
Γ	gamma

List of Tables

List of Figures

CHAPTER 1

Introduction

Chapter 1: Introduction

This introductory chapter provides a general introduction to the thesis and outlines the motivations and research questions of the study. This is followed by a description of the contributions and objectives of the research. Last, this chapter describes the research methodology and thesis organisation.

1.1 General Introduction

Partial differential equations (PDEs) are an essential tool used in various areas of applied mathematics. Solutions to PDEs form the basis of many mathematical models in physics and medicine. Numerical methods for solving scientific and engineering problems are becoming more important and the subject has become an essential part of the training of applied mathematicians, engineers and scientists. Having no exact solution is a critical informational issue for mathematicians—especially those who plan to approximate solutions for nonlinear partial differential equations (NPDEs), except under some initial boundary conditions. One important reason for this is that numerical methods can provide the solution when ordinary analytical methods fail. Many mathematical models for engineering problems are expressed in terms of boundary value problems, which are PDEs with boundary conditions. Four of the most popular techniques for solving PDEs are the B-spline (or basis spline), non-polynomial splines, fractional calculus and Adomian decomposing methods. In the last few decades, numerical techniques have been increasingly used to build mathematical models in engineering research. This thesis focuses on the use of different numerical methods to solve both linear PDEs and some NPDEs. The main objective of the thesis is to study treatments for some types of NPDE using the collocation method with the B-spline and non-polynomial spline. More broadly, this thesis aims to compare approximate solutions for NPDEs with exact solutions. It also seeks to categorise the selected methods and assess their accuracy and efficiency. Further it discusses challenges faced by researchers in this field and emphasises the importance of interdisciplinary effort for advancing the study of use of numerical methods for solving some PDEs. The thesis consists of eight chapters followed by a

list of references considered to be useful to the development and application of the methods discussed in this thesis. A brief description of the contents of each chapter is as follows.

This chapter introduces the main ideas of the thesis including its aim and objectives and outlines the research. Preliminary definitions and a literature review are presented in Chapter 2. Chapter 3 focuses on finding an approximate solution for the dissipative wave equation using the cubic B-spline (CBS) method. Solving the third-order dispersive partial differential equation by using a quartic non-polynomial spline is discussed in Chapter 4. Chapter 5 describes numerical investigations of the coupled nonlinear non-homogeneous Klein–Gordon equations. A new approach to the linear space-fractional telegraph PDE is outlined in Chapter 6. Chapter 7 introduces the adaptive boundary control for the generalised Burgers–Huxley equation (GBHE) with high-order nonlinearity terms. It also outlines the boundary control problem for the unforced GBHE with high-order nonlinearity when the spatial domain is [0, 1]. Finally, Chapter 8 provides a brief conclusion and considerations for possible future research in these areas.

1.2 Motivations and Research Questions

Some important issues relating to NPDEs include finding exact solutions for some initial and boundary conditions. This thesis aims to investigate the numerical treatments for some types of NPDE. The proposed method is the collocation method with the B-spline and the non-polynomials spline.

This thesis also aims to obtain accurate and stable numerical solutions for these NPDEs with integer and fractional orders. Having no exact solution is a critical informational issue for mathematicians—especially those who plan to approximate solutions for NPDEs, except under some initial boundary conditions.

This thesis aims to compare the CBS and non-polynomial spline methods for both the same and separate PDEs. This comparison is an attempt to confirm the suitability of methods applied in this research. More broadly, this research aims to compare approximate solutions for the NPDEs with exact solutions. The thesis also aims to

obtain percentage improvements in terms of accuracy, convergence, stability and so on, by comparison with previous methods.

What makes this study unique is its use of the two methods, the B-spline and the non-polynomial spline, combined to solve some recently discovered types of NPDE.

Specifically, this thesis addresses the following research questions:

1. Why do we use approximate solutions for nonlinear partial differential equations?
2. What methods are required to obtain these approximate solutions?
3. What are the accuracy and stability of such numerical approximations?
4. What are the accuracy and convergence of such numerical approximations?
5. How are nonlinear partial differential equations of fractional order treated?

1.3 Contributions and Objectives of the Research

This thesis contributes to the provision of numerical solutions for NPDEs. The results are analysed by comparing exact solutions under some initial and boundary conditions—when available—with approximates ones. Consequently, this thesis offers researchers and mathematicians the opportunity to assess the stability and effectiveness of methods under consideration. More widely, the findings could be used to identify approximate solutions for NPDEs using up-to-date technology, and saving time and money.

Despite the large body of qualitative literature relating to the treatment of NPDEs, further research is needed. Previous research has commonly sought to solve PDEs using one method; this research, in contrast, explores numerical approximations for NPDEs by using a combination of two methods. Use of this combined methodology is expected to greatly enhance the efficacy of approximations.

The results of this study may be of interest to mathematicians who aim to approximate solutions for certain NPDEs; although there are no exact solutions under some initial and boundary conditions. The results may also be beneficial to the field of numerical analysis in general; and for applications in theoretical physics, biology and

engineering in particular. In addition, the results may help to improve the stability, effectiveness and convergence of numerical solutions of NPDEs encountered in other fields of science or engineering.

1.4 Research Methodology

There are several approximate methods for solving NPDEs. This thesis focuses on two of these: CBS and the non-polynomial methods. The methods are described along with the accuracy and stability of such approximations. The obtained results are then presented.

The study begins by analysing each method in detail. I select some NPDEs that have not been solved using previous methods and address why they were chosen and why approximate solutions are required. Afterwards, I apply CBS methods to the various equations. In addition, I assess the accuracy and stability of these approximations. I then similarly apply another method (non-polynomial spline). I then study the accuracy and stability of these approximations. Finally, arriving at what makes this research different from previous studies, I compare my findings and determine the most accurate methods.

My primary goal is to focus on the polynomial and non-polynomial spline methods to find approximate solutions for some important applications of NPDEs, with integer and fractional orders, and discuss the stability and convergence of such numerical solutions. To achieve this goal, I take the following steps:

1. I consider a series solution as the multiplication of two functions; one dependent on time (t) and the other on discretisation (x).
2. I convert the NPDEs into a system of ordinary differential equations (ODEs).
3. By applying the spline method, I convert the ODEs into a system of algebraic equations.
4. By employing some computer programs (Maple, Mathematica, Matlab), I find approximate solutions.
5. I apply the Von Neumann stability scheme to the existence system after linearisation.

6. I conduct a comparison between the obtained numerical approximations and the exact solutions where they exist.
7. I perform an analysis to demonstrate the accuracy of the numerical method.

1.5 Thesis Organisation

The main objective of this thesis, which consists of eight chapters, is to introduce an analytical and numerical treatment based on the B-spline and non-polynomial spline methods for solving some types of PDE. The thesis organisation is as follows.

This chapter provides a general introduction to the research, outlining the motivation and research questions, contributions and objectives of the research, research methodology and thesis organisation.

Chapter 2 provides some basic definitions, necessary details and mathematical preliminaries relating to the PDEs, which are required to establish the results. This chapter contains four sections. Section 2.1 introduces PDEs and outlines some basic definitions such as PDEs, the order of PDEs, linear PDEs and NPDEs, homogeneous and inhomogeneous PDEs, initial conditions, boundary conditions and classification of second-order PDEs. Section 2.2 provides some background on B-splines including a literature review, an overview of the B-spline method and spline functions, and some basic definitions for the spline function of degree m, the CBS function local truncation error, consistency, total error, convergence, stability and non-polynomial spline functions. Section 2.3 discusses fractional calculus, providing the history of fractional functions and some important definitions and concepts in this field. The last section introduces the Adomian decomposition method (ADM).

Chapter 3 solves the dissipative wave equation using the CBS method to find the best approximation. I begin by considering the governing equation and the derivation of the proposed method. Next, I use the boundary conditions to obtain N+3 equations in N+3 unknowns. I then apply the initial conditions to obtain the independent variables. The Von Neumann stability method is then used to find the method conditionally stable. Finally, I calculate $L\infty$ error norms for an example of the dissipative wave

equation using the Mathematica program and compare the results with those from some previous studies.

Chapter 4 solves a third-order dispersive PDE using the quartic non-polynomial spline method to find the best approximation of this solution. I analyse the effectiveness of the method depending on the quartic formula by using the continuity condition of the first and second derivatives of the non-polynomial spline functions. I then apply the boundary conditions to obtain an ODE, which can be described in the form of a matrix. An error analysis is conducted using Taylor's expansion of the truncation error. I use the Von Neumann stability analysis to discuss the stability of the method. Finally, I present the full numerical results for the non-homogeneous third-order dispersive PDE with the best approximations relative to previous studies.

Chapter 5 provides a brief description of numerical investigations of coupled nonlinear non-homogeneous PDEs using the CBS method. I start with the numerical method, which is based on approximations such as wavelet approximations. I use the values of the cubic function through the knots as well as the values of its first and second derivatives. I examine and analyse the proposed method using the Von Neumann stability method to show that it is conditionally stable. Finally, I introduce a numerical example to solve coupled nonlinear non-homogeneous Klein–Gordon equations. The results are compared with the exact solution and a brief description of the maximum absolute errors is provided.

In Chapter 6, a novel approach based on the quadratic-polynomial spline-based method is used to obtain the numerical solution for the time–space-fractional order telegraph equation. The chapter outlines the derivation of the quadratic-polynomial spline method. Next, the spline relationships are examined through continuity conditions and truncation error. The stability is then discussed theoretically using the Von Neumann stability method. Finally, a numerical example is presented to illustrate the practical implementation of the proposed method. The results are then compared with the exact solutions.

Chapter 7 deals with the boundary control problem for the unforced GBHE with high-order nonlinearity when the spatial domain is [0, 1]. The adaptive boundary control

for the GBHE with high-order nonlinearity terms is introduced. I use an adaptive nonlinear boundary controller to show that it achieves global asymptotic stability in time and to demonstrate convergence of the solution with the trivial solution. I introduce numerical simulations for the controlled equation using the ADM to illustrate the performance of the controller applied to the GBHE.

Finally, Chapter 8 draws conclusions, provides recommendations for future work in this field and highlights some areas for further development.

CHAPTER 2

Preliminary Definitions and Literature Review

Chapter 2: Preliminary Definitions and Literature Review

In this chapter, I provide an introduction to PDEs and describe some commonly used PDEs, various initial conditions and boundary conditions. I then analyse some state-of-the art numerical methods for PDEs and conduct a review of the relevant literature.

2.1 Partial Differential Equations

Fields such as medicine, engineering, physics and biology are significantly influenced by PDEs. Physical phenomena within sound propagation, diffusion, heat transfer, electrodynamics, elasticity, fluid dynamics, optics and electrostatics are outlined via PDEs [1].

2.1.1 Definition of Partial Differential Equations

A PDE is an equation that contains a dependent variable (the unknown function) and its partial derivatives $F\left(x, y, t, u, u_x, u_{xx}, u_y, u_t, u_{xy}, \dots\right) = 0$. It is known that in ODEs, the dependent variable $u = u(x)$ depends only on one independent variable x. Unlike ODEs, the dependent variable in a PDE, such as $u = u(x,t)$ or $u = u(x,y,t)$, must depend on more than one independent variable. If $u = u(x,t)$, then the function u depends on the independent variable x, and on the time variable t. However, if $u = u(x,y,t)$, then the function u depends on the space variables x, y, and on the time variable t. Examples of PDEs are as follows:

$$u_t - k u_{xx} = 0 \tag{2.1.1}$$

$$u_t = k(u_{xx} + u_{yy}) \tag{2.1.2}$$

$$u_t = k(u_{xx} + u_{yy} + u_{zz}) \tag{2.1.3}$$

They describe the heat flow in one-dimensional (1D) space, two-dimensional (2D) space and three-dimensional (3D) space respectively. In Eq. (2.1.1), the dependent variable $u = u(x,t)$ depends on the position x and on the time variable t. However, in Eq. (2.1.2), $u = u(x,y,t)$ depends on three independent variables: the space variables x, y and the time variable t. In Eq. (2.1.3), the dependent variable $u =$

$u(x, y, z, t)$ depends on four independent variables: the space variables x, y and z, and the time variable t [2].

2.1.2 Order of Partial Differential Equations

The order of a PDE is the order of the highest partial derivative that appears in the equation. For example, the following equations are PDEs of first, second, and third order respectively [3]:

$$\begin{aligned} u_x - u_y &= 0 \\ u_{xx} - u_t &= 0 \\ u_y - uu_{xxx} &= 0. \end{aligned} \tag{2.1.4}$$

2.1.3 Linear and Nonlinear Partial Differential Equations

PDEs are classified as linear or nonlinear. A PDE is called linear if:

1. the power of the dependent variable and each partial derivative contained in the equation is one
2. the coefficients of the dependent variable and the coefficients of each partial derivative are constants or independent variables.

If any of these conditions is not satisfied, the equation is called nonlinear [3].

2.1.4 Homogeneous and Inhomogeneous Partial Differential Equations

PDEs are also classified as homogeneous or non-homogeneous. A PDE of any order is called homogeneous if every term in the PDE contains the dependent variable *u* or one of its derivatives; otherwise, it is called an inhomogeneous PDE. This is illustrated by the following examples:

$$\begin{aligned} (a)\ u_t &= 4u_{xx} \\ (b)\ u_t &= u_{xx} + x \\ (c)\ u_{xx} - u_{yy} &= 0 \\ (d)\ u_x + u_y &= u + 4 \end{aligned} \tag{2.1.5}$$

(a) The terms of the equation contain partial derivatives of u only; therefore it is a homogeneous PDE.

(b) The equation is an inhomogeneous PDE because one term contains the independent variable x.

(c) The equation is a homogeneous PDE.

(d) The equation is an inhomogeneous PDE [4].

2.1.5 Initial Conditions

It was indicated that PDEs mostly govern physical phenomena such as heat distribution, wave propagation and quantum mechanics. Most PDEs, such as the diffusion equation and the wave equation, depend on the time t. Accordingly, the initial values of the dependent variable u at starting time $t = 0$ should be prescribed. As discussed later in more detail for the heat case, the initial value $u(t = 0)$, which defines the temperature at the starting time, should be prescribed. For the wave equation, the initial conditions $u(t = 0)$ and $u_t(t = 0)$ should also be prescribed [5].

2.1.6 Boundary Conditions

The exact solution of a PDE is of little use. A particular solution is frequently required that will satisfy prescribed conditions. Given a PDE that controls the mathematical behaviour of a physical phenomenon in a bounded domain D, the dependent variable u is usually prescribed at the boundary of the domain D [5].

2.1.7 Classification of Second-order Partial Differential Equations

A second-order linear PDE with two independent variables x and y in its general form is given by

$$Au_{xx} + Bu_{xy} + Cu_{yy} + Du_x + Eu_y + Fu = G, \qquad (2.1.6)$$

where A, B, C, D, E, F and G are constants or functions of the variables x and y. A second-order PDE such as Eq. (2.1.6) is usually classified into three basic classes of equations, namely [3]:

Parabolic:

A parabolic equation is an equation that satisfies the property

$$B^2 - 4AC = 0. \tag{2.1.7}$$

Examples of parabolic equations are heat flow and diffusion processes equations.

The heat transfer equation is

$$u_t = ku_{xx}. \tag{2.1.8}$$

Hyperbolic:

A hyperbolic equation is an equation that satisfies the property

$$B^2 - 4AC > 0. \tag{2.1.9}$$

Examples of hyperbolic equations are wave propagation equations. The wave equation is:

$$u_{tt} = c^2 u_{xx}. \tag{2.1.10}$$

Elliptic:

An elliptic equation is an equation that satisfies the property

$$B^2 - 4AC < 0. \tag{2.1.11}$$

Examples of elliptic equations are Laplace's and Schrödinger's equations.

The Laplace equation in a 2D space is

$$u_{xx} + u_{yy} = 0. \tag{2.1.12}$$

2.1.8 Some Useful Definitions

Definition: [6]

The l_2 and l_∞ norms for the vector $x = (x_1, x_2, \ldots, x_n)^t \in R^n$ are defined by

$$\|x\|_2 = \left\{\sum_{i=1}^{n} x_i^2\right\}^{1/2} \text{and } \|x\|_\infty = \max_{1\le i\le n}|x_i|. \qquad (2.1.13)$$

Definition: [6]

If $x = (x_1, x_2, \ldots, x_n)^t$ and $y = (y_1, y_2, \ldots, y_n)^t$ are vectors in R^n, the l_2 and l_∞ distances between *x* and y are defined respectively by

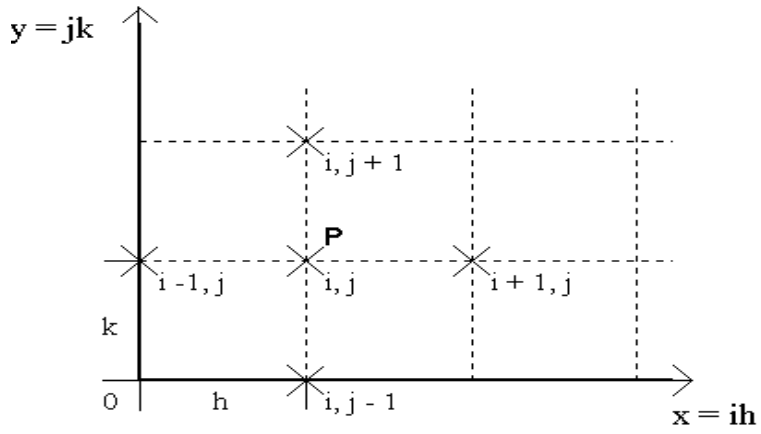
$$\|x - y\|_2 = \{\sum_{i=1}^{n}(x_i - y_i)^2\}^{1/2} \text{ and } \|x - y\|_\infty = \max_{1\le i\le n}|x_i - y_i|. \qquad (2.1.14)$$

Definition: [7]

The set $\varphi = \{\varphi_1, \varphi_2, \varphi_3, \ldots, \varphi_k\}$ is said to be the basis of the set $u = \{u_1, u_2, u_3, \ldots, u_k\}$ if and only if the set $u = span(\varphi)$ and the set φ is linearly independent.

Definition: [8]

The finite difference method involves transforming the partial derivatives into difference equations over a small interval. Assume that *u* is function of independent variables *x* and *y* and we can divide the *x*-*y* plane into equal mesh points using step sizes $\delta x = h$ and $\delta y = k$ as shown below:



We can evaluate *u* at point P by

$$u_p = u(ih, jk) = u_i^j$$

The value of the first derivative at P can be evaluated using three approximations:

Central difference:

$$(u_x)_p = (u_x)_{i,j} \cong \frac{u_{i+1}^j - u_{i-1}^j}{2h}, \left(u_y\right)_p = \left(u_y\right)_{i,j} \cong \frac{u_i^{j+1} - u_i^{j-1}}{2k},$$

Forward difference:

$$(u_x)_p = (u_x)_{i,j} \cong \frac{u_{i+1}^j - u_i^j}{h}, \left(u_y\right)_p = \left(u_y\right)_{i,j} \cong \frac{u_i^{j+1} - u_i^j}{k},$$

Backward difference:

$$(u_x)_p = (u_x)_{i,j} \cong \frac{u_i^j - u_{i-1}^j}{h}, \left(u_y\right)_p = \left(u_y\right)_{i,j} \cong \frac{u_i^j - u_i^{j-1}}{k},$$

The value of the second derivative at P could also be evaluated by

$$(u_{xx})_p = (u_{xx})_{i,j} = \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{h^2}, \left(u_{yy}\right)_p = \left(u_{yy}\right)_{i,j} = \frac{u_i^{j+1} - 2u_i^j + u_i^{j-1}}{k^2}$$

Definition: [8]

We say that the finite difference scheme is consistent with a PDE if the truncation error tends to zero for a fixed point (x, t)in the domain as Δx and $\Delta t \to 0$ independently and $(\frac{\Delta x}{\Delta t})$ is bounded.

Definition: [8]

Let *u* be the exact solution of the partial differential for the independent variables *x* and *t* and let U be a numerical solution for the PDE at mesh point $U(x_i, t_i)$; then the numerical method is said to be convergent when U tends to *u* at a fixed point or along a fixed *t*-level as Δx and Δt both approach zero.

2.2 B-splines

Over the last three decades or thereabouts, the wide-reaching discipline of geometric modelling has seen a surge in usage and recognition of B-splines in surface and curve form. The computationally efficacious algorithms designed for altering B-splines alongside their expedient notation and desirable mathematical features are prime

determinants of their popularity. In cases where the modelling of physical properties and complex data are required to portray a shape, the B-spline form is an effective standard in both contemporary development of dynamic models for physics-based manipulation of solids and surfaces and free-form deformation; and in B'ezeir surfaces and curves as found in the automobile industry through the initial work of B'ezier and de Casteljau [9].

2.2.1 Literature Review

In the 1970s, an upgrade to B'ezier curves was produced in the form of B-spline curves. Riesenfeld pioneered work generating splines containing local support [10]. The overall curve is not affected by modifying the shape of a specific segment of the curve because B-splines possess local support. As a result, shapes generated by B-spline curves can be controlled to a greater degree [11]. The definition of B-spline basis functions of any order was linked to a novel recursive formula in one study [12]. In the 1980s, non-uniform rational B-splines (NURBS) represented an extension of B-splines as per Versprille, Piegel and Tiller [13–15]. Every basis function or control point that enhances shape control is linked to a weight by NURBS. In computer-supported design and graphics, the current standard is the NURBS surface or curve representation.

Finite elements and other such numerical techniques have implemented B-splines because of their benefits, despite them being originally designed for surface and curve representation. B-splines were used in one study to enable the interfacing of finite elements with computer-aided design (CAD) systems [16]. The geometry representation in CAD utilises the same functions as those in finite element analysis using a B-spline. Convergence data for multiple degrees of B-spline functions were presented in a study centred on extending geometric design into mechanical analysis.

In another paper, the finite element approach was improved upon via the use of NURBS [17]. In a finite element analysis, a CAD description of the boundary of a model was utilised. At the NURBS boundary, special piecewise polynomial functions were applied; conventional interpolation functions were implemented in the interior. Through the use of high-order interpolation and a coarser mesh, the capability of

NURBS used in the finite element method (FEM) to attain a precise solution as well as its computational efficacy was noted.

Product shape design was facilitated by the application of NURBS via the finite element approach in the work of Inoue et al. [18]. The incorporation of computer-supported design software and finite element analysis formed part of the research objective of their work. They showed that the geometry of the NURBS surface for the manufactured product was reflected precisely by the shell finite element, and implemented the NURBS finite element approach in bending analyses of shells and plates. In a buckling analysis, fewer errors were observed when using NURBS FEM-approximated solutions than in the standard FEM results. The exact geometry modelled within NURBS FEM may have been responsible for this phenomenon. The application of NURBS within finite elements has been explored in several studies focused on continuity and refinement of the approach [19]. In finite element analyses, the NURBS parameters from CAD are utilised; the exact geometry generated via CAD is sustained while an enhanced finite element solution is found by using degree elevation and/or knot insertion to enable refinement. The accuracy of the solution is seemingly augmented in the majority of scenarios where the smoothness provisioned by NURBS is also augmented.

Additional improvements in the B-spline finite element technique are presented in other studies. A technique for the adaptive refinement of B-spline finite elements was formulated in one of these [20]. That study described the continuity of solution approximation and presentation of multiple refinement techniques. On a similar note, another study described a novel 8-node quadrilateral spline finite element as well as simplifications to products, integrals and derivatives of shape functions [21]. A typical 8-node isoparametric element was found to perform less effectively than the novel form.

Finite elements are not the only source of spline implementation, as demonstrated by their implementation in numerical analyses. A finite difference boundary value problem employed splines in a study by Kumar [22]. The efficacy of splines for solving the boundary value problem as well as the second-order convergence of the spline finite difference technique was noted. Kumar also implemented a finite

difference boundary value problem of fourth-order splines and reported fourth-order convergence [23]. A system of boundary value problems was resolved via the use of cubic splines in another paper [24]. Convergence behaviour was assessed and the technique implemented for several problems. Various techniques utilised to solve the system of boundary value problems were found to be inferior as they were more prone to errors than was the cubic-spline technique used to analyse solutions to multiple fourth-order boundary value problems. Singularly perturbed boundary value problems were solved utilising a fitted mesh B-spline technique in another study [25], where a tridiagonal linear system was developed via the mentioned B-spline technique that also generated a uniform mesh. Existing techniques were found to be inferior to the findings produced by the proposed approach.

The B-spline finite element approximation to PDE surface accuracy was the focus of another study that also discussed data transfer between CAD systems [26]. Enhanced approximation was achieved by examining degree elevation and refinement. No substantial differences were found between non-periodic and periodic B-spline functions in terms of the implementation of surface approximation.

To address a two-point boundary value problem, the finite volume, difference and element techniques were compared with B-spline approximation in one study [27]. Existing techniques were found to be inferior to the B-spline approximation technique in terms of maximum error norms. Thus, those techniques were concluded to be outstripped by B-spline functions with regard to their compatibility for approximating a smooth solution.

Splines have been utilised in multiple implementations of numerical techniques. For example, in one study a 3D spline variational technique was utilised for analysis of a cross-ply laminate with a circular hole [28]. The standard finite element technique and the spline variational technique demonstrated good agreement. Relative to standard finite elements, equivalent interlaminar stress distributions were sustained whereas the number of degrees of freedom was reduced by a factor of 3–5 with the spline variational technique.

The bending of skew plates was examined in one study by implementing a spline element technique [29]. Various numerical techniques were compared in terms of their convergence and accuracy and displayed good agreement with the spline element technique. Additionally, accuracy was enhanced through the use of a mesh grading method and high-ordered splines.

One study implemented finite elements to analyse plates and beams. The authors discussed the variation diminishing property, versatility for modelling distinct boundary conditions, and computational efficacy among the benefits of splines [30]. Finite strip solutions were among the reference techniques demonstrating good agreement. The efficacy of splines was highlighted by the smaller number of degrees of freedom.

Another study implemented finite strip analysis of shear-deformable plates [31]. The research objected was to examine thick, laminated composite plates. The multiple continuity requirements for classical plate analysis were met via the versatility provided by the use of CBS. Existing and analytical numerical solutions showed very good agreement.

Vibration analysis of plates was carried out via the use of a multivariable spline element technique in the work of Peng Cheng et al. [32]. They obtained a solution with high levels of accuracy and emphasised the presence of good convergence features.

Shell analysis was performed via the use of CBS finite elements in one study [33]. In a case where cubic Hermite functions were insufficient, shell analysis was complemented well by the second-derivative continuity of B-splines. In the findings, stress discontinuities were removed by utilising CBSs. The manner in which B-spline usage reduced the number of degrees of freedom was also highlighted.

In one study, the shaping and examination of a torque converter clutch disk were enabled by using a B-spline finite element technique [34]. Master nodes were relocated to drop directly onto the surface of the model via an algorithm that defined the B-spline curve accordingly. An enhanced finite element solution was obtained

through implementation of a mesh smoothing method. A substantial weight decrease was achieved through design of an optimum shape for the disc via the mentioned technique. Tall buildings were examined using spline elements in another study [35]. This approach facilitated the investigation of structures with boundaries and uneven openings, as well as tall buildings of haphazard shape. Computational efficacy was enhanced via the conscientious selection of the order of B-spline functions in distinct dimensions. A handful of elements for static and dynamic analysis providing good approximations were revealed.

A thermistor problem regarding electrical conductivity was investigated in another paper utilising a B-spline finite element technique [36]. A Galerkin technique was utilised in tandem with the CBS. A diverse range of mesh refinements was used to present the data and findings. The known exact solution was converged upon by the approximated solution from the analysis.

The solution to a 1D nonlinear Burgers' equation was approximated by utilising quadratic B-splines in one paper [37]. A set of nonlinear standard differential equations resulted from conversion of the Burgers' equation. A quadratic B-spline finite element technique then served to solve each equation. The B-spline FEM solution also demonstrated high accuracy. Similarly, a CBS finite element technique was used to approximate the solution to Burgers' equation in the work by Gardner et al. [38]. Their findings revealed that other techniques displayed inferior accuracy in comparison with the CBS FEM method.

Panels under mechanical and thermal loading were examined in one study via implementation of CBS collocation techniques [39]. 1D and 2D problems were the focus of the work. The nonlinear induced reaction of the panels was also investigated during the analytical process. The potential acceleration and efficacy of convergence in using B-splines were again highlighted.

2.2.2 B-spline Method

In the mathematical subfield of numerical analysis, the B-spline is a spline function that has minimal support with respect to a given degree, smoothness and domain

partition. Any spline function of given degree can be expressed as a linear combination of B-splines of that degree. Cardinal B-splines have knots that are equidistant from each other. B-splines can be used for curve fitting and numerical differentiation of experimental data. The many types of B-spline include quadratic, cubic, quintic and septic splines, each of which can be applied to obtain an approximate solution for a NPDE. In this study I apply cubic, quintic and septic splines.

2.2.3 Spline Functions

When approximating functions for interpolation or for fitting measured data, it is necessary to have classes of functions that have sufficient flexibility to adapt to a given dataset, and that, at the same time, can be easily evaluated on a computer. Traditionally, polynomials have been used for this purpose. These polynomials have some flexibility and can be computed easily. However, for rapidly changing values of the function to be approximated, the degree of the polynomial must increase, and the result is often a function exhibiting wild oscillations. The situation changes dramatically when the basic interval is divided into subintervals, and the approximating or fitting function is taken to be a piecewise polynomial; that is, the function is represented by a different polynomial over each subinterval. The polynomials are joined together at the interval endpoints (knots) in such a way that a certain degree of smoothness (differentiability) of the resulting function is guaranteed. If the degree of the polynomial is k and the number of subintervals is $n+1$ the resulting function is called a (polynomial) spline function of degree k with n knots.

Spline functions are used in many applications including interpolation, data fitting and numerical solutions for ordinary and partial differential equations (FEM).

In regard to practical problems, spline functions have the following useful properties:

- smooth and flexible
- easy to store and manipulate on a computer
- easy to evaluate, along with their derivatives and integrals
- easy to generalise to higher dimensions.

2.2.4 Some Useful Definitions

Definition: (Spline Function of Degree M)

Let $a = x_0 < x_1 < \ldots < x_n = b$ be a subdivision of the interval $[a, b]$ and $m \in N$. A function $S: [a, b] \to R$ is called a spline of degree m with respect to this subdivision if S is $(m-1)$-times continuously differentiable on $[a, b]$ and if the restriction of S to each subinterval $[x_{j-1}, x_j]$ for $j = 1,2,\ldots,n$ reduces to a polynomial of degree at most values of m[40].

Definition: (The Cubic B-spline Function)

CBS functions $\{\varphi_i(x)\}$ are defined by

$$\varphi_i(x) = \frac{1}{h^3}\begin{cases} (x - x_{i-2})^3 & x \in [x_{i-2}, x_{i-1}] \\ h^3 + 3h^2(x - x_{i-1}) + 3h(x - x_{i-1})^2 - 3(x - x_{i-1})^3 & x \in [x_{i-1}, x_i] \\ h^3 + 3h^2(x_{i+1} - x) + 3h(x_{i+1} - x)^2 - 3(x_{i+1} - x)^3 & x \in [x_i, x_{i+1}] \\ (x_{i+2} - x)^3 & x \in [x_{i+1}, x_{i+2}] \\ 0 & \text{otherwise,} \end{cases}$$

where $h = x_{i+1} - x_i$, $i = 0,1\ldots\ldots, N-1$. The values of the CBS $\varphi_i(x)$ and its first and second derivatives vanish outside the interval (x_{i-2}, x_{i+2}). The values of $\varphi_i(x)$ and its derivatives at the knots are established in Table 2.2.4.1. This type of spline is used to obtain approximate solutions to PDEs (see [40]).

Table 2.2.4.1 The values of $\varphi_i(x)$ and its derivatives with knots at the points shown.

x	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}
$\emptyset_i(x)$	0	1	4	1	0
$\emptyset_i'(x)$	0	$\frac{3}{h}$	0	$\frac{-3}{h}$	0
$\emptyset_i''(x)$	0	$\frac{6}{h^2}$	$\frac{-12}{h^2}$	$\frac{6}{h^2}$	0

Definition:

Let U represent the exact solution for a PDE with independent variables x and t, and u represent the exact solution of the difference equations used to approximate the PDE.

Definition: (Local Truncation Error)

The truncation error is the difference between the differential equation and its approximating difference equation. Let $F_{jn}(u) = 0$ represent the difference equation at the $(j,n)_{th}$ mesh point. If u is replaced by U at the mesh points of the difference equation, then the value of $F_{jn}(U)$ is called the local truncation error at the $(j,n)_{th}$ mesh point: 'We denote it by T_j^n' [41].

Definition: (Consistency)

We say that the finite difference scheme is consistent with the PDE if the truncation error tends to zero for a fixed point (x,t) in the domain as $\Delta x, \Delta t \to 0$ independently and $(\Delta t/\Delta x)$ is bounded [41].

Definition: (Total Error)

Let N be the numerical solution for the difference equation. Then the total error at $(j,n)_{th}$ is

$$U_j^n - N_j^n = (U_j^n - u_j^n) + (u_j^n - N_j^n), \qquad (2.2.1)$$

where $(U_j^n - u_j^n)$ is the discretisation error e_{jn} and $(u_j^n - N_j^n)$ is the round-off error R_{ij} [41].

Definition: (Convergence)

The finite difference equation is said to be convergent when u tends to U at a fixed point or along fixed t-level as $\Delta x = h$ and $\Delta t = k$ both tend to zero. (i.e. the discretisation error $e_{jn} = (U_j^n - u_j^n) \to 0$ as $h \to 0$ [41]. $k \to 0$.)

Definition: (Stability)

Suppose that in a computation involving a difference scheme, an error ξ^0 is introduced at time level t^0, and suppose that no further errors occur. Let ξ^n denote the error resulting from this error at time t^n; then the scheme is stable if $\left|\xi^n\right|$ remains bounded as $n \to \infty$. (i.e. the error must no grow without limit) [41].

2.2.5 Stability by the Fourier Series Method: (Von Neumann's Method)

This method, developed by Von Neumann during World War II, was first discussed in detail by O'Brien, Hyman and Kaplan in a paper published in 1951. It expresses an initial line of errors in terms of a finite Fourier series and considers the growth of a function that reduces to this series for $t = 0$ by the 'variables separable' method identical to that commonly used for deriving analytical solutions for PDEs. If we write $N_j^n = u_j^n + \xi_j^n$ in the difference scheme, and if it is linear, then ξ^n will satisfy the same equation as u_j^ne, which expresses the error as a finite Fourier series of the form $\sum \xi^n\,(t)e^{ikx_j}$, and again if the equation is linear in which case we need consider only the growth of a single form:

$$\xi_j^n = \xi^n e^{ikx_j}, \tag{2.2.2}$$

where $i = \sqrt{-1}$, and let

$$\xi^{n+1} = g\xi^n, \tag{2.2.3}$$

where g is called the growth of an amplification factor. For stability, the Von Neumann condition requires

$$|g| \leq 1 + \mathrm{O}(\Delta t).$$

That is, the error does not increase as *t* increases.

It should be noted that this method applies only to linear difference equations with periodic initial data. The criterion $|g| \leq 1$ is both necessary and sufficient for three or more level equations although it is always necessary. In practice the method often gives useful results even when its application is not fully justified [41].

Definition: (Non-polynomial Spline Functions)

Let $a = x_0 < x_1 < \ldots < x_N = b$ be a subdivision of the interval $[a, b]$. The non-polynomial spline function is defined by

$$p(x, t_n) = \begin{cases} p_0(x, t_n), x \in [x_0, x_1] \\ p_1(x, t_n), x \in [x_1, x_2] \\ \vdots \ \vdots \\ p_{N-1}(x, t_n), x \in [x_{N-1}, x_N] \end{cases},$$

where $p_j(x, t_n)$ is a mixed spline function of the form

$$p_j(x, t_n) = b_{1j}(t_n) \cos \omega (x - x_j) + b_{2j}(t_n) \sin \omega (x - x_j) + a_{1j}(t_n)(x - x_j)^r + a_{2j}(t_n)(x - x_j)^{r-1} + \ldots + a_{rj}(x - x_j) + a_{(r+1)j}, j = 0,1, \ldots, N - 1$$

and r represents the degree of the polynomial part.

Remark:

We can use other functions such as $(\cosh \omega x, \sinh \omega x), (\tanh \omega x, \sec h \omega x)$, $(\cos ch \omega x, \sec h \omega x)$ and $(e^{\omega x}, e^{-\omega x})$ instead of $(\cos \omega x, \sin \omega x)$ in$p_j(x, t_n)$ [42].

2.3 Fractional Calculus

2.3.1 Fractional Calculus History

One definition of fractional calculus is that it is a generalisation of standard integration and differentiation to arbitrary non-integer order. The history of this area dates back to the invention of differential calculus by Newton and Leibniz [43].

Leibniz questioned whether the significance of derivatives with integer order could be generalised to derivatives with non-integer orders, in a letter written in 1695 to L'Hospital. The recipient of the letter responded, questioning what would occur if the order ended up as half. On the birthday of fractional calculus—30 September of the year in which the first letter was written—Leibniz proposed, in response to L'Hospital, that useful consequences could be produced as a result of the paradox when the order equals half [44]. For over 300 years, this interrogation regarding a fractional derivative has remained relevant.

Given that integrals and fractional derivatives are not point or local quantity or property, this subject matter remains enticing. Hence, nonlocal distributed effects and history are taken into account. This could be perceived to signify that the reality of nature is better captured by this subject matter. The description and comprehension of nature are enhanced via the additional dimension provided by fractional calculus, which, in turn, highlights the benefits of making the method widely accessible to the engineering and science community. Interacting with nature in the language of fractional calculus may lead to discovery of greater harmony and thus, efficacy and comprehension. In recent years, multiple disciplines within economics, science and engineering have integrated learning and the application of the mentioned subject matter, which was previously solely the focus of mathematicians [45, 46, 47].

Letnikov, Leibniz, Abel, Weyl, Riemann, Grunwald, Lacroix, Fourier, Caputo and Liouville were among the recognised mathematicians to develop fractal science theory. However, consideration of the fractional derivative as a local operator particular to this theory is a distinct contemporary aspect. The conventional integer order calculus forms part of the superset of fractional differential integral calculus, according to this novel definition, which will lead to multiple new implementations over the next decade or so [48]. Probability, electrical networks, signal processing, coloured noise, electrochemical processes, fluid flow, electromagnetic theory, biology, anomalous diffusion, dielectric polarisation and viscoelasticity materials are among the many fields of engineering and science that have focused on fractional calculus.

The quadrature developed by Chern and Deithelm served as the basis for the fractional backward differences that exemplify one of several numerical solutions to fractional standard differential equations formulated by various academics [49, 50]. A collection of papers by one author served to pioneer the concept of fractional linear multistep techniques [51, 52]. A particular type of Volterra integral equations was numerically adopted by other authors [53]. A novel numerical solution to initial value problems with regard to general linear multi-term differential equations of fractional order possessing fractional derivatives and constant coefficients in the Caputo sense was provided in work utilising a novel algorithm [54]. With respect to the Mittag–

Leffler-type function, the analytical solution to the problem implements the discretised operational calculus and convolution quadrature [55]. Reducing the problem to a system of fractional differential and standard equations each of order at greatest unity enables computation of a numerical approximation for the solution of a linear multi-term fractional differential equation, as demonstrated in one study [56].

In relation to numerical solutions for arbitrary differential equations of fractional order, an algorithm has been proposed [57]. Other works (see [58, 59]) demonstrate the development of a system of differential equations of integer order linked to inverse forms of Abel integral equations via the decomposition of the differential equations. The solutions to nonlinear and linear equations are provided by the algorithm.

A reformulation of the Bagley–Torvik equation system consisted of fractional differential equations of order 0.5 in the work of Diethelm and Ford [60]. Nonlinear multi-term fractional arbitrary order differential equations were solved in another work via the numerical techniques provided [47].

2.3.2 Definitions and Notations

This section presents some basic definitions and properties of fractional calculus theory [61–64].

Definition:

A real function $f(x)$,x > 0, is said to be in the space $C_\mu, \mu \in \Re$, if there exists a real number $p(> \mu)$, such that $f(x) = x^p f_1(x)$ where$f_1(x) \in C(0, \infty)$, and it is said to be in the spaceC_μ^m if $f^m \in C_\mu, m \in N$ [61].

Definition:

The Mittag–Leffler function [62] is a generalisation of the exponential function, first introduced as a one-parameter by the series, and we add to it:

$$M_\alpha(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k+1)}, \alpha > 0, \alpha \in \Re, z \in C.$$

This leads to the following generalisation:

$$M_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k+\beta)}, \alpha, \beta > 0, \alpha, \beta \in \Re, z \in C.$$

Definition:

An incomplete gamma function [63] $\gamma^*(\alpha, z)$ may be defined by

$$\gamma^*(\alpha, z) = e^{-z} \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(k+1+\alpha)}.$$

If $Re\, z > 0$, then $\gamma^*(\alpha, z)$ has the integral representation

$$\gamma^*(\alpha, z) = \frac{1}{\Gamma(\alpha) z^{\alpha}} \int_0^z t^{\alpha-1} e^{-t}\, dt.$$

Functions related to **i**ncomplete gamma functions are called related functions [63].

Definition: (Related Functions) [63]

$$E_z(\alpha, a) = z^{\alpha} e^{az} \gamma^*(\alpha, az) = z^{\alpha} \sum_{k=0}^{\infty} \frac{(az)^k}{\Gamma(k+1+\alpha)},$$

$$C_z(\alpha, a) = z^{\alpha} \sum_{keven}^{\infty} \frac{(-1)^{k/2} (az)^k}{\Gamma(k+1+\alpha)}.$$

If we put $k = 2j$, then

$$C_z(\alpha, a) = z^{\alpha} \sum_{j=0}^{\infty} \frac{(-1)^j (ax)^{2j}}{\Gamma(1+\alpha+2j)}$$

$$S_z(\alpha, a) = z^{\alpha} \sum_{kodd}^{\infty} \frac{(-1)^{(k-1)/2} (az)^k}{\Gamma(k+1+\alpha)}.$$

If we put $k = 2j + 1$, then

$$S_z(\alpha, a) = z^{\alpha} \sum_{j=0}^{\infty} \frac{(-1)^j (ax)^{2j+1}}{\Gamma(2+\alpha+2j)}$$

$$E_z(\alpha, a) = C_z(\alpha, a) + iS_z(\alpha, a).$$

The Riemann–Liouville integral, the Riemann–Liouville derivative and the Caputo derivatives are defined as follows.

Definition:

The Riemann–Liouville fraction integral operator [61–63, 65] of order $\alpha > 0$ of function $f \in C_\mu, \mu \geq -1$is defined as

$$ {}_0^{\alpha}D_x^{-\alpha} f(x) = J^{\alpha} f(x) = \frac{1}{\Gamma(\alpha)} \int_0^x (x-t)^{\alpha-1} f(t)dt, \alpha > 0, x > 0. \qquad (2.3.1) $$

Definition:

The Riemann–Liouville fraction derivative operator [62-63, 65] of order $\alpha > 0$,of function $f \in C_\mu, \mu \geq -1$is defined as

$$ {}_0^{\alpha}D_x^{\alpha} f(x) = D_{RL}^{\alpha} f(x) = \frac{1}{\Gamma(\alpha)} \frac{d^m}{dx^m} \int_0^x (x-t)^{m-\alpha-1} f(t)dt, x > 0, \alpha \in (m-1, m), m \in N. \qquad (2.3.2) $$

Representation:

$$ D_{RL}^{\alpha} f(x) = D^m [J^{m-\alpha}(f(x))], m-1 < \alpha < m, m \in N, x > 0, f \in C_{-1}^m $$

The Riemann–Liouville derivative has certain disadvantages when trying to model real-world phenomena with fractional equations. Therefore, I introduce the modified fractional differential operator D_*^{α} proposed by Caputo in his work on the theory of viscoelasticity.

Definition:

The fractional derivative of $f(x)$ in the Caputo sense [61–62, 64–65] is defined as

$$ D_C^{\alpha} f(x) = \frac{1}{\Gamma(m-\alpha)} \int_0^x (x-t)^{m-\alpha-1} f^{(m)}(x)dt, m-1 < \alpha < m, m \in N, x > 0, f \in C_{-1}^m \qquad (2.3.3) $$

Representation:

$$ D_C^{\alpha} f(x) = J^{m-\alpha}[D^m(f(x))] $$

Caputo's definition is a modification of the Riemann–Liouville definition and has the advantage of dealing properly with initial value problems in which the initial conditions are given in terms of the field variables and their integer order, which is the case in most physical processes.

Properties of Fractional Integral and Derivatives:

I mention only the following properties [63, 66] for $f, g \in C_\mu, \mu \geq -1, \alpha, \beta \geq 0$.

Differintegration Term by Term:

The linearity of differintegral operators means that may be distributed through the terms of a finite sum, that is:

$$\frac{d^\alpha}{[d(x-a)]^\alpha}\sum_{j=0}^{n} f_j\,(x) = \sum_{j=0}^{n}\frac{d^\alpha f_j(x)}{[d(x-a)]^\alpha}.$$

I establish a generalisation of the classical theorem on differintegration term by term.

If the infinite series $\sum_{j=0}^{n} f_j\,(x)$ and $\sum_{j=0}^{n} d^\alpha f_j(x)/[d(x-a)]^\alpha$, the series converge uniformly in $0 < |x-a| < X$, then:

$$\frac{d^\alpha}{[d(x-a)]^\alpha}\sum_{j=0}^{n} f_j\,(x) = \sum_{j=0}^{n}\frac{d^\alpha f_j(x)}{[d(x-a)]^\alpha}, \alpha > 0.$$

Leibniz Rule:

For the Riemann–Liouville integral and derivative [63, 66]

$$D_x^{\pm\alpha}(f(x)g(x)) = \sum_{k=0}^{\infty}\frac{\Gamma(1+\beta)}{\Gamma(1+k)\Gamma(1-k+\beta)}(D^{\alpha-k}f(x))g^{(k)}(x).$$

However, for Caputo's derivative [4]

$$D\_*\,\hat{}\,\alpha\,(f(x)g(x)) = \sum_{k=0}^{\infty}\frac{\Gamma(1+\alpha)}{\Gamma(1+k)\Gamma(1-k+\alpha)}\left(f^{(k)}(x)\right)g^{(\alpha-k)}(x)$$

$$-\sum_{k=0}^{m-1}\frac{x^{k-\alpha}}{\Gamma(k+1-\beta)}((f(x)g(x))^{(k)}(0)).$$

Other Properties of the Operator J^{α} [61–62, 64–65]

For$f, g \in C_{\mu}, \mu \geq -1$ and $\alpha, \beta \geq 0$:

1. $J^{\alpha}[f(x) + g(x)] = J^{\alpha} f(x) + J^{\alpha} g(x)$,
2. 2. $J^{\alpha} J^{\beta} f(x) = J^{\alpha+\beta} f(x)$,
3. 3. $J^{\alpha} J^{\beta} f(x) = J^{\beta} J^{\alpha} f(x)$,

Also, note the following basic properties [61–62]:

Lemma 1: If $m-1 < \alpha < m,\ m \in N,\ x > 0,\ f \in C_{\mu}^{m}, \mu \geq -1$, then

$$D_C^{\alpha} J^{\alpha} f(x) = f(x),$$

and

$$J^{\alpha} D_C^{\alpha} f(x) = f(x) - \sum_{k=0}^{m-1} \frac{x^k}{k!} f^{(k)}(0^+), x > 0.$$

Lemma 2: If $m-1 < \alpha < m,\ m \in \mathrm{Z}^{+},\ x > 0,\ f \in C_{\mu}^{m}, \mu \geq -1$, then

$$D_{RL}^{\alpha} J^{\alpha} f(x) = f(x),$$

and

$$J^{\alpha} D_{RL}^{\alpha} f(x) = f(x) - \sum_{k=0}^{m} \frac{x^{\alpha-k}}{\Gamma(\alpha-k+1)} D_{RL}^{\alpha} f^{(k)}(0^+), x > 0.$$

Theorem 1: If $m-1 < \alpha < m,\ m \in Z^{+},\ x > 0,\ f \in C_{\mu}^{m}, \mu \geq -1$, then

$$D_C^{\alpha} f(x) = D_{RL}^{\alpha} f(x) - \sum_{k=0}^{m-1} f^{(k)}(0^+) \frac{x^{k-\alpha}}{(k-\alpha)!}, x > 0.$$

2.4 The Adomian Method

2.4.1 The Adomian Decomposition Method

Solutions to nonlinear functional equations were provided by an effective method proposed by George Adomian, an American mathematician, in the 1980s. The ADM is the contemporary nomenclature for this method [67, 68, 69].

As described by Adomian (1988), Adomian and Rach (1991) and Adomian (1994), this method is based on decomposition of the nonlinear operator into a series in which the terms are obtained from a polynomial produced through expansion of Adomian polynomials—an analytic function—following the search for a solution representing a series [70]. Validating the convergence of the series of functions and computing the polynomials represents the challenging portion of this technique. Traditional methods are outstripped by the benefits of the ADM in a variety of ways [70, 71]. For example, solutions to nonlinear equations are obtained by averting perturbation.

Discretisation of the solution is not necessary in the decomposition process of the ADM, and this is one of its principal benefits relative to classical techniques. Thus, no large systems of nonlinear or linear equations are produced, unlike with the use of traditional numerical techniques. Further, high accuracy levels regarding the numerical solution are achieved and significantly less computer memory and time are required to discover a solution given its immunity to computation round-off errors. Hence, an efficacious numerical solution is provided by the ADM in the form of a precise approximation of the solution in closed form. A rapidly convergent infinite series with every term calculated expediently is representative of the form this technique generates for its realistic solutions.

With respect to types of integral and differential equations, such as inhomogeneous or homogeneous, nonlinear or linear equations possessing variable coefficients or constant coefficients, the obtention of numerical approximations in closed form is facilitated substantially by the effective ADM [72]. Partial differential, differential delay, differential, integral, integro-differential and algebraic equations are among the broad class of deterministic and stochastic problems within engineering and

science that are formal solutions provided via implementation of this technique. This is exemplified by the utilisation of this method to solve variational problems [73], provide approximate solutions for unusual differential equations [74] and discover a solution to nonlinear vibrations of multi-walled carbon nanotubes [75].

2.4.2 Analysis of the Adomian Decomposition Method

The decomposition method requires that the nonlinear fractional differential equation be expressed in operator form:

$$D_{*t}^{\alpha}u(x,t) + L[u(x,t)] + N(u(x,t)) = g(x,t), x > 0, m-1 \leq \alpha \leq m, m \in N \tag{2.4.1}$$

where L is a linear operator that might include other fractional derivatives of order less than α, $\aleph$ is a nonlinear operator that also might include other fractional derivatives of order less than α, D_{*t}^{α} is the Caputo fractional derivative (CFD) [62, 64–63] of order α and g is the source function.

Applying the operator J^{α} ,the inverse the operator D_{*t}^{α} , to both sides of equation (1.2.1) yields

$$u(x,t) = \sum_{k=0}^{m-1} \frac{\partial^k u}{\partial^k t}(x,0^+)\frac{t^k}{k!} + J^{\alpha}g(x,t) - J^{\alpha}[L[u(x,t)] + N(u(x,t))]. \tag{2.4.2}$$

The ADM [61,67–77] suggests the solution $u(x,t)$ be decomposed into the infinite series of components:

$$u(x,t) = \sum_{n=0}^{\infty} u_n(x,t), \tag{2.4.3}$$

and the nonlinear function in Eq. (2.4.2) be decomposed as follows:

$$N(u(x,t)) = \sum_{n=0}^{\infty} A_n, \tag{2.4.4}$$

where A_n are the so-called Adomian polynomials [76].

Substitution of the decomposition series Eqs. (2.4.3) and (2.4.4) into both sides of Eq. (2.4.2) gives:

$$\sum_{n=0}^{\infty} u_n(x,t) = \sum_{k=0}^{m-1} \frac{\partial^k u}{\partial^k t}(x,0^+)\frac{t^k}{k!} + J^\alpha g(x,t) - J^\alpha[L(\sum_{n=0}^{\infty} u_n(x,t)) + \sum_{n=0}^{\infty} A_n]. \tag{2.4.5}$$

From this equation, the iterates are determined in the following recursive way:

$$\begin{aligned} u_0(x,t) &= \sum_{k=0}^{m-1} \frac{\partial^k u}{\partial^k t}(x,0^+)\frac{t^k}{k!} + J^\alpha g(x,t), \\ u_1(x,t) &= -J^\alpha(Lu_0(x,t) + A_0), \\ u_3(x,t) &= -J^\alpha(Lu_2(x,t) + A_2), \\ &\vdots \\ u_{n+1}(x,t) &= -J^\alpha(Lu_n(x,t) + A_n). \end{aligned} \tag{2.4.6}$$

The Adomian polynomial can be calculated for all forms of nonlinearity according to specific algorithms constructed by Adomian [76].

The general form of formula for Adomian polynomials is

$$A_n = \frac{1}{n!}[\frac{d^n}{d\lambda^n}[N(\sum_{k=0}^{\infty}\lambda^k u_k)]]_{\lambda=0}. \tag{2.4.7}$$

This formula is easy to compute using Mathematica software or by writing a computer code to obtain as many polynomials as needed for calculation of the numerical as well as explicit solutions.

Finally, we approximate the solution $u(x,t)$ via a truncated series:

$$\underset{N\to\infty}{Limt} \sum_{n=0}^{N-1} u_n(x,t) = u(x,t). \tag{2.4.8}$$

However, in many cases the exact solution in a closed form may be obtained. Moreover, the decomposition series solutions generally converge very rapidly. Convergence of the decomposition series has been investigated [77].

CHAPTER 3

Approximate Solution of the Dissipative Wave Equation via the Cubic B-Spline Method

Chapter 3: Approximate Solution of the Dissipative Wave Equation via the Cubic B-Spline Method

3.1 Introduction

In the past few decades, PDEs have attracted considerable attention owing to their ability to model certain physical phenomena. The NPDE is relevant to a wide variety of physical phenomena in several topics in physics, such as water wave theory, fluid dynamics, plasma physics, solid mechanics and nonlinear optics. There are many methods for solving PDEs via numerical solutions. One of these is numerically solving a nonlinear dissipative wave equation using the ADM [78,79]. The CBS has been used by many researchers to solve NPDEs. The most well-known and well-focused results are those of Dağ et al. (2004), who presented a way to solve regularised long wave (RLW) equations. The numerical results obtained in that paper demonstrate that the method is capable of solving RLW equations accurately and reliably [80]. Dağ et al. also published a description of a numerical solution for the 1D Burgers equation in 2005.

Comparison of the calculations with the analytic solution shows that a CBS collocation method is capable of accurately solving the Burgers equation. The proposed method is easy to implement and does not require any inner iteration or corrector to deal with the nonlinear term of the Burgers equation [81]. Khalifa et al. (2008) discussed a modified RLW equation. They applied a collocation method using CBSs to study the solitary waves of their equation, showing that the scheme is marginally stable. Moreover, although the wave does not change, the results showed that the interaction results in a tail of small amplitude in two, and clearly in three, soliton interactions and the conservation laws were reasonably satisfied. The appearance of such a tail can be beneficial for further studies [82]. In 2008, El-Danaf and E.I. Abdel Alaal constructed a non-polynomial spline-based method to obtain numerical solutions for a dissipative wave equation. The numerical results obtained showed that their proposed method retains good accuracy [83]. Later, Mittal and Jain (2012) argued that numerical methods should be proposed to approximate the

solution of the nonlinear parabolic PDE with Neumann's boundary conditions. The numerical results produced by the present method are quite satisfactory and in good agreement with the exact solutions. The computed results justify the advantage of this method. The proposed method can be extended to solve multi-dimensional parabolic equations [84].

In 2015, Zaki developed a numerical method based on quadratic non-polynomial spline functions, which has three coefficients in each sub interval for solving a dissipative wave equation. The results obtained by the proposed technique showed that the approach is easy to implement and computationally attractive. The proposed method was shown to be robust, efficient and easy to implement for linear and nonlinear problems arising in science and engineering [85]. A year later, El-Danaf et al. presented methods for solving generalised regularised long wave (GRLW) equations. The CBSs used to study the solitary waves of GRLW equations showed that the scheme is unconditionally stable. Also, the approximate numerical solutions obtained showed good accuracy compared with the exact solutions [86].

Hepson and Dağ, in their 2017 research, implemented a numerical technique to obtain approximate solutions to Fisher's equation. The method is capable of producing fair solutions for Fisher's equation and can be used as an alternative to the method's accompanying B-spline functions [87]. In 2017, Iqbal et al.'s proposed numerical technique is based on the CBS collocation method. Their version uses a new approximation for the second-order derivative. The proposed scheme is based on the CBS collocation method equipped with a new approximation for the second-order derivative, and produces fifth-order accurate results. The proposed method also generates a piecewise spline solution in the presence of singularity, which can be used to obtain a numerical solution at any point in the domain and is not restricted to the values at the selected knots, unlike existing finite difference methods [88].

A year later, Başhan (2018) studied numerical solutions for the third-order nonlinear Korteweg–de Vries (KdV) equation by using modified CBSs in five different test problems. The performance and accuracy of the modified CBS method was shown by calculating and comparing the L_2 and L_∞ error norms with earlier work. A stability analysis was performed for all of the test problems, and all eigenvalues are in

convenience with stability criteria. Thus, the modified cubic B-spline differential quadrature (MCBC-DQM) method may be useful for obtaining numerical solutions for other important nonlinear problems [89]. In research conducted in 2019 by Başhan, the MCBC-DQM method was successfully implemented for a numerical solution to the nonlinear Kawahara equation to obtain the first-, third- and fifth-order derivative approximations. Four test problems were investigated separately. These newly obtained results clearly indicate that the MCBC-DQM method can be used to produce numerical solutions to the Kawahara equation with high accuracy [90].

More recently, Iqbal et al. studied the Galerkin method based on a CBS function, where the shape and weight functions were applied to find a numerical solution for the 1D-coupled nonlinear Schrödinger equation. The use of the CBS Galerkin method produced smooth solutions without numerical smearing in 2020 [91]. In the same year, Ahmed et al. (2020) used a non-polynomial spline function to obtain numerical solutions to a dissipative wave equation at the middle points for a lattice in the space direction; at the same time, a finite difference method was used in the time direction. The presented method was shown to be conditionally stable. The approximating results proved to have good agreement with the true solutions; hence the method can be used to find approximate solutions for these types of problem [92].

In the current work, I propose a mathematical treatment for the nonlinear dissipative wave equation, utilising the collocation technique with CBS shape functions. For the mathematical methodology, the time derivatives are achieved by using the typical finite difference method. The technique is shown to be conditionally stable by applying the Von Neumann stability investigation procedure. I test the precision of the proposed strategy by conducting an examination of the mathematical outcomes and the specific arrangement of the condition.

3.2 The Governing Equation and the Derivation of the Proposed Method

This section is concerned with applying the CBS method to develop a mathematical strategy for approximating the specific arrangement of a nonlinear dissipative wave equation [78] of the structure

$$u_{tt} - u_{xx} + 2u_t u = \eta(x,t), \eta(x,t) = -2\sin^2 x \sin t \cos t. \qquad (3.2.1)$$

Under the boundary and initial conditions

$$u_{xx}(a,t) = 0, u_{xx}(b,t) = 0,$$

$$u(x,0) = \sin x, u_t(x,0) = 0. \qquad (3.2.2)$$

The interval $[a,b]$ can be divided into equal subintervals $[x_{i-1}, x_i]$, $i = 0,1,\dots,N+1$, where $x_i = a + ih$, and $h = \frac{b-a}{n}$.

Let the CBS basis functions $\emptyset_i(x)$ be given as

$$\emptyset_i(x) = \frac{1}{h^3}\begin{cases} (x - x_{i-2})^3 & x \in [x_{i-2}, x_{i-1}] \\ h^3 + 3h^2(x - x_{i-1}) + 3h(x - x_{i-1})^2 - 3(x - x_{i-1})^3 & x \in [x_{i-1}, x_i] \\ h^3 + 3h^2(x_{i+1} - x) + 3h(x_{i+1} - x)^2 - 3(x_{i+1} - x)^3 & x \in [x_i, x_{i+1}] \\ (x_{i-2} - x)^3 & x \in [x_{i+1}, x_{i+2}] \\ 0 & Otherwise \end{cases}$$

where $\{\emptyset_i\}$ for $i = 0,1,\dots,N+1$ are the basis for the function defined over the interval $[a,b]$; this implies that the estimates of the CBS $\emptyset_i(x)$ and its derivatives vanish outside the interval $[x_{i-2}, x_{i+2}]$, $i = 0,1,\dots\dots N$.

Table 3.2.1 The values of $\emptyset_i(x)$ and their derivative within the interval $[x_{i-2}, x_{i+2}]$.

x	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}
$\emptyset_i(x)$	0	1	4	1	0
$\emptyset_i'(x)$	0	3/h	0	-3/h	0
$\emptyset_i''(x)$	0	$6/h^2$	$-12/h^2$	$6/h^2$	0

We obtain three ODEs as follows:

$$\begin{cases} U_j = U(x_j,t) = g_{j-1} + 4g_j + g_{j+1}, \\ U_j' = U'(x_j,t) = \frac{3}{h}(g_{j+1} - g_{j-1}), \\ U_j'' = U''(x_j,t) = \frac{6}{h^2}(g_{j-1} - 2g_j + g_{j+1}). \end{cases}$$

The mathematical treatment for Eq. (3.2.1) by the collocation method with CBSs involves tracking down an inexact arrangement $U_N(x,t)$ to the exact solution $u(x,t)$. Set the approximate solution $U_N(x,t)$ as follows:

$$U_N(x,t) = \sum_{i=-1}^{N+1} \omega_i(t)\emptyset_i(x), \tag{3.2.3}$$

where $\omega_i(t)$ are time-dependent parameters that can be resolved utilising the boundary conditions

$$(U_{xx})_N(a,t) = 0, (U_{xx})_N(b,t) = 0, \tag{3.2.4}$$

and the collocation form of Eq. (3.2.1)

$$(U_{tt})_N(x_j,t) - (U_{xx})_N(x_j,t) + 2(U)_N(x_j,t)(U_t)_N(x_j,t) = \eta(x_j,t). \tag{3.2.5}$$

By substituting Eq. (3.2.3) into Eq. (3.2.5), we obtain

$$\sum_{i=-1}^{N+1} \frac{d^2\omega_i(t)}{dt^2}\emptyset_i(x) - \sum_{i=-1}^{N+1} \omega_i\,(t)\emptyset_i''(x_j) + 2\omega_i(t)\emptyset_i(x_j)\sum_{\delta=-1}^{N+1} \frac{d\omega_\delta(t)}{dt}\emptyset_\delta(x_j) = \eta_j^n(x,t). \tag{3.2.6}$$

Applying the finite difference method results in

$$\omega_i^n = \frac{\omega_i^{n+1}+\omega_i^{n-1}}{2}, \frac{d^2\omega}{dt^2} = \frac{\omega_i^{n-1}-2\omega_i^n+\omega_i^{n+1}}{k^2}, \text{ where } k = \Delta t. \tag{3.2.7}$$

By substituting Eq. (3.2.7) into Eq. (3.2.6) and simplifying the results, we obtain

$$\sum_{i=-1}^{N+1}\left[\omega_i^{n-1} - 2\omega_i^n + \omega_i^{n+1}\right]\emptyset_i(x_j) - k^2\sum_{i=-1}^{N+1}\omega_i\,(t)\,\emptyset_i''(x_j) + k^2\left[\omega_i^{n+1} + \omega_i^{n-1}\right]\emptyset_i(x_j)\sum_{\delta=-1}^{N+1}\frac{d\omega_\delta(t)}{dt}\,\emptyset_\delta(x_j) = k^2\eta_j^n(x,t). \tag{3.2.8}$$

Using the values of $\emptyset_i(x_j)$ and $\emptyset_i''(x_j)$ from Table 3.2.1

$$\omega_{i-1}^{n-1} - 2\omega_{i-1}^n + \omega_{i-1}^{n+1} + 4\omega_i^{n-1} - 8\omega_i^n + 4\omega_i^{n+1} + \omega_{i+1}^{n-1} - 2\omega_{i+1}^n + \omega_{i+1}^{n+1} - \frac{6}{h^2}\,k^2\omega_{i-1}^n$$

$$+\frac{12}{h^2}\,k^2\omega_i^n-\frac{6}{h^2}\,k^2\omega_{i+1}^n+\,k^2\left[\omega_{i-1}^{n+1}+\omega_{i-1}^{n-1}\right]\emptyset_i(x_j)\frac{d\omega_\delta(t)}{dt}$$
$$+4\,k^2\left[\omega_i^{n+1}+\omega_i^{n-1}\right]\emptyset_i(x_j)\frac{d\omega_\delta(t)}{dt}$$
$$+k^2\left[\omega_{i+1}^{n+1}+\omega_{i+1}^{n-1}\right]\emptyset_i(x_j)\frac{d\omega_\delta(t)}{dt}=k^2\eta_j^n(x,t),$$

$$(1+\,k^2\,\emptyset_i(x_j)\frac{d\omega_\delta(t)}{dt})\omega_{i-1}^{n+1}+(\,4+\,4\,k^2\,\emptyset_i(x_j)\frac{d\omega_\delta(t)}{dt})\omega_i^{n+1}+(\,1$$
$$+\,k^2\,\emptyset_i(x_j)\frac{d\omega_\delta(t)}{dt})\omega_{i+1}^{n+1}$$

$$+(-2\,-\frac{6}{h^2}\,k^2)\omega_{i-1}^n+(\,-8+\frac{12}{h^2}\,k^2)\omega_i^n+(\,-2-\,\frac{6}{h^2}\,k^2)\omega_{i+1}^n+(1$$
$$+\,k^2\,\emptyset_i(x_j)\frac{d\omega_\delta(t)}{dt}\,)\omega_{i-1}^{n-1}$$

$$+(4+\,4\,k^2\,\emptyset_i(x_j)\frac{d\omega_\delta(t)}{dt})\omega_i^{n-1}+(1+\,k^2\,\emptyset_i(x_j)\frac{d\omega_\delta(t)}{dt})\omega_{i+1}^{n-1}=k^2\eta_i^n(x,t),$$

Eq. (3.2.8) can be determined at $x_j, j=0,1,2,\ldots,N$, so that

$$a_i\omega_{i-1}^{n+1}+b_i\omega_i^{n+1}+c_i\omega_{i+1}^{n+1}=-d_i\omega_{i-1}^n-e_i\omega_i^n-f_i\omega_{i+1}^n-n_i\omega_{i-1}^{n-1}-s_i\omega_i^{n-1}-$$
$$l_i\omega_{i+1}^{n-1}+k^2\eta_i^n(x,t), \qquad (3.2.9)$$

where

$$a_i=1+k^2z_{i-1}, d_i=-2-\frac{6}{h^2}k^2, n_i=1+k^2z_{i-1},$$

$$b_i=4+4k^2z_{i-1}, e_i=-8+12\frac{k^2}{h^2}, s_i=4+4k^2z_{i-1}, \qquad (3.2.10)$$

$$c_i=1+k^2z_{i-1}, f_i=-2-\frac{6}{h^2}k^2, l_i=2+k^2z_{i-1}.$$

with $z_{i-1}=\frac{\partial U_N(x_i,t_n)}{\partial t}=[\frac{\omega_{i-1}^n-\omega_{i-1}^{n-1}}{k}+4\frac{\omega_i^n-\omega_i^{n-1}}{k}+\frac{\omega_{i+1}^n-\omega_{i+1}^{n-1}}{k}]$,

for all $i=0,1,2,\ldots,N$. The nonlinear logarithmic system in Eq. (3.2.9) contains (N+1) equations of (N+3) unknowns. To find the solution for this system, we need

two additional conditions that are obtained from the conditions in Eq. (3.2.4) as follows:

$$\frac{6}{h^2}\omega_{-1} - \frac{12}{h^2}\omega_0 + \frac{6}{h^2}\omega_1 = 0,$$

$$\frac{6}{h^2}\omega_{N-1} - \frac{12}{h^2}\omega_N + \frac{6}{h^2}\omega_{N+1} = 0. \qquad (3.2.11)$$

System Eq. (3.2.9) and additional equations in (3.2.11) have (N+3) equations with (N+3) unknowns, so we can identify the time-dependent variables ω_i in the matrix form:

$$A\omega^{n+1} = -B\omega^n - C\omega^{n-1} + k^2\eta_i^n(x,t), \qquad (3.2.12)$$

where

$$A = \begin{bmatrix} \frac{6}{h^2} & \frac{-12}{h^2} & \frac{6}{h^2} & 0 & \dots & 0 \\ a_0 & b_0 & c_0 & 0 & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & . \\ . & . & \dots & .. & \dots & 0 \\ 0 & \dots & 0 & a_N & b_N & c_N \\ 0 & \dots & 0 & \frac{6}{h^2} & \frac{-12}{h^2} & \frac{6}{h^2} \end{bmatrix},$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 \\ d_0 & e_0 & f_0 & 0 & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & . \\ . & . & \dots & .. & \dots & 0 \\ 0 & \dots & 0 & d_N & e_N & f_N \\ 0 & \dots & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 \\ n_0 & s_0 & l_0 & 0 & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & . \\ . & . & \dots & .. & \dots & 0 \\ 0 & \dots & 0 & n_N & s_N & l_N \\ 0 & \dots & 0 & 0 & 0 & 0 \end{bmatrix}.$$

3.3 The Initial State

In this section, we apply the first initial condition:

$$u(x,0) = \sin x \qquad (3.3.1)$$

The initial conditions can be communicated as

$$(U_x)_N(a,0) = u_x(a,0),$$

$$U_N(x_j,0) = \sum_{i=-1}^{N+1} \emptyset_i\,(x_j)\,\omega_i^0, j = 0,1,2,\ldots,N, \tag{3.3.2}$$

$$(U_x)_N(b,0) = u_x(b,0).$$

By using the values of $\emptyset_i$ and their derivatives in Table 2.2.4.1, the system in (3.3.2) takes the structure

$$-3\omega_{-1}^0 + 3\omega_1^0 = hu_x(a,0),$$

$$\omega_{i-1}^0 + 4\omega_i^0 + \omega_{i+1}^0 = u(x_j,0), j = 0,1,2,\ldots,N, \tag{3.3.3}$$

$$-3\omega_{N-1}^0 + 3\omega_{N+1}^0 = hu_x(b,0).$$

I rewrite the system (3.3.3) in matrix form:

$$Mv = q \tag{3.3.4}$$

where

$$M = \begin{bmatrix} -3 & 0 & 3 & 0 & \ldots & 0 \\ 1 & 4 & 1 & 0 & \ldots & 0 \\ 0 & \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & .. & \ldots & 0 \\ 0 & \ldots & 0 & 1 & 4 & 1 \\ 0 & \ldots & 0 & -3 & 0 & 3 \end{bmatrix},$$

and $v = (\omega_{-1}^0, \omega_0^0, .., \omega_N^0, \omega_{N+1}^0)^T$, $q = (hu_x(a,0), u(x_0,0), \ldots\ldots, u(x_N,0), hu_x(b,0))^T$.

To find the second initial condition using Taylor expansion to $U_N(x,t_i)$ at $t = t_0$ we apply

$$U_N(x,t_1) = U_N(x,t_0) + k\frac{\partial U_N(x,t_0)}{\partial t} + \frac{k^2}{2!}\frac{\partial^2 U_N(x,t_0)}{\partial t^2} + O(k^3),\ k = t_1 - t_0. \tag{3.3.5}$$

Setting $t_0 = 0$, we obtain

$$U_N(x,t_1) = U_N(x,0) + k\frac{\partial U_N(x,0)}{\partial t} + \frac{k^2}{2!}\frac{\partial^2 U_N(x,0)}{\partial t^2} + O(k^3). \tag{3.3.6}$$

Subbing Eq. (3.2.1) into Eq. (3.3.6) results in

$$U_N(x,t_1) \approx U_N(x,0) + \frac{k^2}{2!}\left(\frac{\partial^2 U_N(x,0)}{\partial x^2} - 2u\frac{\partial U_N(x,0)}{\partial t} + \eta(x,0)\right). \tag{3.3.7}$$

After simplifying, Eq. (3.3.7) becomes

$$U_N(x,t_1) \approx U_N(x,0) + \frac{k^2}{2!}\frac{\partial^2 U_N(x,0)}{\partial x^2}, \tag{3.3.8}$$

where $\eta(x,0) = 0$.

Substituting Eq. (3.3.1) and initial condition (3.2.2) into Eq. (3.3.8), we obtain

$$\sum_{i=-1}^{N+1} \varphi(x_j)\omega_i^1 \approx \eta(x_j), j = 0,1,\dots,N, \tag{3.3.9}$$

where

$$\eta(x_j) = \sin x_j - \frac{k^2}{2!}\sin x_j.$$

To complete this system, we differentiate Eq. (3.3.9) with respect to *x*, and compute its value at the ends of the range, which gives the following system:

$$\begin{aligned} -3\omega_{-1}^1 + 3\omega_1^1 &= h\eta'(x_0), \\ \omega_{i-1}^1 + 4\omega_i^1 + \omega_{i+1}^1 &= h\eta(x_j), \\ -3\omega_{N-1}^1 + 3\omega_{N+1}^1 &= h\eta'(x_N). \end{aligned} \tag{3.3.10}$$

The system (3.3.10) can be expressed in a matrix equation form as $My = H$, where

$$M = \begin{bmatrix} -3 & 0 & 3 & 0 & \dots. & 0 \\ 1 & 4 & 1 & 0 & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & 0 & 1 & 4 & 1 \\ 0 & \dots & 0 & -3 & 0 & 3 \end{bmatrix},$$

and

$$y = (\omega_{-1}^1, \omega_0^1, \ldots, \omega_N^1, \omega_{N+1}^1)^T,$$

while *H* has the form:

$$H = \left(h\eta'(x_0), h\eta(x_0), \ldots\ldots\ldots\ldots, h\eta(x_N), h\eta'(x_N)\right)^T.$$

3.4 Stability Analysis

The Von Neumann stability analysis for system (3.2.9) takes effect after linearising the nonlinear term as

$$z_{i-1} = d + 4d + d = (6d), m = 6d.$$

Then the Von Neumann stability analysis takes the form

$$\omega_j^n = \varepsilon^n \, exp(\, q\sigma jh), q = \sqrt{-1}, \tag{3.4.1}$$

where σ is the wave number and h is the element size. At $x = x_i$, Eq. (3.2.9) can be written as

$$a_i\omega_{i-1}^{n+1} + b_i\omega_i^{n+1} + c_i\omega_{i+1}^{n+1} = -d_i\omega_{i-1}^n - e_i\omega_i^n - f_i\omega_{i+1}^n - n_i\omega_{i-1}^{n-1} - s_i\omega_i^{n-1} - l_i\omega_{i+1}^{n-1} + k^2\eta_i^n. \tag{3.4.2}$$

Substituting Eq. (3.4.1) into the recurrence relationship (3.4.2) results in

$$a_j\varepsilon^{n+1}\exp\left(q\emptyset h(j-1)\right) + b_j\varepsilon^{n+1}\exp(q\emptyset hj) + c_j\varepsilon^{n+1}\exp\left(q\emptyset h(j+1)\right) + d_j\varepsilon^n\exp\left(q\emptyset h(j-1)\right) + e_j\varepsilon^n\exp(q\emptyset hj) + f_j\varepsilon^n\exp\left(q\emptyset h(j+1)\right) + n_j\varepsilon^{n-1}\exp\left(q\emptyset h(j-1)\right) + s_j\varepsilon^{n-1}\exp(q\emptyset hj) + l_j\varepsilon^{n-1}\exp\left(q\emptyset h(j+1)\right) = k^2\eta_j^n.$$

Dividing both sides by $\varepsilon^{n-1}\, exp(\, jq\phi h)$, results in

$$a_j\varepsilon^2\exp(-jq\emptyset h) + b_j\varepsilon^2 + c_j\varepsilon^2\exp(jq\emptyset h) + d_j\varepsilon\exp(-jq\emptyset h) + e_j\varepsilon + f_j\varepsilon\exp(jq\emptyset h) + n_j\exp(-jq\emptyset h) + s_j + l_j\exp(jq\emptyset h) = \frac{k^2\eta_j^n}{\varepsilon^{n-1}\, exp(jq\phi h)}.$$

Using Euler's formula, results in

$$exp[\,q\varphi] = \cos\varphi + q\sin\varphi\,, \quad \varphi = \phi h,$$

$$a_j\varepsilon^2(cos\emptyset h - qsin\emptyset h) + b_j\varepsilon^2 + c_j\varepsilon^2(cos\emptyset h + qsin\emptyset h) + d_j\varepsilon(cos\emptyset h - qsin\emptyset h) + e_j\varepsilon$$

$$+f_j\varepsilon(cos\emptyset h + qsin\emptyset h) + n_j(cos\emptyset h - qsin\emptyset h) + s_j + l_j(cos\emptyset h + qsin\emptyset h)$$

$$= \frac{k^2\eta_j^n}{\varepsilon^{n-1}\,exp(\,jq\phi h)},$$

$$\varepsilon^2\big[\,a_j cos\emptyset h - a_j qsin\emptyset h + b_j + c_j cos\emptyset h + c_j qsin\emptyset h\,\big]$$

$$+\varepsilon\,\big[d_j cos\emptyset h - d_j qsin\emptyset h + e_j + f_j cos\emptyset h + f_j qsin\emptyset h\big]$$

$$+[\,n_j cos\emptyset h - n_j qsin\emptyset h + s_j + l_j cos\emptyset h + l_j qsin\emptyset h] = \frac{k^2\eta_j^n}{\varepsilon^{n-1}\,exp(\,jq\phi h)}$$

$$\varepsilon^2\big[\big(c_j + a_j\big)\cos\phi\, h + b_j + q\big(c_j - a_j\big)\sin\phi\, h\big] + \varepsilon\big[\big(f_j + d_j\big)\cos\phi\, h + e_j + q\big(f_j - d_j\big)\sin\phi\, h\big] + \big[\big(l_j + n_j\big)\cos\phi\, h + s_j + q\big(l_j - n_j\big)\sin\phi\, h\big] = 0. \quad (3.4.3)$$

Thus, we have

$$\varepsilon^2[(2 + 2k^2m)\cos\phi\, h + (4 + 4k^2m)] + \varepsilon[(-4 - 2r_1)\cos\phi\, h + (-8 + 2r_1)]$$
$$+[(2 + 2k^2m)\cos\phi\, h + (4 + 4k^2m)] = 0, m = 6d, \quad (3.4.4)$$

where $r_1 = [\frac{6}{h^2}k^2]$.

Dividing Eq. (3.4.4) by $[(2 + 2k^2m)\cos\phi\, h + (4 + 4k^2m)]$ gives the equation

$$\varepsilon^2 + \varepsilon\frac{[(-4-2r_1)\cos\phi h + (-8+2r_1)]}{[(2+2k^2m)\cos\phi h + (4+4k^2m)]} + 1 = 0. \quad (3.4.5)$$

Eq. (3.4.5) can be written as:

$$\varepsilon^2 + 2\beta\varepsilon + 1 = 0, \quad (3.4.6)$$

where

$$\beta = \frac{[(-2-r_1)\cos\phi h + (-4+r_1)]}{[(2+2k^2m)\cos\phi h + (4+4k^2m)]}.$$

Eq. (3.4.6) is a quadratic in ε and hence will have two roots; that is $\varepsilon = -\beta \pm \sqrt{\beta^2 - 1}$.

For stability, $|\varepsilon| \leq 1$. Now, from Eq. (3.4.6) we see that the result of the two estimations of ε should rise to solidarity, which results in the following three cases.

Case 1: On the off chance that the two roots are equivalent to solidarity, which infers that the segregate of the Eq. (3.4.6) is zero.

Case 2: One of the two roots is more prominent than solidarity. At that point, the separate is more noteworthy than nothing. This implies that the steadiness condition, (|ε| ≤ 1), is not fulfilled.

Case 3: The discriminate is less than zero, that is $\beta^2 - 1 < 0$.

Thus for stability:

$$-1 \leq \beta \leq 1. \tag{3.4.7}$$

Using Eq. (3.4.7), the above inequality becomes

$$-1 \leq \frac{[(-2-r_1)\cos\phi h+(-4+r_1)]}{[(2+2k^2m)\cos\phi h+(4+4k^2m)]} \leq 1. \tag{3.4.8}$$

The right inequality (3.4.8) takes the form

$$\frac{[(-2-r_1)\cos\phi h+(-4+r_1)]}{[(2+2k^2m)\cos\phi h+(4+4k^2m)]} \leq 1. \tag{3.4.9}$$

$$[(-2 - r_1)\cos\phi\, h + (-4 + r_1)] \leq [(2 + 2k^2m)\cos\phi\, h + (4 + 4k^2m)]$$

$$r_1 \leq [(2 + 2k^2m)\cos\phi\, h + (4 + 4k^2m) + (2 + r_1)\cos\phi\, h + 4]$$

$$r_1 \leq (4 + 2k^2m + r_1)\cos\phi\, h + 4k^2m + 8.$$

After simplifying inequality (3.4.9), we obtain

$$\frac{6}{h^2}k^2 \leq \left[\left(\frac{6}{h^2}k^2 + 4 + 2k^2m\right)\cos\phi\, h + (8 + 4k^2m)\right],$$

or

$$6 \leq \left[\left(6 + 4\frac{h^2}{k^2} + 2h^2m\right)cos\,\phi\, h + \left(8\frac{h^2}{k^2} + 4h^2m\right)\right]. \qquad (3.4.10)$$

Then, using the relationship $cos\,\phi\, h = 1 - 2\,sin^2\frac{\phi h}{2}$, inequality (3.4.10) reduces to

$$6 \leq \left[\left(6 + 12\frac{h^2}{k^2} + 6h^2m\right) - \left(12 + 8\frac{h^2}{k^2} + 4h^2m\right)sin^2\frac{\phi h}{2}\right]. \qquad (3.4.11)$$

After simplifying inequality (3.4.11), we obtain

$$12\frac{h^2}{k^2} + 6h^2m \geq \left(12 + 8\frac{h^2}{k^2} + 4h^2m\right)sin^2\frac{\phi h}{2},$$

which is satisfied for $k << h$, if h is small enough. However, the left inequality (3.4.8) becomes

$$-1 \leq \frac{[(-2-r_1)\,cos\,\phi h + (-4+r_1)]}{[(2+2k^2m)\,cos\,\phi h + (4+4k^2m)]}. \qquad (3.4.12)$$

$$(-1)[(2 + 2k^2m)\,cos\,\phi\, h + (4 + 4k^2m)] \leq [(-2 - r_1)\,cos\,\phi\, h + (-4 + r_1)]$$

$$[(2 + 2k^2m)\,cos\,\phi\, h + (4 + 4k^2m)] \geq [(2 + r_1)\,cos\,\phi\, h + (4 - r_1)]$$

$$[(2k^2m - r_1)\,cos\,\phi\, h] \geq [(-4k^2m - r_1)]$$

$$[(r_1 - 2k^2m)\,cos\,\phi\, h] \leq [(4k^2m + r_1)].$$

After simplifying inequality (3.4.12), we obtain

$$\left(-2k^2m + \frac{6}{h^2}k^2\right)cos\,\phi\, h \leq \frac{6}{h^2}k^2 + 4k^2m,$$

or

$$\left(\frac{-mh^2}{3} + 1\right)cos\,\phi\, h \leq 1 + \frac{2mh^2}{3}. \qquad (3.4.13)$$

Using the relationship $cos\,\phi\, h = 1 - 2\,sin^2\frac{\phi h}{2}$, inequality (3.4.13) becomes

$$\left(\frac{-mh^2}{3} + 1\right) + \left(\frac{2mh^2}{3} - 2\right)sin^2\frac{\phi h}{2} \leq \left(1 + \frac{2mh^2}{3}\right), \qquad (3.4.14)$$

if h is small enough. Thus the method is conditionally stable.

3.5 Numerical Illustration

I apply the CBS method to obtain a numerical solution for the dissipative equation for one standard issue. The precision of the proposed mathematical technique is estimated by registering the L_∞ error norm. The exact solution to the dissipative Eq. (3.2.1) obtained in [78] is given by

$$u(x,t) = cos\, t\, sin\, x\,, 0 \le x \le \pi, t \ge 0.$$

I use the following conditions:

$$u(x,0) = sin\, x,$$

$$u_{xx}(0,t) = 0,$$

$$u_{xx}(\pi,t) = 0.$$

I enter the acquired mathematical outcomes in Tables 3.5.1–3.5.6.

Table 3.5.1 Comparison between the numerical and exact solutions at $t = 0.2, k = 0.002, h = \frac{\pi}{20}$.

x	Numerical Solution	Exact Solution
0.1π	0.302857	0.303116
0.2π	0.576069	0.576509
0.3π	0.792891	0.793442
0.4π	0.932099	0.932707
0.5π	0.980067	0.980692
0.6π	0.932099	0.932707
0.7π	0.792891	0.793442
0.8π	0.576069	0.576509
0.9π	0.302857	0.303116

Table 3.5.2 The $L_\infty error$ for the numerical and exact solutions when $k = 0.001, h = \frac{\pi}{20}$ from t=0.05 to t=0.2.

Time	0.05	0.1	0.15	0.20
$L_\infty error[Our]$	2.0097×10^{-5}	3.1051×10^{-5}	3.3695×10^{-5}	2.8898×10^{-5}
$L_\infty error[92]$	2.5236×10^{-4}	9.8616×10^{-4}	2.1532×10^{-3}	3.7090×10^{-3}

Table 3.5.3 The $L_\infty error$ for the numerical and exact solutions when $k = 0.01, h = \frac{\pi}{20}$ from t=0.5 to t=2.0.

Time	0.5	1.0	1.5	2.0
$L_\infty error[Our]$	6.2131×10^{-4}	6.2454×10^{-4}	1.92996×10^{-3}	3.87406×10^{-3}
$L_\infty error[92]$	8.7356×10^{-4}	2.5274×10^{-3}	4.4852×10^{-3}	7.5875×10^{-3}

Table 3.5.4 The L_∞ error for the numerical and exact solutions when $k = 0.01, h = \frac{\pi}{20}$ from *t*=6.0 to *t*=9.0.

Time	6.0	7.0	8.0	9.0
$L_\infty error$	2.9063×10^{-2}	2.7145×10^{-2}	1.7337×10^{-2}	3.1307×10^{-2}

Table 3.5.5 The $L_\infty error$ for the numerical and exact solutions for a big time when $k = 0.01, h = \frac{\pi}{20}$ from *t*=10.0 to *t*=40.0.

Time	10.0	20.0	30.0	40.0
$L_\infty error$	3.4557×10^{-2}	3.3574×10^{-2}	1.4119×10^{-2}	2.3355×10^{-2}

Table 3.5.6 Comparison between the numerical and exact solutions at $t = 2, k = 0.002, h = \frac{\pi}{20}$.

x	Numerical Solution	Exact Solution
0.1π	−0.128596	−0.129169
0.2π	−0.244605	−0.245756
0.3π	−0.33667	−0.338348
0.4π	−0.395779	−0.397833
0.5π	−0.416147	−0.418337
0.6π	−0.395779	−0.397833
0.7π	−0.33667	−0.338348
0.8π	−0.244605	−0.245756
0.9π	−0.128596	−0.129169

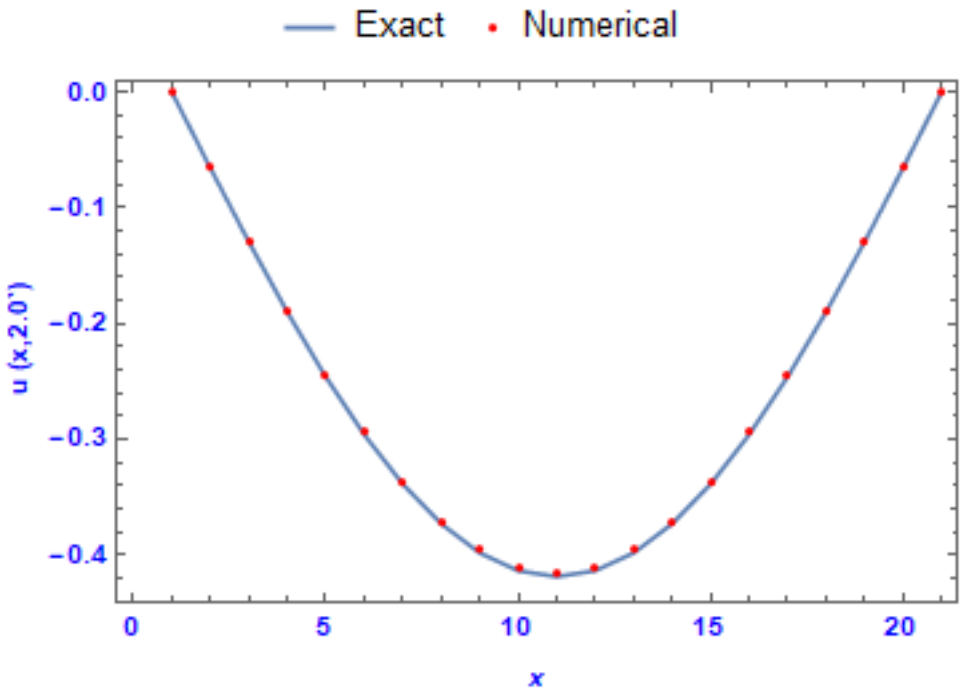


Figure 3.5.1 The exact and numerical results for time *t*=2.0 with *k*=0.01.

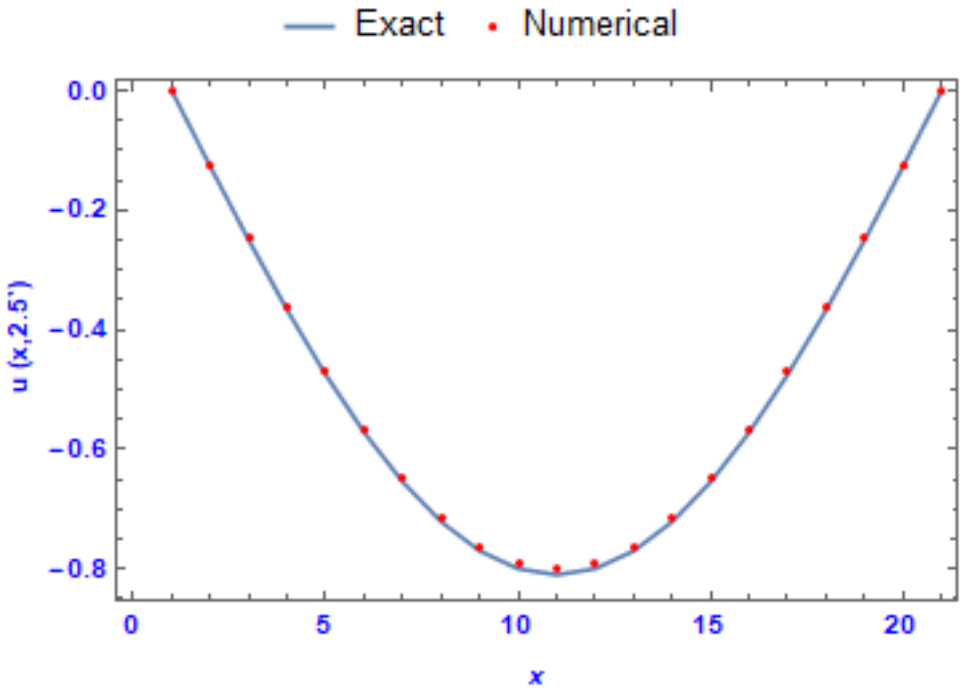


Figure 3.5.2 The exact and numerical results for time *t*=2.5 with *k*=0.01.

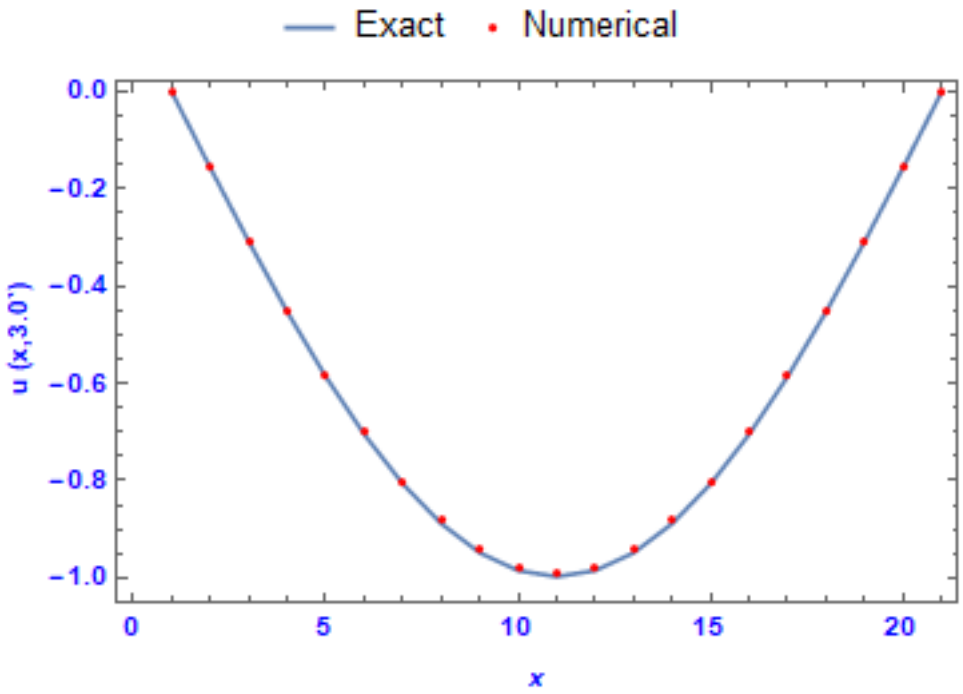


Figure 3.5.3 The exact and numerical results for time *t*=3.0 with *k*=0.01.

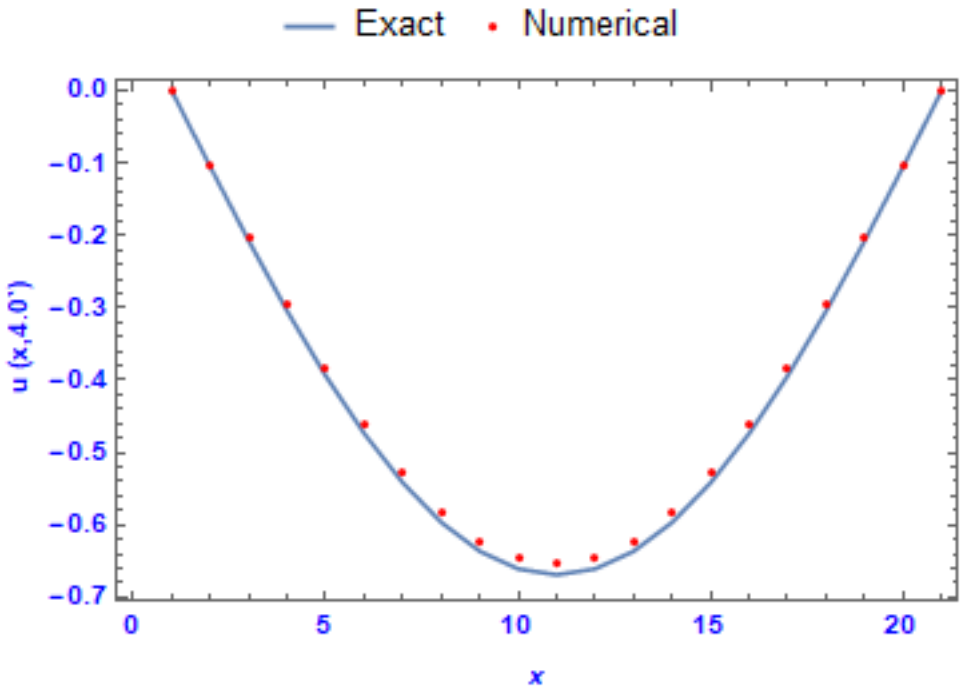


Figure 3.5.4 The exact and numerical results for time *t*=4.0 with *k*=0.01.

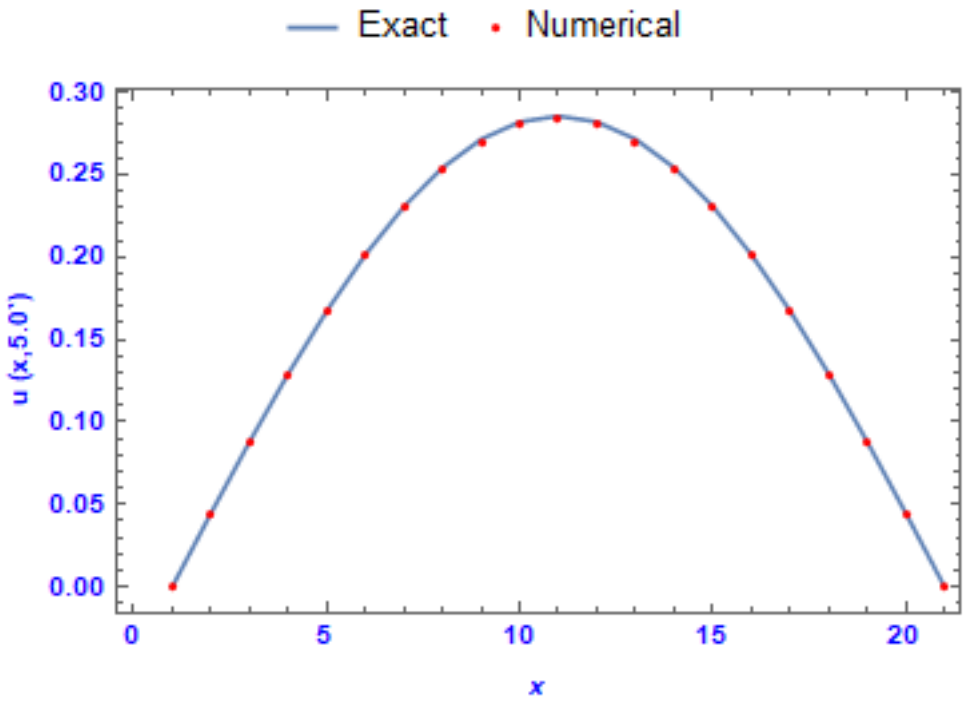


Figure 3.5.5 The exact and numerical results for time *t*=5.0 with *k*=0.01.

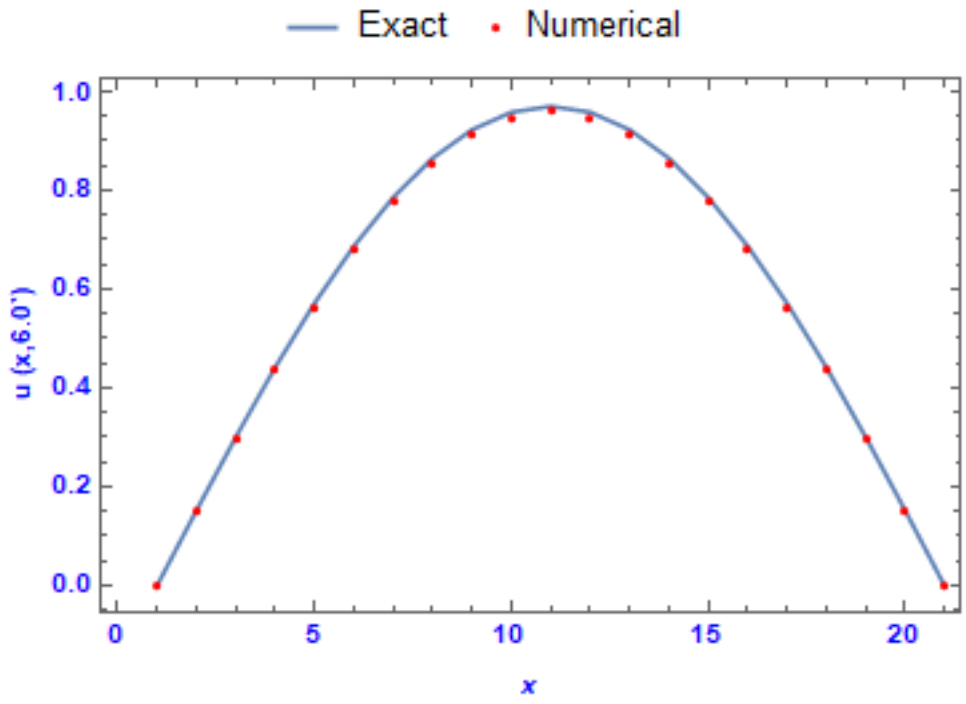


Figure 3.5.6 The exact and numerical results for time *t*=6.0 with *k*=0.01.

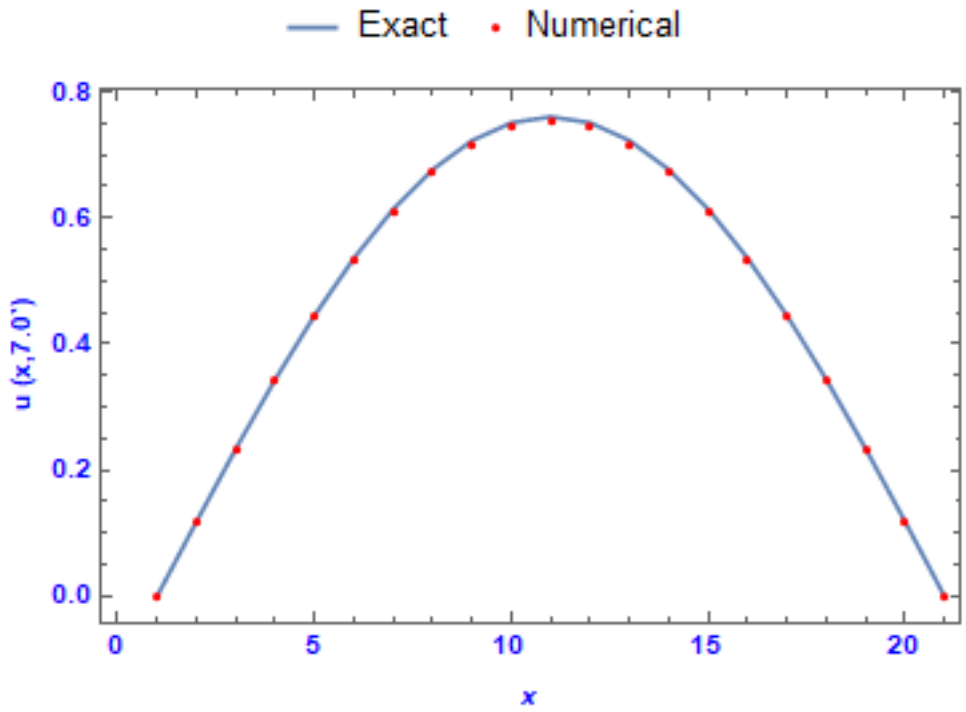


Figure 3.5.7 The exact and numerical results for time *t*=7.0 with *k*=0.01.

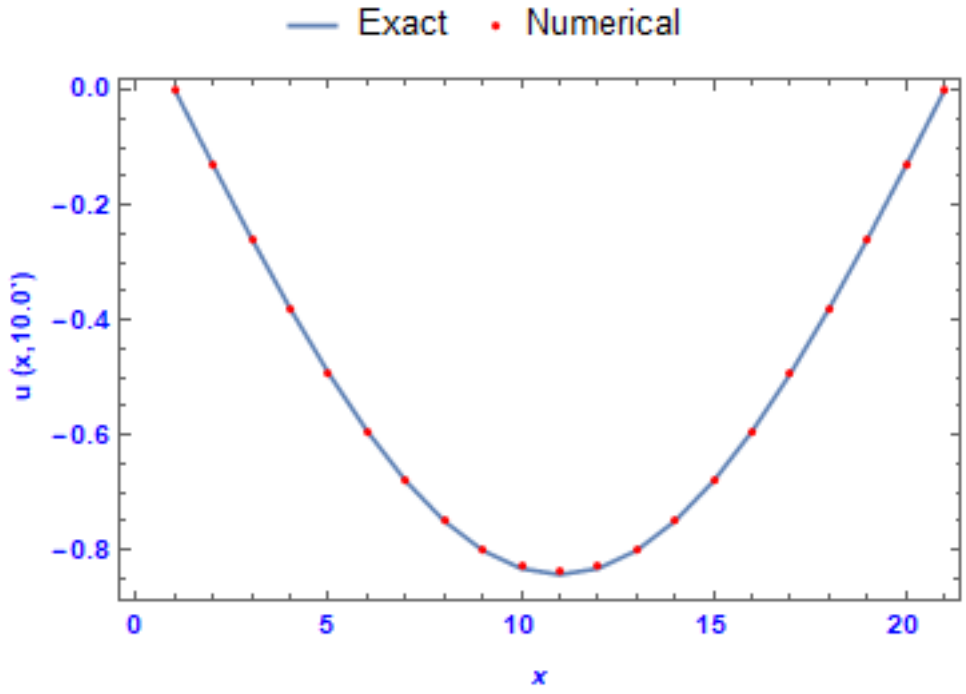


Figure 3.5.8 The exact and numerical results for time *t*=10.0 with *k*=0.01.

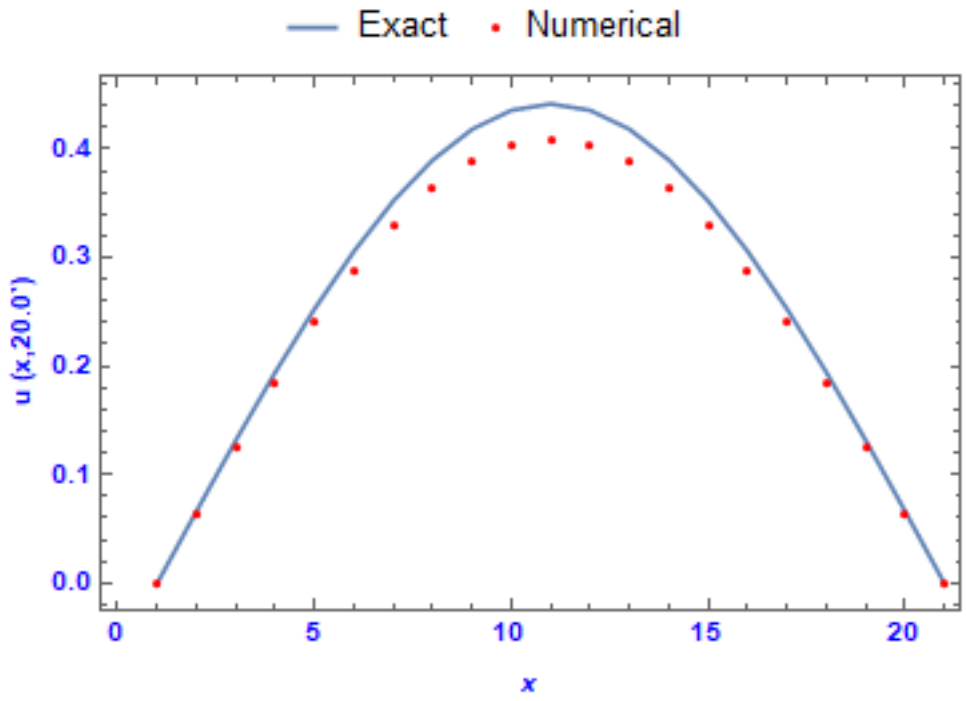


Figure 3.5.9 The exact and numerical results for time *t*=20.0 with *k*=0.01.

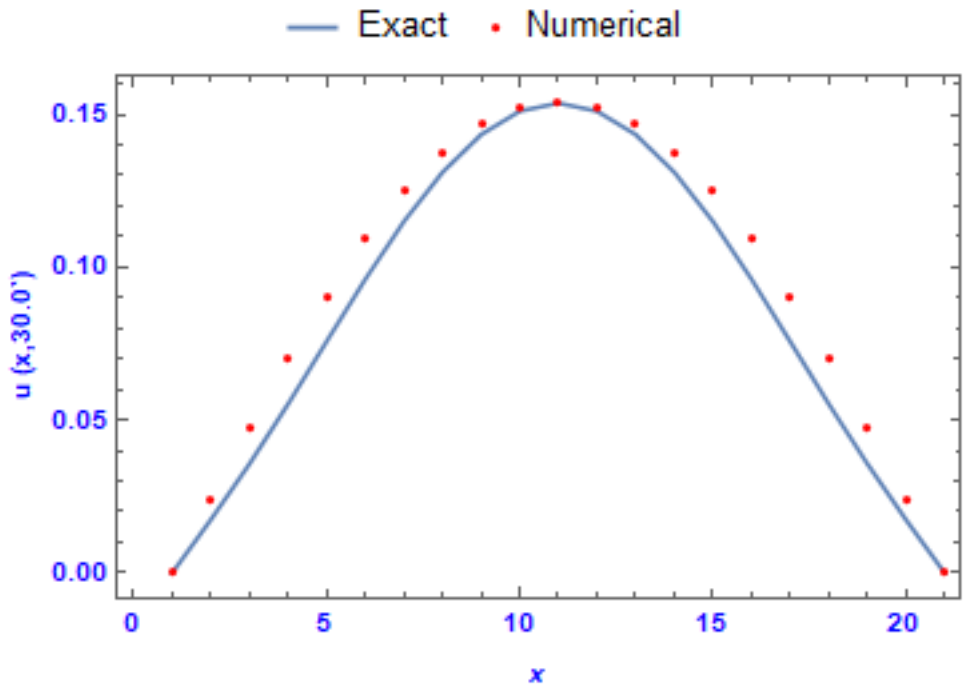


Figure 3.5.10 The exact and numerical results for time *t*=30.0 with *k*=0.01.

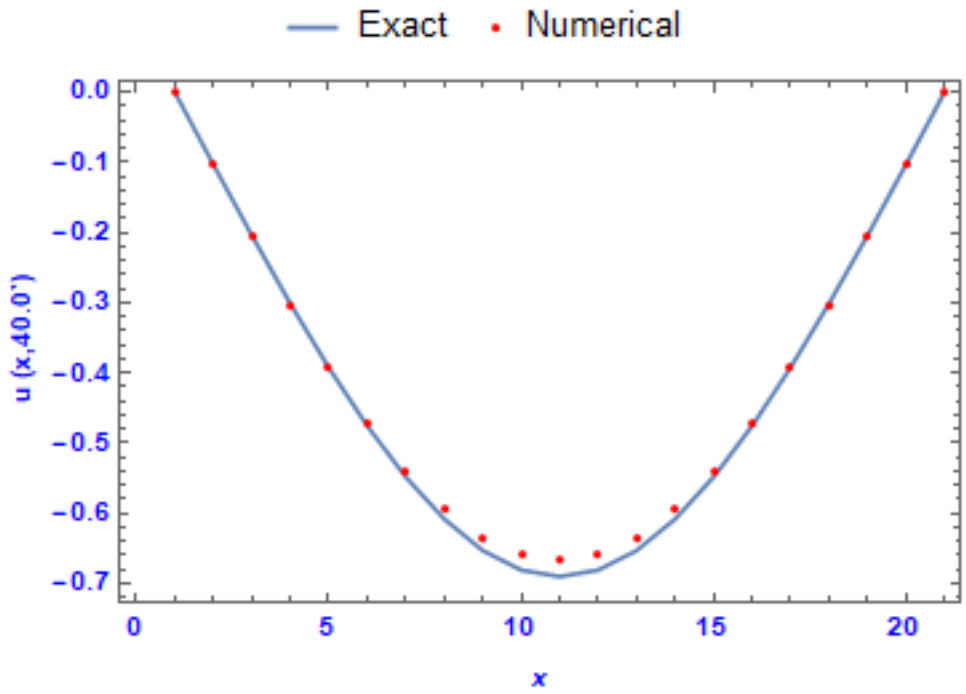


Figure 3.5.11 The exact and numerical results for time *t*=40.0 with *k*=0.01.

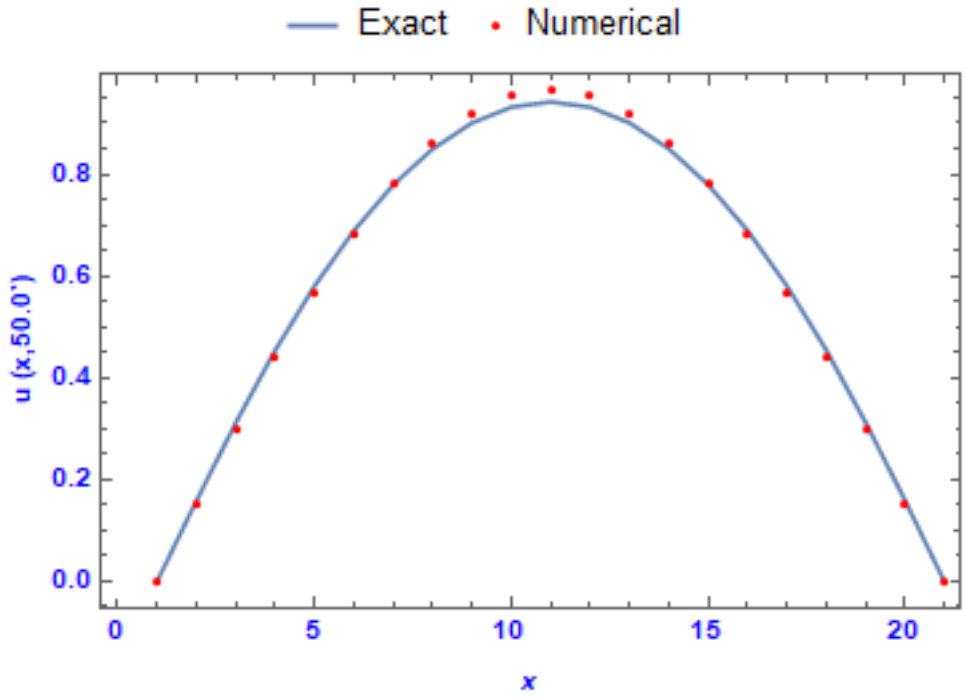


Figure 3.5.12 The exact and numerical results for time *t*=50.0 with *k*=0.01.

Tables 3.5.2–3.5.6 show that the smaller $\Delta t = k$ **(**than the value of *h***),** the better the accuracy. The numerical approximations are still acceptable within the large time.

Figures 3.5.13–3.5.16 present the 3D numerical solutions of the dispersive equation for various times with the same discretisations (***h***).

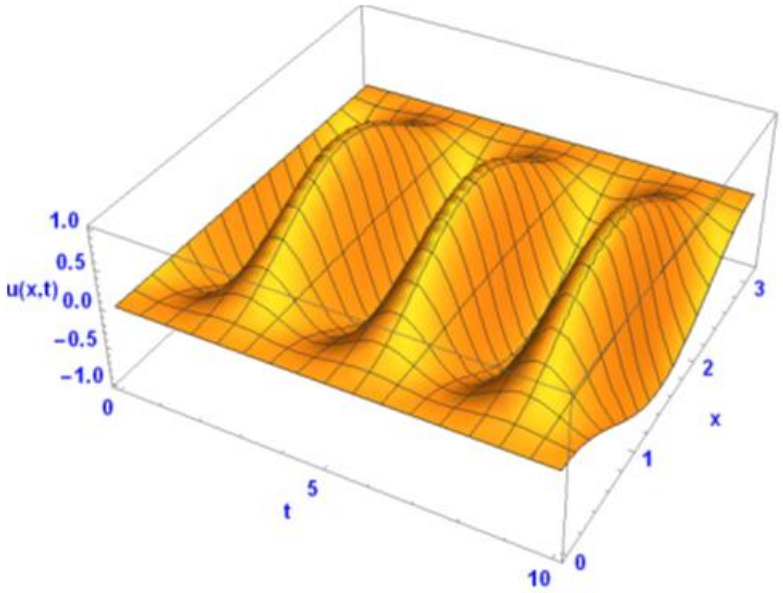


Figure 3.5.13 3D representation of the behaviour of the numerical solutions of the dissipative wave equation from time *t*=0.0 to *t*=10.0

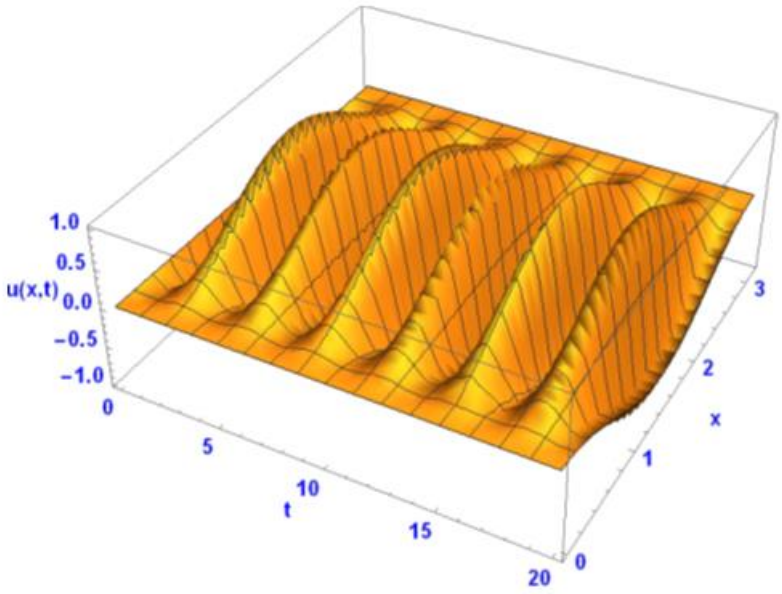


Figure 3.5.14 3D representation of the behaviour of the numerical solutions of the dissipative wave equation from time *t*=0.0 to *t*=20.0

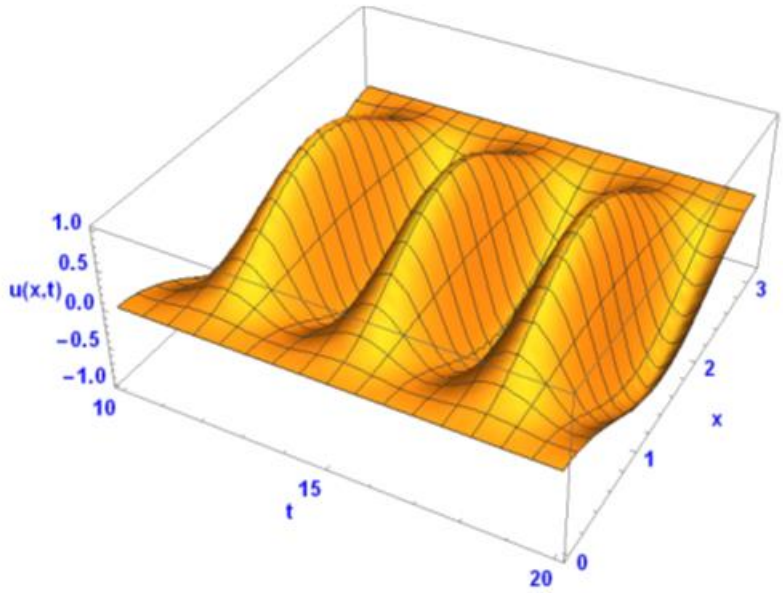


Figure 3.5.15 3D representation of the behaviour of the numerical solutions of the dissipative wave equation from time *t*=10.0 to *t*=20.0

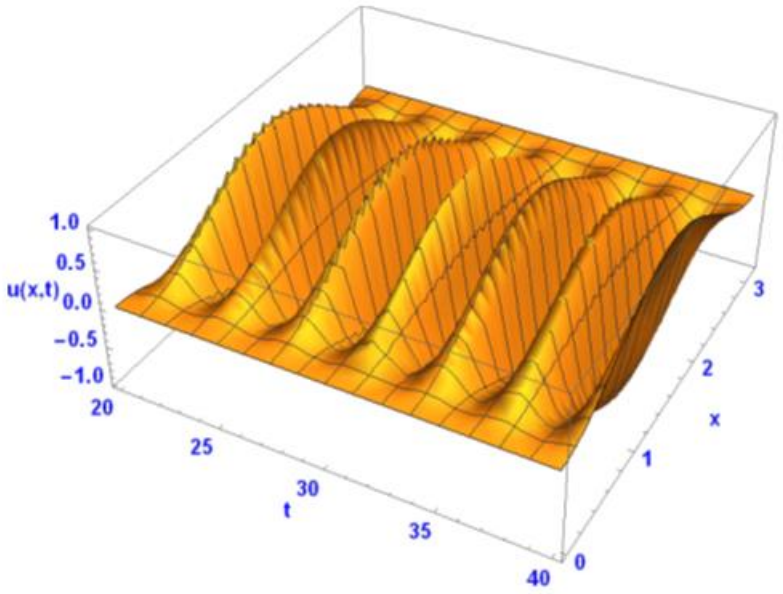


Figure 3.5.16 3D representation of the behaviour of the numerical solutions of the dissipative wave equation from time *t*=20.0 to *t*=40.0

3.6 Conclusion

In this chapter, a numerical solution for the nonlinear dissipative wave equation was proposed, utilising a collocation strategy with the CBS. To illustrate the method and demonstrate its convergence and applicability computationally, I applied the Von Neumann stability method. The stability analysis investigation showed that the method is conditionally stable. The performance and accuracy of the present method were shown by calculating and comparing the L_∞ error norms with earlier work. The obtained invariants are considered acceptable in comparison with some earlier studies. The numerical results produced by the present method are quite satisfactory and show good agreement with the exact solutions. The computed results demonstrate the advantages of this method. As seen in Tables 3.5.2 and 3.5.3, the results are better than those in [92]. The estimated mathematical arrangements achieve great precision with the specific arrangements, particularly when Δt is more modest than the estimate of h.

CHAPTER 4

Quartic Non-polynomial Splines for Solving Third-Order Dispersive Partial Differential Equations

Chapter 4: Quartic Non-polynomial Splines for Solving Third-Order Dispersive Partial Differential Equations

4.1 Introduction

Third-order singularly perturbed boundary value problems occur frequently in many areas of applied sciences such as solid mechanics, quantum mechanics, chemical reactor theory, Newtonian fluid mechanics, optimal control, convection diffusion processes, hydrodynamics and aerodynamics. These problems have various important applications in fluid dynamics. The field of nonlinear dispersive waves has undergone enormous development since the work of Stokes, Boussinesq, Korteweg and de Vries, all of whom studied water wave problems in the nineteenth century. In the 1960s, researchers developed effective asymptotic methods for deriving nonlinear wave equations such as the KdV equation that govern a broad class of physical phenomena [93].

Some approaches to solving NPDEs have been described in the recent literature; the most prominent of which are non-polynomial spline methods. The non-polynomial spline used for solving NPDEs has been employed by many researchers. The most well-known and well-focused results are those presented by Ramadan et al. (2005), who used a numerical method for approximation of the Burgers equation [94]. Shock waves and blowup arising in third-order nonlinear dispersive equations were studied in 2008 by Galaktionov and Pohozaev [95]. In [96, 97], criteria for deriving stability conditions for the difference method were considered for the numerical solution to a third-order linear dispersive equation. Tirmizi et al. (2008) used quartic non-polynomial spline functions to develop a class of numerical methods for solving self-adjoint singularly perturbed problems [98]. In 2011, Taiwo and Ogunlaran developed a numerical technique for solving linear fourth-order boundary value problems, which were initially reduced to a system of second-order boundary value problems [99]. In research by Lin (2014), a numerical method based on splines in tension was developed for solving RLW equations. The method was tested by using single solitary waves, the interaction of two solitary waves, and solitary waves with Maxwellian

initial condition [100]. In the same year, Mustafa and Ilhame discussed the method of lines as applied to the boundary value problem for a third-order PDE [101]. In 2017, El-Danaf et al. considered the GRLW equation. They studied the interaction of solitons when no analytic solution is known during the interaction. The Maxwellian initial condition for the GRLW equation was used [102]. A year later, Li et al. solved the time-fractional NLS equation [103]. In 2018, Sultana et al. presented a new three-level implicit method, which was developed to solve linear and nonlinear third-order dispersive PDEs [104]. In 2019, Shahna demonstrated how to solve fourth-order boundary value problems whose highest-order derivative is multiplied by a small perturbation parameter [105]. In this chapter, a novel approach, based on the use of non-polynomial splines to solve a third-order dispersive PDE is proposed**.** The third-order dispersive PDE used is [106]

$$\frac{\partial \eta}{\partial t} + \frac{\partial^3 \eta}{\partial x^3} = g(x,t), \qquad a \le x \le b, \quad t > 0, \tag{4.1.1}$$

where $g(x,t)$ is a source term. The boundary conditions associated with Eq. (4.1.1) are assumed to be of the form

$$\eta(a,t) = \beta_1(t), \quad \eta(b,t) = \beta_2(t), \ \eta_{xx}(b,t) = \beta_3(t) \quad t > 0\ , \tag{4.1.2}$$

and the initial condition is

$$\eta(x,0) = f(x), \qquad a \le x \le b. \tag{4.1.3}$$

The spline functions proposed, as defined in [107], have the form

$$T_4 = span\{1, x, x^2, sin\,\omega\, x, cos\,\omega\, x\},$$

where ω is the frequency of the trigonometric part of the spline functions, which are used to increase the accuracy of the method.

4.2 Analysis of the Method

The first step in the non-polynomial spline method is to create a grid with two mesh constants h and k. The grid points for this situation are (x_i, t_j) where $x_i = a + ih$, $i = 0,1,\ldots, N+1$ and $t_j = jk$, $j = 0,1,\ldots$.

Let $Z_i^{\,j}$ be an approximation to $\eta(x_i, t_j)$, obtained by the segment $P_i(x, t_j)$ of the mixed spline function passing through the points $(x_i, Z_i^{\,j})$ and $(x_{i+1}, Z_{i+1}^{\,j})$. Each segment has the form

$$P_i(x,t_j) = a_i(t_j)\ \cos\omega(x-x_i) + b_i(t_j)\ \sin\omega(x-x_i) + c_i(t_j)\ (x-x_i)^2 + d_i(t_j)(x-x_i) + e_i(t_j). \tag{4.2.1}$$

for each $i = 0,1,\ldots, N$. To obtain expressions for the coefficients of Eq. (4.2.1) in terms of $Z_i^{\,j}$, $Z_{i+1}^{\,j}$, $M_i^{\,j}$, $S_i^{\,j}$, and $S_i^{\,j+1}$, we first define

$$P_i(x_i,t_j) = Z_i^j,\quad P_i(x_{i+1},t_j) = Z_{i+1}^j,\quad P_i^{(1)}(x_i) = M_i^j,\quad P_i^{(3)}(x_i,t_j) = S_i^j, and\quad P_i^{(3)}(x_{i+1},t_j) = S_{i+1}^j. \tag{4.2.2}$$

Using Eqs. (4.2.1) and (4.2.2), we obtain:

$$\begin{aligned} a_i + e_i &= Z_i^j, \\ a_i\cos\theta + b_i\sin\theta + h^2c_i + hd_i + e_i &= Z_{i+1}^j, \\ b_i\omega + d_i &= M_i^j, \\ -\omega^3 b_i &= S_i^j, \\ a_i\omega^3\sin\theta - b_i\omega^3\cos\theta &= S_{i+1}^j, \end{aligned} \tag{4.2.3}$$

where $a_i \equiv a_i(t_j)$, $b_i \equiv b_i(t_j)$, $c_i \equiv c_i(t_j)$, $d_i \equiv d_i(t_j)$, $e_i \equiv e_i(t_j)$, and $\theta = \omega h$.

By solving the last five equations in (4.2.3), we obtain the following; from the fourth equation in (4.3.2):

$$b_i = -\frac{S_i^j}{w^3} = -h^3 \frac{S_i^j}{\theta^3}.$$

Then, from the last equation in (4.3.2) we obtain

$$a_i \omega^3 \sin\theta - b_i \omega^3 \cos\theta = S_{i+1}^j$$

$$a_i = \frac{S_{i+1}^j + b_i \omega^3 \cos\theta}{\omega^3 \sin\theta}$$

$$= \frac{S_{i+1}^j - S_i^j \cos\theta}{(\frac{\theta}{h})^3 \sin\theta}$$

$$= h^3 \frac{S_{i+1}^j - S_i^j \cos\theta}{\theta^3 \sin\theta}.$$

By the same manner we obtain the other unknown coefficients:

$$a_i \cos\theta + b_i \sin\theta + h^2 c_i + h d_i + e_i = Z_{i+1}^j$$

$$c_i = \frac{Z_{i+1}^j}{h^2} - \frac{a_i \cos\theta}{h^2} - \frac{b_i \sin\theta}{h^2} - \frac{h d_i}{h^2} - \frac{e_i}{h^2}$$

$$= \frac{Z_{i+1}^j}{h^2} - \frac{a_i}{h^2} + \frac{a_i}{h^2} - \frac{e_i}{h^2} - \frac{a_i \cos\theta}{h^2} - \frac{b_i \cos\theta}{h^2 \sin\theta} + \frac{b_i \cos\theta}{h^2 \sin\theta} - \frac{b_i \sin^2\theta}{h^2 \sin\theta} - \frac{\theta b_i}{h^2} + \frac{\theta b_i}{h^2} - \frac{h d_i}{h^2}$$

$$= \frac{Z_{i+1}^j}{h^2} - \frac{(a_i + e_i)}{h^2} + \frac{a_i}{h^2} - \frac{b_i \cos\theta}{h^2 \sin\theta} - \frac{a_i \cos\theta}{h^2} + \frac{b_i \cos\theta}{h^2 \sin\theta} - \frac{b_i (1 - \cos^2\theta)}{h^2 \sin\theta} - \frac{h d_i}{h^2} - \frac{\theta b_i}{h^2} + \frac{\theta b_i}{h^2}$$

$$= \frac{Z_{i+1}^j}{h^2} - \frac{Z_i^j}{h^2} + \frac{a_i h \theta^3 \sin\theta}{h^2 h \theta^3 \sin\theta} - \frac{b_i \cos\theta h \theta^3}{h^2 \sin\theta h \theta^3} - \frac{a_i \cos\theta h \theta^3 \sin\theta}{h^2 h \theta^3 \sin\theta} + \frac{b_i \cos\theta h \theta^3}{h^2 \sin\theta h \theta^3}$$

$$- \frac{b_i h \theta^3}{h^2 \sin\theta h \theta^3} + \frac{b_i \cos^2\theta h \theta^3}{h^2 \sin\theta h \theta^3} - \frac{\theta b_i}{h^2} - \frac{h d_i}{h^2} + \frac{\theta b_i h \theta^2}{h^2 h \theta^2}$$

$$= \frac{Z_{i+1}^j}{h^2} - \frac{Z_i^j}{h^2} + \frac{h a_i w^3 \sin\theta}{\theta^3 \sin\theta} - \frac{h b_i w^3 \cos\theta}{\theta^3 \sin\theta} - \frac{h a_i w^3 \cos\theta \sin\theta}{\theta^3 \sin\theta} + \frac{h b_i w^3 \cos^2\theta}{\theta^3 \sin\theta}$$

$$-\frac{h\,b_i w^3}{\theta^3 \sin\theta}+\frac{h\,b_i w^3\cos\theta}{\theta^3\sin\theta}-\frac{b_i w}{h}-\frac{d_i}{h}+\frac{h\,b_i w^3}{\theta^2}$$

$$=\frac{Z_{i+1}^j}{h^2}-\frac{Z_i^j}{h^2}+\frac{h(a_i w^3\sin\theta-b_i w^3\cos\theta)}{\theta^3\sin\theta}-\frac{h\ \cos\theta\,(a_i w^3\sin\theta-b_i w^3\cos\theta)}{\theta^3\sin\theta}$$

$$+\frac{h(-b_i w^3)}{\theta^3\sin\theta}-\frac{h(-b_i w^3)\cos\theta}{\theta^3\sin\theta}-\frac{(b_i w+d_i)}{h}-\frac{h\,(-b_i w^3)}{\theta^2}$$

$$=\frac{Z_{i+1}^j-Z_i^j}{h^2}+\frac{h\,S_{i+1}^j}{\theta^3\sin\theta}-\frac{h\ \cos\theta\,S_{i+1}^j}{\theta^3\sin\theta}+\frac{h\,S_i^j}{\theta^3\sin\theta}-\frac{h\cos\theta S_i^j}{\theta^3\sin\theta}-\frac{M_i^j}{h}-\frac{h\,S_i^j}{\theta^2}$$

$$=\frac{Z_{i+1}^j-Z_i^j}{h^2}+\frac{h\,S_{i+1}^j(1-cos\theta)+h\,S_i^j(1-cos\theta)}{\theta^3\sin\theta}-\frac{M_i^j}{h}-\frac{h\,S_i^j}{\theta^2}$$

$$=\frac{Z_{i+1}^j-Z_i^j}{h^2}+\frac{h\,(1-cos\theta)(S_{i+1}^j+S_i^j)}{\theta^3\sin\theta}-\frac{M_i^j}{h}-\frac{h\,S_i^j}{\theta^2}$$

$$b_i\omega+d_i=M_i^j$$

$$d_i=M_i^j-b_i\omega=M_i^j-(-h^3\frac{S_i^j}{\theta^3})\frac{\theta}{h}=M_i^j+h^2\frac{S_i^j}{\theta^2}$$

$$a_i+e_i=Z_i^j$$

$$e_i=Z_i^j-h^3\frac{S_{i+1}^j-S_i^j\cos\theta}{\theta^3\sin\theta}$$

$$a_i=h^3\frac{S_{i+1}^j-S_i^j\cos\theta}{\theta^3\sin\theta},\ b_i=-h^3\frac{S_i^j}{\theta^3},$$

$$c_i=\frac{Z_{i+1}^j-Z_i^j}{h^2}+\frac{h\left(S_{i+1}^j+S_i^j\right)(1-cos\,\theta)}{\theta^3\sin\theta}-\frac{M_i^j}{h}-\frac{hS_i^j}{\theta^2} \qquad (4.2.4)$$

$$d_i=M_i^j+h^2\frac{S_i^j}{\theta^2},\ e_i=Z_i^j-h^3\frac{S_{i+1}^j-S_i^j\cos\theta}{\theta^3\sin\theta}.$$

Using the continuity condition of the first and second derivatives at $x = x_i$ —that is, $P_i^{(n)}(x_i,t_j)=P_{i-1}^{(n)}(x_i,t_j)$ where *n*=1 and 2—we obtain the following relations:

$$\omega b_i+d_i=-a_{i-1}\omega\sin\theta+b_{i-1}\omega\cos\theta+2hc_{i-1}+d_{i-1},$$

$$-\omega^2 a_i + 2c_i = -a_{i-1}\omega^2 \cos\theta - b_{i-1}\omega^2 \sin\theta + 2c_{i-1}. \qquad (4.2.5)$$

Using expressions from (4.2.4), the equations in (4.2.5) become

$$M_i^j + M_{i-1}^j = \frac{2}{h}\left(Z_i^j - Z_{i-1}^j\right) - \frac{h^2}{\theta^2}\left(S_{i-1}^j + S_i^j\right) + \frac{2h^2}{\theta^3 \sin\theta}\left(S_{i-1}^j + S_i^j\right)(1 - \cos\theta) \qquad (4.2.6)$$

$$M_i^j - M_{i-1}^j = \frac{h^2 S_i^j \cos\theta}{\theta \sin\theta} - \frac{h^2}{2\theta \sin\theta}\left(S_{i-1}^j + S_{i+1}^j\right) + \frac{1}{h}\left(Z_{i-1}^j - 2Z_i^j + Z_{i+1}^j\right) + \frac{h^2}{\theta^3 \sin\theta}(1 - \cos\theta)\left(S_{i+1}^j - S_{i-1}^j\right) + \frac{h^2}{\theta^2}\left(S_{i-1}^j - S_i^j\right). \qquad (4.2.7)$$

Adding Eqs. (4.2.6) and (4.2.7), gives

$$M_i^j = \frac{1}{2h}\left(Z_{i+1}^j - Z_{i-1}^j\right) - \frac{h^2}{\theta^2}S_i^j + \frac{h^2}{2\theta^3 \sin\theta}(1 - \cos\theta)\left(S_{i-1}^j + 2S_i^j + S_{i+1}^j\right) + \frac{h^2 S_i^j \cos\theta}{2\theta \sin\theta} - \frac{h^2}{4\theta \sin\theta}\left(S_{i-1}^j + S_{i+1}^j\right). \qquad (4.2.8)$$

Similarly,

$$M_{i-1}^j = \frac{1}{2h}\left(Z_i^j - Z_{i-2}^j\right) - \frac{h^2}{\theta^2}S_{i-1}^j + \frac{h^2}{2\theta^3 \sin\theta}(1 - \cos\theta)\left(S_{i-2}^j + 2S_{i-1}^j + S_i^j\right) + \frac{h^2 S_{i-1}^j \cos\theta}{2\theta \sin\theta} - \frac{h^2}{4\theta \sin\theta}\left(S_{i-2}^j + S_i^j\right). \qquad (4.2.9)$$

Using M_i^j and M_{i-1}^j in Eq. (4.2.7) gives the following relation:

$$\frac{1}{2h}\left(Z_{i+1}^j - Z_{i-1}^j + Z_i^j - Z_{i-2}^j\right) - \frac{h^2}{\theta^2}\left(S_i^j + S_{i-1}^j\right) + \frac{h^2(1 - \cos\theta)}{2\theta^3 \sin\theta}\left(S_{i-2}^j + 3S_{i-1}^j + 3S_i^j + S_{i+1}^j\right) + \frac{h^2\left(S_i^j + S_{i-1}^j\right)\cos\theta}{2\theta \sin\theta} - \frac{h^2}{4\theta \sin\theta}\left(S_{i-2}^j + S_{i-1}^j + S_i^j + S_{i+1}^j\right) = \frac{2}{h}\left(Z_i^j - Z_{i-1}^j\right) - \frac{h^2}{\theta^2}\left(S_{i-1}^j + S_i^j\right) + \frac{2h^2}{\theta^3 \sin\theta}\left(S_{i-1}^j + S_i^j\right)(1 - \cos\theta),$$

or

$$-Z_{i-2}^{j} - h^3 S_{i-2}^{j}\left(\frac{cos\,\theta - 1}{\theta^3\, sin\,\theta} + \frac{1}{2\theta\, sin\,\theta}\right) + 3Z_{i-1}^{j}$$
$$- h^3 S_{i-1}^{j}\left(\frac{1 - cos\,\theta}{\theta^3\, sin\,\theta} - \frac{cos\,\theta}{\theta\, sin\,\theta} + \frac{1}{2\theta\, sin\,\theta}\right)$$

$$-3Z_i^j - h^3 S_i^j\left(\frac{1 - cos\,\theta}{\theta^3\, sin\,\theta} - \frac{cos\,\theta}{\theta\, sin\,\theta} + \frac{1}{2\theta\, sin\,\theta}\right) + Z_{i+1}^j - h^3 S_{i+1}^j\left(\frac{cos\,\theta - 1}{\theta^3\, sin\,\theta} + \frac{1}{2\theta\, sin\,\theta}\right)$$

$$= 0,\ i = 2, \dots, N.$$

This equation can be rewritten in the following simple form:

$$-Z_{i-2}^j + 3Z_{i-1}^j - 3Z_i^j + Z_{i+1}^j = \alpha S_{i-2}^j + \beta S_{i-1}^j + \beta S_i^j + \alpha S_{i+1}^j,$$

$$i = 2, \dots, N, \qquad (4.2.10)$$

where

$$\alpha = h^3\left(\frac{cos\,\theta - 1}{\theta^3\, sin\,\theta} + \frac{1}{2\theta\, sin\,\theta}\right),\ and\ \beta = h^3\left(\frac{1 - cos\,\theta}{\theta^3\, sin\,\theta} - \frac{cos\,\theta}{\theta\, sin\,\theta} + \frac{1}{2\theta\, sin\,\theta}\right).$$

Remark:

As $\omega \to 0$, that is $\theta \to 0$, $(\alpha, \beta) \to \left(\frac{h^3}{24}, \frac{11h^3}{24}\right)$, and system (4.2.10) reduces to the ordinary quartic spline:

$$-Z_{i-2}^j + 3Z_{i-1}^j - 3Z_i^j + Z_{i+1}^j = \frac{h^3}{24}\left(S_{i-2}^j + 11S_{i-1}^j + 11S_i^j + S_{i+1}^j\right).$$

Using Eq. (4.1.1), we can write S_{i-2}^j,S_{i-1}^j, S_i^j and S_{i+1}^j, in the form

$$S_{i-2}^j = \frac{\partial^3 Z_{i-2}^j}{\partial x^3} = \left(g_{i-2}^j - \frac{\partial Z_{i-2}^j}{\partial t}\right),$$

$$S_{i-1}^j = \frac{\partial^3 Z_{i-1}^j}{\partial x^3} = \left(g_{i-1}^j - \frac{\partial Z_{i-1}^j}{\partial t}\right),$$

$$S_i^j = \frac{\partial^3 Z_i^j}{\partial x^3} = \left(g_i^j - \frac{\partial Z_i^j}{\partial t}\right),$$

$$S_{i+1}^j = \frac{\partial^3 Z_{i+1}^j}{\partial x^3} = \left(g_{i+1}^j - \frac{\partial Z_{i+1}^j}{\partial t}\right).$$

These equations can be discretised in the form

$$S_{i-2}^{j} \approx \left(g_{i-2}^{j} - \left(\frac{Z_{i-2}^{j} - Z_{i-2}^{j-1}}{k} \right) \right),$$

$$S_{i-1}^{j} \approx \left(g_{i-1}^{j} - \left(\frac{Z_{i-1}^{j} - Z_{i-1}^{j-1}}{k} \right) \right),$$

$$S_{i}^{j} \approx \left(g_{i}^{j} - \left(\frac{Z_{i}^{j} - Z_{i}^{j-1}}{k} \right) \right),$$

$$S_{i+1}^{j} \approx \left(g_{i+1}^{j} - \left(\frac{Z_{i+1}^{j} - Z_{i+1}^{j-1}}{k} \right) \right). \qquad (4.2.11)$$

The use of equations from (4.2.10) in Eq. (4.2.11) gives the following system:

$$-Z_{i-2}^{j} + 3Z_{i-1}^{j} - 3Z_{i}^{j} + Z_{i+1}^{j} = \alpha \left(g_{i-2}^{j} - \frac{Z_{i-2}^{j} - Z_{i-2}^{j-1}}{k} \right) + \beta \left(g_{i-1}^{j} - \frac{Z_{i-1}^{j} - Z_{i-1}^{j-1}}{k} \right) + \beta \left(g_{i}^{j} - \frac{Z_{i}^{j} - Z_{i}^{j-1}}{k} \right) + \alpha \left(g_{i+1}^{j} - \frac{Z_{i+1}^{j} - Z_{i+1}^{j-1}}{k} \right),$$

or

$$A_i Z_{i-2}^{j} + B_i Z_{i-1}^{j} + C_i Z_{i}^{j} + D_i Z_{i+1}^{j} = \alpha Z_{i-2}^{j-1} + \beta Z_{i-1}^{j-1} + \beta Z_{i}^{j-1} + \alpha Z_{i+1}^{j-1} + \delta_i^{j}, \quad i = 2, \ldots, N, \qquad (4.2.12)$$

where

$$A_i = -k + \alpha \text{ , } B_i = 3k + \beta, C_i = -3k + \beta, A_i = -k + \alpha$$

$$\text{and } \delta_i^{j} = k\left(\alpha g_{i-2}^{j} + \beta g_{i-1}^{j} + \beta g_{i}^{j} + \alpha g_{i+1}^{j}\right).$$

System (4.2.12) consists of $N-1$ equations in unknowns Z_i^j, $i = 0, \ldots, N+1$. To obtain a solution to this system, we need three additional equations. Two equations are obtained from the first two parts in (4.1.2):

$$Z_0^j = \eta(a,t) = \beta_1(t), \quad Z_{N+1}^j = \eta(b,t) = \beta_2(t). \qquad (4.2.13)$$

The third equation can be obtained from the third part of (4.1.2); that is $\frac{\partial^2 Z_{N+1}^j}{\partial x^2} = \eta_{xx}(b,t) = \beta_3(t)$, which can be discretised as

$$-10Z_{N-4}^j + 61Z_{N-3}^j - 156Z_{N-2}^j + 214Z_{N-1}^j - 154Z_N^j + 45Z_{N+1}^j \approx$$
$$12h^2 \frac{\partial^2 Z_{N+1}^j}{\partial x^2} = 12h^2\beta_3, \quad j \geq 0. \tag{4.2.14}$$

Writing the equations in (4.2.12)–(4.2.14) in matrix form gives

$$Q\ Z^j = Q^*\ Z^{j-1} + r^j, \tag{4.2.15}$$

where

$$Z^j = \left(Z_0^j,\ \ Z_1^j,\ \ Z_2^j,\ \ \ldots \qquad Z_{N-1}^j,\ \ Z_N^j,\ \ Z_{N+1}^j\right)^t,$$

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & \ldots\ldots & 0 \\ A_2 & B_2 & C_2 & D_2 & 0 & 0 & 0 & \ldots\ldots & 0 \\ 0 & A_3 & B_3 & C_3 & D_3 & 0 & 0 & \ldots\ldots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & 0 & 0 & A_{N-1} & B_{N-1} & C_{N-1} & D_{N-1} & 0 \\ 0 & \ldots\ldots & 0 & 0 & 0 & A_N & B_N & C_N & D_N \\ 0 & \ldots\ldots 0 & 0 & -10 & 61 & -156 & 214 & -154 & 45 \\ 0 & \ldots\ldots & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$Q^{*}=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ldots\ldots & 0 \\ \alpha & \beta & \beta & \alpha & 0 & 0 & 0 & \ldots\ldots & 0 \\ 0 & \alpha & \beta & \beta & \alpha & 0 & 0 & \ldots\ldots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & 0 & 0 & \alpha & \beta & \beta & \alpha & 0 \\ 0 & \ldots\ldots & 0 & 0 & 0 & \alpha & \beta & \beta & \alpha \\ 0 & \ldots\ldots & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ldots\ldots & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

and $r^{j}=\left(\beta_1(t_j),\delta_2^j,\ldots,\delta_N^j,12h^2\beta_3(t_j),\beta_2(t_j)\right)^{t}$. The initial condition $\eta(x,t_0)=f(x)$, for each $a\le x\le b$, implies that $Z_i^0=f(x_i)$, for each $i=0,1,\ldots,N+1$. These values can be used in Eq. (4.2.15) to find the value of Z_i^1, for each $i=0,1,\ldots,N+1$. If the procedure is reapplied once all the approximations Z_i^1 are known, the values of $Z_i^2, Z_i^3,\ldots$ can be obtained in a similar manner.

4.3 Error Analysis

Using Eq. (4.2.12), we obtain the truncation error:

$$T_i^j=A_i\eta_{i-2}^j+B_i\eta_{i-1}^j+C_i\eta_i^j+D_i\eta_{i+1}^j-\alpha\eta_{i-2}^{j-1}-\beta\eta_{i-1}^{j-1}-\beta\eta_i^{j-1}-\alpha\eta_{i+1}^{j-1}-\delta_i^j, \tag{4.3.1}$$

where

$$A_i=-k+\alpha,\ B_i=3k+\beta, C_i=-3k+\beta,\ A_i=-k+\alpha \text{ and}$$

$$\delta_i^j=k\left(\alpha g_{i-2}^j+\beta g_{i-1}^j+\beta g_i^j+\alpha g_{i+1}^j\right)$$

Expanding Eq. (4.3.1) in the Taylor series, in terms of $\eta(x_i,t_j)$ and its derivatives, we obtain the following expression:

$$T_i^j = A_i\left(1-\frac{2h}{1!}D_x+\frac{(2h)^2}{2!}D_x^2-\cdots\right)\eta_i^j + B_i\left(1-hD_x+\frac{h^2}{2!}D_x^2-\cdots\right)\eta_i^j +$$

$$C_i\eta_i^j + D_i\left(1+hD_x+\frac{h^2}{2!}D_x^2+\frac{h^3}{3!}D_x^3+\cdots\right)\eta_i^j -$$

$$\alpha\left(1+(-kD_t-2hD_x)+\frac{1}{2!}(-kD_t-2hD_x)^2+\cdots\right)\eta_i^j -$$

$$\beta\left(1+(-kD_t-hD_x)+\frac{1}{2!}(-kD_t-hD_x)^2+\cdots\right)\eta_i^j -$$

$$\beta\left(1-kD_t+\frac{k^2}{2!}D_t^2-\frac{k^3}{3!}D_t^3+\cdots\right)\eta_i^j -$$

$$\alpha\left(1+(-kD_t+hD_x)+\frac{1}{2!}(-kD_t+hD_x)^2+\cdots\right)\eta_i^j -$$

$$k\alpha\left(1-2hD_x+\frac{(2h)^2}{2!}D_x^2-\cdots\right)g_i^j - k\beta\left(1-hD_x+\frac{h^2}{2!}D_x^2-\cdots\right)g_i^j -$$

$$k\beta g_i^j - k\alpha\left(1+hD_x+\tfrac{h^2}{2!}D_x^2+\cdots\right)g_i^j,$$

where $g_i^j = D_t\eta_i^j + D_x^3\eta_i^j$. After simple calculations, we obtain:

$$T_i^j = \alpha\left(1-\frac{2h}{1!}D_x+\frac{(2h)^2}{2!}D_x^2-\cdots\right)\eta_i^j - k\left(1-\frac{2h}{1!}D_x+\frac{(2h)^2}{2!}D_x^2-\cdots\right)\eta_i^j$$

$$+\beta\left(1-hD_x+\frac{h^2}{2!}D_x^2-\cdots\right)\eta_i^j + 3k\left(1-hD_x+\frac{h^2}{2!}D_x^2-\cdots\right)\eta_i^j + \beta\eta_i^j - 3k\eta_i^j$$

$$+\alpha\left(1+hD_x+\frac{h^2}{2!}D_x^2+\frac{h^3}{3!}D_x^3+\cdots\right)\eta_i^j + k\left(1+hD_x+\frac{h^2}{2!}D_x^2+\frac{h^3}{3!}D_x^3+\cdots\right)\eta_i^j$$

$$-\alpha\left(1+(-kD_t-2hD_x)+\frac{1}{2!}(-kD_t-2hD_x)^2+\cdots\right)\eta_i^j$$

$$-\beta\left(1+(-kD_t-hD_x)+\frac{1}{2!}(-kD_t-hD_x)^2+\cdots\right)\eta_i^j$$

$$-\beta\left(1-kD_t+\frac{k^2}{2!}D_t^2-\frac{k^3}{3!}D_t^3+\cdots\right)\eta_i^j$$

$$-\alpha\left(1+(-kD_t+hD_x)+\frac{1}{2!}(-kD_t+hD_x)^2+\cdots\right)\eta_i^j$$

$$-k\alpha\left(1-2hD_x+\frac{(2h)^2}{2!}D_x^2-\cdots\right)g_i^j - k\beta\left(1-hD_x+\frac{h^2}{2!}D_x^2-\cdots\right)g_i^j$$

$$-k\beta g_i^j - k\alpha\left(1+hD_x+\tfrac{h^2}{2!}D_x^2+\cdots\right)g_i^j.$$

The local truncation error can be simplified to

$$T_i^j = -k\left(-\frac{2h}{1!}D_x + \frac{(2h)^2}{2!}D_x^2 - \cdots\right)\eta_i^j + 3k\left(-hD_x + \frac{h^2}{2!}D_x^2 - \cdots\right)\eta_i^j +$$

$$k\left(hD_x + \frac{h^2}{2!}D_x^2 + \frac{h^3}{3!}D_x^3 + \cdots\right)\eta_i^j - 2(\beta+\alpha)\left(-kD_t + \frac{k^2}{2!}D_t^2 - \frac{k^3}{3!}D_t^3 + \cdots\right)\eta_i^j +$$

$$(\beta+\alpha)h\left(\frac{-1}{2!}\binom{2}{1}kD_t + \frac{1}{3!}\binom{3}{1}k^2D_t^2 - \cdots\right)D_x\eta_i^j +$$

$$(\beta+5\alpha)h^2\left(\frac{1}{3!}\binom{3}{2}kD_t - \frac{1}{4!}\binom{4}{2}k^2D_t^2 + \cdots\right)D_x^2\eta_i^j +$$

$$(\beta+7\alpha)h^3\left(\frac{-1}{4!}\binom{4}{3}kD_t + \frac{1}{5!}\binom{5}{3}k^2D_t^2 - \cdots\right)D_x^3\eta_i^j + \cdots -$$

$$k\alpha\left(-2hD_x + \frac{(2h)^2}{2!}D_x^2 - \cdots\right)(D_t + D_x^3)\eta_i^j - k\beta\left(-hD_x + \frac{h^2}{2!}D_x^2 - \cdots\right)(D_t + D_x^3)\eta_i^j$$

$$-$$

$$k\alpha\left(hD_x + \tfrac{h^2}{2!}D_x^2 + \cdots\right)(D_t + D_x^3)\eta_i^j - 2k(\beta+\alpha)(D_t + D_x^3)\eta_i^j.$$

$$T_i^j = k\big(h^3 - 2(\beta+\alpha)\big)D_x^3\eta_i^j + kh\left(\frac{-h^3}{2} + (\beta+\alpha)\right)D_x^4\eta_i^j$$

$$+kh^2\left(\frac{h^3}{4} - \frac{1}{2}(\beta+5\alpha)\right)D_x^5\eta_i^j + kh^3\left(\frac{-h^3}{12} + \frac{1}{6}(\beta+7\alpha)\right)D_x^6\eta_i^j$$

$$+kh^4\left(\frac{h^3}{40} - \frac{1}{24}(\beta+17\alpha)\right)D_x^7\eta_i^j + \cdots +$$

$$2(\beta+\alpha)\left(-\frac{k^2}{2!}D_t^2 + \frac{k^3}{3!}D_t^3 - \cdots\right)\eta_i^j$$

$$+\ (\beta+\alpha)h\left(\frac{1}{3!}\binom{3}{1}k^2D_t^2 - \frac{1}{4!}\binom{4}{1}k^3D_t^3 + \cdots\right)D_x\eta_i^j$$

$$+(\beta+5\alpha)h^2\left(-\frac{1}{4!}\binom{4}{2}k^2D_t^2 + \frac{1}{5!}\binom{5}{2}k^3D3_t - \cdots\right)D_x^2\eta_i^j$$

$$+(\beta+7\alpha)h^3\left(\tfrac{1}{5!}\binom{5}{3}k^2D_t^2 - \tfrac{1}{6!}\binom{6}{3}k^3D_t^3 + \cdots\right)D_x^3\eta_i^j + \ldots \qquad (4.3.2)$$

For $\beta+\alpha = \frac{h^3}{2}$, the local truncation error is of order $o(kh^2 + k^2h^3)$ but for $\beta + \alpha = \frac{h^3}{2}$, and $\alpha = 0$ it is of $o(kh^4 + k^2h^3)$.

Remark:

The previous relations, which enable us to choose α, *and* β, can be obtained using simple calculations by expanding Eq. (4.3.1) in terms of u_i^j and its derivatives, which is the local truncation error of Eq. (4.3.1), as follows:

$$T^*{}_i^j = -\eta_{i-2}^j + 3\eta_{i-1}^j - 3\eta_i^j + \eta_{i+1}^j - \alpha D_x^3\eta_{i-2}^j - \beta D_x^3\eta_{i-1}^j - \beta D_x^3\eta_i^j - \alpha D_x^3\eta_{i+1}^j,$$

$$T_i^j = \left(h^3 - 2(\beta+\alpha)\right)D_x^3\eta_i^j + h\left(\frac{-h^3}{2} + (\beta+\alpha)\right)D_x^4\eta_i^j + h^2\left(\frac{h^3}{4} - \frac{1}{2}(\beta+5\alpha)\right)D_x^5\eta_i^j + h^3\left(\frac{-h^3}{12} + \frac{1}{6}(\beta+7\alpha)\right)D_x^6\eta_i^j + h^4\left(\frac{h^3}{40} - \frac{1}{24}(\beta+17\alpha)\right)D_x^7\eta_i^j + \cdots$$

4.4 Stability Analysis

Using the Von Neumann method, the stability of the method can be investigated. According to this method, the solution of the difference (4.2.12) can be written in the form:

$$Z_i^j = \zeta^j\, exp(q\phi ih), \qquad (4.4.1)$$

where φ is the wave number, $q = \sqrt{-1}$, h is the element size, and ζ^j is the amplification factor at time level j. Inserting the latter expression for Z_i^j in scheme (4.2.12), we obtain the characteristic equation in the form

$$\zeta^j\{A_i\, exp((i-2)q\phi h) + B_i\, exp\big((i-1)q\phi h\big) + C_i\, exp(iq\phi h) + D_i\, exp\big((i+1)q\phi h\big)\} = \zeta^{j-1}\{\alpha\, exp((i-2)q\phi h) + \beta\, exp\big((i-1)q\phi h\big) + \beta\, exp(iq\phi h) + \alpha\, exp\big((i+1)q\phi h\big)\}.$$

$$A_i = -k + \alpha, B_i = 3k + \beta, C_i = -3k + \beta,\ A_i = k + \alpha.$$

Dividing both sides by

$$\zeta^{j-1}\{A_i\, exp((i-2)q\phi h) +: B_i\, exp\big((i-1)q\phi h\big) + C_i\, exp(iq\phi h) + D_i\, exp\big((i+1)q\phi h\big)\}$$

results in

$$\frac{\zeta^j}{\zeta^{j-1}}$$

$$= \frac{\alpha\, exp((i-2)q\phi h) + \beta\, exp\big((i-1)q\phi h\big) + \beta\, exp(iq\phi h) + \alpha\, exp\big((i+1)q\phi h\big)}{A_i\, exp((i-2)q\phi h) + B_i\, exp\big((i-1)q\phi h\big) + C_i\, exp(iq\phi h) + D_i\, exp\big((i+1)q\phi h\big)\}}$$

Dividing by exp $(iq\phi h)$ then results in

$$\zeta = \frac{\alpha\, exp(-2q\phi h) + \beta\, exp(-q\phi h) + \beta + \alpha\, exp(q\phi h)}{A_i\, exp(-2q\phi h) + B_i\, exp(-q\phi h) + C_i + D_i\, exp(q\phi h)} \qquad (4.4.2)$$

Using Euler's formula; results in

$$exp[\, q\varphi] = cos\, \varphi + q\, sin\, \varphi\,, \quad \varphi = \phi h,$$

$$\zeta = \frac{\alpha(cos\emptyset^2 - 2qsin\emptyset cos\emptyset - sin\emptyset^2) + \beta(cos\emptyset - qsin\emptyset) + \beta + \alpha(cos\emptyset + qsin\emptyset)}{A_i(cos2\emptyset - qsin2\emptyset) + B_i(cos\emptyset - qsin\emptyset) + C_i + D_i(cos\emptyset + qsin\emptyset)}$$

$$= \frac{\alpha(cos2\emptyset - qsin2\emptyset) + \beta(cos\emptyset - qsin\emptyset) + \beta + \alpha(cos\emptyset + qsin\emptyset)}{(\alpha - k)(cos2\emptyset - qsin2\emptyset) + (3k + \beta)(cos\emptyset - qsin\emptyset) + (\beta - 3k) + (\alpha + k)(cos\emptyset + qsin\emptyset)}$$

$$= \frac{\alpha cos2\emptyset - \alpha qsin2\emptyset + \beta cos\emptyset - \beta qsin\emptyset + \beta + \alpha cos\emptyset + \alpha qsin\emptyset}{(\alpha - k)cos2\emptyset - q(\alpha - k)sin2\emptyset + (3k + \beta)cos\emptyset - q(3k + \beta)sin\emptyset + (\beta - 3k) + (\alpha + k)cos\emptyset + q(\alpha + k)sin\emptyset}$$

$$= \frac{\alpha cos2\emptyset + (\alpha + \beta)cos\emptyset + \beta + (\alpha q - \beta q)sin\emptyset - \alpha qsin2\emptyset}{(\alpha - k)cos2\emptyset - (\alpha + \beta + 4k)cos\emptyset + (\beta - 3k) - q(\alpha - k)sin2\emptyset + q(\alpha - \beta - 2k)sin\emptyset}$$

$$= \frac{\alpha cos2\emptyset + (\alpha + \beta)cos\emptyset + \beta + q[(-\alpha sin2\emptyset) + (\alpha - \beta)sin\emptyset]}{(\alpha - k)cos2\emptyset - (\alpha + \beta + 4k)cos\emptyset + (\beta - 3k) + q[(k - \alpha)sin2\emptyset + (\alpha - \beta - 2k)sin\emptyset]}$$

Eq. (4.4.2) becomes:

$$\zeta = \frac{X^* + qY^*}{X + qY}, \qquad (4.4.3)$$

where

$$X^* = \alpha\, cos\, 2\, \varphi + (\beta + \alpha)\, cos\, \varphi + \beta,$$

$$Y^* = -\alpha\, sin\, 2\, \varphi + (\alpha - \beta)\, sin\, \varphi,$$

$$X = (\alpha - k)\, cos\, 2\, \varphi + (\beta + \alpha + 4k)\, cos\, \varphi + (\beta - 3k), \qquad (4.4.4)$$

$$Y = (k - \alpha)\, sin\, 2\, \varphi + (-2k + \alpha - \beta)\, sin\, \varphi.$$

Using $\cos 2\phi = 1 - 2\sin^2 \phi$, and $\cos\phi = 1 - 2\sin^2 \frac{\phi}{2}$, we can rewrite the equations in (4.4.4) as

$$X^* = \alpha cos2\emptyset + (\beta + \alpha)cos\emptyset + \beta$$

$$= \alpha cos2\emptyset + \alpha - \alpha + \alpha cos\emptyset + \beta cos\emptyset + \beta$$

$$= \alpha(cos2\emptyset - 1) + \alpha(cos\emptyset + 1) + \beta(cos\emptyset + 1)$$

$$= \alpha(-2sin\emptyset^2) + \alpha\left(2cos^2\frac{\emptyset}{2}\right) + \beta\left(2cos^2\frac{\emptyset}{2}\right)$$

$$= -2\alpha\left(2sin\frac{\emptyset}{2}cos\frac{\emptyset}{2}\right)^2 + \alpha\left(2cos^2\frac{\emptyset}{2}\right) + \beta(2cos^2\frac{\emptyset}{2})$$

$$= 2\alpha cos^2\frac{\emptyset}{2} + 2\beta cos^2\frac{\emptyset}{2} - 8\alpha sin^2\frac{\emptyset}{2}cos^2\frac{\emptyset}{2}$$

$$= 2\left(cos^2\tfrac{\emptyset}{2}\right)\left(\beta + \alpha - 4\alpha\, sin^2\tfrac{\emptyset}{2}\right),$$

which take the form

$$X^* = 2\left(cos^2\tfrac{\varphi}{2}\right)\left(\beta + \alpha - 4\alpha\, sin^2\tfrac{\varphi}{2}\right),$$

$$Y^* = -\alpha\, sin\, 2\, \emptyset + (\alpha - \beta)\, sin\, \emptyset$$

$$= -\alpha(2sin\emptyset cos\emptyset) + \alpha sin\emptyset - \beta sin\emptyset$$

$$= -2\alpha sin\emptyset cos\emptyset + 2\alpha sin\emptyset - \alpha sin\emptyset - \beta sin\emptyset$$

$$= sin\emptyset[-2\alpha cos\emptyset + 2\alpha - \alpha - \beta]$$

$$= sin\emptyset[-\alpha - \beta - 2\alpha(cos\emptyset - 1)]$$

$$= sin\emptyset\left[-\alpha - \beta - 2\alpha\left(-2\, sin^2\frac{\emptyset}{2}\right)\right]$$

$$= sin\emptyset\left[-\alpha - \beta + 4\alpha\, sin^2\frac{\emptyset}{2}\right]$$

$$= (2sin\frac{\emptyset}{2}cos\frac{\emptyset}{2})(-1)[\alpha + \beta - 4\alpha sin^2\frac{\emptyset}{2}]$$

$$= -2\left(sin\frac{\emptyset}{2}cos\frac{\emptyset}{2}\right)\left(\beta + \alpha - 4\alpha\, sin^2\frac{\emptyset}{2}\right).$$

Then, the final form of Y^* is as follows:

$$Y^* = -2\left(sin\frac{\varphi}{2}cos\frac{\varphi}{2}\right)\left(\beta + \alpha - 4\alpha\, sin^2\frac{\varphi}{2}\right),$$

$$X = (\alpha - k)\cos 2\emptyset + (\beta + \alpha + 4k)\cos\emptyset + (\beta - 3k)$$

$$= \alpha\cos 2\emptyset - k\cos 2\emptyset + \alpha cos\emptyset + \beta cos\emptyset + 4k\cos\emptyset + \beta - 3k$$

$$= \alpha cos2\emptyset + \alpha - \alpha + \alpha cos\emptyset + \beta cos\emptyset + \beta - kcos2\emptyset + 4kcos\emptyset - 3k$$

$$= \alpha(cos2\emptyset - 1) + \alpha(cos\emptyset + 1) + \alpha(cos\emptyset + 1) - 2k - kcos2\emptyset - k + 4kcos\emptyset$$

$$= \alpha(-2sin^2\emptyset) + \alpha(cos\emptyset + 1) + \alpha(cos\emptyset + 1) - 2k - kcos2\emptyset - k + 4kcos\emptyset$$

$$= -2\alpha sin^2\emptyset + (\alpha + \beta)(cos\emptyset + 1) - 2k - k(cos\emptyset + 1) + 4kcos\emptyset$$

$$= -2\alpha(2sin\frac{\emptyset}{2}cos\frac{\emptyset}{2})^2 + (\alpha + \beta)\left(2cos^2\frac{\emptyset}{2}\right) - 2k - k(2cos^2\emptyset) + 4kcos\emptyset$$

$$= -8\alpha\left(sin^2\frac{\emptyset}{2}cos^2\frac{\emptyset}{2}\right)^2 + 2(\alpha + \beta)\left(cos^2\frac{\emptyset}{2}\right) - 2k(1 + cos^2\emptyset - 2cos\emptyset)$$

$$= -8\alpha\left(sin^2\frac{\emptyset}{2}cos^2\frac{\emptyset}{2}\right)^2 + 2(\alpha + \beta)\left(cos^2\frac{\emptyset}{2}\right) - 2k(1 - cos\emptyset)^2$$

$$= -8\alpha\left(sin^2\frac{\emptyset}{2}cos^2\frac{\emptyset}{2}\right)^2 + 2(\alpha + \beta)\left(cos^2\frac{\emptyset}{2}\right) - 2k(2sin^2\frac{\emptyset}{2})^2$$

$$= 2\left(cos^2\frac{\emptyset}{2}\right)\left(\alpha + \beta - 4\alpha sin^2\frac{\emptyset}{2}\right) - 8ksin^4\frac{\emptyset}{2}$$

$$X = \left[2\left(cos^2\frac{\varphi}{2}\right)\left(\beta + \alpha - 4\alpha\, sin^2\frac{\varphi}{2}\right) - 8k\, sin^4\frac{\varphi}{2}\right] = X^* + T,$$

$$Y = (k - \alpha)\sin 2\emptyset + (-2k + \alpha - \beta)\sin\emptyset$$

$$= k\sin 2\emptyset - \alpha\sin 2\emptyset + \alpha\sin\emptyset - \beta sin\emptyset - 2ksin\emptyset$$

$$= k\,sin\,2\,\emptyset - 2ksin\emptyset - \alpha sin2\emptyset + 2\alpha sin\emptyset - \alpha sin\emptyset - \beta\,sin\,\emptyset$$

$$= (\alpha - k)(2sin\emptyset - sin2\emptyset) - \alpha sin\emptyset - \beta sin\emptyset$$

$$= (\alpha - k)(2sin\emptyset - 2sin\emptyset cos\emptyset) - 2\alpha sin\frac{\emptyset}{2}cos\frac{\emptyset}{2} - 2\beta in\frac{\emptyset}{2}cos\frac{\emptyset}{2}$$

$$= (\alpha - k)(2sin\emptyset)(1 - cos\emptyset) - 2\alpha sin\frac{\emptyset}{2}cos\frac{\emptyset}{2} - 2\beta in\frac{\emptyset}{2}cos\frac{\emptyset}{2}$$

$$= (2\alpha sin\emptyset)(1 - cos\emptyset) - (2ksin\emptyset)(1 - cos\emptyset) - 2\alpha sin\frac{\emptyset}{2}cos\frac{\emptyset}{2} - 2\beta in\frac{\emptyset}{2}cos\frac{\emptyset}{2}$$

$$= 4\alpha\left(sin\frac{\emptyset}{2}cos\frac{\emptyset}{2}\right)\left(2sin^2\frac{\emptyset}{2}\right) - 4k\left(sin\frac{\emptyset}{2}cos\frac{\emptyset}{2}\right)\left(2sin^2\frac{\emptyset}{2}\right) - 2\alpha sin\frac{\emptyset}{2}cos\frac{\emptyset}{2} - 2\beta in\frac{\emptyset}{2}cos\frac{\emptyset}{2}$$

$$= 8\alpha\left(sin^3\frac{\emptyset}{2}cos\frac{\emptyset}{2}\right) - 8k\left(sin^3\frac{\emptyset}{2}cos\frac{\emptyset}{2}\right) - 2\alpha sin\frac{\emptyset}{2}cos\frac{\emptyset}{2} - 2\beta in\frac{\emptyset}{2}cos\frac{\emptyset}{2}$$

$$= -2\left(sin\frac{\emptyset}{2}cos\frac{\emptyset}{2}\right)(\alpha + \beta - 4\alpha sin^2\frac{\emptyset}{2}) - 8k\left(sin^3\frac{\emptyset}{2}cos\frac{\emptyset}{2}\right)$$

$$Y = -2\left(sin\frac{\varphi}{2}cos\frac{\varphi}{2}\right)\left(\beta + \alpha - 4\alpha\,sin^2\frac{\varphi}{2}\right) - 8k\,cos\frac{\varphi}{2}sin^3\frac{\varphi}{2} = Y^* + G, \quad (4.4.5)$$

where $T = -8k\,sin^4\frac{\varphi}{2}$, $G = -8\,kcos\frac{\varphi}{2}sin^3\frac{\varphi}{2}$.

Using Eq. (4.4.3), we obtain

$$|\zeta| = \sqrt{\frac{X^{*2}+Y^{*2}}{X^2+Y^2}}. \quad (4.4.6)$$

Eq. (4.4.6) enables us to rewrite the last equation in the form

$$|\zeta| = \sqrt{\frac{X^{*2}+Y^{*2}}{(X^*+T)^2+(Y^*+G)^2}},$$

or

$$|\zeta| = \sqrt{\frac{X^{*2}+Y^{*2}}{X^{*2}+T^2+Y^{*2}+G^2+\delta}}, \quad (4.4.7)$$

where $\delta = 2X^*T + 2Y^*G$. Using (4.4.7), δ becomes

$$\delta = 2(2\left(cos^2\frac{\emptyset}{2}\right)\left(\alpha+\beta - 4\alpha sin^2\frac{\emptyset}{2}\right)(-8k\, sin^4\frac{\emptyset}{2}) + 2(-2(sin\frac{\emptyset}{2}cos\frac{\emptyset}{2})(\alpha+\beta - 4\alpha sin^2\frac{\emptyset}{2})(-8\,kcos\frac{\varphi}{2}sin^3\frac{\emptyset}{2})$$

$$= -32k\left(sin^4\frac{\emptyset}{2}cos^2\frac{\emptyset}{2}\right)\left(\alpha+\beta - 4\alpha sin^2\frac{\emptyset}{2}\right) + 32k(sin\frac{\emptyset}{2}cos\frac{\emptyset}{2})(\alpha+\beta - 4\alpha sin^2\frac{\emptyset}{2})(cos\frac{\varphi}{2}sin^3\frac{\emptyset}{2})$$

$$= -32k\left(sin^4\frac{\emptyset}{2}cos^2\frac{\emptyset}{2}\right)\left(\alpha+\beta - 4\alpha sin^2\frac{\emptyset}{2}\right) + 32k\left(sin^4\frac{\emptyset}{2}cos^2\frac{\emptyset}{2}\right)(\alpha+\beta - 4\alpha sin^2\frac{\emptyset}{2})$$

$$= 0$$

$$\delta = -32k\, cos^2\tfrac{\varphi}{2}sin^4\tfrac{\varphi}{2}\left(\beta+\alpha-4\alpha\, sin^2\tfrac{\varphi}{2}\right) + 32\, cos^2\tfrac{\varphi}{2}sin^4\tfrac{\varphi}{2}\left(\beta+\alpha-4\alpha\, sin^2\tfrac{\varphi}{2}\right) = 0.$$

This result enables us to write Eq. (4.4.7) as

$$|\zeta| = \sqrt{\frac{X^{*2}+Y^{*2}}{X^{*2}+Y^{*2}+T^2+G^2}}. \qquad (4.4.8)$$

For stability, we must have $|\zeta| \le 1$ (otherwise ζ^j in Eq. (4.4.1) would grow in an unbounded manner). Using Eq. (4.4.8), we can say that the stability condition, that is $|\zeta| \le 1$, is satisfied.

4.5 Numerical Example

In this section**,** we obtain numerical solutions to Eq. (4.1.1) for a numerical example. Consider the non-homogeneous third-order dispersive PDE [105]:

$$\frac{\partial\eta}{\partial t} + \frac{\partial^3\eta}{\partial x^3} = -sin\,(\pi x)sin\;t - \pi^3 cos\,(\pi x)cos\;t,\; 0 \le x \le 1, t \ge 0,$$

with boundary conditions

$$\eta(0,t) = \eta(1,t) = \eta_{xx}(1,t) = 0,\; t > 0,$$

and the initial condition

$$\eta(x,0) = \sin\ \pi x,\ 0 \le x \le 1.$$

The exact solution of this problem is

$$\eta(x,t) = \sin\ \pi x \cos\ t$$

The numerical results obtained are listed in the following tables, where all calculations are carried out using Mathematica. The accuracy of the method is measured by computing the L∞ - error norm, Max. Absolute error, as shown in Tables 4.5.1–4.5.3. Tables 4.5.4–4.5.5 show the numerical and exact solutions for $h = 0.025, k = 0.0005, and\ \beta = -\alpha + \frac{h^3}{2}$.

Table 4.5.1 The $L_\infty error$ for the numerical and exact solutions when $h = 0.025, k = 0.0005, \alpha = 0, and\ \beta = -\alpha + \frac{h^3}{2}$.

Time	0.500	1.500	2.00	2.500
$L_\infty error$	4.59312× 10^{-6}	5.05911× 10^{-7}	2.01782× 10^{-6}	4.047 ×10^{-6}

Table 4.5.2 The $L_\infty error$ for the numerical and exact solutions when $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, and\ \beta = -\alpha + \frac{h^3}{2}$.

Time	0.500	1.500	2.00	2.500
$L_\infty error$	7.28473× 10^{-5}	9.00942× 10^{-6}	3.09094× 10^{-5}	6.31829×10^{-5}

Table 4.5.3 The $L_\infty error$ for the numerical and exact solutions when $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{24}, and\ \beta = -\alpha + \frac{h^3}{2}$.

Time	0.500	1.500	2.00	2.500
$L_\infty error$	4.63835× 10^{-4}	5.7401× 10^{-5}	1.9661×10^{-4}	4.02159×10^{-4}

The reason that the accuracy in Table 4.5.1 is the best is that for $\alpha = 0, \beta = -\alpha + \frac{h^3}{2}$, the local truncation error is of order $o(kh^4 + k^2h^3)$ but for $\beta + \alpha = \frac{h^3}{2}$, it is of order $o(kh^2 + k^2h^3)$.

Table 4.5.4 Comparison between the numerical and exact solutions when $h = 0.025, t = 2, \alpha = 0, and\ \beta = -\alpha + \frac{h^3}{2}$.

x	Exact Solution	Numerical Solution
0.1 π	−0.128596	−0.129457
0.2 π	−0.244605	−0.2446030
0.3 π	−0.336669	−0.3366680
0.4 π	−0.395779	−0.3957770
0.5 π	−0.416147	−0.4161450
0.6 π	−0.395779	−0.3957780
0.7 π	−0.336669	−0.3366700
0.8 π	−0.244605	−0.2446046
0.9 π	−0.128596	−0.129457

Table 4.5.5 The $L_\infty error$ for the numerical and exact solutions when $h = \frac{\pi}{20}, k = 0.002, and\ the\ time\ steps\ from\ t = 1.9\ to\ t = 2.1.$

Time	1.9	2	2.1
$L_\infty error$	2.9822×10^{-3}	3.31405×10^{-3}	3.68684×10^{-3}

Table 4.5.6 Comparison between the numerical and exact solutions when $h = 0.025, t = 2, \alpha = \frac{h^3}{160}, and\ \beta = -\alpha + \frac{h^3}{2}.$

x	Exact Solution	Numerical Solution
0.1π	−0.128596	−0.129126
0.2π	−0.244605	−0.2445870
0.3π	−0.336669	−0.3366450
0.4π	−0.395779	−0.3957500
0.5π	−0.416147	−0.4161160
0.6π	−0.395779	−0.3957500
0.7π	−0.336669	−0.3366450
0.8π	−0.244605	−0.2445870
0.9π	−0.128596	−0.129126

Table 4.5.7 The $L_\infty error$ for the numerical and exact solutions when $h = \frac{\pi}{20}, k = 0.0004, and\ the\ time\ steps\ from\ t = 1.9\ to\ t = 2.1.$

Time	1.9	2	2.1
$L_\infty error$	1.86724×10^{-3}	2.08023×10^{-3}	2.31321×10^{-3}

Figures 4.5.1–4.5.12 show the relationships between the numerical and exact solutions for the dispersive equation for various times and the same discretisations (*h*).

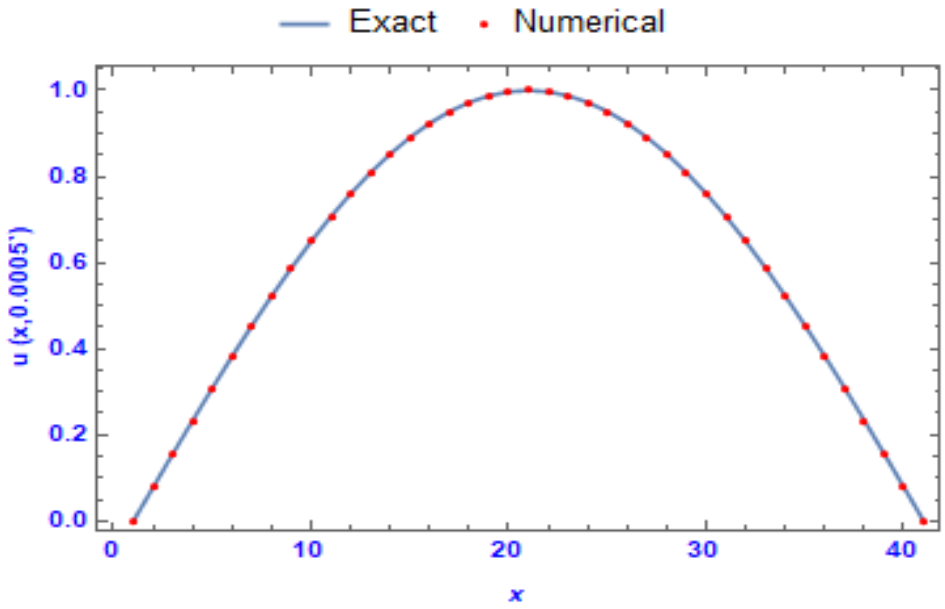


Figure 4.5.1 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 0.0005$.

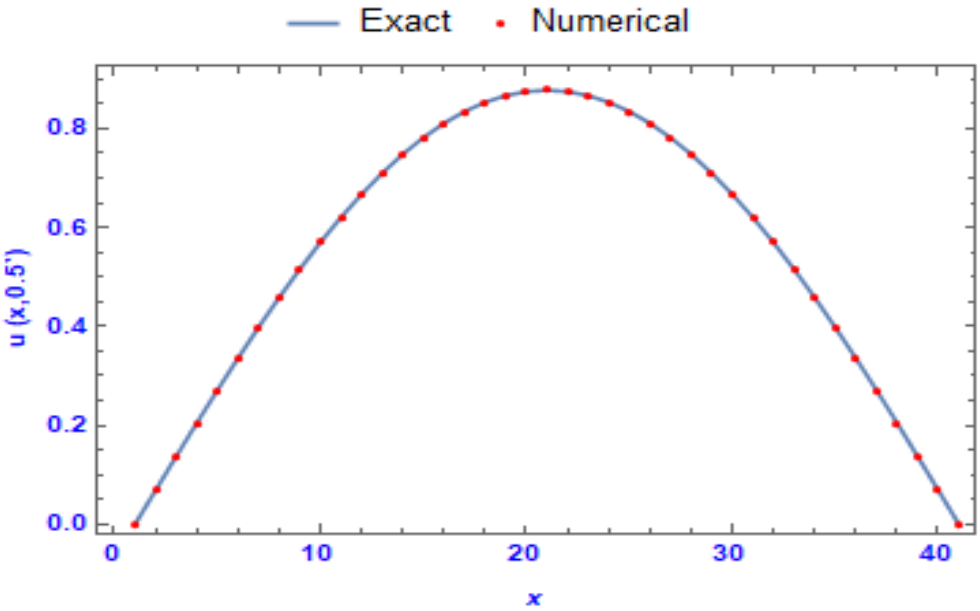


Figure 4.5.2 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 0.5$.

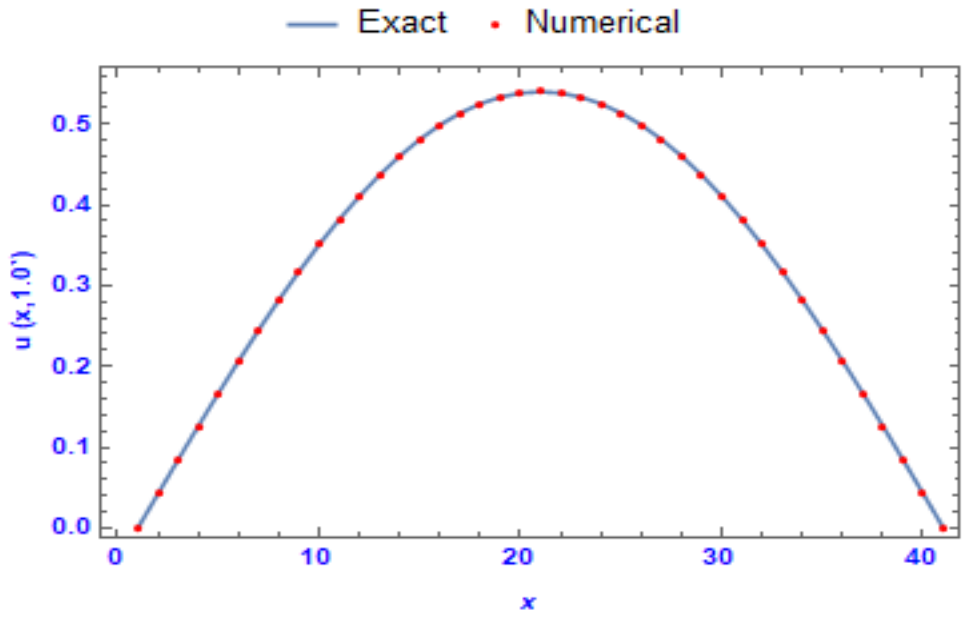


Figure 4.5.3 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 1.0$.

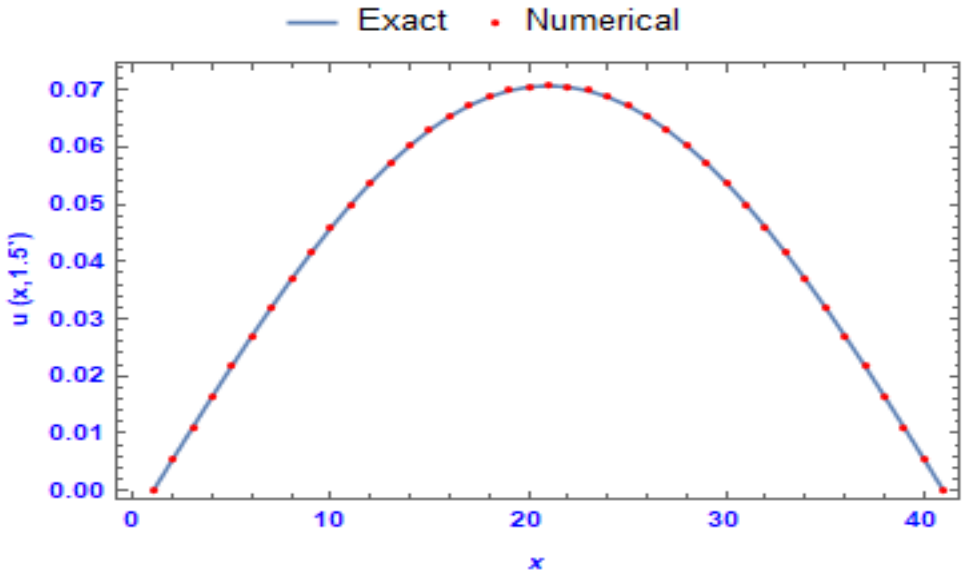


Figure 4.5.4 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 1.5$.

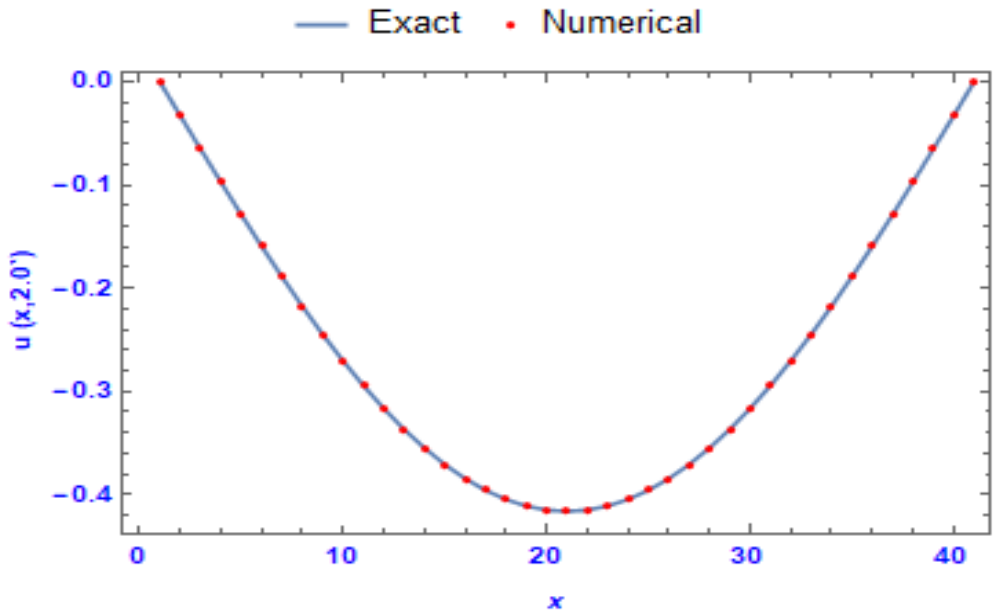


Figure 4.5.5 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 2.0$.

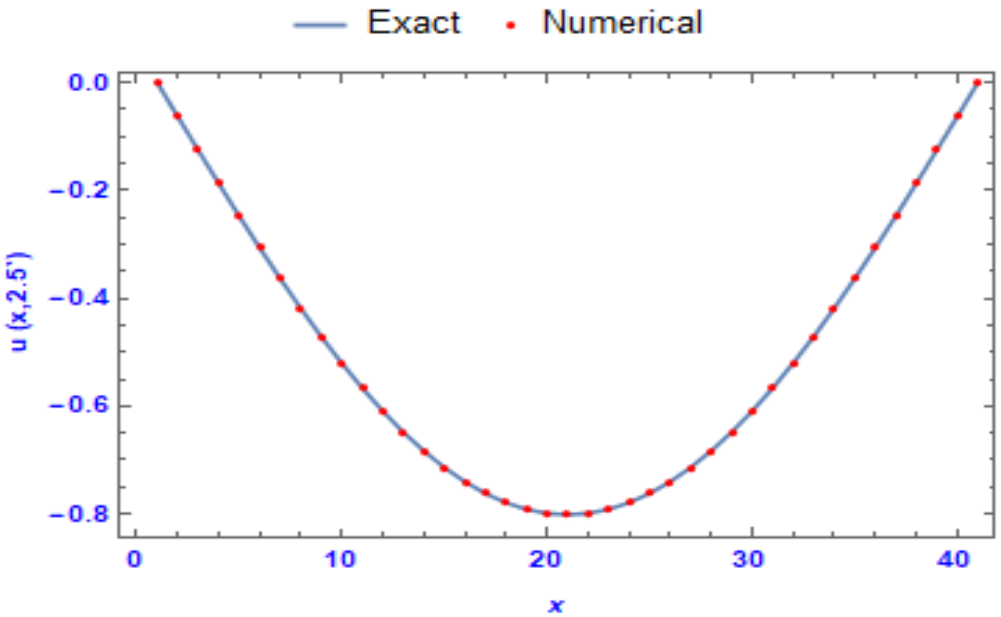


Figure 4.5.6 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 2.5$.

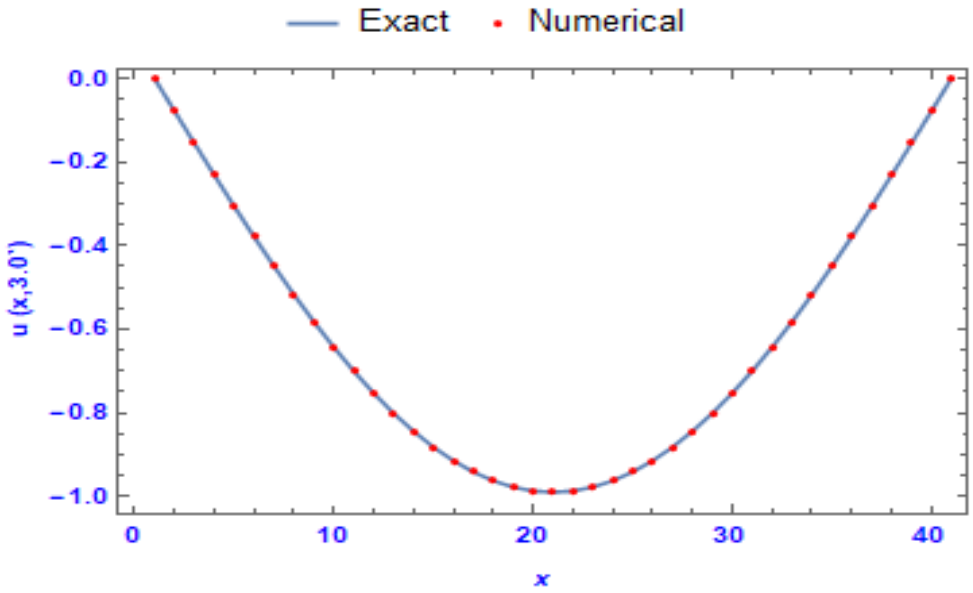


Figure 4.5.7 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 3.0$.

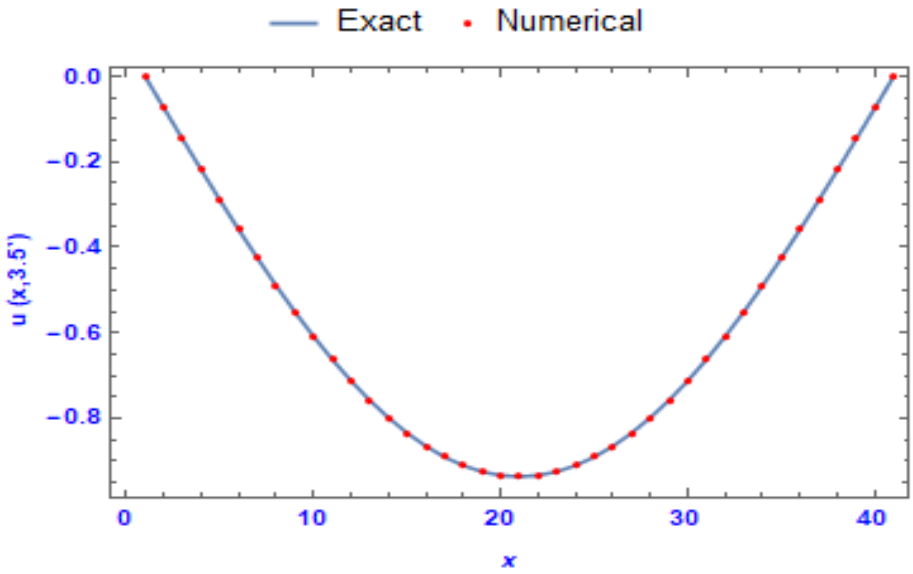


Figure 4.5.8 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 3.5$.

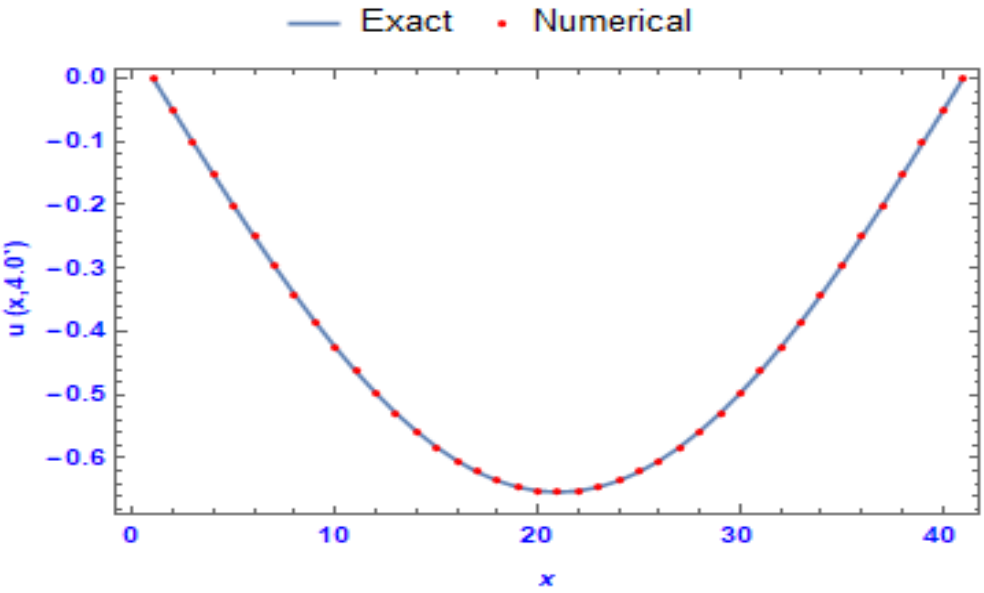


Figure 4.5.9 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 4.0$.

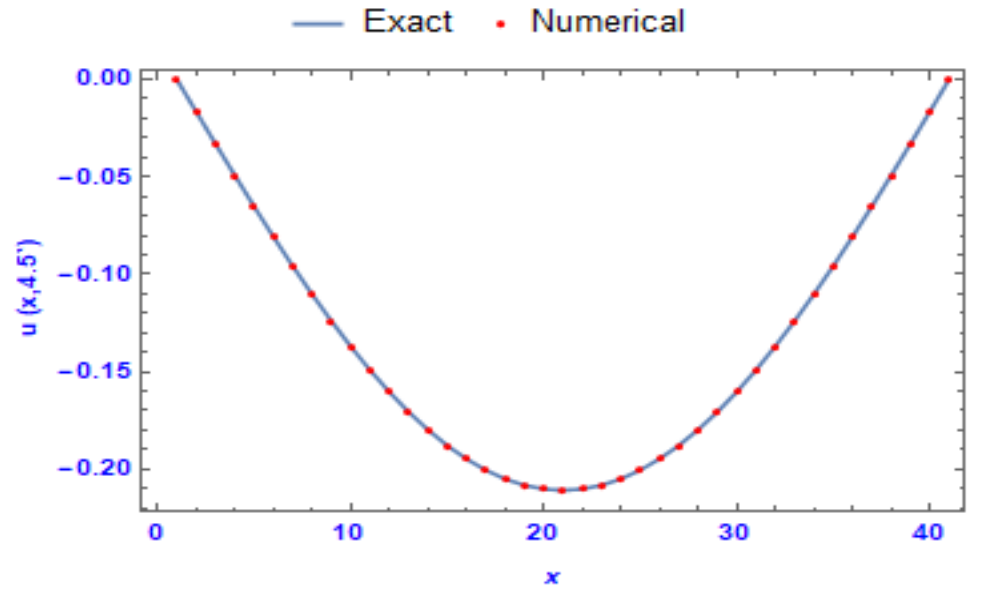


Figure 4.5.10 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 4.5$.

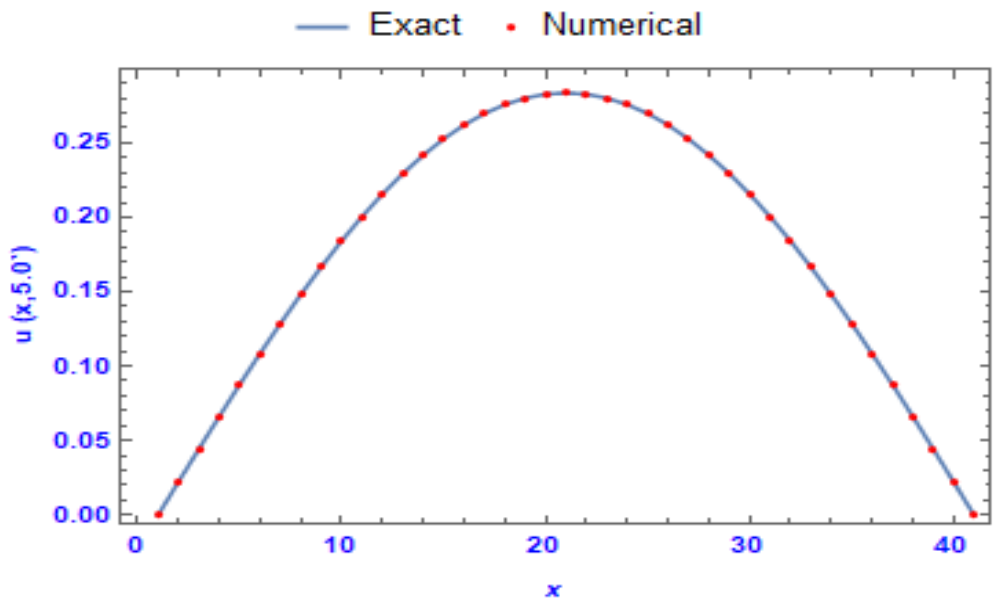


Figure 4.5.11 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 5.0.$

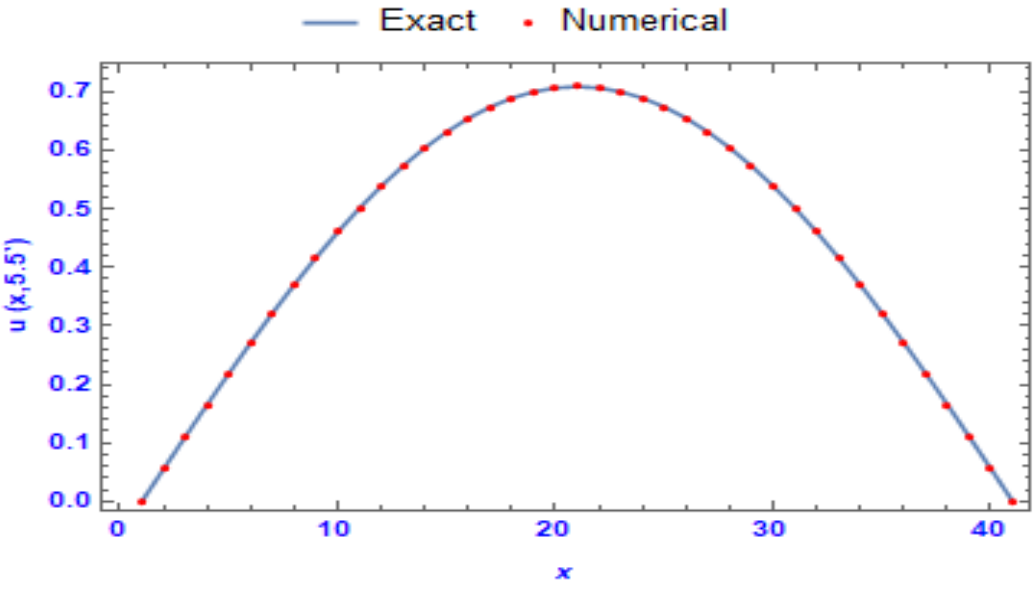


Figure 4.5.12 The relationship between the numerical and exact solutions of the dispersive equation at $h = 0.025, k = 0.0005, \alpha = \frac{h^3}{160}, \beta = -\alpha + \frac{h^3}{2}, and\ t = 5.5.$

Figures 4.5.13–4.5.16 show 3D numerical solutions to the dispersive equation for various times and the same discretisations (h).

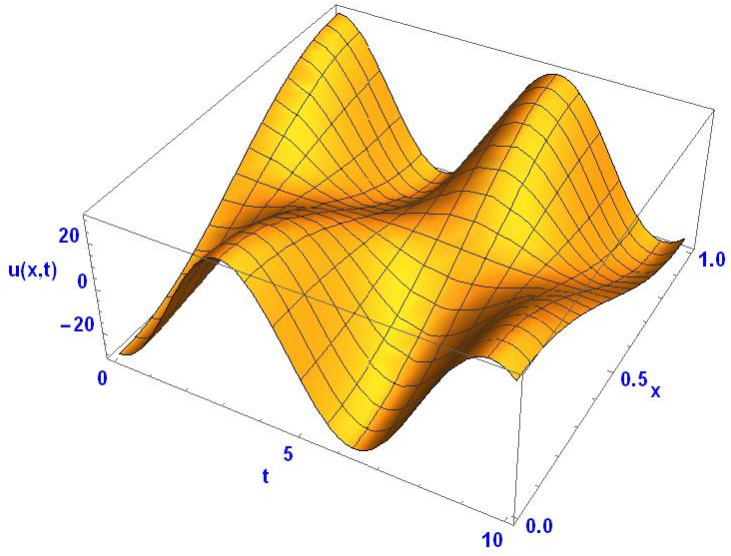


Figure 4.5.13 3D representation of the behaviour of the numerical solutions to the dispersive equation at time t=0.00 to t=10.0.

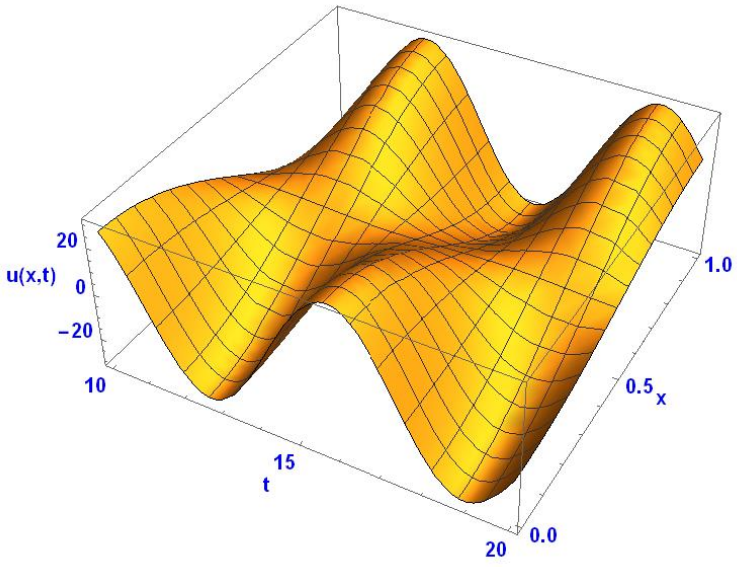


Figure 4.5.14 3D representation of the behaviour of the numerical solutions to the dispersive equation at time t=10.0 to t=20.0.

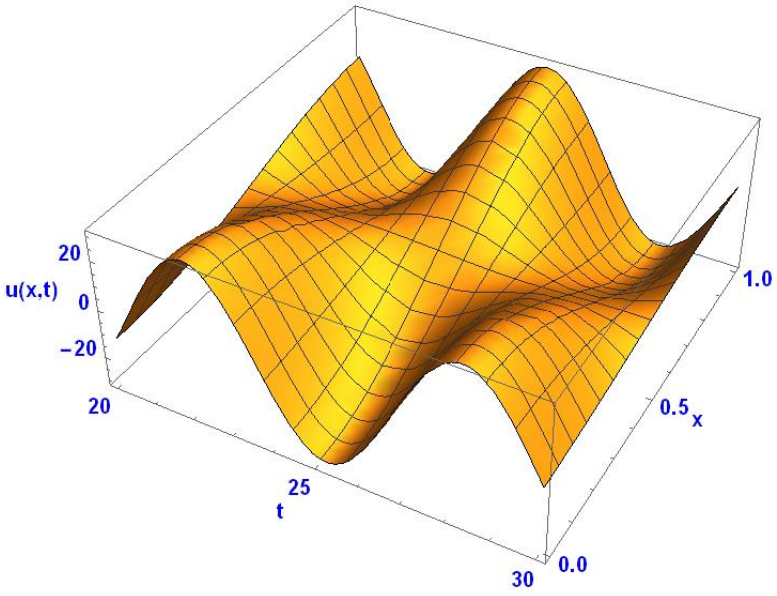


Figure 4.5.15 3D representation of the behaviour of the numerical solutions to the dispersive equation at time *t*=20.0 to *t*=30.0.

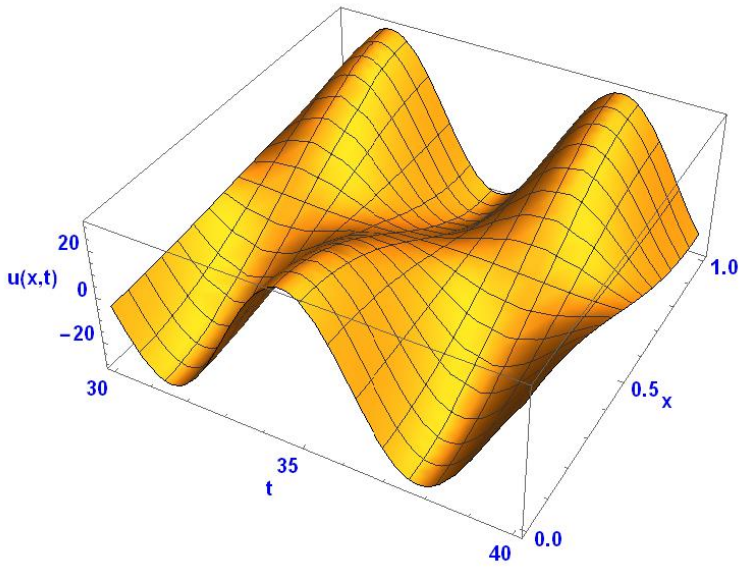


Figure 4.5.16 3D representation of the behaviour of the numerical solutions to the dispersive equation at times *t*=30.0 to *t*=40.0.

4.6 Concluding Remarks

This chapter was devoted to the use of quartic non-polynomial spline functions for solving third-order dispersive PDEs. Recent trends in computational mathematics, mathematical physics and mechanics show the common use of spline functions to solve such problems. The results obtained are very encouraging. It was shown that the L_∞ error norms confirm theoretical convergence. The convergence analysis of the method proved that the scheme is third-order convergent. Also, the method was shown to be unconditionally stable. The numerical examples illustrated that the non-polynomial spline functions are more adaptable in approximating functions. The graphs comparing exact and approximate solutions for the numerical examples show the superiority of the method compared with [100].

CHAPTER 5

Numerical Investigation of Coupled Nonlinear Non-Homogeneous Partial Differential Equations

Chapter 5: Numerical Investigation of Coupled Nonlinear Non-Homogeneous Partial Differential Equations

5.1 Introduction

Nonlinear phenomena modelled by PDEs are relevant to many areas of scientific fields including solid state physics, plasma physics, fluid dynamics, mathematical biology and chemical kinetics. The Klein–Gordon equation is one of the most important mathematical models in quantum field theory. The nonlinear Klein–Gordon equation (NKGE) is used to model many nonlinear phenomena. It appears in theoretical physics, particularly in the area of relativistic quantum mechanics. This equation is a relativistic version of the Schrödinger equation, which describes scalar spineless particles [78,101–108]. Many researchers have used various numerical methods to solve the NKGE. In 2002, Masmoudi and Nakanishi showed that solutions for the NKGE can be described using a system of two coupled NLS equations as the speed of light tends to infinity, in the strong topology of the energy space [109]. Later, John (2004) argued in favour of a numerical study of a particular form of the NKGE, based on resonant structures within the NKGE, The NKGE has been solved numerically using finite difference methods in one spatial dimension, with the asymmetric double-well potential as its nonlinear term [110].

In 2007, Khusnutdinova asserted that a system of coupled Klein–Gordon equations is a model for 1D nonlinear wave processes in two-component media; for example, long longitudinal waves in elastic bi-layers, where nonlinearity comes only from the bonding material. He proposed general properties for the model (i.e. group classification, conservation laws, invariant solutions) and special solutions exhibiting an energy exchange between the two physical components of the system [111]. A year later, Dehghan and Shokri proposed a numerical scheme to solve the 1D NKGE with quadratic and cubic nonlinearity. Their scheme used collocation points and approximated the solution using thin-plate spline radial basis functions [112].

In 2010, Sassaman studied coupled Klein–Gordon equations in (1+1) and (1+2) dimensions [113]. Li, in 2011, proposed a 1D lattice Boltzmann scheme with an

amending function for the NKGE. With the Taylor and Chapman–Enskog expansion, the NKGE was recovered correctly from the lattice Boltzmann equation [114]. Wu and Ge studied the Klein–Gordon equation coupled with the Maxwell equation in rotationally symmetric bounded domains when a non-homogeneous term breaks the symmetry of the associated functional. Under suitable assumptions about nonlinear perturbation, they obtained infinite radially symmetric solutions to the non-homogeneous Klein–Gordon–Maxwell system [115].

In 2013, Krämer in their diploma study demonstrated the derivation of an approximate solution to the NKGE via the method of multiple scales. This method follows the concept of expanding the solution into a perturbation series, including multiple temporal and spatial scales [116]. In the same year, Chen and Li proved the existence of multiple solutions for the non-homogeneous Klein–Gordon equation coupled with Born–Infeld theory [117]. Guo et al. (2015) presented a numerical analysis of the 1D Klein–Gordon equation with quadratic and cubic nonlinearity, using the element-free reproducing kernel particle Ritz method [118]. Also in 2015, Sarboland and Aminataei provided a numerical scheme to approximate solutions to the NKGE by applying the multiquadric quasi-interpolation and the integrated radial basis function network schemes [119]. A year later, Raza et al. presented a scheme for numerical approximation of solutions to the 1D NKGE. They used a common approach to find a solution for a nonlinear system by first linearising the equations through successive substitution—that is, by applying the Newton iteration method—and then solving a linear least squares problem [120]. Rashidinia and Jokar used polynomial wavelets to find the numerical solution for a NKGE in 2016 [121]. In 2018, Shi and Chen studied the multiplicity of positive solutions for a class of non-homogeneous Klein–Gordon-Maxwell equations. They proved the existence of two positive solutions through use of Ekeland's variational principle and the Mountain Pass Theorem [122]. Recently, in 2021, Ghazi and Tawfiq considered a new approach to solve a type of PDE by using coupled Laplace transformation with a decomposition method to find the exact solution for a nonlinear, non-homogenous equation with initial conditions [123]. Here we consider a nonlinear Klein–Gordon PDE in the following form [78]:

$$\frac{\partial^2 u}{\partial x^2}(x,t) + \frac{\partial^2 u}{\partial t^2}(x,t) + v(x,t)u(x,t) = f(x,t), \quad (5.1.1)$$

$$\frac{\partial^2 v}{\partial x^2}(x,t) + \frac{\partial^2 v}{\partial t^2}(x,t) + u(x,t)v(x,t) = g(x,t), \quad (5.1.2)$$

for $a \leq x \leq b$ and $t \geq 0$, subject to the conditions

$$u(a,t) = \varepsilon_1(t), u(b,t) = \varepsilon_2(x), \quad (5.1.3)$$

$$v(a,t) = \rho_1(t), v(b,t) = \rho_2(t), \quad (5.1.4)$$

$$u(x,0) = \tau_1(x), \frac{\partial u}{\partial t}(x,0) = \tau_2(x), \quad (5.1.5)$$

$$v(x,0) = \sigma_1(x), \frac{\partial v}{\partial t}(x,0) = \sigma_2(x). \quad (5.1.6)$$

5.2 The Numerical Method

To approximate $u(x,t)$ and $v(x,t)$ through collocation using a CBS, let the region $R = [a,b] \times [0,\infty]$ be discretised by a set of points R_{ij}, which are the vertices of a grid of points (x_i, t_j), where $x_i = a + ih, h = \Delta x$ for $i = 0,1,\dots,n$ and $t_j = jk, k = \Delta t$ for $j = 0,1,\dots$; and let $\emptyset_i(x)$ be CBSs with knots at $x_{-2} < x_{-1} < \cdots < x_{n+1} < x_{n+2}$, where

$$\emptyset_i(x) = \frac{1}{h^3}\begin{cases} (x - x_{i-2})^3, \text{if } x \in [x_{i-2}, x_{i-1}] \\ h^3 + 3h^2(x - x_{i-1}) + 3h(x - x_{i-1})^2 - 3(x - x_{i-1})^3, \text{if } x \in [x_{i-1}, x_i] \\ h^3 + 3h^2(x_{i+1} - x) + 3h(x_{i+1} - x)^2 - 3(x_{i+1} - x)^3, \text{if } x \in [x_i, x_{i+1}] \\ (x_{i+2} - x)^3, \text{if } x \in [x_{i+1}, x_{i+2}] \\ 0, otherwise \end{cases}.$$

Table 5.2.1 presents values of $\emptyset_i(x)$ and its derivatives at the knots. Since $\emptyset_i(x)$ and its first and second derivatives vanish outside the interval (x_{i-2}, x_{i+2}), there is no need to tabulate $\emptyset_i$ for other values of x.

Table 5.2.1 The values of $\emptyset_i(x)$ and their derivative within the interval $[x_{i-2}, x_{i+2}]$.

x	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}
$\emptyset_i(x)$	0	1	4	1	0

$\emptyset_i'(x)$	0	3/h	0	-3/h	0
$\emptyset_i''(x)$	0	$6/h^2$	$-12/h^2$	$6/h^2$	0

The collocation method for approximately solving Eqs. (5.1.1) and (5.1.2) involves seeking approximations U and V of u and v from the finite dimensional subspace of $C^2[a,b]$, which is spanned by the linearly independent set of CBSs $\{\emptyset_{-1}, \emptyset_0, \emptyset_1, \ldots, \emptyset_n, \emptyset_{n+1}\}$ having the forms

$$U(x,t) = \beta_{-1}(t)\emptyset_{-1}(x) + \beta_0(t)\emptyset_0(x) + \cdots + \beta_n(t)\emptyset_n(x) + \beta_{n+1}(t)\emptyset_{n+1}(x), \tag{5.2.1}$$

$$V(x,t) = \alpha_{-1}(t)\emptyset_{-1}(x) + \alpha_0(t)\emptyset_0(x) + \cdots + \alpha_n(t)\emptyset_n(x) + \alpha_{n+1}(t)\emptyset_{n+1}(x), \tag{5.2.2}$$

such that

$$\frac{\partial^2 U}{\partial x^2}(x_i,t_j) + \frac{\partial^2 U}{\partial t^2}(x_i,t_j) + V(x_i,t_j)U(x_i,t_j) = f(x_i,t_j), \tag{5.2.3}$$

$$\frac{\partial^2 V}{\partial x^2}(x_i,t_j) + \frac{\partial^2 V}{\partial t^2}(x_i,t_j) + U(x_i,t_j)V(x_i,t_j) = g(x_i,t_j), \tag{5.2.4}$$

for $a \le x \le b$ and $t \ge 0$, subject to the conditions

$$U(a,t_j) = \varepsilon_1(t_j), U(b,t) = \varepsilon_2(t_j), \tag{5.2.5}$$

$$V(a,t_j) = \rho_1(t_j), V(b,t) = \rho_2(t_j), \tag{5.2.6}$$

$$U(x_i,0) = \tau_1(x_i), \frac{\partial U}{\partial t}(x_i,0) = \tau_2(x_i), \tag{5.2.7}$$

$$V(x_i,0) = \sigma_1(x_i), \frac{\partial V}{\partial t}(x_i,0) = \sigma_2(x_i), \tag{5.2.8}$$

$i = 0,1,\ldots,n$ and $j = 0,1,\ldots$. Substituting Eqs. (5.2.1) and (5.2.2) into Eqs. (5.2.3) and (5.2.4) gives

$$\sum_{m=-1}^{n+1}\beta_m(t_j)\,\emptyset_m''(x_i) + \sum_{m=-1}^{n+1}\frac{d^2\beta_m}{dt^2}(t_j)\,\emptyset_m(x_i) + V_{i,j}\sum_{m=-1}^{n+1}\beta_m(t_j)\emptyset_m(x_i) = f_{i,j}, \tag{5.2.9}$$

$$\sum_{m=-1}^{n+1} \alpha_m\left(t_j\right) \emptyset_m''(x_i) + \sum_{m=-1}^{n+1} \frac{d^2\alpha_m}{dt^2}\left(t_j\right) \emptyset_m(x_i) + U_{i,j} \sum_{m=-1}^{n+1} \alpha_m\left(t_j\right)\emptyset_m(x_i) = g_{i,j}, \tag{5.2.10}$$

where

$$U_{i,j} = \sum_{m=-1}^{n+1} \beta_m\left(t_j\right)\emptyset_m(x_i) \text{ and } V_{i,j} = \sum_{m=-1}^{n+1} \alpha_m\left(t_j\right)\emptyset_m(x_i).$$

Using the values in Table 5.2.1 in Eqs. (5.2.9) and (5.2.10) gives:

$$\frac{6}{h^2}\left(\beta_{i-1,j} - 2\beta_{i,j} + \beta_{i+1,j}\right) + \left(\frac{d^2\beta_{i-1,j}}{dt^2} + 4\frac{d^2\beta_{i,j}}{dt^2} + \frac{d^2\beta_{i+1,j}}{dt^2}\right) + V_{i,j}\left(\beta_{i-1,j} + 4\beta_{i,j} + \beta_{i+1,j}\right) = f_{i,j}\,, \tag{5.2.11}$$

$$\frac{6}{h^2}\left(\alpha_{i-1,j} - 2\alpha_{i,j} + \alpha_{i+1,j}\right) + \left(\frac{d^2\alpha_{i-1,j}}{dt^2} + 4\frac{d^2\alpha_{i,j}}{dt^2} + \frac{d^2\alpha_{i+1,j}}{dt^2}\right) + U_{i,j}\left(\alpha_{i-1,j} + 4\alpha_{i,j} + \alpha_{i+1,j}\right) = g_{i,j}\,, \tag{5.2.12}$$

where

$$V_{i,j} = \alpha_{i-1,j} + 4\alpha_{i,j} + \alpha_{i+1,j}, U_{i,j} = \beta_{i-1,j} + 4\beta_{i,j} + \beta_{i+1,j},\ \beta_{i,j} = \beta_i\left(t_j\right) \text{ and } \alpha_{i,j} = \alpha_i\left(t_j\right).$$

The central finite difference approximations

$$\beta_{i,j} \cong \frac{\beta_{i,j-1}+\beta_{i,j+1}}{2}, \frac{d^2\beta_{i,j}}{dt^2} \cong \frac{\beta_{i,j-1}-2\beta_{i,j}+\beta_{i,j+1}}{k^2}, \tag{5.2.13}$$

$$\alpha_{i,j} \cong \frac{\alpha_{i,j-1}+\alpha_{i,j+1}}{2}, \text{and } \frac{d^2\alpha_{i,j}}{dt^2} \cong \frac{\alpha_{i,j-1}-2\alpha_{i,j}+\alpha_{i,j+1}}{k^2}, \tag{5.2.14}$$

can be substituted into Eqs. (5.2.11) and (5.2.12) to give the following systems:

$$A_{i,j}\beta_{i-1,j+1} + B_{i,j}\beta_{i,j+1} + A_{i,j}\beta_{i+1,j+1} =$$

$$\frac{2}{k^2}\left(\beta_{i-1,j} + 4\beta_{i,j} + \beta_{i+1,j}\right) - A_{i,j}\left(\beta_{i-1,j-1} + \beta_{i+1,j-1}\right) - B_{i,j}\beta_{i,j-1} + f_{i,j}, \tag{5.2.15}$$

and

$$C_{i,j}\alpha_{i-1,j+1} + D_{i,j}\alpha_{i,j+1} + C_{i,j}\alpha_{i+1,j+1} = \frac{2}{k^2}\left(\alpha_{i-1,j} + 4\alpha_{i,j} + \alpha_{i+1,j}\right) - C_{i,j}\left(\alpha_{i-1,j-1} + \alpha_{i+1,j-1}\right) - D_{i,j}\alpha_{i,j-1} + g_{i,j}\,, \tag{5.2.16}$$

where

$$A_{i,j} = \frac{3}{h^2} + \frac{1}{k^2} + 0.5V_{i,j}, B_{i,j} = \frac{-6}{h^2} + \frac{4}{k^2} + 2V_{i,j},$$

$$C_{i,j} = \frac{3}{h^2} + \frac{1}{k^2} + 0.5U_{i,j}, D_{i,j} = \frac{-6}{h^2} + \frac{4}{k^2} + 2U_{i,j},$$

$$V_{i,j} = \alpha_{i-1,j} + 4\alpha_{i,j} + \alpha_{i+1,j}, U_{i,j} = \beta_{i-1,j} + 4\beta_{i,j} + \beta_{i+1,j},$$

for each $i = 0,1, \ldots, n$ and $j = 1,2, \ldots$.

Systems (5.2.15) and (5.2.16) have to be complemented by the boundary conditions:

$$U(a,t_j) = U(x_0,t_j) = \varepsilon_1(t_j), U(b,t) = U(x_n,t) = \varepsilon_2(t_j),$$

$$V(a,t_j) = V(x_0,t_j) = \rho_1(t_j), V(b,t) = V(x_n,t_j) = \rho_2(t_j).$$

Using Eqs. (5.2.1) and (5.2.2) and Table 2.2.4.1 these conditions give

$$\beta_{-1,j} + 4\beta_{0,j} + \beta_{1,j} = \varepsilon_1(t_j), \tag{5.2.17}$$

$$\beta_{n-1,j} + 4\beta_{n,j} + \beta_{n+1,j} = \varepsilon_2(t_j), \tag{5.2.18}$$

$$\alpha_{-1,j} + 4\alpha_{0,j} + \alpha_{1,j} = \rho_1(t_j), \tag{5.2.19}$$

$$\alpha_{n-1,j} + 4\alpha_{n,j} + \alpha_{n+1,j} = \rho_2(t_j), \tag{5.2.20}$$

for each $j = 0,1, \ldots$.

Eliminating $\beta_{-1,j}$ and $\alpha_{-1,j}$ from the first equation of (5.2.15) results in

$$A_{0,j}\beta_{-1,j+1} + B_{0,j}\beta_{0,j+1} + A_{0,j}\beta_{1,j+1} =$$

$$\frac{2}{k^2}\left(\beta_{-1,j} + 4\beta_{0,j} + \beta_{1,j}\right) - A_{0,j}\left(\beta_{-1,j-1} + \beta_{1,j-1}\right) - B_{0,j}\beta_{0,j-1} + f_{0,j}\,.$$

and from Eqs. (5.2.17) and (5.2.19),

$$\beta_{-1,j} + 4\beta_{0,j} + \beta_{1,j} = \varepsilon_1(t_j), j = 0,1, \ldots.$$

$$\alpha_{-1,j} + 4\alpha_{0,j} + \alpha_{1,j} = \rho_1(t_j), j = 0,1, \ldots.$$

where

$A_{0,j} = \frac{3}{h^2} + \frac{1}{k^2} + 0.5V_{0,j}, B_{0,j} = \frac{-6}{h^2} + \frac{4}{k^2} + 2V_{0,j}$, and $V_{0,j} = \alpha_{-1,j} + 4\alpha_{0,j} + \alpha_{1,j}$.

I find

$$A_{0,j}(\beta_{-1,j+1} + \beta_{1,j+1}) + B_{0,j}\beta_{0,j+1} =$$

$$\frac{2}{k^2}\varepsilon_1(t_j) - A_{0,j}(\varepsilon_1(t_{j-1}) - 4\beta_{0,j-1}) - B_{0,j}\beta_{0,j-1} + f_{0,j}.$$

$$A_{0,j}\left(\varepsilon_1(t_{j+1})\right) - 4A_{0,j}\beta_{0,j+1} + B_{0,j}\beta_{0,j+1} =$$

$$\frac{2}{k^2}\varepsilon_1(t_j) - A_{0,j}\varepsilon_1(t_{j-1}) + 4A_{0,j}\beta_{0,j-1} - B_{0,j}\beta_{0,j-1} + f_{0,j}.$$

$$-4\left(\frac{3}{h^2} + \frac{1}{k^2} + 0.5V_{0,j}\right)\beta_{0,j+1} + \left(\frac{-6}{h^2} + \frac{4}{k^2} + 2V_{0,j}\right)\beta_{0,j+1} = \frac{2}{k^2}\varepsilon_1(t_j)$$

$$-A_{0,j}\left(\varepsilon_1(t_{j-1}) - \varepsilon_1(t_{j+1})\right) + 4\left(\frac{3}{h^2} + \frac{1}{k^2} + 0.5V_{0,j}\right)\beta_{0,j-1}$$

$$-\left(\frac{-6}{h^2} + \frac{4}{k^2} + 2V_{0,j}\right)\beta_{0,j-1} + f_{0,j}.$$

$$\frac{-18}{h^2}\beta_{0,j+1} = \frac{18}{h^2}\beta_{0,j-1} + \frac{2}{k^2}\varepsilon_1(t_j) - \left(\frac{3}{h^2} + \frac{1}{k^2} + \frac{1}{2}\rho_1(t_j)\right)\left(\varepsilon_1(t_{j+1}) + \varepsilon_1(t_{j-1})\right) + f_{0,j.}$$

(5.2.21)

Similarly, eliminating $\beta_{n+1,j}$ and $\alpha_{n+1,j}$ from the last of Eq. (5.2.15) and Eqs. (5.2.18) and (5.2.20) results in

$$\frac{-18}{h^2}\beta_{n,j+1} = \frac{18}{h^2}\beta_{n,j-1} + \frac{2}{k^2}\varepsilon_2(t_j) - \left(\frac{3}{h^2} + \frac{1}{k^2} + \frac{1}{2}\rho_2(t_j)\right)\left(\varepsilon_2(t_{j+1}) + \varepsilon_2(t_{j-1})\right) + f_{n,j}.$$

(5.2.22)

Eqs. (5.2.21), (5.2.22) and (5.2.15) for $i = 1,2, \dots, n-1$ can be written in matrix form:

$$A\beta = d \text{ (5.2.23)}$$

where

$$A = \begin{bmatrix} L & 0 & 0 & \dots & & & & 0 \\ A_{1,j} & B_{1,j} & A_{1,j} & 0 & \dots & & & 0 \\ 0 & A_{2,j} & B_{2,j} & A_{2,j} & 0 & \dots & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & & \vdots \\ 0 & \dots & & 0 & A_{n-2,j} & B_{n-2,j} & A_{n-2,j} & 0 \\ 0 & \dots & & & 0 & A_{n-1,j} & B_{n-1,j} & A_{n-1,j} \\ 0 & \dots & & & & 0 & 0 & L \end{bmatrix},$$

where

$$L = \frac{-18}{h^2}, A_{i,j} = \frac{3}{h^2} + \frac{1}{k^2} + 0.5V_{i,j}, B_{i,j} = \frac{-6}{h^2} + \frac{4}{k^2} + 2V_{i,j},$$

$$V_{i,j} = \alpha_{i-1,j} + 4\alpha_{i,j} + \alpha_{i+1,j}, \beta = \left(\beta_{0,j+1}, \beta_{1,j+1}, \dots, \beta_{n,j+1}\right)^t,$$

$$d = (d_0, d_1, \dots, d_n)^t,$$

$$d_0 = \frac{18}{h^2}\beta_{0,j-1} + \frac{2}{k^2}\varepsilon_1(t_j) - \left(\frac{3}{h^2} + \frac{1}{k^2} + \frac{1}{2}\rho_1(t_j)\right)\left(\varepsilon_1(t_{j+1}) + \varepsilon_1(t_{j-1})\right) + f_{0,j},$$

$$d_n = \frac{18}{h^2}\beta_{n,j-1} + \frac{2}{k^2}\varepsilon_2(t_j) - \left(\frac{3}{h^2} + \frac{1}{k^2} + \frac{1}{2}\rho_2(t_j)\right)\left(\varepsilon_2(t_{j+1}) + \varepsilon_2(t_{j-1})\right) + f_{n,j},$$

$$d_i = \frac{2}{k^2}\left(\beta_{i-1,j} + 4\beta_{i,j} + \beta_{i+1,j}\right) - A_{i,j}\left(\beta_{i-1,j-1} + \beta_{i+1,j-1}\right) - B_{i,j}\beta_{i,j-1} + f_{i,j},$$

for each $i = 1,2, \dots, n-1$ and $j = 1,2, \dots$.

A similar system can be developed by eliminating $\alpha_{-1,j}, \beta_{-1,j}, \alpha_{n+1,j}$, and $\beta_{n+1,j}$ from the first and the last equations of (5.2.16). The result is the following system:

$$C\alpha = w \text{ (5.2.24)}$$

where

$$C = \begin{bmatrix} L & 0 & 0 & \dots & & & & 0 \\ C_{1,j} & D_{1,j} & C_{1,j} & 0 & \dots & & & 0 \\ 0 & C_{2,j} & D_{2,j} & C_{2,j} & 0 & \dots & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & & \vdots \\ 0 & \dots & & 0 & C_{n-2,j} & D_{n-2,j} & C_{n-2,j} & 0 \\ 0 & \dots & & & 0 & C_{n-1,j} & D_{n-1,j} & C_{n-1,j} \\ 0 & \dots & & & & 0 & 0 & L \end{bmatrix},$$

where

$$L = \frac{-18}{h^2}, C_{i,j} = \frac{3}{h^2} + \frac{1}{k^2} + 0.5U_{i,j}, D_{i,j} = \frac{-6}{h^2} + \frac{4}{k^2} + 2U_{i,j},$$

$$U_{i,j} = \beta_{i-1,j} + 4\beta_{i,j} + \beta_{i+1,j},\ \alpha = \left(\alpha_{0,j+1}, \alpha_{1,j+1}, \dots, \alpha_{n,j+1}\right)^t,$$

$$w = (w_0, w_1, \dots, w_n)^t,$$

$$w_0 = \frac{18}{h^2}\alpha_{0,j-1} + \frac{2}{k^2}\rho_1(t_j) - \left(\frac{3}{h^2} + \frac{1}{k^2} + \frac{1}{2}\varepsilon_1(t_j)\right)\left(\rho_1(t_{j+1}) + \rho_1(t_{j-1})\right) + g_{0,j},$$

$$w_n = \frac{18}{h^2}\alpha_{n,j-1} + \frac{2}{k^2}\rho_2(t_j) - \left(\frac{3}{h^2} + \frac{1}{k^2} + \frac{1}{2}\varepsilon_2(t_j)\right)\left(\rho_2(t_{j+1}) + \rho_2(t_{j-1})\right) + g_{n,j},$$

$$w_i = \frac{2}{k^2}\left(\alpha_{i-1,j} + 4\alpha_{i,j} + \alpha_{i+1,j}\right) - C_{i,j}\left(\alpha_{i-1,j-1} + \alpha_{i+1,j-1}\right) - D_{i,j}\alpha_{i,j-1} + g_{i,j},$$

for each $i = 1,2, \dots, n-1$ and $j = 1,2, \dots$.

Since $A_{i,j}$ and $B_{i,j}$ are given by $A_{i,j} = \frac{3}{h^2} + \frac{1}{k^2} + 0.5V_{i,j}$, and $B_{i,j} = \frac{-6}{h^2} + \frac{4}{k^2} + 2V_{i,j}$, it is easy to see that $B_{i,j} = 2\left(\frac{1}{k^2} + 0.5V_{i,j} - \frac{6}{h^2}\right) + 2A_{i,j}$. Taking absolute values with sufficiently small k, we have

$$\left|B_{i,j}\right| = \left|2\left(\frac{1}{k^2} + V_{i,j} - \frac{6}{h^2}\right) + 2A_{i,j}\right| = 2\left(\frac{1}{k^2} + V_{i,j} - \frac{6}{h^2}\right) + 2A_{i,j}.$$

Therefore,

$$\left|B_{i,j}\right| > 2A_{i,j} = \left|A_{i,j}\right| + \left|A_{i,j}\right|.$$

From this we observe that A is diagonally dominant and thus nonsingular by Gershgorin's theorem. Since A is nonsingular, system (5.2.23) has a unique solution. Similarly, we can prove that system (5.2.24) has a unique solution if k is chosen to be small enough.

Systems (5.2.23) and (5.2.24) imply that the $(j+1)st$ time step requires values from the $(j)st$ and $(j-1)st$ time steps. This produces a minor starting problem since values for $j = 0$ are given by the first parts of initial conditions (5.2.7) and (5.2.8):

$$\frac{\partial U}{\partial x}(x_0, 0) = \frac{d\tau_1}{dx}(x_0),$$

$$U(x_i, 0) = \tau_1(x_i), i = 0,1, \dots, n. \tag{5.2.25}$$

$$\frac{\partial U}{\partial x}(x_n, 0) = \frac{d\tau_1}{dx}(x_n),$$

$$\frac{\partial V}{\partial x}(x_0, 0) = \frac{d\sigma_1}{dx}(x_0),$$

$$V(x_i, 0) = \sigma_1(x_i), i = 0,1, \dots, n. \tag{5.2.26}$$

$$\frac{\partial V}{\partial x}(x_n, 0) = \frac{d\sigma_1}{dx}(x_n),$$

which can be rewritten by using Eqs. (5.2.1) and (5.2.2) and Table 5.2.1 in the following matrix forms:

$$A^*\beta^* = d^* \tag{5.2.27}$$

where

$$A^* = \begin{bmatrix} \frac{-3}{h} & 0 & \frac{3}{h} & \dots & & & & 0 \\ 1 & 4 & 1 & 0 & \dots & & & 0 \\ 0 & 1 & 4 & 1 & 0 & \dots & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & & \vdots \\ 0 & \dots & & 0 & 1 & 4 & 1 & 0 \\ 0 & \dots & & & 0 & 1 & 4 & 1 \\ 0 & \dots & & & & \frac{-3}{h} & 0 & \frac{3}{h} \end{bmatrix},$$

$\beta^* = \left(\beta_{-1,0}, \beta_{0,0}, \dots, \beta_{n+1,0}\right)^t$, and $d^* = \left(\frac{d\tau_1}{dx}(x_0), \tau_1(x_0), \tau_1(x_1) \dots, \tau_1(x_n), \frac{d\tau_1}{dx}(x_n)\right)^t$

and

$$C^*\alpha^* = w^* \tag{5.2.28}$$

where

$$C^* = \begin{bmatrix} \frac{-3}{h} & 0 & \frac{3}{h} & \dots & & & & 0 \\ 1 & 4 & 1 & 0 & \dots & & & 0 \\ 0 & 1 & 4 & 1 & 0 & \dots & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & & \vdots \\ 0 & \dots & & 0 & 1 & 4 & 1 & 0 \\ 0 & \dots & & & 0 & 1 & 4 & 1 \\ 0 & \dots & & & & \frac{-3}{h} & 0 & \frac{3}{h} \end{bmatrix},$$

$$\alpha^* = \left(\alpha_{-1,0}, \alpha_{0,0}, \dots, \alpha_{n+1,0}\right)^t,$$

and

$$w^* = \left(\frac{d\sigma_1}{dx}(x_0), \sigma(x_0), \sigma_1(x_1) \dots, \sigma_1(x_n), \frac{d\sigma_1}{dx}(x_n)\right)^t.$$

However, the values for $j = 1$, which are needed in (5.2.23) and (5.2.24) to compute $\left(\beta_{-1,2}, \beta_{0,2}, \dots, \beta_{n+1,2}\right)^t$ and $\left(\alpha_{-1,2}, \alpha_{0,2}, \dots, \alpha_{n+1,2}\right)^t$ must be obtained from the second parts of (5.2.7) and (5.2.8):

$$\frac{\partial U}{\partial t}(x_i, 0) = \tau_2(x_i), i = 0,1, \dots, n,$$

$$\frac{\partial V}{\partial t}(x_i, 0) = \sigma_2(x_i), i = 0,1, \dots, n.$$

By using a second Maclaurin polynomial in t for U::

$$U(x_i, t_1) \cong U(x_i, 0) + k\frac{\partial U}{\partial t}(x_i, 0) + \frac{k^2}{2}\frac{\partial^2 U}{\partial \mathrm{t}^2}(x_i, 0) =$$

$$\tau_1(x_i) + k\tau_2(x_i) + \frac{k^2}{2}\frac{\partial^2 U}{\partial \mathrm{t}^2}(x_i, 0),$$

and Eq. (5.2.3) calculated at $t = 0$; that is,

$$\frac{\partial^2 U}{\partial t^2}(x_i, 0) = -\frac{\partial^2 U}{\partial x^2}(x_i, 0) - V(x_i, 0)\, U(x_i, 0) + f(x_i, 0)$$

$$= -\frac{d^2\tau_1}{dx^2}(x_i) - \sigma_1(x_i)\tau_1(x_i) + f_{i,0},$$

we obtain the following system

$$U(x_i, t_1) = \tau_1(x_i) + k\tau_2(x_i) + \frac{k^2}{2}\left(-\frac{d^2\tau_1}{dx^2}(x_i) - \sigma_1(x_i)\tau_1(x_i) + f_{i,0},\right),$$

for each $i = 0,1, \dots, n$. After using the values in Table 2.2.4.1 this equation gives the system

$$\beta_{i-1,1} + 4\beta_{i,1} + \beta_{i+1,1} = \tau_1(x_i) + k\tau_2(x_i) + \frac{k^2}{2}\left(-\frac{d^2\tau_1}{dx^2}(x_i) - \sigma_1(x_i)\tau_1(x_i) + f_{i,0},\right), \tag{5.2.29}$$

for each $i = 0,1, \dots, n$. System (5.2.29) consists of n+1 equations and n+3 unknowns $\left(\beta_{-1,1}, \beta_{0,1}, \dots, \beta_{n+1,1}\right)^t$. Solving the last system requires two additional equations:

$$\frac{\partial^2 U}{\partial x \partial t}(x_0, 0) = \frac{d\tau_2}{dx}(x_0),$$

$$\frac{\partial^2 U}{\partial x \partial t}(x_n, 0) = \frac{d\tau_2}{dx}(x_n).$$

Table 5.2.1 enables us to rewrite the last two equations in the forms

$$\frac{-3}{h}\frac{d\beta_{-1}}{dt}(0) + \frac{3}{h}\frac{d\beta_1}{dt}(0) = \frac{d\tau_2}{dx}(x_0),$$

$$\frac{-3}{h}\frac{d\beta_{n-1}}{dt}(0) + \frac{3}{h}\frac{d\beta_{n+1}}{dt}(0) = \frac{d\tau_2}{dx}(x_n).$$

The use of the forward-difference formula,

$$\frac{d\beta_i}{dt}(0) = \frac{\beta_{i,1} - \beta_{i,0}}{k},$$

in the last two equations results in the equations

$$\frac{-3}{h}\beta_{-1,1} + \frac{3}{h}\beta_{1,1} = \frac{-3}{h}\beta_{-1,0} + \frac{3}{h}\beta_{1,0} + k\frac{d\tau_2}{dx}(x_0), \quad (5.2.30)$$

$$\frac{-3}{h}\beta_{n-1,1} + \frac{3}{h}\beta_{n+1,1} = \frac{-3}{h}\beta_{n-1,0} + \frac{3}{h}\beta_{n+1,0} + k\frac{d\tau_2}{dx}(x_n). \quad (5.2.31)$$

Eqs. (5.2.30), (5.2.31) and (5.2.29) can be written in matrix form:

$$A^{**}\beta^{**} = d^{**} \quad (5.2.32)$$

where

$$A^{**} = \begin{bmatrix} \frac{-3}{h} & 0 & \frac{3}{h} & \dots & & & & 0 \\ 1 & 4 & 1 & 0 & \dots & & & 0 \\ 0 & 1 & 4 & 1 & 0 & \dots & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & & \vdots \\ 0 & \dots & & 0 & 1 & 4 & 1 & 0 \\ 0 & \dots & & & 0 & 1 & 4 & 1 \\ 0 & \dots & & & & \frac{-3}{h} & 0 & \frac{3}{h} \end{bmatrix},$$

$\beta^{**} = \left(\beta_{-1,1}, \beta_{0,1}, \dots, \beta_{n+1,1}\right)^t$, and $d^{**} = (d_{-1}, d_0, \dots, d_n, d_{n+1})^t$,

$$d_i^{**} = \tau_1(x_i) + k\tau_2(x_i) + \frac{k^2}{2}\left(-\frac{d^2\tau_1}{dx^2}(x_i) - \sigma_1(x_i)\tau_1(x_i) + f_{i,0},\right), \text{for } i = 0,1,\dots,n,$$

$d_{-1}^{**} = \frac{-3}{h}\beta_{-1,0} + \frac{3}{h}\beta_{1,0} + k\frac{d\tau_2}{dx}(x_0)$, $d_{n+1}^{**} = \frac{-3}{h}\beta_{n-1,0} + \frac{3}{h}\beta_{n+1,0} + k\frac{d\tau_2}{dx}(x_n)$.

A similar system can be developed for computing $\left(\alpha_{-1,1}, \alpha_{0,1}, \dots, \alpha_{n+1,1}\right)^t$:

$$C^{**}\alpha^{**} = w^{**} \quad (5.2.33)$$

where

$$C^{**} = \begin{bmatrix} \frac{-3}{h} & 0 & \frac{3}{h} & \dots & & & & 0 \\ 1 & 4 & 1 & 0 & \dots & & & 0 \\ 0 & 1 & 4 & 1 & 0 & \dots & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & & \vdots \\ 0 & \dots & & 0 & 1 & 4 & 1 & 0 \\ 0 & \dots & & & 0 & 1 & 4 & 1 \\ 0 & \dots & & & & \frac{-3}{h} & 0 & \frac{3}{h} \end{bmatrix},$$

$$\alpha^{**} = \left(\alpha_{-1,1}, \alpha_{0,1}, \dots, \alpha_{n+1,1}\right)^t, \text{ and } w^{**} = (w_{-1}, w_0, \dots, w_n, w_{n+1})^t,$$

$$w_i^{**} = \sigma_1(x_i) + k\sigma_2(x_i) + \frac{k^2}{2}\left(-\frac{d^2\sigma_1}{dx^2}(x_i) - \tau_1(x_i)\sigma_1(x_i) + g_{i,0},\right), \text{for } i = 0,1, \dots, n,$$

$$w_{-1}^{**} = \frac{-3}{h}\alpha_{-1,0} + \frac{3}{h}\alpha_{1,0} + k\frac{d\sigma_2}{dx}(x_0), w_{n+1}^{**} = \frac{-3}{h}\alpha_{n-1,0} + \frac{3}{h}\alpha_{n+1,0} + k\frac{d\sigma_2}{dx}(x_n).$$

Since matrices A^*, C^*, A^{**} and C^{**} are strictly diagonally dominant, it follows from Gershgorin's theorem that A^*, C^*, A^{**} and C^{**} are nonsingular. Hence, systems (5.2.27), (5.2.28), (5.2.32) and (5.2.33) have unique solutions.

5.3 Stability Analysis

For stability analysis, I use the Von Neumann technique. To do this, I must linearise the nonlinear term $v(x,t)u(x,t)$ in Eq. (5.1.1) by making $v(x,t)$ locally constant, which is equivalent to assuming that the corresponding value $V_{i,j}$ is equal to a local constant V^* in system (5.2.15). According to the Von Neumann technique,

$$\beta_{i,j} = \xi^j exp(q\varphi hi), \tag{5.3.1}$$

where $q^2 = -1, \varphi$ is the mode number, h is the element size, and ξ is the amplification factor. Inserting the last expression for $\beta_{i,j}$ into system (5.2.15) gives the following characteristic equation:

$$A_{i,j}\,\beta_{i-1,j+1} + \; B_{i,j}\;\beta_{i,j+1} + \; A_{i,j}\,\beta_{i+1,j+1} = \frac{2}{k^2}(\,\beta_{i-1,j} + 4\,\beta_{i,j} + \;\beta_{i+1,j})$$

$$- A_{i,j}(\,\beta_{i-1,j-1} + \;\beta_{i+1,j-1}) - \;B_{i,j}\;\beta_{i,j-1} + \;f_{i,j},$$

$$A_{i,j}\xi^{j+1}exp(q\varphi h(i-1)) + \ B_{i,j}\,\xi^{j+1}exp(q\varphi hi) + \ A_{i,j}\xi^{j+1}exp(q\varphi h(i+1))$$

$$= \frac{2}{k^2}(\xi^j exp(q\varphi h(i-1)) + 4\xi^j exp(q\varphi hi) + \xi^j exp(q\varphi h(i+1)))$$

$$- A_{i,j}\left(\xi^{j-1}exp\big(q\varphi h(i-1)\big) + \xi^{j-1}exp\big(q\varphi h(i+1)\big)\right) - \ B_{i,j}\,\xi^{j-1}exp(q\varphi hi),$$

$$\xi^{j+1}\big\{\,A_{i,j}\{exp(q\varphi h(i-1)) + exp(q\varphi h(i+1))\} + \ B_{i,j}exp(q\varphi hi)\big\}$$
$$= \frac{2}{k^2}\xi^j\big\{\{exp(q\varphi h(i-1)) + exp(q\varphi h(i+1))\} + 4exp(q\varphi hi)\big\}$$
$$- \ \xi^{j-1}\big\{\,A_{i,j}\big\{exp\big(q\varphi h(i-1)\big) + exp\big(q\varphi h(i+1)\big)\big\} + \ B_{i,j}exp(q\varphi hi)\big\},$$

where

$$A_{i,j} = \frac{3}{h^2} + \frac{1}{k^2} + 0.5V^*, B_{i,j} = \frac{-6}{h^2} + \frac{4}{k^2} + 2V^*.$$

After dividing by $\xi^{j-1}exp(q\varphi hi)$, this equation becomes

$$\xi^2\big\{\,A_{i,j}\{exp(-q\varphi h) + exp(q\varphi h)\} + \ B_{i,j}\big\} =$$

$$\tfrac{2}{k^2}\xi\big\{\{exp(-q\varphi h) + exp(q\varphi h)\} + 4\big\} - \big\{\,A_{i,j}\{exp(-q\varphi h) + exp(q\varphi h)\} + \ B_{i,j}\big\}.$$

After dividing by $\big\{A_{i,j}\{exp(-q\varphi h) + exp(q\varphi h)\} + \ B_{i,j}\big\}$, this equation becomes

$$\xi^2 = \frac{2\xi}{k^2}\frac{\big\{\{exp(-q\varphi h) + exp(q\varphi h)\} + 4\big\}}{\big\{\,A_{i,j}\{exp(-q\varphi h) + exp(q\varphi h)\} + \ B_{i,j}\big\}} - 1$$

$$\xi^2 = \frac{2\xi}{k^2}\frac{\cos(\varphi h) - qsin(\varphi h) + cos(\varphi h) + qsin(\varphi h) + 4}{\big\{\,A_{i,j}\cos(\varphi h) - \ A_{i,j}qsin(\varphi h) + \ A_{i,j}cos(\varphi h) + \ A_{i,j}qsin(\varphi h) + \ B_{i,j}\big\}} - 1$$

$$\xi^2 = \frac{2\xi}{k^2}\frac{2\cos(\varphi h) + 4}{\big\{\,2A_{i,j}\cos(\varphi h) + \ B_{i,j}\big\}} - 1$$

$$\xi^2 - \frac{2\xi}{k^2}\frac{2\cos(\varphi h) + 4}{\big\{\,2A_{i,j}\cos(\varphi h) + \ B_{i,j}\big\}} + 1 = 0$$

Simple calculations enable us to write

$$\xi^2 + 2\mu\xi + 1 = 0, \qquad (5.3.2)$$

where

$$\mu = \frac{-(4+2cos\phi)}{k^2\left(B_{i,j}+2\,A_{i,j}cos\phi\right)},$$

the two roots of the characteristic Eq. (5.3.2) are

$$\xi_{\pm} = \frac{-2\mu\pm\sqrt{(2\mu)^2-4}}{2} = -\mu \pm \sqrt{\mu^2-1}.$$

The necessary and sufficient condition for stability is $\left|\xi_{\pm}\right| \leq 1$. If $\mu^2 - 1 \leq 0$, it is easy to verify that $\left|\xi_{\pm}\right| \leq 1$, and if $\mu^2 - 1 > 0$ then either ξ_- or ξ_+ do not satisfy the condition $\left|\xi_{\pm}\right| \leq 1$.

Hence, the necessary and sufficient condition for stability is

$$\mu^2 - 1 \leq 0.$$

In other words, we must have

$$-1 \leq \mu \leq 1$$

$$-1 \leq \frac{(4+2cos\phi)}{k^2\left(B_{i,j}+2\,A_{i,j}cos\phi\right)} \leq 1$$

$$-1 \leq \frac{(4+2cos\phi)}{k^2\left(\frac{-6}{h^2}+\frac{4}{k^2}+2V^*+2(\frac{3}{h^2}+\frac{1}{k^2}+\frac{1}{2}V^*)cos\phi\right)} \leq 1$$

$$-1 \leq \frac{(4+2cos\phi)}{\frac{-6}{h^2}k^2+4+2k^2V^*+\frac{6k^2}{h^2}cos\phi+2cos\phi+k^2V^*cos\phi} \leq 1,$$

which implies that

$$-1 \leq \frac{(4+2cos\phi)}{(4+2cos\phi)+k^2\left\{\left(\frac{-6}{h^2}+2V^*\right)+\left(\frac{6}{h^2}+V^*\right)cos\phi\right\}} \leq 1$$

When k is sufficiently small, such that $k^2 \to 0$, this inequality is satisfied. Thus, stability analysis of system (5.2.16) can be developed.

5.4 Numerical Results

Consider the coupled NPDE:

$$\frac{\partial^2 u}{\partial x^2}(x,t)+\frac{\partial^2 u}{\partial t^2}(x,t)+v(x,t)u(x,t)=-(\pi^2+1)sin\pi x\ cost+\sin^2\pi x\ \cos^2 t,$$

$$\frac{\partial^2 v}{\partial x^2}(x,t)+\frac{\partial^2 v}{\partial t^2}(x,t)+u(x,t)v(x,t)=-(\pi^2+1)sin\pi x\ cost+\sin^2\pi x\ \cos^2 t,$$

(5.4.1)

for $0 \le x \le 1$ and $t \ge 0$, subject to the conditions

$$u(0,t)=0, u(1,t)=0,\ v(0,t)=0, v(1,t)=0, \qquad (5.4.2)$$

$$u(x,0)=sin\pi x, \frac{\partial u}{\partial t}(x,0)=0, v(x,0)=sin\pi x, \frac{\partial v}{\partial t}(x,0)=0. \qquad (5.4.3)$$

The exact solution of system (5.4.1) is

$$u(x,t)=sin\pi x\ cost,$$

$$v(x,t)=sin\pi x\ cost. \qquad (5.4.4)$$

The results are presented in Tables 5.4.1–5.4.6. Tables 5.4.1–5.4.4 list the maximum absolute errors with $k=0.0005$ and $h=0.1$ and with $k=0.00005$ and $h=0.01$. Tables 5.4.5–5.4.6 list numerical solutions $V_{i,j}=\alpha_{i-1,j}+4\alpha_{i,j}+\alpha_{i+1,j}$ and $U_{i,j}=\beta_{i-1,j}+4\beta_{i,j}+\beta_{i+1,j}$ and exact solutions $u_{i,j}$ and $v_{i,j}$with $k=0.0005\ and\ h=0.1$.

Table 5.4.1 Maximum absolute error $k = \Delta t = 0.0005$ and $h = \Delta x = 0.1$.

Time	0.1	0.3	0.4	0.5
$\mathbf{Max}\lvert U_{i,j} - u_{i,j}\rvert$	4.0749×10^{-4}	3.8535×10^{-3}	7.1709×10^{-3}	1.3249×10^{-2}

Table 5.4.2 Maximum absolute error $k = \Delta t = 0.0005$ and $h = \Delta x = 0.1$.

Time	0.1	0.3	0.4	0.5
$\mathbf{Max}\lvert V_{i,j} - v_{i,j}\rvert$	4.0749×10^{-4}	3.8535×10^{-3}	7.1709×10^{-3}	1.3249×10^{-2}

Table 5.4.3 Maximum absolute error $k = \Delta t = 0.00005$ and $h = \Delta x = 0.01$.

Time	0.0005	0.005	0.054
$\mathbf{Max}\lvert U_{i,j} - u_{i,j}\rvert$	9.1326×10^{-11}	1.0047×10^{-8}	1.4709×10^{-6}

Table 5.4.4 Maximum absolute error $k = \Delta t = 0.00005$ and $h = \Delta x = 0.01$.

Time	0.0005	0.005	0.054
$\mathbf{Max}\lvert V_{i,j} - v_{i,j}\rvert$	9.1326×10^{-11}	1.0047×10^{-8}	1.4709×10^{-6}

Table 5.4.5 $u_{i,j}$ and $U_{i,j}$ with $t = 0.3, k = \Delta t = 0.0005$ and $h = \Delta x = 0.1$.

x	$u_{i,j}$	$U_{i,j}$
0.2	0.5615326992848998	**0.563810889237005**
0.4	0.90857899323744	**0.9122450638802360**
0.6	0.90857899323744	**0.9122450638156732**
0.8	0.5615326992848999	**0.5638108891169986**

Table 5.4.6 $v_{i,j}$ and $V_{i,j}$ with $t = 0.3, k = \Delta t = 0.0005$ and $h = \Delta x = 0.1$.

x	$v_{i,j}$	$V_{i,j}$
0.2	0.5615326992848998	**0.563810889237005**
0.4	0.90857899323744	**0.9122450638802360**
0.6	0.90857899323744	**0.9122450638156732**
0.8	0.5615326992848999	**0.5638108891169986**

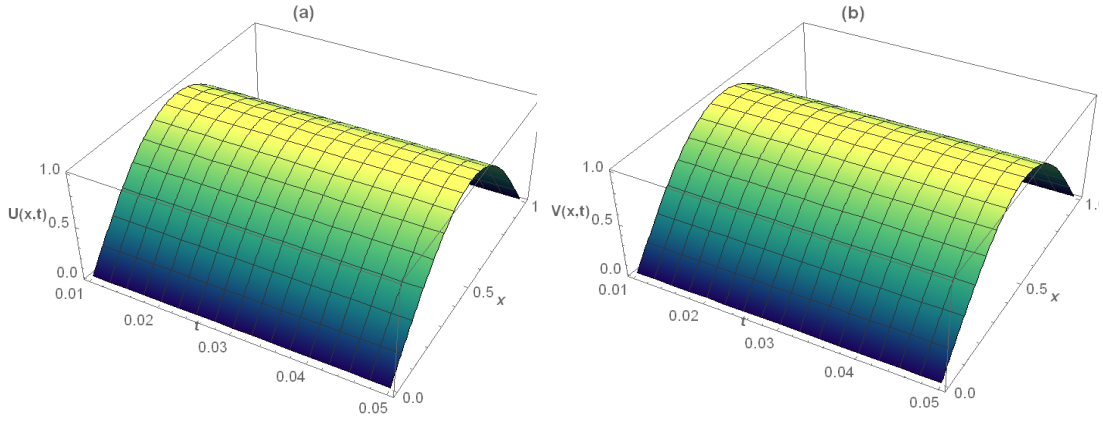


Figure 5.4.1 Graphs of approximate solutions at $\Delta t = 0.0005$ and $h = 0.01$ for $U(x,t)$ part (a) and $V(x,t)$ part (b).

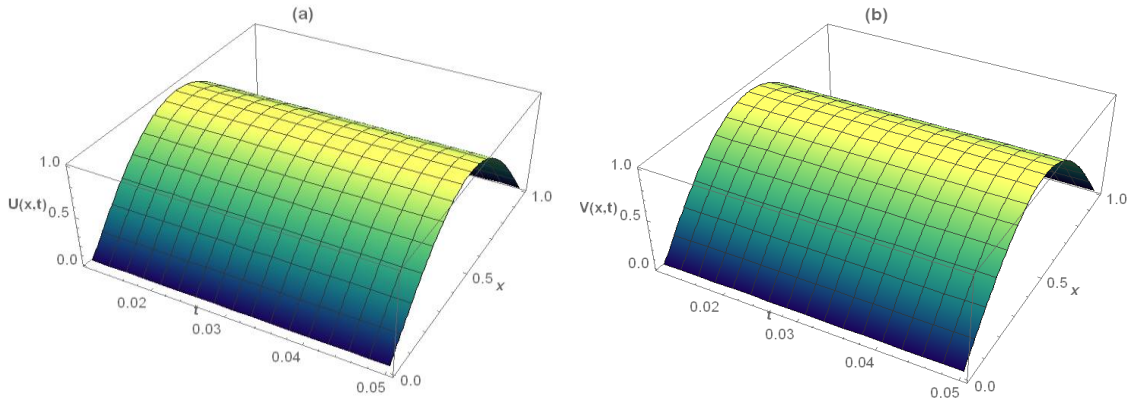


Figure 5.4.2 Graphs of approximate solutions at $\Delta t = 0.00005$ and $h = 0.01$ for $U(x,t)$ part (a) and $V(x,t)$ part (b).

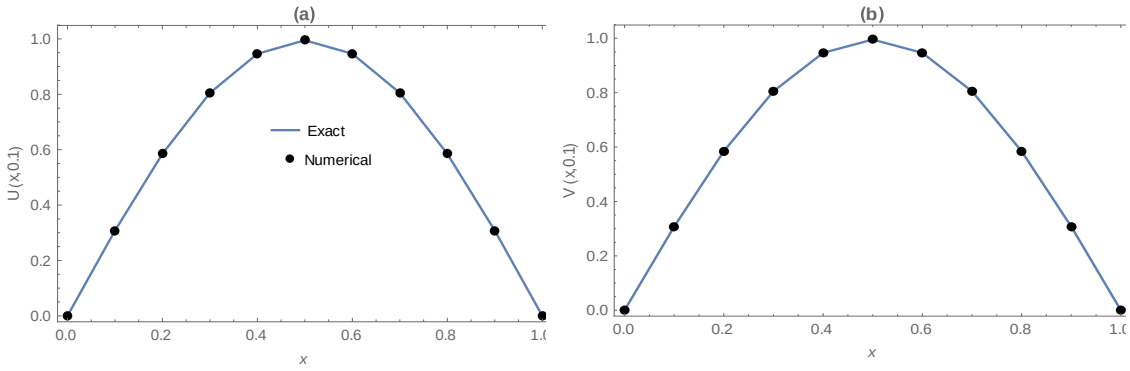


Figure 5.4.3 Graphs of exact and approximate solutions at $\Delta t = 0.0005$ and $h = 0.1$ for $U(x,t)$ part (a) and $V(x,t)$ part (b).

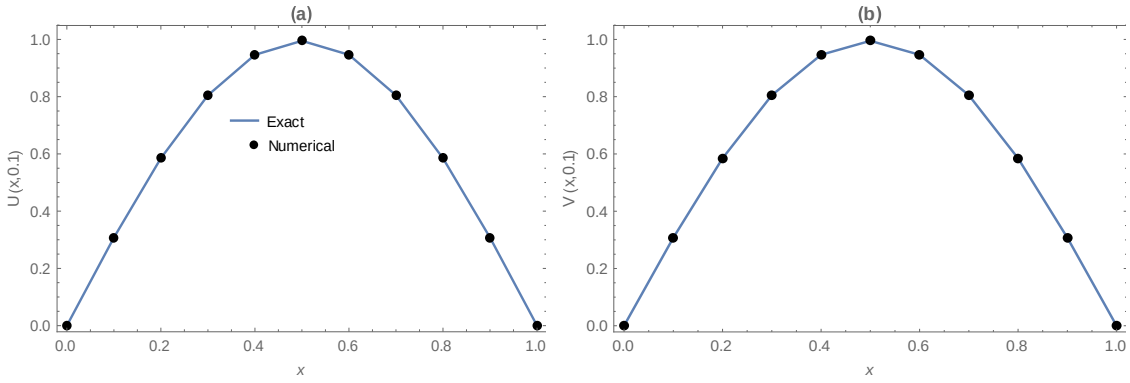


Figure 5.4.4 Graphs of exact and approximate solutions at $\Delta t = 0.00005$ and $h = 0.1$ for $U(x,t)$ part (a) and $V(x,t)$ part (b).

5.5 Conclusion

In this chapter, systematic use of the collocation analysis method was successful in identifying approximate solutions for coupled nonlinear non-homogeneous Klein–Gordon PDEs. I applied the Von Neumann stability method and found that the proposed method is conditionally stable. The basic approach employed here can be widely utilised to solve other strongly nonlinear evolution problems. I provided a numerical example to examine the accuracy and efficiency of the proposed method. It is evident from the example that the approximate solution is very close to the exact solution. The observed errors are summarised in tables to verify the stability of the

presented scheme. Its accuracy was demonstrated by calculating L_∞ error norms. The obtained numerical results show that the present method is a remarkably successful numerical technique for solving coupled nonlinear non-homogeneous Klein–Gordon equations, which makes it useful for a wide range of applications.

CHAPTER 6

Computational Analysis for Solving the Linear Space-fractional Telegraph Equation

Chapter 6: Computational Analysis for Solving the Linear Space-fractional Telegraph Equation

6.1 Introduction

During the 1980s, fractional calculus attracted the attention of many researchers and explicit applications began to appear in several fields, including physics, chemical and industrial mathematics, processing and control theory, fluid mechanics quantum evolution of complex systems, and viscoelastic mechanics [124]. In our global society, communication systems play a vital role. This, it is essential to investigate the fractional telegraph equation (FTE), which, as a typical fractional diffusion-wave equation, is applied in signal analysis for transmission, propagation of electrical signals, modelling of the reaction diffusion, random walk of suspension flows and so on [125–126].

Many researchers have recently investigated telegraph equations of fractional order. In 2008, Odibat and Momani developed a new generalisation of the 2D differential transform method (DTM), thus extending the application of the method to linear PDEs with space- and time-fractional derivatives [127]. Garg et al. (2011) applied the generalised DTM to solve a space–time FTE [128]. A year later, Zhao et al. (2012) used the fractional difference method in the temporal direction and the FEM in the spatial direction to numerically solve the time–space-fractional order telegraph equation [129]. Later, Aguilar and Baleanu, in their 2014 research, considered the fractional differential equation for a transmission line without losses in terms of the CFD [130]. In 2015, Varaki et al. used the homotropy analysis method (HAM) to obtain approximate solutions for space–time FTEs [131]. In the same year, Lopushanska and Rapita established unique solvability in an inverse problem for a semi-linear FTE with regularised fractional derivatives, with respect to time on a bounded cylindrical domain [132]. Research by Khan et al. (2018) established a new and efficient analytical scheme for a space FTE, by means of a fractional Sumudu decomposition method [133]. In the same year, Kamran et al. proposed a local meshless method, coupled with the Laplace transform, for a time FTE [134].

In 2018, L. Wei et al. introduced and analysed a flexible numerical method for the time FTE. The solution was discretised with a new finite difference scheme in time and a local discontinuous Galerkin method in space [135]. A year later, Mohammadian et al. (2019) developed the so-called generalised DTM to derive a semi-analytical solution for fractional PDEs, which involves the Riesz space-fractional derivative [136]. In the same year, Akram et al. presented and discussed a finite difference scheme based on a combination of the extended cubic B-spline method and the CFD for numerical solution of a time FTE [137]. In 2019, Kumar et al. presented a finite difference scheme for the generalised time-FTE, which was defined using generalised fractional derivative terms [138].

In 2019, research conducted by Ali et al. involved development of a new iterative method for solving the 2D hyperbolic telegraph fractional differential equation, which is central to the mathematical modelling of transmission lines satisfying a certain relationship between voltage and current waves in specific distance and time [139]. Hosseininia and Heydari in 2019 investigated a novel version of the nonlinear 2D telegraph equation involving variable-order time-fractional derivatives in the Atangana–Baleanu–Caputo sense, with a Mittag–Leffler nonsingular kernel [140]. Bouaouid et al. (2019) used the cosine family of linear operators to prove the existence, uniqueness and stability of an integral solution to a nonlocal telegraph equation in terms of the conformable time-fractional derivative. Moreover, they provided an implicit fundamental solution in terms of the classical trigonometric functions [141]. In 2020, Mohammadian et al. used the fractional reduced DTM to derive a semi-analytical solution for fractional PDEs that involve Riesz space-fractional derivatives [142]. In research by Wu and Yang (2020), a class of pure alternating segment explicit–implicit and implicit–explicit parallel difference methods were constructed for time FTEs [143]. In the same year, Hamada introduced a new model of fractional telegraph point reactor kinetics to approximate the time-dependent Boltzmann transport equation, considering new terms including the time derivative of the reactivity and fractional integral of the neutron density [144].

Recently, in 2021, Devi and Jakhar used a modified decomposition method, namely the Sumudu–Adomian decomposition method, to find exact and approximate

solutions for fractional order telegraph equations [145]. Hamza et al. (2021) discussed accurate and convergent numerical solutions for linear space–time matching telegraph fractional equations by means of a double Sumudu matching transformation method [146]. Azhar et al. (2021) studied fractional order telegraph equations via the natural transform decomposition method with nonsingular kernel derivatives. They applied natural transformation on FTEs followed by inverse natural transformation to achieve a solution to the equation [147]. Ibrahim and Bijiga (2021) presented a time FTE as an optimisation problem, with a neural network method, to solve a time FTE [148]. Vieira et al. (2021) considered the Cauchy problem for a time FTE of distributed order in Rn × R+. Employing the Fourier, Laplace and Mellin transforms, a representation of the fundamental solution to this equation in terms of convolutions involving the Fox H-function was obtained [149]. Khater et al. (2021) investigated numerical solutions to the fractional nonlinear telegraph equation by employing five numerical techniques—ADM, El Kalla, CBS, extended CBS and exponential CBS schemes—to explain the match between analytical and approximate solutions [150]. Nikan et al. (2021) focused on the numerical solution to the nonlinear time FTE, which was formulated in the Caputo sense. This model is a useful description of the neutron transport process inside the core of a nuclear reactor [151].

In this context, this chapter proposes a quadratic-polynomial spline-based method to obtain the numerical solution to a time–space-fractional order telegraph equation in the form

$$\frac{\partial^{\alpha} u}{\partial x^{\alpha}} = \frac{\partial^{2} u}{\partial t^{2}} + \frac{\partial u}{\partial t} + u \quad x > 0, \quad 1 < \alpha \leq 2, \tag{6.1.1}$$

subject to boundary conditions

$$u(a,t) = \beta_1(t), \quad u(b,t) = \beta_2(t), \quad t > 0\ . \tag{6.1.2}$$

and initial conditions

$$u(x,0) = f_1(x), \quad \frac{\partial u(x,0)}{\partial t} = f_2(x), \quad a \leq x \leq b. \tag{6.1.3}$$

The space-fractional partial derivative of order α in Eq. (6.1.1) is considered in the Caputo sense, defined by [128–129]

$$\frac{\partial^{\alpha}}{\partial x^{\alpha}}u(x,t_j)=\frac{1}{\Gamma(n-\alpha)}\int_g^x\frac{\partial^n u(s,t_j)}{\partial x^n}(x-s)^{n-\alpha-1}ds,\quad n-1\ <\alpha\leq n. \quad (6.1.4)$$

6.2 Derivation of the Method

To set up the quadratic-polynomial spline method, we select an integer $N>0$ and time step size $k>0$. With $h=\frac{b-a}{N}$, then mesh points (x_i,t_j) are $x_i=a+ih$, for each $i=0,1,\dots,N$ and $t_j=jk,\quad k=\Delta t$ for each $j=0,1,\dots$

Let ${Z_i}^j$ be an approximation to $u(x_i,t_j)$ obtained by the segment $P_i(x,t_j)$ of the spline function passing through the points (x_i,Z_i^j) and (x_{i+1},Z_{i+1}^j). Each segment has the form [40]

$$P_i(x,t_j)=a_i(t_j)\ (x-x_i)^2+b_i(t_j)\ (x-x_i)+c_i(t_j). \quad (6.2.1)$$

for each $i=0,1,\dots,N-1$. To obtain expressions for the coefficients of Eq. (6.2.1) in terms of ${Z_{i+1/2}}^j$, ${D_i}^j$, and $S_{i+1/2}^j$, we first define

$$P_i\left(x_{i+\frac{1}{2}},t_j\right)=a_i(t_j)\ (x_{i+\frac{1}{2}}-x_i)^2+b_i(t_j)\ (x_{i+\frac{1}{2}}-x_i)+c_i(t_j)$$

$$=a_i(t_j)\ (x_{\frac{1}{2}})^2+b_i(t_j)\ (x_{\frac{1}{2}})+c_i(t_j)$$

$$=a_i(t_j)\ (\frac{1}{2}h)^2+b_i(t_j)\ (\frac{1}{2}h)+c_i(t_j)$$

$$=\frac{h^2}{4}a_i+\frac{h}{2}b_i+c_i=Z_{i+1/2}^j,$$

$$P_i(x_{i+1/2},t_j)=Z_{i+1/2}^j \quad (6.2.2)$$

$$P_i^{(1)}(x,t_j)=2\ a_i(t_j)\ (x-x_i)+b_i\left(t_j\right),$$

$$P_i^{(1)}\left(x_i,t_j\right)=2\ a_i(t_j)\ (x_i-x_i)+b_i(t_j)$$

$$=b_i,$$

$$P_i^{(1)}(x_i,t_j)=D_i^j \quad (6.2.3)$$

$$P_i^{(\alpha)}(x_{i+1/2}, t_j) = \frac{\partial^\alpha}{\partial x^\alpha} P_i(x_{i+1/2}, t_j) = S_{i+1/2}^j, \quad 1 < \alpha \leq 2, \quad x_i < x_{i+1/2} \leq x_{i+1}. \text{ (6.2.4)}$$

where $a_i \equiv a_i(t_j),\ b_i \equiv b_i(t_j),\ c_i \equiv c_i(t_j),\ d_i \equiv d_i(t_j)$ and $\theta = \omega h$. Eqs. (6.2.1), (6.2.2) and (6.2.3), give

$$\frac{h^2}{4} a_i + \frac{h}{2} b_i + c_i = Z_{i+1/2}^j. \tag{6.2.5}$$

$$b_i = D_i^j. \tag{6.2.6}$$

Using Eqs. (6.1.4), (6.2.1) and (6.2.4), we obtain

$$\frac{\partial^\alpha}{\partial x^\alpha} u(x_{i+1/2}, t_j) = \frac{1}{\Gamma(2-\alpha)} \int_{x_i}^{x_{i+1/2}} \frac{\partial^2 P_i(s,t_j)}{\partial x^2} \left(x_{i+1/2} - s\right)^{1-\alpha} ds = S_{i+1/2}^j.$$

This equation can be simplified as

$$\frac{\partial P_i(x, t_j)}{\partial x} = 2a_i(x - x_i),$$

$$\frac{2^2 P_i(x, t_j)}{\partial x^2} = 2a_i \text{ , then}$$

$$\frac{1}{\Gamma(2-\alpha)} \int_{x_i}^{x_{i+\frac{1}{2}}} (2a_i) \left(x_{i+\frac{1}{2}} - s\right)^{1-\alpha} ds$$

$$= \frac{-2a_i}{\Gamma(2-\alpha)} \left[\frac{\left(x_{i+\frac{1}{2}} - s\right)^{2-\alpha}}{2-\alpha} \right]_{x_i}^{x_i+\frac{1}{2}}$$

$$= \frac{-2a_i}{\Gamma(2-\alpha)} \left[- \frac{\left(x_{i+\frac{1}{2}} - x_i\right)^{2-\alpha}}{2-\alpha} \right]$$

$$= \frac{2ai}{\Gamma(2-\alpha)} \cdot \frac{(h)^{2-\alpha}}{2-\alpha}$$

$$= \frac{2}{\Gamma(3-\alpha)} (h)^{2-\alpha} (a_i),$$

$$where\ \Gamma(3-\alpha) = (2-\alpha)\Gamma(2-\alpha)$$

$$\mu\ a_i = S^j_{i+1/2} \tag{6.2.7}$$

where $\mu = \frac{2}{\Gamma(3-\alpha)}\left(\frac{h}{2}\right)^{2-\alpha}$. By solving Eqs. (6.2.5), (6.2.6) and (6.2.7), we obtain the following expressions:

$$a_i = \frac{\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha-2} S^j_{i+1/2},$$

$$b_i = D^j_i,$$

$$c_i \quad = z^j_{i+\frac{1}{2}} - \frac{h^2}{4}a_i - \frac{h}{2}b_i$$

$$= -\frac{h^2}{4}\left(\frac{\Gamma(3-\alpha)}{2}h^{(2-\alpha)}s_{i+\frac{1}{2}}\right) - \frac{h}{2}D^j_i + Z^j_{i+\frac{1}{2}}$$

$$= Z^j_{i+\frac{1}{2}} - \frac{h}{2}D^j_i - \frac{1}{2}\,\Gamma(3-\alpha)\left(\frac{h}{2}\right)^{\alpha} s_{i+\frac{1}{2}}$$

Then,

$$c_i = -\frac{1}{2}\Gamma(3-\alpha)\left(\frac{h}{2}\right)^{\alpha} S^j_{i+1/2} - \frac{h}{2}D^j_i + Z^j_{i+1/2}. \tag{6.2.8}$$

6.3 Spline Relationships

I use the following continuity conditions at $x = x_i$:

$$P_i(x_i, t_j) = P_{i-1}(x_i, t_j) \Rightarrow c_i = h^2 a_{i-1} + h b_{i-1} + c_{i-1}. \tag{6.3.1}$$

$$P^{(1)}_i(x_i, t_j) = P^{(1)}_{i-1}(x_i, t_j) \Rightarrow b_i = 2h a_{i-1} + b_{i-1} \tag{6.3.2}$$

Using the expressions in Eq. (6.2.8), Eqs. (6.3.1) and (6.3.2) become

$$Z^j_{i+\frac{1}{2}} - \frac{h}{2}D^j_i - \frac{1}{2}\,\Gamma(3-\alpha)\left(\frac{h}{2}\right)^{\alpha} s_{i+\frac{1}{2}} =$$

$$h^2\,\frac{\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha-2} S^j_{i-\frac{1}{2}} + h\,D^j_{i-1} - \frac{1}{2}\Gamma(3-\alpha)\left(\frac{h}{2}\right)^{\alpha} S^j_{i-\frac{1}{2}} - \frac{h}{2}D^j_{i-1} + Z^j_{i-\frac{1}{2}}$$

$$Z^j_{i+\frac{1}{2}} - Z^j_{i-\frac{1}{2}} - \frac{1}{2}\,\Gamma(3-\alpha)\left(\frac{h}{2}\right)^{\alpha} s_{i+\frac{1}{2}} + \frac{1}{2}\Gamma(3-\alpha)\left(\frac{h}{2}\right)^{\alpha} S^j_{i-\frac{1}{2}} - \frac{h}{2}D^j_i + \frac{h}{2}D^j_{i-1} =$$

$$h^2\,\frac{\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha-2} S^j_{i-\frac{1}{2}} + h\,D^j_{i-1}$$

$$Z^j_{i+\frac{1}{2}} - Z^j_{i-\frac{1}{2}} - \frac{1}{2}\,\Gamma(3-\alpha)\left(\frac{h}{2}\right)^{\alpha}\left(s_{i+\frac{1}{2}} - s_{i-\frac{1}{2}}\right) - \frac{h}{2}\left(D^j_i - D^j_{i-1}\right) =$$

$$\left(\frac{h}{2}\right)^2 (2)^2\,\frac{\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha-2} S^j_{i-\frac{1}{2}} + h\,D^j_{i-1}$$

Then,

$$Z^j_{i+1/2} - Z^j_{i-1/2} - \frac{\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha}\left(S^j_{i+1/2} - S^j_{i-1/2}\right) - \frac{h}{2}\left(D^j_i - D^j_{i-1}\right) =$$

$$\frac{4\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha} S^j_{i-1/2} + hD^j_{i-1} \tag{6.3.3}$$

$$D^j_i - D^j_{i-1} = \frac{4\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha-1} S^j_{i-1/2} \tag{6.3.4}$$

Solving for D^j_{i-1} results in

$$hD^j_{i-1} = \left(Z^j_{i+1/2} - Z^j_{i-1/2}\right) - \frac{\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha} S^j_{i+\frac{1}{2}} - \frac{7\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha} S^j_{i-\frac{1}{2}}. \tag{6.3.5}$$

Similarly,

$$hD^j_i = \left(Z^j_{i+3/2} - Z^j_{i+1/2}\right) - \frac{\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha} S^j_{i+3/2} - \frac{7\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha} S^j_{i+1/2.} \tag{6.3.6}$$

Using expressions in Eqs. (6.3.5) and (6.3.6), Eq. (6.3.4) becomes

$$Z^j_{i+3/2} - 2Z^j_{i+\frac{1}{2}} + Z^j_{i-\frac{1}{2}} = \delta(S^j_{i+\frac{3}{2}} + 6S^j_{i+\frac{1}{2}} + S^j_{i-\frac{1}{2}}), \quad i = 1,2,\dots,N-2, \tag{6.3.7}$$

where

$$\delta = \frac{\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha}.$$

As $\alpha \to 2$, system (6.3.7) reduces to

$$Z_{i+3/2}^{j} - 2Z_{i+1/2}^{j} + Z_{i-1/2}^{j} = \frac{h^2}{8}(S_{i+3/2}^{j} + 6S_{i+1/2}^{j} + S_{i-1/2}^{j}), \quad i = 1,2,\dots,N-2. \tag{6.3.8}$$

Remark:

The truncation error for Eq. (6.3.7); that is,

$$T^{*}{}_{i}^{j} = \left(u_{i-1/2}^{j} + u_{i+3/2}^{j}\right) - 2u_{i+1/2}^{j} - \delta\left(D_x^2 u_{i-1/2}^{j} + D_x^2 u_{i+3/2}^{j}\right) - 6\delta D_x^2 u_{i+1/2}^{j}$$

can be obtained by expanding this equation in a Taylor series in terms of $u(x_{i+1/2}, t_j)$ and its derivatives as follows:

$$T^{*}{}_{i}^{j} = (h^2 - 8\delta)D_x^2 u_{i+1/2}^{j} + \left(\frac{h^4}{12} - \delta h^2\right)D_x^4 u_{i+1/2}^{j} + \left(\frac{h^6}{360} - \frac{\delta h^4}{12}\right)D_x^6 u_{i+1/2}^{j} + \cdots.$$

Since $\delta = \frac{\Gamma(3-\alpha)}{2}\left(\frac{h}{2}\right)^{\alpha}$, the last expression can be simplified as

$$T^{*}{}_{i}^{j} = h^{\alpha}(h^{2-\alpha} - 8\theta)D_x^2 u_{i+1/2}^{j} + h^{2+\alpha}\left(\frac{h^{2-\alpha}}{12} - \theta\right)D_x^4 u_{i+1/2}^{j} +$$

$$h^{4+\alpha}\left(\frac{h^{2-\alpha}}{360} - \frac{\theta}{12}\right)D_x^6 u_{i+1/2}^{j} + \cdots,$$

where $\theta = \frac{\Gamma(3-\alpha)}{2^{\alpha+1}}$. From this expression of the local truncation error, our scheme is

$$O(h^{\alpha}),\ 1 < \alpha \leq 2$$

$$S_i^j = \frac{\partial^{\alpha} Z_i^j}{\partial x^{\alpha}} = \frac{\partial^2 Z_i^j}{\partial t^2} + \frac{\partial Z_i^j}{\partial t} + Z_i^j \tag{6.3.9}$$

$$S_i^j = \frac{\partial^{\alpha} Z_i^j}{\partial x^{\alpha}} \approx \frac{Z_i^{j+1} - 2Z_i^j + Z_i^{j-1}}{k^2} + \frac{Z_i^{j+1} - Z_i^{j-1}}{2k} + Z_i^j, \tag{6.3.10}$$

which can be discretised as follows:

$$S_{i-1/2}^{j} = \frac{\partial^{\alpha} Z_{i-1/2}^{j}}{\partial x^{\alpha}} \approx \frac{Z_{i-1/2}^{j+1} - 2Z_{i-1/2}^{j} + Z_{i-1/2}^{j-1}}{k^2} + \frac{Z_{i-1/2}^{j+1} - Z_{i-1/2}^{j-1}}{2k} + Z_{i-1/2}^{j}$$

$$S_{i+1/2}^{j} = \frac{\partial^{\alpha} Z_{i+1/2}^{j}}{\partial x^{\alpha}} \approx \frac{Z_{i+1/2}^{j+1} - 2Z_{i+1/2}^{j} + Z_{i+1/2}^{j-1}}{k^2} + \frac{Z_{i+1/2}^{j+1} - Z_{i+1/2}^{j-1}}{2k} + Z_{i+1/2}^{j} \quad (6.3.11)$$

$$S_{i+3/2}^{j} = \frac{\partial^{\alpha} Z_{i+3/2}^{j}}{\partial x^{\alpha}} \approx \frac{Z_{i+3/2}^{j+1} - 2Z_{i+3/2}^{j} + Z_{i+3/2}^{j-1}}{k^2} + \frac{Z_{i+3/2}^{j+1} - Z_{i+3/2}^{j-1}}{2k} + Z_{i+3/2}^{j}$$

Using the formulae in (6.3.11) in (6.3.7) gives the following useful systems:

$$A\ Z_{i-\frac{1}{2}}^{j+1} + B\ Z_{i+\frac{1}{2}}^{j+1} + A\ Z_{i+\frac{3}{2}}^{j+1} = A^{*} Z_{i-\frac{1}{2}}^{j} +$$

$$B^{*} Z_{i+1/2}^{j} + A^{*} Z_{i+3/2}^{j} + \hat{A}\ Z_{i-1/2}^{j-1} + \hat{B}\ Z_{i+1/2}^{j-1} + \hat{C}\ Z_{i+3/2,}^{j-1} \qquad (6.3.12)$$

where

$$A = \frac{\delta}{k^2} + \frac{\delta}{2k}, A^{*} = 1 + \frac{2\delta}{k^2} - \delta, \hat{A} = \frac{-\delta}{k^2} + \frac{\delta}{2k},$$

$$B = \frac{6\delta}{k^2} + \frac{3\delta}{k}, B^{*} = -2 + \frac{12\delta}{k^2} - 6\delta \text{ and } \hat{B} = \frac{-6\delta}{k^2} + \frac{3\delta}{k}. \qquad (6.3.13)$$

System (6.3.12) consists of *N−2* equations in *N* unknowns. To obtain a solution to this system, we need two additional equations. Using the boundary conditions (6.1.2), that are $Z_0^j = \beta_1(t), Z_{N+1}^j = \beta_2(t)$, we obtain the following equation, supposing that $Z_{1/2}^j$ is linearly interpolated between Z_0^j and $Z_{3/2}^j$:

$$-3Z_{1/2}^{j} + Z_{3/2}^{j} = -2Z_0^{j} = -2\beta_1, \quad j \geq 0. \qquad (6.3.14)$$

In a similar manner.

$$Z_{N-3/2}^{j} - 3Z_{N-1/2}^{j} = -2Z_N^{j} = -2\beta_2, \quad j \geq 0. \qquad (6.3.15)$$

Eq. (6.3.12) implies that the *(j+1)*st time step requires values from the *(j)*st and *(j-1)*st time steps. This produces a minor starting problem, since values for *j=0* are given by the first part in Eq. (6.1.3):

$$Z_i^0 = u(x_i, 0) = f_1(x_i), \quad i = 1, \ldots, N. \qquad (6.3.16)$$

but values for *j=0*, which are needed in Eq. (6.3.13) to compute Z_i^1, must be obtained from the first part in (6.1.3):

$$\frac{\partial Z_i^0}{\partial t} = u_t(x_i, 0) = f_2(x_i), \quad i = 1, \ldots, N.$$

One approach is to replace $\frac{\partial Z_i^0}{\partial t}$ by a forward-difference approximation:

$$f_2(x_i) = \frac{\partial Z_i^0}{\partial t} = \frac{Z_i^1 - Z_i^0}{k} + o(k), \tag{6.3.17}$$

which gives

$$Z_i^1 = Z_i^0 + k f_2(x_i), \qquad i = 1, \ldots., N. \tag{6.3.18}$$

6.4 Stability Analysis

The Von Neumann technique is used to investigate the stability of systems (6.3.11) and (6.3.12). Key to the Von Neumann analysis is the assumption of a solution in the following form:

$$Z_i^j = \zeta_j e^{(q\emptyset h)}, \tag{6.4.1}$$

Where ϕ is the wave number, $q = \sqrt{-1}$, h is the element size and ζ is the amplification factor of the scheme. The use of Eqs. (6.4.1) and (6.3.12) provides a characteristic equation in the form

$$\zeta^{j+1}\{ A\, e^{((i-1)q\phi h)} + B\ e^{(iq\phi h)} + A\, e^{((i+1)q\emptyset h)}\} =$$

$$\zeta^{j}\{A^*\, e^{((i-1)q\phi h)} + B^*\, e^{(iq\phi h)} + A^* e^{((i+1)q\emptyset h)}\} +$$

$$\zeta^{j-1}\{\hat{A}\, e^{((i-1)q\phi h)} + \hat{B}\ e^{(iq\phi h)} + \hat{A}\ e^{((i+1)q\emptyset h)}\}. \tag{6.4.2}$$

By dividing both sides of the last equation by $e^{(iq\emptyset h)}$ and cancelling the common term, which isζ^{j-1}, Eq. (6.4.2) becomes

$$\zeta^{2}\{ A\, e^{(-q\phi h)} + B\ + A\ e^{(q\emptyset h)}\} -$$

$$\zeta\left\{A^{*}\, e^{(-q\phi h)} + B^{*} + A^{*}\, e^{(q\emptyset h)}\right\} -$$

$$\left\{\hat{A}\, e^{(-q\phi h)} + \hat{B}\ + \hat{A}\ \ e^{(q\emptyset h)}\right\} = 0, \qquad (6.4.3)$$

where

$$A = \frac{\delta}{k^2} + \frac{\delta}{2k}, A^{*} = 1 + \frac{2\delta}{k^2} - \delta, \hat{A} = \frac{-\delta}{k^2} + \frac{\delta}{2k}$$

$$B = \frac{6\delta}{k^2} + \frac{3\delta}{k}, B^{*} = -2 + \frac{12\delta}{k^2} - 6\delta, \text{ and } \hat{B} = \frac{-6\delta}{k^2} + \frac{3\delta}{k}.$$

This equation can be rewritten in the simple form

$$a\zeta^2 \quad + b\zeta + c = 0 \qquad (6.4.4)$$

where

$$a = \left(A\, e^{(-q\phi h)} + B\ + A\ e^{(q\phi h)}\right),\ b = -\left(A^{*}\, e^{(-q\phi h)} + B^{*} + A^{*} e^{(q\phi h)}\right),$$

$$and\ c = -\left(\hat{A}\, e^{(-q\phi h)} + \hat{B}\ + \hat{A}\ e^{(q\phi h)}\right)$$

or

$$a = B + 2A\cos\varphi\,, b = -B^{*} - 2A^{*}\cos\varphi\quad, c = -\hat{B} - 2\hat{A}\cos\varphi\,, \varphi = h\phi$$

or

$$a = \frac{\delta}{k}(3 + cos\,\varphi)\left(\frac{2}{k} + 1\right), b = 2(1 - cos\,\varphi) - 2\delta(3 + cos\,\varphi)\left(\frac{2}{k^2} - 1\right),$$

$$c = \frac{\delta}{k}(3 + cos\,\varphi)\left(\frac{2}{k} - 1\right)$$

Eq. (6.4.4) is a quadratic in ζ and thus it will have two roots; that is:

$$\varsigma_{\pm} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$\varsigma_{\pm} = \sqrt{\frac{c}{a}}\left(-\psi \pm \sqrt{\psi^2 - 1}\right), \psi = \frac{b}{2\sqrt{ac}}$$

Stability requires that $\left|\zeta_{\pm}\right| \leq 1$. So, we have three cases.

Case 1: The discriminant of the quadratic Eq. (6.4.4) is zero; that is, $\psi^2 - 1 = 0$, in which case

$$\varsigma_{\pm} = \pm\sqrt{\frac{c}{a}} = \pm\sqrt{\frac{2-k}{2+k}},\ 0 < k < 1$$

and the stability condition, $|\zeta_{\pm}| \leq 1$, is satisfied.

Case 2: The discriminant is less than zero; that is $\psi^2 - 1 < 0$, in which case

$$\varsigma_{\pm} = \sqrt{\frac{c}{a}}\left(-\psi \pm q\sqrt{1-\psi^2}\right) = \sqrt{\frac{2-k}{2+k}}\left(-\psi \pm q\sqrt{1-\psi^2}\right) \Rightarrow$$

the stability condition, $|\zeta_{\pm}| \leq 1$, is satisfied.

Case 3: The discriminant is greater than zero. This means that either ζ_+ or ζ_- does not satisfy the stability condition. Thus, for stability we must have $\psi^2 - 1 \leq 0$:

$$1 \leq \psi \leq 1 \tag{6.4.5}$$

$$1 \leq \frac{b}{2\sqrt{ac}} \leq 1$$

Since $\sqrt{ac} > 0 \Rightarrow -2\sqrt{ac} \leq b \leq 2\sqrt{ac}$,

$$-\frac{2\delta}{k^2}(3 + cos\,\varphi)\sqrt{4-k^2} \leq 2(1 - cos\,\varphi) - 2\delta(3 + cos\,\varphi)\left(\frac{2}{k^2} - 1\right) \leq$$

$$\frac{2\delta}{k^2}(3 + cos\,\varphi)\sqrt{4-k^2}.$$

The right side in the above inequality takes the form

$$2(1 - cos\,\varphi) \leq \frac{2\delta}{k^2}(3 + cos\,\varphi)\left(\sqrt{4-k^2} + 2 - k^2\right),$$

which is satisfied for $k << \mathcal{S}$ where h is small enough.

However, the left side of the above inequality takes the form

$$-2(1 - cos\,\varphi) \leq \frac{2\delta}{k^2}(3 + cos\,\varphi)\left(\sqrt{4-k^2} - 2 + k^2\right),$$

which is satisfied for $k << \delta$ where h is small enough, and the method is thus conditionally stable.

6.5 Numerical Example

In this section, a numerical example is included to illustrate the practical implementation of the proposed method. Consider the following linear space FTE [127]:

$$\frac{\partial^{1.5}u}{\partial x^{1.5}} = \frac{\partial^2 u}{\partial t^2} + \frac{\partial u}{\partial t} + u, x > 0. \tag{6.5.1}$$

subject to the initial condition

$$u(x, 0) = 0, \tag{6.5.2}$$

boundary conditions

$$u(0.0125, t) \approx exp(-t)\,(1 + 0.0125) + \frac{0.0125^{1.5}}{\Gamma(5/2)} + \frac{0.0125^{2.5}}{\Gamma(7/2)} + \frac{0.0125^{3}}{\Gamma(4)} + \frac{0.0125^{4}}{\Gamma(5)} + \ldots \tag{6.5.3}$$

and

$$u(1.0125, t) \approx exp(-t)\,(1 + 1.0125) + \frac{1.0125^{1.5}}{\Gamma(5/2)} + \frac{1.0125^{2.5}}{\Gamma(7/2)} + \frac{1.0125^{3}}{\Gamma(4)} + \frac{1.0125^{4}}{\Gamma(5)} + \ldots . \tag{6.5.4}$$

The exact solution is

$$u(x, t) \approx exp(-t)\left(1 + x + \frac{x^{1.5}}{\Gamma(5/2)} + \frac{x^{2.5}}{\Gamma(7/2)} + \frac{x^{3}}{\Gamma(4)} + \frac{x^{4}}{\Gamma(5)} + \ldots\right). \tag{6.5.5}$$

Tables 6.5.1–6.5.3 illustrate the comparison between the proposed method, developed in Section 6.4 and existing methods [127] and [145] with $k = 0.00005, h = 0.025, t = 0.05,0.1,0.15$ and $\alpha = 1.5$.

Tables 6.5.4–6.5.5 compare the method developed in Section 6.4, with existing methods [127] and [145] with $k = 0.00005, h = 0.025, t = 0.05,0.1$, and $\alpha = 1.75$.

Table 6.5.1 Comparison between the proposed numerical method and methods [127] and [145] when *t*=0.05, *k*=0.000005, and *h*=0.025 and $\alpha = 1.5$.

x	Proposed Method	Methods [127] and [145]
0.1	1.0689295078552934	1.0700487208006241
0.2	1.2105809555003169	1.2119422776213813
0.3	1.3713891366890514	1.3729692926753612
0.4	1.5513544745941052	1.5531657691651113
0.5	1.7514676554067925	1.7535302292267538
0.6	1.9732140327892282	1.9755523719324375
0.7	2.2184148686049125	2.2210569535598963
0.8	2.4891614801652615	2.4921386344095513
0.9	2.7877830486712691	2.7911317256723061
1.0	3.1196537361432172	3.1205956925765412

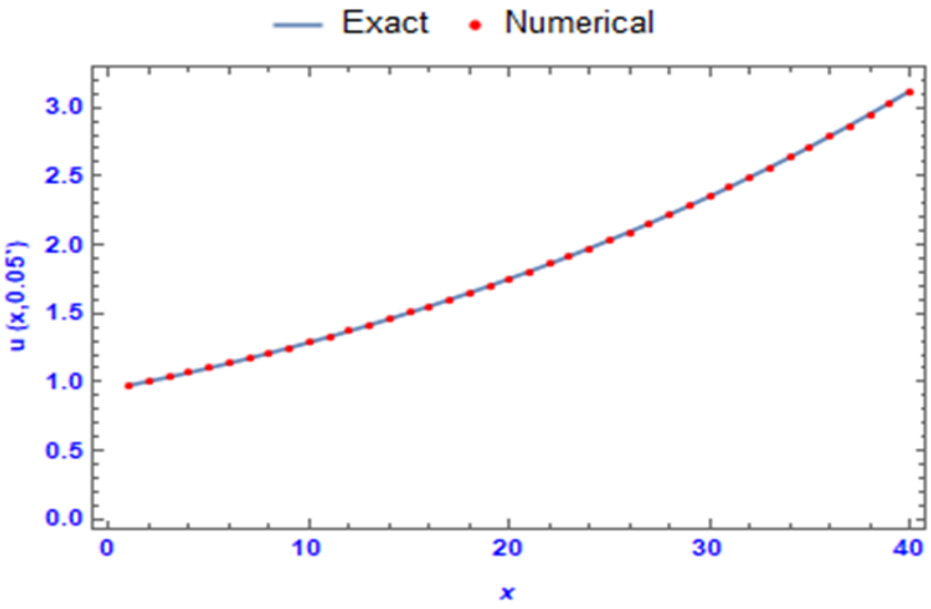


Figure 6.5.1 Comparison between the proposed method and method [127] and [145] when *t*=0.05, *k*=0.000005, and *h*=0.025 and $\alpha = 1.5$.

Table 6.5.2 Comparison between the proposed numerical method and methods [127] and [145] when t=0.1, k=0.000005, and h=0.025 and $\alpha = 1.5$.

x	Proposed Method	Methods [127] and [145]
0.1	1.0139125288963764	1.0178618288749026
0.2	1.1480559941970572	1.1528351552698712
0.3	1.3004476445553682	1.3060087901287358
0.4	1.4710342588062175	1.4774169807571376
0.5	1.6607354930479643	1.6680095507919713
0.6	1.8709521529921287	1.8792035458243128
0.7	2.1034072173406435	2.1127347277180891
0.8	2.3600811054248516	2.3705955989853926
0.9	2.6431880681647740	2.6550066251169517
1.0	2.9649893224621964	2.9684024447489910

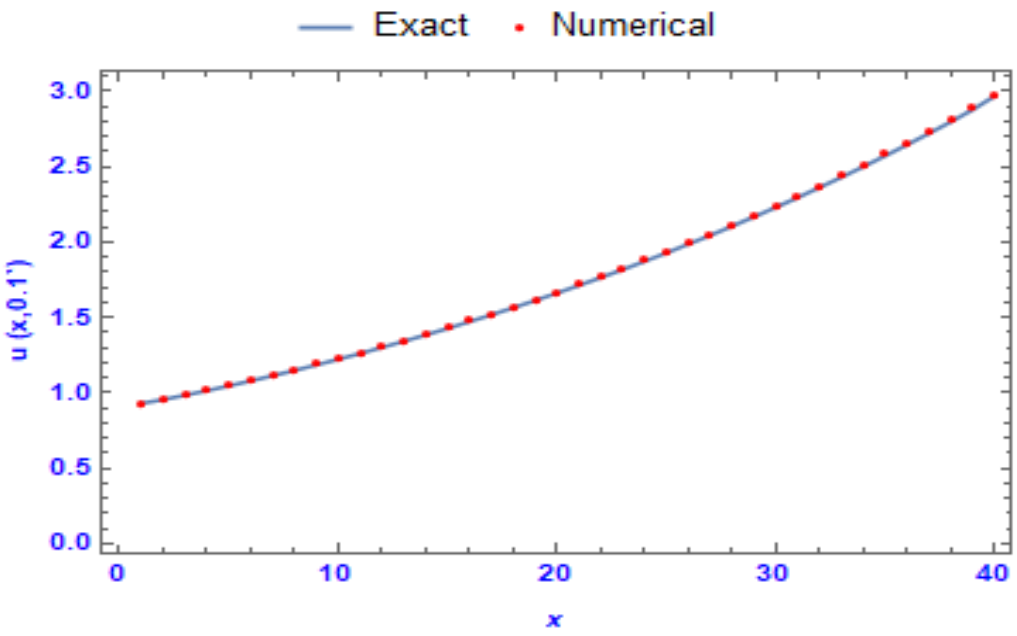


Figure 6.5.2 Comparison between the proposed method and method [127] and [145] when t=0.1, k=0.000005, and h=0.025 and $\alpha = 1.5$.

Table 6.5.3 Comparison between the proposed numerical method and methods [127] and [145] when t=0.15, k=0.000005, and h=0.025 and $\alpha = 1.5$.

x	Proposed Method	Methods [127] and [145]
0.1	0.9596371905024942	0.9682201217019183
0.2	1.0862170007301811	1.0966107212915508
0.3	1.2302168726205736	1.2423139898270312
0.4	1.3914776194134673	1.4053625043531945
0.5	1.5708357143835627	1.5866597650615402
0.6	1.7696035239597443	1.7875537074141623
0.7	1.9894042815310826	2.0096954391699512
0.8	2.2321069494266474	2.2549802873468003
0.9	2.4998194340024652	2.5255204240555812
1.0	2.8167740246016137	2.8236317492050944

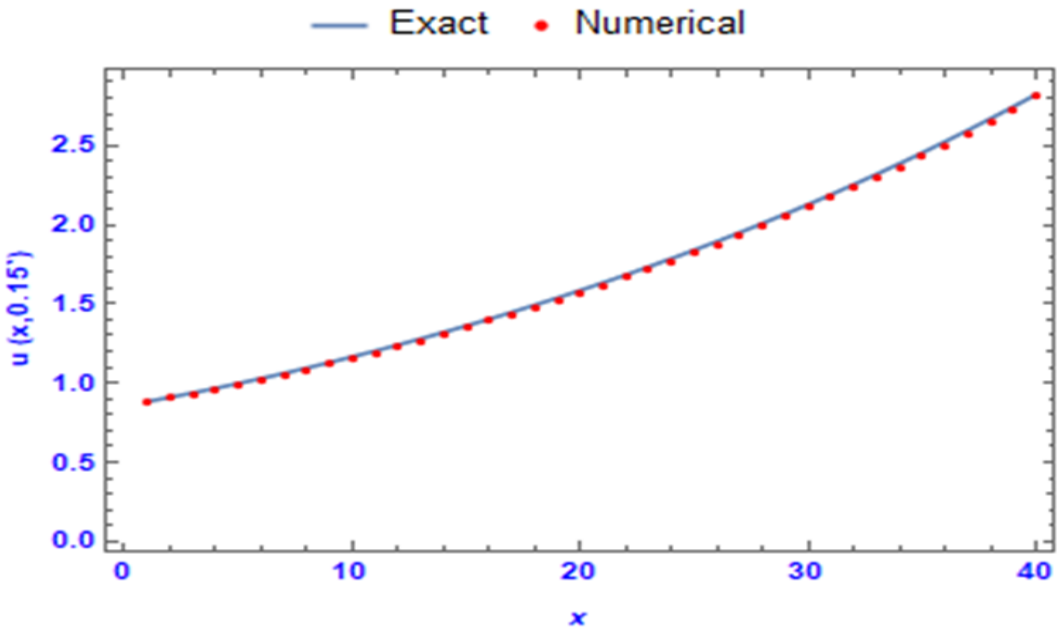


Figure 6.5.3 Comparison between the proposed method and methods [127] and [145] when t=0.15, k=0.000005, and h=0.025 and $\alpha = 1.5$.

Table 6.5.4 Comparison between the proposed numerical method and methods [127] and [145] when t=0.05, k=0.00005, and h=0.025, $\alpha = 1.75$.

x	Proposed Method	Methods [127] and [145]
0.1	0.9591480315753628	1.0572785260392232
0.2	1.0867445813092063	1.1797304399209085
0.3	1.2307489176630863	1.3176660343983828
0.4	1.3920372585030207	1.4716641084951654
0.5	1.5714366396546564	1.6428130928171421
0.6	1.7702550524339082	1.8325254529378252
0.7	1.9901139102638157	2.0424754500651385
0.8	2.2328814918984262	2.2745763567746264
0.9	2.5005404809568174	2.5309742073637795
1.0	2.8087858338014686	2.8140496901826731

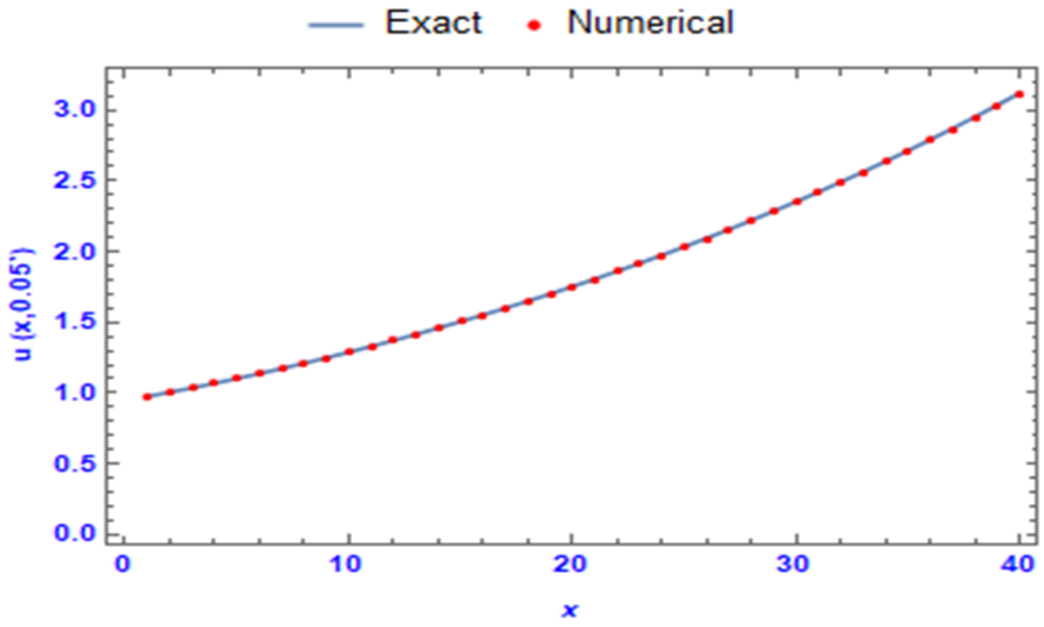


Figure 6.5.4 Comparison between the proposed numerical method and methods [127] and [145] when t=0.05, k=0.00005, and h=0.025, $\alpha = 1.75$.

Table 6.5.5 Comparison between the proposed numerical method and methods [127] and [145] when t=0.1, k=0.00005, and h=0.025, $\alpha = 1.75$.

x	Proposed Method	Methods [127] and [145]
0.1	0.9597673920033094	1.0057144438612532
0.2	1.0867461330050554	1.1221943074319412
0.3	1.2307489144320851	1.2534027035849116
0.4	1.3920372584910621	1.3998902029822125
0.5	1.5714366396549502	1.5626921528426882
0.6	1.7702550524472353	1.7431521319809578
0.7	1.9901139126215512	1.9428627469222985
0.8	2.2328811392604192	2.1636439588376586
0.9	2.4997581380145126	2.4075371386967985
1.0	2.6931301646725034	2.6768068673088763

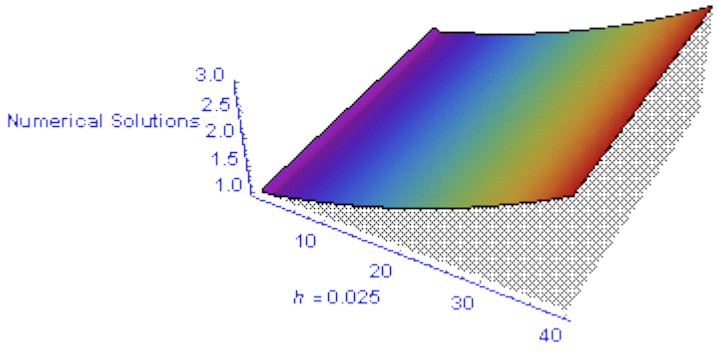


Figure 6.5.5 The 3D behaviour of the numerical solutions from t=0.0005 to t=0.05, k=0.0005, and h=0.025, $\alpha = 1.75$.

Tables 6.5.1–6.5.5 and Figures 6.5.1–6.5.5 demonstrate that the obtained approximate numerical solutions are in good agreement with the approximate solutions obtained using methods [127] and [145] for all values of x and t.

6.6 Conclusion

In this chapter, a numerical treatment was proposed for a linear space FTE using quadratic-polynomial splines. Through application of the Von Neumann stability analysis, the developed method was shown to be conditionally stable. The numerical example worked well and confirmed the theoretical analysis, which demonstrates that the numerical scheme is effective and reliable for the time–space-fractional order telegraph equation. The approximate numerical solutions obtained are in good agreement with the approximate solutions obtained using method [127] and [145] for all values of *x* and *t*. The proposed scheme for local truncation error is of $O(h^{\alpha}), 1 < \alpha \leq 2$, and it can be concluded that this technique is both very powerful and efficient for finding the approximate solutions for a large class of linear PDEs of fractional order.

CHAPTER 7

Adaptive Boundary Control for the Dynamics of the Generalised Burgers–Huxley Equation

Chapter 7: Adaptive Boundary Control for the Dynamics of the Generalised Burgers–Huxley Equation

7.1 Introduction

NPDEs have been widely studied by researchers over the years and have since become ubiquitous in nature [152]. Exact solutions rarely exist for NPDEs; thus, there has been much attention devoted recently to the search for better and more efficient methods for determining a solution—approximate or exact, analytical or numerical—to nonlinear models [153]. Of the plethora of NPDEs, the Burgers–Huxley equation is finding an increasing number of useful applications in different fields. The Burgers–Huxley equation is a well-known NPDE that simulates nonlinear wave phenomena in physics, biology, economics and ecology [154]. It finds application in many fields such as biology, nonlinear acoustics, metallurgy, chemistry, combustion, mathematics and engineering, as per Satsuma et al. [155]. It is a special type of nonlinear advection-diffusion reaction problem that is of importance in applications in mechanical engineering, material sciences and neurophysiology. Examples include particle transport, wall motion in liquid crystals [156], dynamics of ferroelectric materials [157] and action potential propagation in nerve fibres [158]. Further, some of these reaction processes involve fascinating phenomena such as busting oscillation, population genetics and bifurcation [159–164].

The GBHE model has application in relation to propagating signals in the nervous system, elasticity, gas dynamics and heat conduction [165]. The Burgers–Huxley equation was first introduced to describe turbulence in one space dimension, and has been used in several other physical contexts; for instance sound waves in viscous media [166].

Many methods have been developed to solve the Burgers–Huxley equation, including the ADM [167–169]. El-Danaf discussed some analytic properties of the GBHE such as the translation property and the steady state solution of the equation [170]. Using the first integral method, Xijun Deng studied travelling wave solutions to the GBHE in 2008 [171]. A year later, the HAM was applied to obtain approximate analytical

solutions for the GBHE and the Huxley equation, by Sami Bataineh et al. [172]. In 2010, Smaoui et al. designed three adaptive control laws for the forced generalised Korteweg–de Vries–Burgers (GKdVB) equation when either the kinematic viscosity, dynamic viscosity or both were unknown [173]. In the same year, Biazar and Mohammadi applied the DTM to the GBHE and some special cases of the equation such as the Huxley and Fitzhugh–Nagoma equations [174]. Bratsos, in his 2011 research, proposed an implicit finite difference scheme based on fourth-order rational approximants to the matrix exponential term for the numerical solution of the Burgers–Huxley equation [175]. Macías-Díaz et al. (2011) developed a non-standard finite difference scheme to approximate the solution of the GBHE from fluid dynamics [176]. In 2013, El-Kady et al. introduced treatments for the GBHE that were dependent on cardinal Chebyshev and Legendre basis functions with the Galerkin method [177]. In the same year, Ray and Gupta solved the GBHE and Huxley equations using the Haar wavelet method [178]. Liu et al. (2013) used the double exp-function method to obtain a two-soliton solution for the GBHE [179]. A year later, Emad applied a relatively new semi-analytic technique, the reduced DTM to solve the GBHE and some special cases [180]. In 2015, Ervin et al. published a paper outlining a finite element scheme capable of preserving the non-negative and bounded solutions of the GBHE [181]. Inan (2016) applied an implicit exponential finite difference method to compute the numerical solutions of the nonlinear generalised Huxley equation [182]. Kumar and Singh proposed a numerical scheme for solution of the GBHE using an improved nodal integral method in 2016 [183]. In the same year, Machado et al. introduced an algorithm based on adopting the approximate analytical solution of the Cauchy problem for the Burgers–Huxley equation [184]. In 2017, Inan presented an explicit exponential finite difference method to solve the generalised forms of the Huxley and Burgers–Huxley equations [185]. In 2018, Wasim et al. introduced a new numerical technique for solving nonlinear generalised Burgers–Fisher and Burgers–Huxley equations using the hybrid B-spline collocation method [186]. Appadu et al. (2019) obtained numerical solutions to the Burgers–Huxley equation with specified initial and boundary conditions using two novel non-standard finite difference schemes and two exponential finite difference schemes [187]. In the same year, Fu discussed the

persistence of travelling wavefronts in a GBHE with long-range diffusion [188]. A year later, Sun and Zhu developed a kind of CBS quasi-interpolation, which was used to solve Burgers–Huxley equations [189]. In 2020, Khan et al. demonstrated how to use the new auxiliary method for solitary wave solutions of the GBHE [190]. Kumar and Mohan in 2020 introduced an analytical global solvability as well as asymptotic analysis of the stochastic GBHE perturbed by space–time white noise in a bounded interval of R [191]. Kushner (2020) constructed similar dynamics for the classical Burgers–Huxley equation and then used them to construct new exact solutions [192]. More recently, Ebiwareme (2021) proposed the tanh–coth and Banach contraction methods to solve the Burgers–Huxley and Kuramoto–Sivashinsky (KS) equations [193]. In the same year, Mohan and Khan considered the forced GBHE and established the existence an uniqueness of a global weak solution using a Faedo–Galerkin approximation method [194].

Many researchers have worked on the control problems for the Burgers, KS, KdV and KdVB equations (see [195]–[198]). In [199]–[201], the authors obtained a nonlinear robust boundary control for the KS equation and a nonlinear robust stabilisation for the GKdVB using the boundary control. In [202] and [203], Smaoui et al. obtained a nonlinear boundary control for the generalised Burgers and GKdVB equations. In [204] and [205], Smaoui et al. controlled the dynamics of the Burgers and GKdVB equations using an adaptive boundary control. In [206], Smaoui and El-Gamil produced a paper dealing with the adaptive control of the unforced GKdVB equation using three adaptive control laws.

The GBHE takes the form

$$\frac{\partial u}{\partial t} + \alpha u^{\delta}\frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial x^2} = \beta u\left(1-u^{\delta}\right)\left(u^{\delta}-\gamma\right), 0 \le x \le 1, t \ge 0, \qquad (7.1.1)$$

where α, β, γ and δ are parameters and where $\beta \ge 0, \delta > 0, \gamma \in (0,1)$.

In population dynamics, $u(x,t)$ represent the population density, γ is the species carrying capacity, α stands for the speed of advection and β is a parameter that describes a nonlinear source. When a certain condition is imposed on the parameter, the GBHE is reduced to many parabolic evolution equations of physical insight.

These equations describe different phenomena in mathematical physics, biomathematics, chemistry and mechanics [207]. Eq. (7.1.1) models the interaction between reaction mechanisms, convection effects and diffusion transports [208,209]. The Burgers equation is a very interesting model because of the nonlinear advection $u^{\delta}u_x$ term, dissipation u_{xx} term, and the shock wave behaviour when the Reynolds number is very large [210].

In this chapter, an adaptive boundary control is developed for the GBHE (7.1.1) with high-order nonlinearity:

$$u_t + \propto u^{\delta}u_x - \nu u_{xx} = \beta u\left(1 - u^{\delta}\right)\left(u^{\delta} - \gamma\right), 0 \le x \le 1, t \ge 0$$

with the initial condition $u(x,0) = f_0(x)$

and boundary conditions

$$au(0,t) + bu_x(0,t) = \omega_1(t),$$

$$cu(1,t) + du_x(1,t) = \omega_2(t). \qquad (7.1.2)$$

7.2 Preliminaries

In this section, I present some basic propositions and lemmas that become useful in the next sections.

Proposition: (Gronwall–Bellman Inequality) [211]

Let $\gamma(t) : [a,b] \to \mathbb{R}$ and $\alpha(t) : [a,b] \to \mathbb{R}$ be two continuous functions and let $\beta(t) \ge 0$ be a non-negative integrable function on the same interval. If $\gamma(t)$ satisfies

$$\gamma(t) \le \alpha(t) + \int_a^t \beta(s)\gamma(s)ds, a \le t \le b. \qquad (7.2.1)$$

and if the function $\alpha(t)$ is non-decreasing, then

$$\gamma(t) \le \alpha(t) \exp\left(\int_a^t \beta(\tau)d\tau\right), \text{for } a \le t \le b. \qquad (7.2.2)$$

Lemma 1: [212]

Let $\beta < 0. If\ u(x,t) \in L^2(0,\infty)$; then

$$\int_0^t exp(\beta(t-\tau))u^2(1,\tau)d\tau \to 0. \quad (7.2.3)$$

Lemma 2: [212]

Let $\beta < 0$, if $u(x,t) \in L^{2\propto+2}(0,\infty)$; then

$$\int_0^t exp(\beta(t-\tau))u^{2\propto+2}(1,\tau)d\tau \to 0 \text{ as } t \to \infty. \quad (7.2.4)$$

7.3 Global Exponential Stability of the Generalised Burgers–Huxley Equation With Zero Dirichlet Conditions

In this section, I state and prove a theorem to show these types of equations are globally exponential stable in $L^2[0,1]$ under zero Dirichlet boundary conditions.

Theorem 1:

Let δ be a positive integer, $\nu > 0$ and $\gamma \leq 1$; then the GBHE with zero Dirichlet boundary conditions is globally exponential stable in $L^2(0, 1)$.

Proof:

Multiplying both sides of Eq. (7.1.1) by $2u(x,t)$ results in

$$2uu_t + 2 \propto u^{\delta+1}u_x - 2\mathcal{V}uu_{xx} = -2\beta u^2(u^\delta - 1)(u^\delta - \gamma), 0 \leq x \leq 1, t \geq 0. \quad (7.3.1)$$

By integrating Eq. (7.3.1) from 0 to 1,

$$\frac{d}{dt}\int_0^1 u^2dx + 2\alpha\int_0^1 u^{\delta+1}u_x dx - 2\nu\int_0^1 uu_{xx}dx = -2\beta\int_0^1 u^2(u^\delta - 1)(u^\delta - \gamma)dx, \quad (7.3.2)$$

$$\frac{d}{dt}\|u\|^2 + \frac{2\alpha}{\delta+2}[u^{\delta+2}(1,t) - u^{\delta+2}(0,t)] - 2\nu[u(1,t)u_x(1,t) - u(0,t)u_x(0,t)] + 2\nu\|u_x\|^2 = -2\beta\int_0^1(u^{2\delta+2} - (\gamma+1)u^{\delta+2} + \gamma u^2)dx. \quad (7.3.3)$$

Using the Dirichlet boundary condition $u(0,t) = u(1,t) = 0$ on Eq. (7.3.3), we have

$$\frac{d}{dt}\|u\|^2 + 2\nu\|u_x\|^2 = -2\beta\int_0^1 (u^{2\delta+2} - (\gamma+1)u^{\delta+2} + \gamma u^2)dx. \quad (7.3.4)$$

$$\frac{d}{dt}\|u\|^2 + 2\nu\|u_x\|^2 = -2\beta\gamma\|u\|^2 + 2\beta(\gamma+1)\int_0^1 uu^{\delta+1}dx - 2\beta\|u^{\delta+1}\|^2 \quad (7.3.5)$$

Using the Cauchy Shwartz and Young inequalities, we have

$$2\beta(\gamma+1)\int_0^1 uu^{\delta+1}dx \le \beta(\gamma+1)(\|u\|^2 + \|u^{\delta+1}\|^2) \quad (7.3.6)$$

From Eq. (7.3.1) and inequality (7.3.6), we have

$$\frac{d}{dt}\|u\|^2 + 2\nu\|u_x\|^2 = -2\beta\gamma\|u\|^2 + 2\beta(\gamma+1)\int_0^1 uu^{\delta+1}dx - 2\beta\|u^{\delta+1}\|^2$$

$$\le -2\beta\gamma\|u\|^2 + \beta(\gamma+1)\|u\|^2 + \|u^{\delta+1}\|^2 - 2\beta\|u^{\delta+1}\|^2$$

$$= (-\beta\gamma+\beta)\|u\|^2 + \|u^{\delta+1}\|^2 - 2\beta\|u^{\delta+1}\|^2$$

$$\frac{d}{dt}\|u\|^2 + 2\nu\|u_x\|^2 \le (-\beta\gamma+\beta)\|u\|^2 + (\beta\gamma-\beta)\|u^{\delta+1}\|^2 \quad (7.3.7)$$

Since $\|u^{\delta+1}\|^2 \le \|u^{\delta}\|^2\|u\|^2$, $\|u^{\delta}\|^2 \le \|u\|^{2\delta}$, we have:

$$\|u^{\delta+1}\|^2 \le \|u\|^{2\delta+2}, \quad (7.3.8)$$

which gives

$$\frac{d}{dt}\|u\|^2 \le (-\beta\gamma+\beta)\|u\|^2 + (\beta\gamma-\beta)\|u^{\delta+1}\|^2 - 2\nu\|u_x\|^2$$

$$= -\beta\gamma\|u\|^2 + \beta\|u\|^2 + \beta\gamma\|u^{\delta+1}\|^2 - \beta\|u^{\delta+1}\|^2 - 2\nu\|u_x\|^2$$

$$\le -\beta\gamma\|u\|^2 + \beta\|u\|^2 + \beta\gamma\|u\|^{2\delta+2} - \beta\|u\|^{2\delta+2} - 2\nu\|u_x\|^2$$

$$= (\beta-\beta\gamma)\|u\|^2 + (\beta\gamma-\beta)\|u\|^{2\delta+2} - 2\nu\|u_x\|^2$$

$$= \beta(1-\gamma)[\|u\|^2 + \|u\|^{2\delta+2}] - 2\nu\|u_x\|^2$$

$$\frac{d}{dt}\|u\|^2 \leq -2\nu\|u_x\|^2 + \beta(1-\gamma)\left[\|u\|^2 + \|u\|^{2\delta+2}\right] \quad (7.3.9)$$

Since $\|u\|^{2\delta+2} \geq \|u\|^2$, then

$$\frac{d}{dt}\|u\|^2 \leq -2\nu\|u_x\|^2 \quad (7.3.10)$$

Using the Poincare inequality [213], we obtain

$$-2\nu\|u_x\|^2 \leq \frac{-\upsilon}{2}\|u\|^2. \quad (7.3.11)$$

By the basic comparison of inequality (7.3.11) with first-order differential inequalities, we have

$$\|u\|^2 \leq \|u_0\|^2 \exp(-2\nu t). \quad (7.3.12)$$

Therefore, $\|u(x,t)\|$ converges to zero exponentially when $t \to \infty$.

7.4 Construction of the Adaptive Boundary Control for the Generalised Burgers–Huxley Equation

In this section, I build an adaptive boundary control for Eq. (7.1.1) as follows.

Theorem 2:

Let $\delta > 0, \gamma \leq 1$; then the solution $u(x,t)$ of Eq. (7.1.1) with initial condition $f_0(x) \in H^3(0,1)$, which satisfies the boundary conditions (7.1.2) such that a, b, c, d are arbitrary constants with the property $\|u(.,t)\| \to 0 \; as \; t \to \infty$.

Proof:

If $u(0,t).u(1,t)$ are locally existing in $L^{2\propto+2}(0,1)$ and the control functions $\omega_1(t), \omega_2(t)$ are given by

$$\omega_1(t) = k_1(t)u^{2\delta+1}(0,t) + k_2(t)u^{\delta+1}(0,t) + k_3(t)u(0,t), \quad (7.4.1)$$

$$\omega_2\,(t)\;=\;k_4\,(t)u^{2\delta+1}\,(1,t)\;+\;k_5\,(t)u^{\delta+1}\,(1,t)\;+\;k_6\,(t)u(1,t), \quad (7.4.2)$$

such that $k_n(t), n\;=\;1,2,\ldots,6$ are bounded for any $t \geq 0$.

First, note that the following are true:

$$k'_1\,(t)\;=\;r_1u^{2\delta+2}\,(0,t), k'_2\,(t)\;=\;r_2u^{\delta+2}\,(0,t),\;\; k'_3\,(t)\;=\;r_3u^{2\delta+2}\,(0,t),$$

$$k'_4(t)=\;-r_4u^{2\delta+2}(1,t), k'_5(t)=\;-r_5u^{\delta+2}\,(1,t),\; k'_6\,(t)\;=\;-r_6\,u(1,t),$$

such that $r_n>\;0, n\;=\;1,2,\ldots,6,$

Now, I proceed with proving the theorem.

Consider the following Lyapunov function candidate [214]:

$$V\,(t)\;=\int_0^1 u^2(x,t)dx \quad (7.4.3)$$

Operating on Eq. (7.4.3) with a differential operator with respect to *t* and using Eq. (7.1.1) gives

$$V'(t)=\int_0^1 u(x,t)u_t(x,t)dx=\int_0^1 u\left(-\propto u^\delta u_x+\upsilon u_{xx}+\beta u\left(u^\delta-1\right)\left(u^\delta-\gamma\right)\right)dx. \quad (7.4.4)$$

Thus,

$$V'(t)=-\alpha\int_0^1 u^{\delta+1}u_x dx+\upsilon\int_0^1 uu_{xx}dx+\beta\int_0^1 u^2\left(u^\delta-1\right)\left(u^\delta-\gamma\right)dx$$

$$=\frac{-\alpha}{\delta+2}\left[u^{\delta+2}(1,t)-u^{\delta+2}(0,t)\right]+\upsilon[u(1,t)u_x(1,t)-u(0,t)u_x(0,t)]-\upsilon\left\|u_x\right\|^2$$

$$=+\beta\int_0^1\left(\left(u^{\delta+1}\right)^2-(\gamma+1)uu^{\delta+1}+\gamma u^2\right)dx. \quad (7.4.5)$$

Now, using the Cauchy Shwartz and Young inequalities, we have

$$\beta(\gamma+1)\int_0^1 uu^{\delta+1}dx\leq\frac{\beta(\gamma+1)}{2}\left(\left\|u\right\|^2+\left\|u^{\delta+1}\right\|^2\right), \quad (7.4.6)$$

From the Poincare inequality, we obtain

$$-v\|u_x\|^2 \leq \frac{-v}{4}\|u\|^2 + \frac{v}{2}u^2(0,t) \tag{7.4.7}$$

$$V'(t) \leq \frac{-\alpha}{\delta+2}\left[u^{\delta+2}(1,t) - u^{\delta+2}(0,t)\right] + v[u(1,t)u_x(1,t) - u(0,t)u_x(0,t)] + \frac{v}{2}u^2(0,t)$$

$$-\frac{v}{4}\|u\|^2 + \beta\int_0^1 \left(\left(u^{\delta+1}\right)^2 - (\gamma+1)uu^{\delta+1} + \gamma u^2\right)dx.$$

From Eq. (7.3.4),

$$\frac{d}{dt}\|u\|^2 + 2v\|u_x\|^2 = -2\beta\int_0^1 \left(u^{2\delta+2} - (\gamma+1)u^{\delta+2} + \gamma u^2\right)dx.$$

Then, we get

$$-\frac{v}{4}\|u\|^2 - \frac{1}{2}\frac{d}{dt}\|u\|^2 - v\|u_x\|^2$$

From Eq. (7.3.9),

$$\frac{d}{dt}\|u\|^2 \leq -2v\|u_x\|^2 + \beta(1-\gamma)\left[\|u\|^2 - \|u\|^{2\delta+2}\right]$$

Then, we get

$$-\frac{v}{4}\|u\|^2 - \frac{1}{2}(-2v\|u_x\|^2 + \beta(1-\gamma)\left[\|u\|^2 - \|u\|^{2\delta+2}\right]) - v\|u_x\|^2$$

$$-\frac{v}{4}\|u\|^2 + v\|u_x\|^2 - \frac{1}{2}\beta(1-\gamma)\left[\|u\|^2 - \|u\|^{2\delta+2}\right]) - v\|u_x\|^2$$

$$-\frac{v}{4}\|u\|^2 - \frac{1}{2}\beta(1-\gamma)\|u\|^2 + \frac{1}{2}\beta(1-\gamma)\|u\|^{2\delta+2}$$

$$\beta\left(1 - \frac{\gamma+1}{2}\right)\|u^{\delta+1}\|^2 + \left(-\beta\gamma - \frac{v}{4} - \frac{\beta\gamma}{2} + \frac{\beta}{2}\right)\|u\|^2 \tag{7.4.8}$$

Then, at $\gamma \leq 1$, we have

$$V'(t) \leq \frac{-\alpha}{\delta+2}\left[u^{\delta+2}(1,t) - u^{\delta+2}(0,t)\right] + \upsilon[u(1,t)u_x(1,t) - u(0,t)u_x(0,t)] + \frac{\upsilon}{2}u^2(0,t) + \left(-\frac{\upsilon}{4}\right)\|u\|^2. \quad (7.4.9)$$

Using Eq. (7.1.2), we have

$$u_x(0,t) = \frac{1}{a}\big(\omega_1(t) - bu(0,t)\big), \quad (7.4.10)$$

$$u_x(1,t) = \frac{1}{c}\big(\omega_2(t) - du(1,t)\big), \quad (7.4.11)$$

Using inequality (7.4.9) and Eqs. (7.4.10) and (7.4.11), we have

$$V'(t) \leq \left(-\frac{\upsilon}{4}\right)\|u\|^2 + \frac{\upsilon}{2}u^2(0,t) - \frac{-\alpha}{\delta+2}\left[u^{\delta+2}(1,t) - u^{\delta+2}(0,t)\right] + \upsilon\Big[u(1,t)(\tfrac{1}{c}(\omega_2(t) - du(1,t))) - u(0,t)(\tfrac{1}{a}(\omega_1(t) - bu(0,t)))\Big]. \quad (7.4.12)$$

Substituting by the suggested values of $\omega_1(t), \omega_2(t)$, we get

$$V'(t) \leq \left(-\frac{\upsilon}{4}\right)\|u\|^2 + \frac{\upsilon}{2}u^2(0,t) - \frac{-\alpha}{\delta+2}\left[u^{\delta+2}(1,t) - u^{\delta+2}(0,t)\right] + \upsilon u(1,t)\left[(\tfrac{1}{c}((k_4(t)u^{2\delta+2}(1,t) + k_5(t)u^{\delta+2}(1,t) + k_6(t)u(1,t))) - \frac{d}{c}u(1,t)\right] - \upsilon u(0,t)\left[(\tfrac{1}{c}((k_1(t)u^{2\delta+2}(0,t) + k_2(t)u^{\delta+2}(0,t) + k_3(t)u(0,t))) - \frac{b}{a}u(0,t))\right]. \quad (7.4.13)$$

We introduce the non-negative energy function $E(t)$, as follows:

$$E(t) = V(t) + \frac{\upsilon}{2ar_1}(k_1(t))^2 + \frac{a}{2\upsilon r_2}\left(\frac{\upsilon}{a}k_2(t) - \frac{\alpha}{\delta+2}\right)^2 + \frac{a}{2\upsilon r_3}\left(\frac{\upsilon}{a}k_3(t) - \frac{\upsilon b}{a} - \frac{\upsilon}{2}\right)^2 + \frac{\upsilon}{2cr_4}(k_4(t))^2 + \frac{c}{2\upsilon r_5}\left(\frac{\upsilon}{c}k_5(t) - \frac{\alpha}{\delta+2}\right)^2 + \frac{c}{2\upsilon r_6}\left(\frac{\upsilon}{c}k_6(t) - \frac{\upsilon d}{c}\right)^2 \quad (7.4.14)$$

Evaluating the time derivative of $E(t)$ and substituting $V'(t)$ from inequality (7.4.13) and $k'_n(t)$ into Eq. (7.4.14), we have

$$E'(t) \leq \frac{-\upsilon}{4}\|u\|^2, \quad (7.4.15)$$

This implies that $E(t) \leq E(0)$. Since $u(0,t)$ and $u(1,t) \in L^{2\propto+2}(0,\infty)$, it follows that $k_j(t)$ can be defined as continuous functions on $(0,\infty)$. Then, Eq.(7.4.14) and inequality (7.4.15) imply that $k_j(t), j = 1,\ldots,6$ are bounded, which implies that

$$u(i,t) \in L^2(0,\infty) \cap L^{2\propto+2}(0,\infty), i = 0,1$$

I also show the global asymptotic stability of Eqs. (7.1.1) and (7.1.2). Using the Gronwall inequality on inequality (7.2.1), we have

$$\begin{aligned} V(t) \leq V(0)\exp\left(\frac{-\upsilon}{4}t\right) \\ + \upsilon\int_0^t \left[\frac{-k_1(\tau)}{a}u^{2\delta+2}(0,\tau) + \left(\frac{\alpha}{(\delta+2)\upsilon} - \frac{k_2(\tau)}{a}\right)u^{\delta+2}(0,\tau) + (\frac{1}{2} + \frac{b}{a}\right. \\ \left. - \frac{k_3(\tau)}{a})u^2(0,\tau)\right]\exp\left(\frac{-\upsilon}{4}(t-\tau)\right)d\tau \\ + \upsilon\int_0^t \left[\frac{k_4(\tau)}{c}u^{2\delta+2}(1,\tau) + \left(\frac{-\alpha}{(\delta+2)\upsilon} + \frac{k_5(\tau)}{c}\right)u^{\delta+2}(1,\tau) + (\frac{-d}{c}\right. \\ \left. - \frac{k_6(\tau)}{c})u^2(1,\tau)\right]\exp\left(\frac{-\upsilon}{4}(t-\tau)\right)d\tau. \end{aligned}$$

Next, using Lemma 1 and Lemma 2, we predict that $\|u(.,t)\| \to 0\ as\ t \to \infty$.

7.5 Adomian Decomposition Method for the Initial Boundary Value Problem [215]

Consider the nonlinear initial boundary value problem of PDE in the following general operator form:

$$Lu(x,t) = Ru(x,t) + Nu(x,t) + gu(x,t), 0 < \alpha \leq 1, \qquad (7.5.1)$$

with initial condition $u(x,0) = f_0(x)$, and boundary conditions $u(0,t) = p(t)$ and $u(1,t) = q(t)$, where $L = \frac{\partial}{\partial t}$,, is the highest partial derivative with respect to t, R is a linear operator, $N(u)$ is the nonlinear term and $g(x,t)$ is the source function. Operating on both sides of Eq. (7.5.1) with the inverse operator L^{-1} gives

$$u(x,t) = \phi + L^{-1}\big(g(x,t)\big) + L^{-1}\big(Ru(x,t) + Nu(x,t)\big) \qquad (7.5.2)$$

where the first part from the right-hand side of Eq. (7.5.2) is obtained from the solution to the homogenous differential equation $L\phi = 0$.

The ADM defines the solution $u(x,t)$ as an infinite series in the form

$$u(x,t) = \sum_{n=0}^{\infty} u_n(x,t) \tag{7.5.3}$$

where the components $u_n(x,t)$ can be obtained in recursive form. The nonlinear term $N(u)$ can be decomposed by an infinite series of polynomials given by

$$N(u) = \sum_{n=0}^{\infty} A_n, \tag{7.5.4}$$

The formula for Adomian polynomials is

$$A_n = \frac{1}{n!}\left[\frac{d^n}{d\lambda^n} N\left(\sum_{i=0}^{\infty} \lambda^i u_i\right)\right]_{\lambda=0}, n = 0,1,2, ... \tag{7.5.5}$$

Substituting from Eqs. (7.5.3) and (7.5.4) into Eq. (7.5.3) gives

$$\sum_{n=0}^{\infty} u_n(x,t) = \varphi + L^{-1}\big(g(x,t)\big) + L^{-1}\left(\sum_{n=0}^{\infty} u_n + \sum_{n=0}^{\infty} A_n\right). \tag{7.5.6}$$

Substituting the initial conditions, we obtain the components $u_n(x,t)$ of the solution using the following formula

$$u_0(x,t) = f_0(x) + \phi + L^{-1}\big(g(x,t)\big),$$

$$u_{n+1}(x,t) = L^{-1}(Ru_n + A_n), n \geq 0. \tag{7.5.7}$$

The initial solution can be written as

$$u_0(x,t) = f_0(x), \tag{7.5.8}$$

I construct a new successive approximate solution $u_n^*(x,t)$ as follows:

$$u_n^*(x,t) = u_n(x,t) + (1-x)[p(t) - u_n(0,t)] + x[q(t) - u_n(1,t)], n = 0,1,2, ... \tag{7.5.9}$$

$$u_{n+1}^*(x,t) = L^{-1}(Ru_n^* + A_n^*), \tag{7.5.10}$$

such that

$$A_n^* = \frac{1}{n!}\left[\frac{d^n}{d\lambda^n} N\left(\sum_{i=0}^{\infty} \lambda^i u_i^*\right)\right]_{\lambda=0}, n = 0,1,2, ...$$

Using Eqs. (7.5.8)–(.5.10), we obtain the approximate solution,

$$u(x,t) = \sum_{n=0}^{\infty} u_n(x,t). \qquad (7.5.11)$$

7.6 Numerical Example

Using the ADM algorithm presented here in Eq. (7.1.1), when $\alpha = \beta = 1$, $\gamma = 0.001$ and $\delta = 2$, I solve the GBHE without control as outlined in Tables 7.6.1–7.6.7, with time t=0, 0.5,1,2,3,4 and 5. Table 7.6.8 presents the absolute errors for the GBHE using the ADM when $t = 0$ to t = $1, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1;$

Table 7.6.1 Comparison between the numerical and exact solutions for the GBHE when $t = 0, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

x	Numerical Solution	Exact Solution
0	0.0005	0.0005
0.1	0.000521699	0.000521699
0.2	0.000543317	0.000543317
0.3	0.000564773	0.000564773
0.4	0.000585989	0.000585989
0.5	0.00060689	0.00060689
0.6	0.000627407	0.000627407
0.7	0.000647476	0.000647476
0.8	0.000667037	0.000667037
0.9	0.000686039	0.000686039
1.0	0.000704437	0.000704437

Table 7.6.2 Comparison between the numerical and exact solutions for the GBHE when $t = 0.5, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

x	Numerical Solution	Exact Solution
0	0.000500415	0.000637703
0.1	0.000519779	0.00065752
0.2	0.000539102	0.000676802
0.3	0.000558343	0.000695501
0.4	0.000577459	0.000713576
0.5	0.000596404	0.000730993
0.6	0.000615127	0.000747726

0.7	0.000633579	0.000763754
0.8	0.000651706	0.000779064
0.9	0.000669457	0.000793651
1.0	0.000686782	0.000807512

Table 7.6.3 Comparison between the numerical and exact solutions for the GBHE when $t = 1, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

x	**Numerical Solution**	**Exact Solution**
0	0.000500772	0.00075599
0.1	0.00051801	0.000771653
0.2	0.000535386	0.000786595
0.3	0.000553023	0.000800811
0.4	0.000571005	0.000814304
0.5	0.000589367	0.00082708
0.6	0.00060809	0.000839151
0.7	0.000627102	0.000850532
0.8	0.000646283	0.00086124
0.9	0.000665479	0.000871298
1.0	0.000684512	0.000880727

Table 7.6.4 Comparison between the numerical and exact solutions for the GBHE when $t = 2, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

x	**Numerical Solution**	**Exact Solution**
0	0.000501326	0.000905649
0.1	0.000495311	0.000912814
0.2	0.00049128	0.000919482
0.3	0.000491024	0.000925683
0.4	0.000495977	0.000931441
0.5	0.000507089	0.000936784
0.6	0.000524761	0.000941736

0.7	0.000548837	0.000946323
0.8	0.000578658	0.000950567
0.9	0.000613156	0.000954492
1.0	0.000650983	0.000958119

Table 7.6.5 Comparison between the numerical and exact solutions for the GBHE when $t = 3, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

x	**Numerical Solution**	**Exact Solution**
0	0.000501689	0.000967468
0.1	0.000414376	0.000970093
0.2	0.000334647	0.000972513
0.3	0.000269394	0.000974741
0.4	0.000224215	0.000976794
0.5	0.000202957	0.000978683
0.6	0.000207464	0.000980422
0.7	0.00023755	0.000982021
0.8	0.000291162	0.000983492
0.9	0.000364719	0.000984844
1.0	0.00045356	0.000986088

Table 7.6.6 Comparison between the numerical and exact solutions for the GBHE when $t = 4, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

x	**Numerical Solution**	**Exact Solution**
0	0.000501889	0.000989263
0.1	0.000235977	0.000990147
0.2	−0.0000108612	0.00099096
0.3	−0.000221241	0.000991705
0.4	−0.000380963	0.00099239
0.5	−0.000480123	0.000993019
0.6	−0.000513747	0.000993596

0.7	−0.00048188	0.000994125
0.8	−0.00038919	0.000994611
0.9	−0.000244166	0.000995058
1.0	−0.0000580347	0.000995467

Table 7.6.7 Comparison between the numerical and exact solutions for the GBHE when $t = 5, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

x	Numerical Solution	Exact Solution
0	0.000501954	0.000996509
0.1	−0.0000791146	0.000996799
0.2	−0.000621592	0.000997064
0.3	−0.00109026	0.000997308
0.4	−0.00145624	0.000997531
0.5	−0.00169924	0.000997736
0.6	−0.00180881	0.000997924
0.7	−0.00178456	0.000998096
0.8	−0.00163537	0.000998254
0.9	−0.00137782	0.000998399
1.0	−0.00103406	0.000998532

Table 7.6.8 Comparison between the numerical and exact solutions for the GBHE when $t = 0$ to $t = 1, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

t	Absolute Error
0	0.
0.1	0.0000401684
0.2	0.0000785384
0.3	0.000115102
0.4	0.00014986
0.5	0.000182822
0.6	0.00021401

0.7	0.00024346
0.8	0.000271219
0.9	0.000297349
1.0	0.000321925

To illustrate the behaviour of the numerical and exact solutions for the GBHE at various times, I present 2D figures in Figures 7.6.1–7.6.7, when $t = 0, 0.5,1,2,3,4$ and 5. 3D figures are presented in Figures 7.6.8–7.6.12. Figure 7.6.13 compares the numerical and exact solutions with control for the GBHE when $\gamma = 0.001; \delta = 2; \alpha = 1;\ \beta = 1$ from $t = 0$ to $t = 6$.

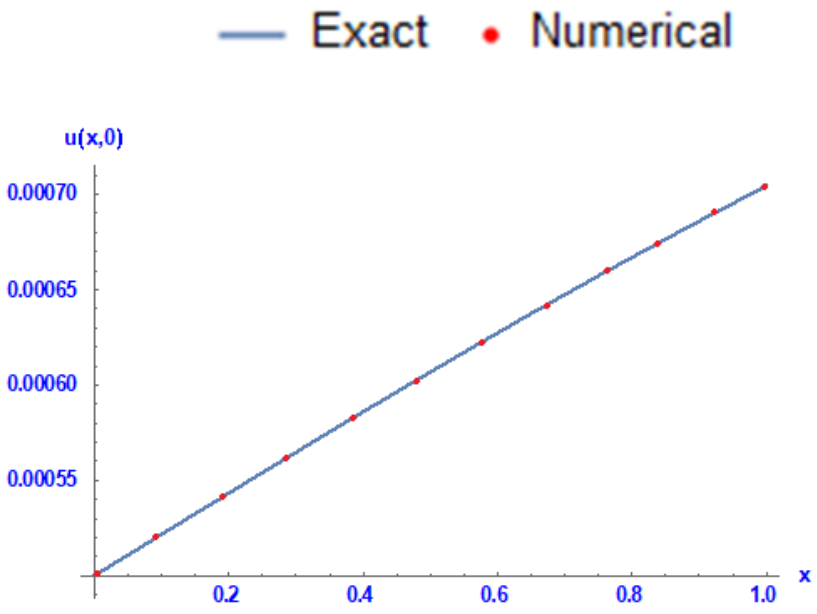


Figure 7.6.1 Comparison between the numerical and exact solutions for the GBHE when $t = 0, \gamma = 0.001; \delta = 2; \alpha = 1;\ \beta = 1$.

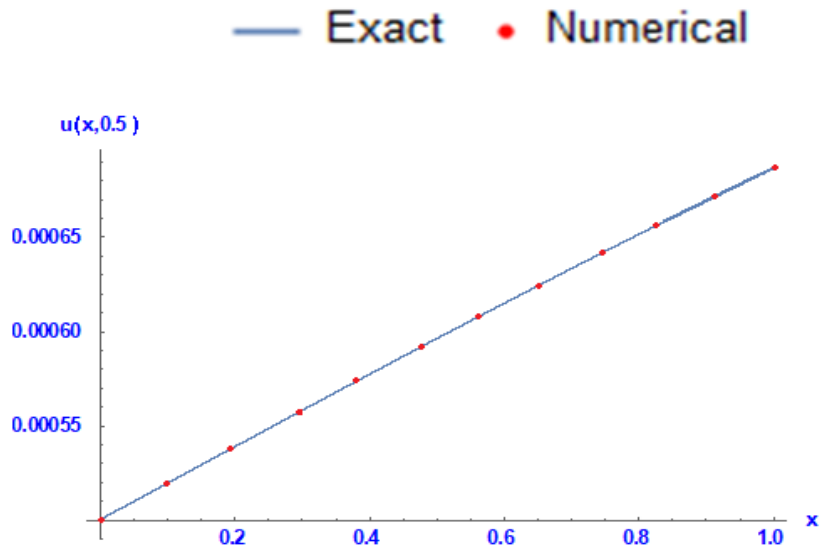


Figure 7.6.2 Comparison between the numerical and exact solutions for the GBHE when $t = 0.5, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

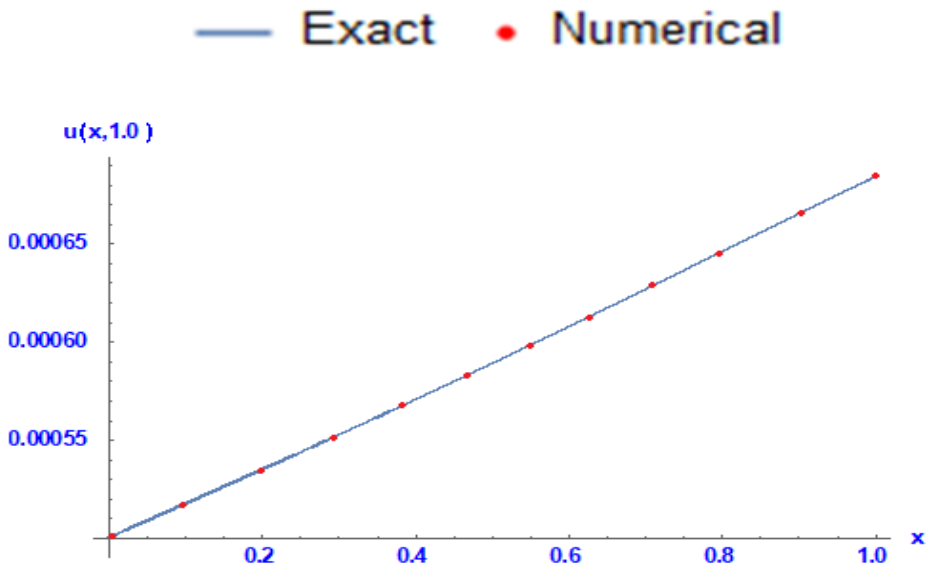


Figure 7.6.3 Comparison between the numerical and exact solutions for the GBHE when $t = 1, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

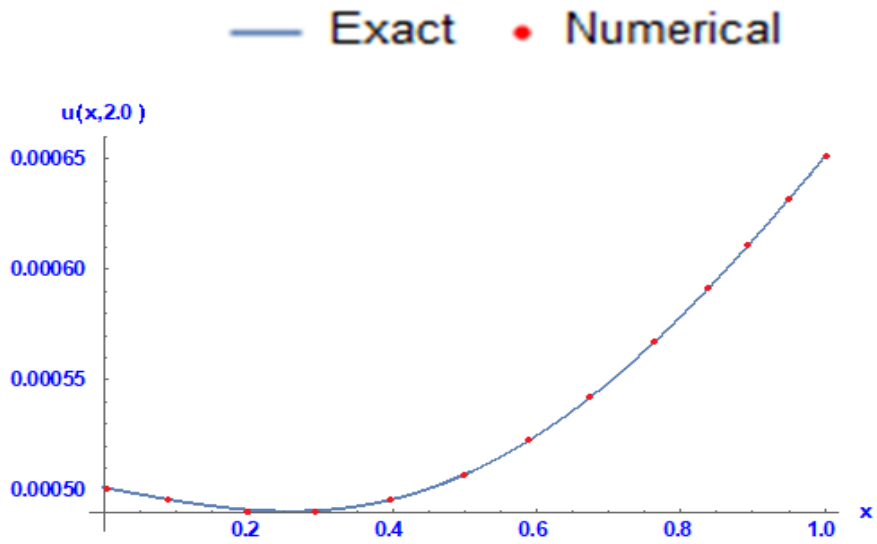


Figure 7.6.4 Comparison between the numerical and exact solutions for the GBHE when $t = 2, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

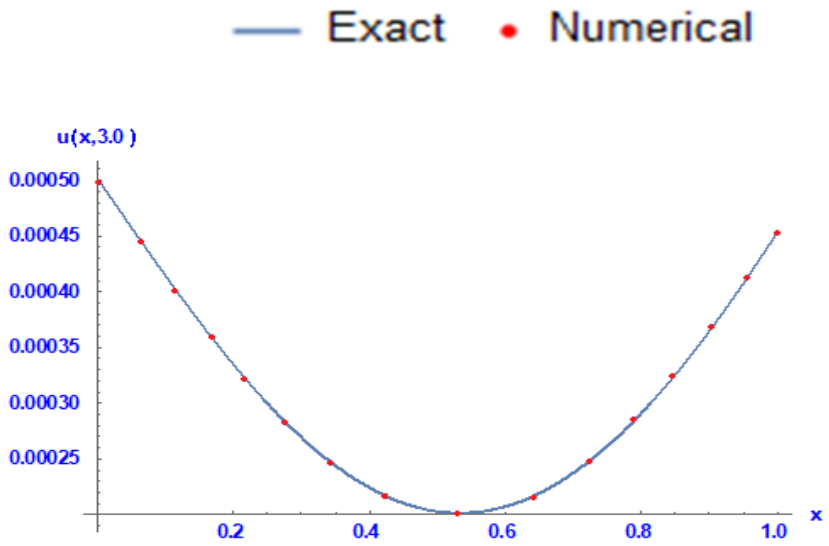


Figure 7.6.5 Comparison between the numerical and exact solutions for the GBHE when $t = 3, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

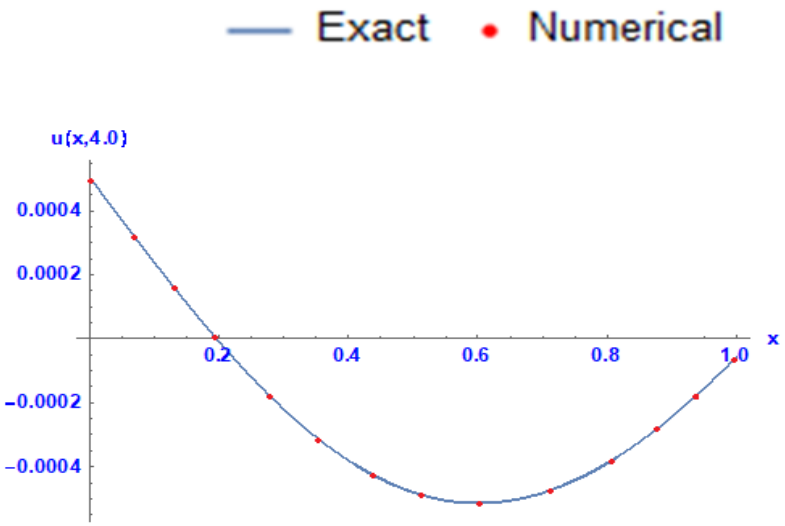


Figure 7.6.6 Comparison between the numerical and exact solutions for the GBHE when $t = 4, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

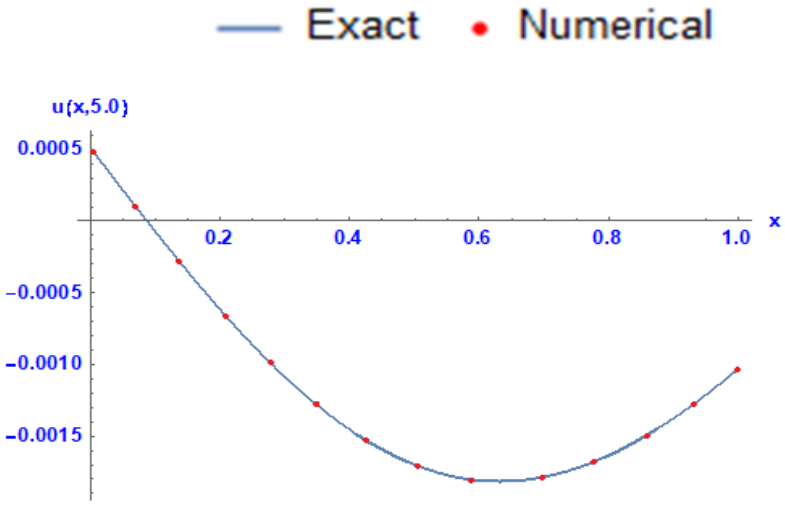


Figure 7.6.7 Comparison between the numerical and exact solutions for the GBHE when $t = 5, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

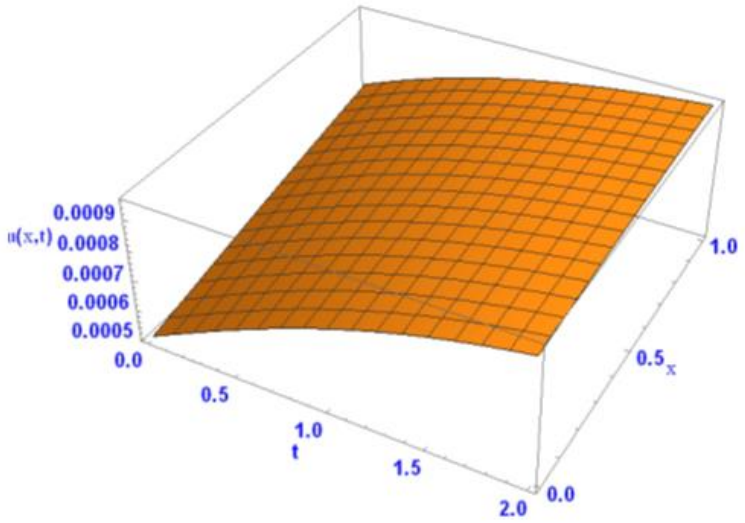


Figure 7.6.8 3D representation of the behaviour of the numerical solutions for the GBHE when $t = 0$ to $t = 2, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

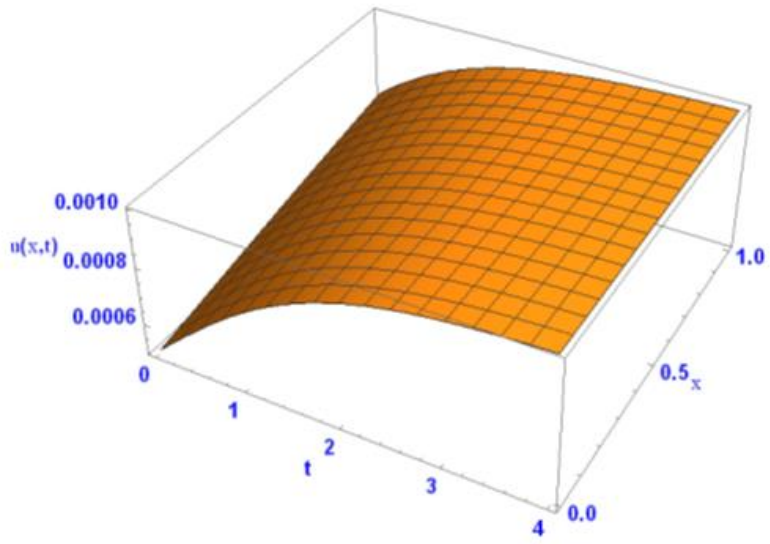


Figure 7.6.9 3D representation of the behaviour of the numerical solutions for the GBHE when $t = 0$ to $t = 4, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

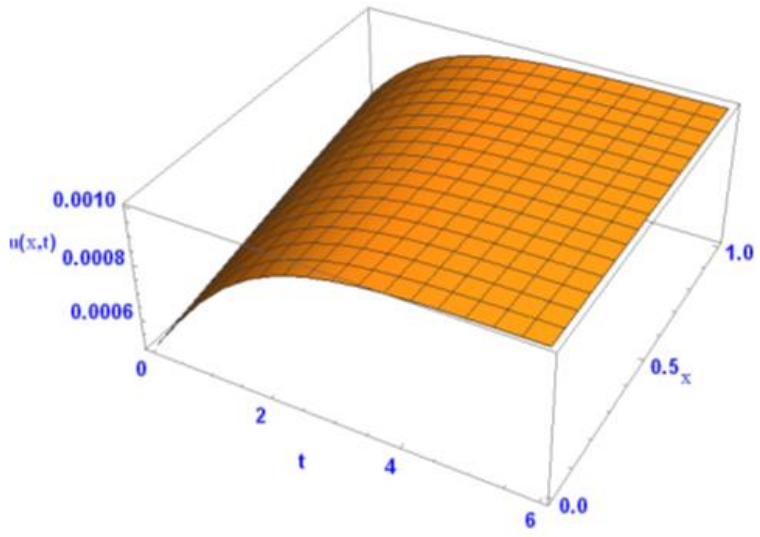


Figure 7.6.10 3D representation of the behaviour of the numerical solutions for the GBHE when $t = 0$ to $t = 6, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

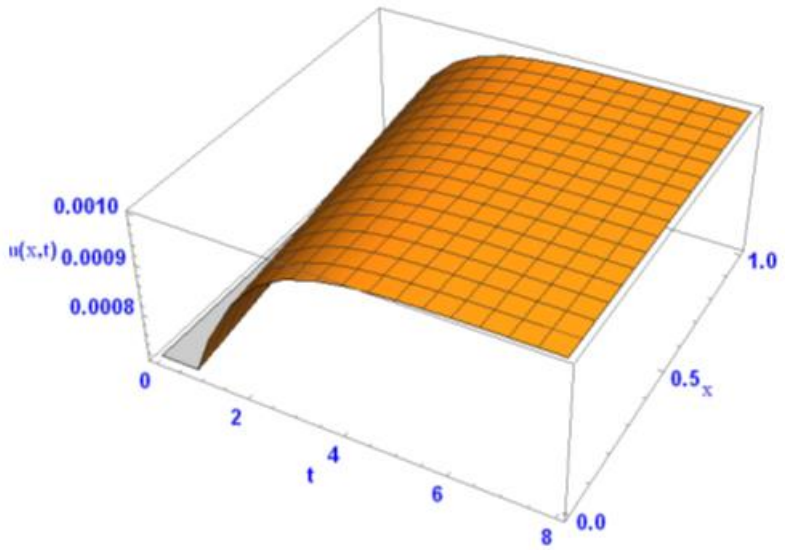


Figure 7.6.11 3D representation of the behaviour of the numerical solutions for the GBHE when $t = 0$ to $t = 8, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

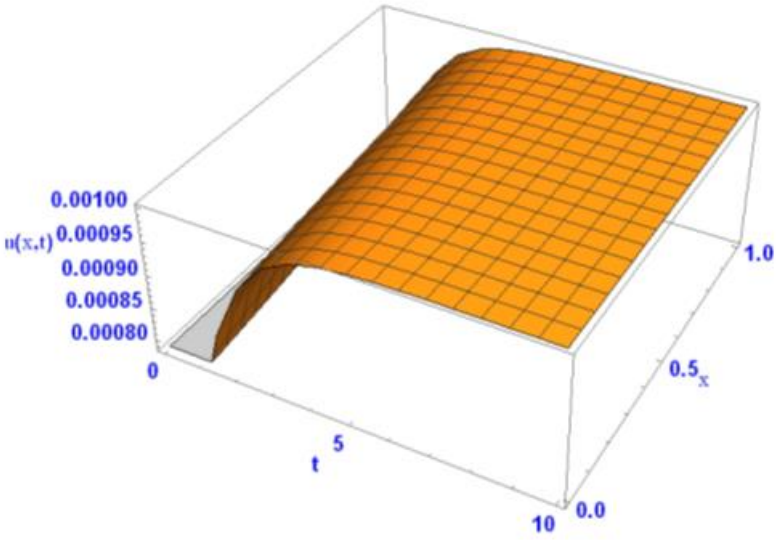


Figure 7.6.12 3D representation of the behaviour of the numerical solutions for the GBHE when $t = 0$ to $t = 10, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

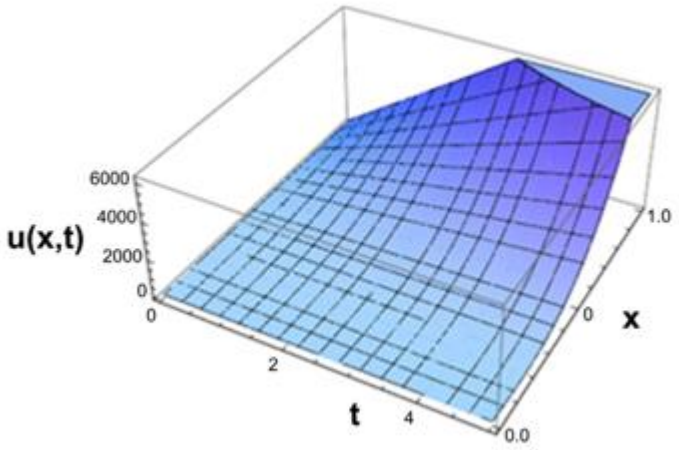


Figure 7.6.13 The ADM truncated solution $u(x,t)$ using the suggested boundary control, for the numerical and exact solution for the GBHE when $t = 0$ to $t = 6, \gamma = 0.001; \delta = 2; \alpha = 1; \beta = 1$.

7.7 Conclusion

In this chapter, I introduced adaptive boundary control for the GBHE with high-order nonlinearity terms. I proved that this type of GBHE is globally exponential stable in

$L^2[0,1]$, under zero Dirichlet boundary conditions. I developed an adaptive boundary control for the GBHE, finding the solution $u(x,t)$ to the GBHE using initial solution $f_0(x) \in H^3(0,1)$ and some boundary conditions having the property $\|u(.,t)\| \rightarrow 0 \; as \; t \rightarrow \infty$. Finally, the ADM was used to illustrate the performance of the controller applied to the GBHE.

CHAPTER 8

Conclusions and Future Work

Chapter 8: Conclusions and Future Work

8.1 Introduction

This final chapter of the thesis presents the overall conclusions based on the approximate solutions for the PDEs presented in previous chapters. It also outlines the significant practical contributions of treatment for some PDEs through numerical analysis, which is a useful tool that can be used to identify better methods. This chapter also shares some recommendations for future research.

8.2 Conclusion

This chapter discusses the general conclusions relating to the numerical treatment of PDEs and recommends further avenues for research.

Chapter 1 provides a general overview of the topic of analysis, motivation for the study, research questions, contributions, objectives, methodology and thesis structure.

Chapter 2 consists of four sections. The first discusses the selected PDEs and provides the key definitions and concepts related to their analysis. Sections 2.2, 2.3 and 2.4 use various numerical methods—the B-spline, fractional calculus and ADM—to solve the selected PDEs.

Chapter 3's objective is to discuss the use of the CBS polynomial method to identify the approximate solutions to the nonlinear dispersive wave equation. The CBS method for solving the dissipative wave equation involves four steps:

1. Analysis of the method

   The approximate solution to the dissipative wave equation is considered as

   $$U_N(x,t) = \sum_{i=0}^{n} w_i(t_j)\, \phi_i(x_j),$$

   where $\phi_i(x_j)$ are the spline functions and $w_i(t_j)$ are the unknowns to be determined. The dissipative wave equation is reduced to one system of ODE.

The boundary conditions are used to obtain N+3 equations in N+3 unknowns. The initial conditions are applied to obtain the independent variables:

$$(\omega_{-1}^0, \omega_0^0, .., \omega_N^0, \omega_{N+1}^0)^T$$

2. The initial state

After deriving the following system of algebraic equations,

$$A\omega^{n+1} = -B\omega^n - C\omega^{n-1} + k^2\eta_i^n(x,t)$$

two initial conditions are applied to obtain the following system:

$$\omega_{i-1}^0 + 4\omega_i^0 + \omega_{i+1}^0 = u(x_j, 0),$$
$$j = 0,1,2,\ldots,N.$$

Adding the following two initial conditions completes the system of equations:

$$-3\omega_{-1}^0 + 3\omega_1^0 = hu_x(a,0),$$
$$-3\omega_{N-1}^0 + 3\omega_{N+1}^0 = hu_x(b,0).$$

3. The stability analysis

The Von Neumann stability analysis for system

$$A\omega^{n+1} = -B\omega^n - C\omega^{n-1} + k^2\eta_i^n(x,t),$$

takes effect after linearising the nonlinear terms,

$$z_{i-1} = d + 4d + d = (6d), m = 6d,$$

which take the form

$$\omega_j^n = \varepsilon^n \, exp(\, q\sigma jh), q = \sqrt{-1}$$

I obtain the characteristic equation

$$\varepsilon^2 + 2\beta\varepsilon + 1 = 0,$$

if *h* is small enough; thus the method is conditionally stable.

The stability analysis investigation shows that the method is conditionally stable. In addition, the performance and accuracy of the present method are evidenced by calculating and comparing the L∞ error norms for a greater time, at 10, 20,..., up to 50. I have not identified any studies that compute values for large numbers; they are usually only computed up to a value of 5.0.

4. Numerical examples

The form of the dissipative wave equation

$$u_{tt} - u_{xx} + 2u_t u = g(x, t),$$
$$g(x, t) = -2sin^2 x \, sint \, cost,$$

has the boundary and initial conditions

$$u_{xx}\ (a,\ t) = 0, u_{xx}(b,\ t) = 0,$$
$$u(x,\ 0) = sinx, u_t(x,\ 0) = 0.$$

The numerical results produced by the present method are quite satisfactory and show good agreement with the exact solutions.

Chapter 4 identifies the numerical solution for the dispersive PDE using the non-polynomial splines method. The non-polynomial spline method for solving the dispersive partial equation includes four steps:

1. The analysis of the method

   The approximate solution to the dispersive wave equation is considered as

   $$P_i(x, t_j) = a_i(t_j)\ \ cos\,\omega\,(x - x_i) + b_i(t_j)\ \ sin\,\omega\,(x - x_i) + c_i(t_j)\ \ (x - x_i)^2 + d_i(t_j)(x - x_i) + e_i(t_j),$$

   where $p_j(x,\ t_n)$ are the non-polynomial spline functions and $a_i(t_j), b_i(t_j)$, $c(t_j)$, $d_i(t_j)$ and $e(t_j)$ are unknowns to be determined. Using the continuity condition of the first and second derivatives of $p_j(x,\ t_n)$ at (x_j) and applying the boundary conditions, I obtain

   $$Q\ Z^j = Q^* Z^{j-1} + r^j.$$

2. The error analysis

   The truncation error

   $T_i^j = A_i\eta_{i-2}^j + B_i\eta_{i-1}^j + C_i\eta_i^j + D_i\eta_{i+1}^j - \alpha\eta_{i-2}^{j-1} - \beta\eta_{i-1}^{j-1} - \beta\eta_i^{j-1} - \alpha\eta_{i+1}^{j-1} - \delta_i^j,$

   Taylor series, in terms of $\eta(x_i, y_i)$ and its derivatives, is used to obtain

   $$\beta + \alpha = \frac{h^3}{2}$$

   The local truncation error is of order $o(kh^4 + k^2h^3)$ for $\beta + \alpha = \frac{h^3}{2}$, but for $\beta + \alpha = \frac{h^3}{2}$ and $\alpha = 0$ the local truncation error is of order $o(kh^2 + k^2h^3)$.

3. The stability analysis

   Using the Von Neumann method, the stability of the method is investigated as

   $$Z_i^j = \zeta^j\, exp(q\phi ih),$$

where φ is the wave number, $q = \sqrt{-1}$, h is the element size, and ζ^j is the amplification factor at time level j. I prove that the stability conditions are satisfied to be conditionally stable for small *h* and time step *t.*

4. Numerical examples

   The form of the third-order dispersive equation is

$$\frac{\partial \mu}{\partial t} + \frac{\partial^3 \mu}{\partial x^3} = g(x,\ t), a \le x \le b, t > 0,$$
$$g(x,\ t) = -\sin(\pi x)\, sint - \pi^3 \cos(\pi x)\, cost,$$
$$0 \le x \le 1, t \ge 0.$$

   with boundary and initial conditions

$$\mu(0,\ t) = \mu(1,\ t) = \mu_{xx}(1,\ t) = 0, t > 0,$$
$$\mu(x,\ 0) = sin\pi x, 0 \le x \le 1.$$

   The convergence analysis of the method proves that the proposed scheme is third-order convergent. The method is also shown to be unconditionally stable. The results from a comparison of the exact and approximate solutions in the numerical examples show the superiority of the method compared with those outlined in the literature review. I also show that the L-∞ error norms confirm theoretical convergence.

Chapter 5 uses the CBS method to solve coupled nonlinear non-homogeneous PDEs. The CBS method to solve the coupled system Klein–Gordon equation includes four steps:

1. The analysis of the method

   The collocation method for approximately solving coupled nonlinear non-homogeneous Klein–Gordon equations by approximating *u(x,t)* and *v(x,t)* with CBS to *U(x,t)* and *V(x,t)* is

$$U(x,t) = \sum_{i=-1}^{N+1} \emptyset_i(x)\beta_i(t),$$
$$V(x,t) = \sum_{i=-1}^{N+1} \emptyset_i(x)\alpha_i(t)$$

   The Kline–Gordan equation is reduced to two systems of ODEs. The boundary conditions are used to obtain N+3 equations in N+3 unknowns for the two systems $A\beta = d$ and $C\alpha = w$. These two systems are completed by

using the boundary conditions to obtain two systems with N+3 unknowns in N+3 equations.

2. The initial state

   The initial conditions are applied to obtain the independent variables

   $$(\beta_{-1}^0, \beta_0^0, .., \beta_N^0, \beta_{N+1}^0)^T \text{ and } (\alpha_{-1}^0, \alpha_0^0, .., \alpha_N^0, \alpha_{N+1}^0)^T$$

   Finally, we obtain the following two coupled systems:

   $$A^{**}\beta^{**} = d^{**} \text{ and } C^{**}\alpha^{**} = w^{**}$$

   where A^{**} and C^{**} are nonsingular $n \times n$ matrices.

3. The stability analysis

   For stability analysis, I use the Von Neumann technique. First, I linearise the nonlinear term $v(x,t)u(x,t)$, and then

   $$\beta_{i,j} = \xi^j exp(q\varphi hi)$$

   I get

   $$\begin{aligned}\xi^{j+1}\{ A_{i,j}\{exp(q\varphi h(i-1)) + exp(q\varphi h(i+1))\} + B_{i,j}exp(q\varphi hi)\} \\ = \frac{2}{k^2}\xi^j\{\{exp(q\varphi h(i-1)) + exp(q\varphi h(i+1))\} + 4exp(q\varphi hi)\} \\ - \xi^{j-1}\{ A_{i,j}\{exp(q\varphi h(i-1)) + exp(q\varphi h(i+1))\} \\ + B_{i,j}exp(q\varphi hi)\}.\end{aligned}$$

   After some calculations:

   $$\varepsilon^2 + 2\mu\varepsilon + 1 = 0,$$

   where k is sufficiently small, such that $k^2 \to 0$, and stability is satisfied.

4. Numerical examples

   The form of the coupled nonlinear Klein–Gordon PDE:

   $$\begin{aligned}\frac{\partial^2 u}{\partial x^2}(x,t) + \frac{\partial^2 u}{\partial t^2}(x,t) + v(x,t)u(x,t) = f(x,t), \\ \frac{\partial^2 v}{\partial x^2}(x,t) + \frac{\partial^2 v}{\partial t^2}(x,t) + u(x,t)v(x,t) = g(x,t),\end{aligned}$$

   for $a \le x \le b$ and $t \ge 0$, is subject to the conditions

   $$\begin{aligned}u(a,t) = \varepsilon_1(t), u(b,t) = \varepsilon_2(x), \\ v(a,t) = \rho_1(t), v(b,t) = \rho_2(t), \\ u(x,0) = \tau_1(x), \frac{\partial u}{\partial t}(x,0) = \tau_2(x), \\ v(x,0) = \sigma_1(x), \frac{\partial v}{\partial t}(x,0) = \sigma_2(x).\end{aligned}$$

I find that the proposed method is conditionally stable. The method's accuracy is demonstrated by calculating L∞ error norms. The numerical results obtained show that the present method is a remarkably successful numerical technique for solving coupled nonlinear non-homogeneous Klein–Gordon equations, which makes it useful for a wide range of applications.

Chapter 6 proposes a numerical solution for the time–space-fractional order telegraph equation through the use of a quadratic-polynomial spline-based method. The method to solve the FTE involves four steps:

1. The analysis of the method

   The approximate solution of FTE is considered as

   $$P_i(x,t_j) = a_i(t_j)\ (x-x_i)^2 + b_i(t_j)\ (x-x_i) + c_i(t_j).$$

   Then,

   $$P_i^{(\alpha)}(x_{i+1/2},t_j) = \frac{\partial^\alpha}{\partial x^\alpha} P_i(x_{i+1/2},t_j) = S_{i+1/2}^j, \quad 1<\alpha\le 2, \quad x_i < x_{i+1/2} \le x_{i+1}.$$

2. Using the continuity condition of the first and second derivatives of $p_j(x,\ t_n)$ at (x_j) and applying the boundary conditions I obtain

   $$Z_{i+3/2}^j - 2Z_{i+1/2}^j + Z_{i-1/2}^j = \frac{h^2}{8}\left(S_{i+3/2}^j + 6S_{i+1/2}^j + S_{i-1/2}^j\right),$$

3. The error analysis

   The truncation error is

   $$T^{*}{}_i^j = h^\alpha(h^{2-\alpha} - 8\theta)D_x^2 u_{i+1/2}^j + h^{2+\alpha}\left(\frac{h^{2-\alpha}}{12} - \theta\right)D_x^4 u_{i+1/2}^j$$
   $$+ h^{4+\alpha}\left(\frac{h^{2-\alpha}}{360} - \frac{\theta}{12}\right)D_x^6 u_{i+1/2}^j + \cdots$$

   where

   $$\theta = \frac{\Gamma(3-\alpha)}{2^{\alpha+1}}.$$

   From this expression of the local truncation error, the scheme is of order

   $$O(h^\alpha),\ 1<\alpha\le 2.$$

4. The stability analysis

   Using the Von Neumann method, the stability of the method is investigated as

   $$Z_i^j = \zeta^j\, exp(q\phi ih),$$

where φ is the wave number, $q=\sqrt{-1}$, h is the element size, and ζ^j is the amplification factor at time level j. We obtain a characteristic equation

$$a\zeta^2+\mathrm{b}\ \zeta+\mathrm{c}=0$$

with which I prove that the stability conditions are satisfied to be conditionally stable for small h and time step t.

5. Numerical examples

The form of the time–space-fractional order telegraph equation is

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

subject to boundary conditions

$$u(a,t)=\beta_1(t), \quad u(b,t)=\beta_2(t), \quad t>0\ .$$

and initial conditions

$$u(x,0)=f_1(x), \quad \frac{\partial u(x,0)}{\partial t}=f_2(x), \quad a\leq x\leq b.$$

The developed method is shown to be conditionally stable. The theoretical analysis demonstrates that the numerical scheme is effective and reliable for the time–space-fractional order telegraph equation. The approximate numerical solutions obtained are in good agreement with the approximate solutions outlined in the literature review. It can be concluded that this technique is both powerful and efficient in finding approximate solutions for a large class of linear PDEs of fractional order.

Chapter 7 develops an adaptive boundary control for the GBHE with high-order nonlinearity. In addition, a solution is found for the GBHE using the ADM**.** Adaptive boundary control is developed for the GBHE using the ADM as follows:

1. The analytical approximate solution of the GBHE is

$$u_t+\propto u^\delta u_x-\nu u_{xx}=\beta u\left(1-u^\delta\right)\left(u^\delta-\gamma\right).$$

1.1 The nonlinear initial value problem of the PDE is put in the following general operator, with the initial condition and boundary conditions:

$$Lu(x,t)=Ru(x,t)+Nu(x,t)+gu(x,t),$$

1.2 The inverse operator is used on both sides of the GBHE:

$$u(x,t) = \emptyset + L^{-1}(g(x,t)$$

$$+ L^{-1}((Ru(x,t) + Nu(x,t))$$

1.3 The solution is applied to the ADM using the infinite series. Then, the Adomian polynomials are applied:

$$A_n = \frac{1}{n!}\left[\frac{d^n}{d\lambda^n} N(\sum_{i=0}^{\infty} \lambda^i u_i)\right]_{\lambda=0}, \text{ n=0,1,…}$$

2. The initial state and analytical solution is found using the initial condition

$$u(x,\ 0) = f_0(x).$$

Substituting the initial conditions, I obtain the components $u_n(x,t)$ of the solution using the following formula:

$$u_0(x,\ t) = f_0(x) + \phi + L^{-1}\big(g(x,t)\big)$$

Then,

$$u_{n+1}(x,\ t) = L^{-1}(Ru_n + A_n), n \geq 0$$

where

$$A_n = \frac{1}{n!}\left[\frac{d^n}{d\lambda^n} N(\sum_{i=0}^{\infty} \lambda^i u_i)\right]_{\lambda=0}, \text{ n=0,1,…}$$

and

$$L^{-1}(.) = \int_0^t (.)dt$$

So, I obtain the sequences A_0, A_1,…. Then, I obtain $u_n(x,\ t)$, the analytical solution to the GBHE.

3. Numerical example

The GBHEs are given the form

$$u_t + \propto u^{\delta} u_x - \nu u_{xx} = \beta u\left(1 - u^{\delta}\right)\left(u^{\delta} - \gamma\right), 0 \leq x \leq 1, t \geq 0$$

with initial condition

$$u(x, 0) = f_0(x)$$

and boundary conditions

$$au(0, t) + bu_x(0, t) = \omega_1(t),$$

$$cu(1, t) + du_x(1, t) = \omega_2(t).$$

I prove that this type of GBHE is globally exponential stable in $L^2[0,1]$, under zero Dirichlet boundary conditions. In addition, using an adaptive nonlinear parametric controller, I show that the solution is convergent on the trivial solution and that it achieves global asymptotic stability in time.

Finally, **Chapter 8** provides a summary of this thesis and considerations for possible future work in these areas.

8.3 Recommendations for Future Research

I suggest here some ideas for related work in the future:

1. Use of local fractional analytical methods to solve wave equations with local fractional derivatives, using the:
    - iterative method
    - integral iterative method
    - ADM
    - new iterative method.
2. FEM for NPDE with quintic B-spline and septic B-spline methods
3. numerical solutions for integro-PDEs.

REFERENCES

References

[1] D. Bleecker, and G. Csordas, *Basic partial differential equations*, Chapman and Hall, New York, 1995.

[2] F. John, *Partial differential equations*, Springer-Verlag, New York, 1982. https://doi.org/10.1007/978-1-4684-9333-7

[3] S. J. Farlow, *Partial differential equations for scientists and engineers*, Dover, New York, 1993.

[4] G. B. Whitham, *Linear and nonlinear waves*, John Wiley, New York, 1976.

[5] L. Lam, *Nonlinear physics for beginners*, World Scientific, Singapore, 1998. https://doi.org/10.1142/1037

[6] F. B. Hildebrand, *Introduction to numerical analysis*, 2nd Edition, Dover Books on Mathematics, Dover Publications, Inc., USA, 1987.

[7] G. Strang, *Introduction to linear algebra*, 5th Edition, Wellesley-Cambridge Press, Cambridge: CUP, 2016.

[8] G. D. Smith, *Numerical solution of partial differential equations: finite difference methods*, Oxford Applied Mathematics and Computing Science Series, 1986.

[9] C. B. Owens, *Implementation of B-splines in a conventional finite element framework*, Masters thesis, Texas A&M University, 2009, Available electronically from .https://hdl handle.net/1969.1/ETD-TAMU-2009-05-469.

[10] R. Riesenfeld, *Application of B-spline approximation to geometric problems of computer aided design*, Proc. May 19-22, 1975, National Computer Conference and Exposition, Syracuse University, vol. 59, no. January 1973, pp. 17–19, 1973, https://ui.adsabs.harvard.edu/abs/1975SPIE5917R.

[11] D. F. Rogers, *An introduction to NURBS: with historical perspective*, Morgan Kaufman Publishers, New York, 2001.

[12] C. de Boor, "On the calculation with B-splines," *J. Approx. Theory*, vol. 6, pp. 50–62, 1972. https://doi.org/10.1016/0021-9045(72)90080-9

[13] K. Verspille, *Computer-aided design applications of the rational B- spline approximation form*, PhD thesis, Syracuse University, 1975.

[14] L. Piegel, and W. Tiller, "Curve and surface constructions using rational B-splines," *CAD*, vol. 19, pp. 485–498, 1987. https://doi.org/10.1016/0010-4485(87)90234-X

[15] W. Tiller, "Rational B-splines for curve and surface representation," *IEEE Comput. Graph. App*., vol. 3, no. 6, pp. 61–69, 1983. https://doi.org/10.1109/MCG.1983.263244

[16] P. Kagan, A. Fischer, and P. Bar-Yoseph, "New B-spline finite element approach for geometrical design and mechanical analysis," *Int. J. Numer. Methods Eng*., vol. 41, pp. 435–458, 1998. https://doi.org/10.1002/(SICI)1097-0207(19980215)41:3%3C435::AID-NME292%3E3.0.CO;2-U

[17] R. Sevilla, S. Fernandez-Mendez, and A. Huerta, "NURBS-enhanced finite element method (NEFEM)," *Int. J. Numer. Methods Eng*., vol. 76, no. 1, pp. 56–83, 2008. https://doi.org/10.1002/nme.2311

[18] K. Inoue, Y. Kikuchi, and T. Masuyama, "A NURBS finite element method for product shape design," *J. Eng. Des*., vol. 16, no. 2, pp. 157– 174, 2005. https://doi.org/10.1080/01405110500033127

[19] J. Cottrell, T. Hughes, and A. Reali, "Studies of refinement and continuity in isogeometric structural analysis," *Comput. Methods Appl. Mech. Eng*., vol. 196, pp. 4160–4183, 2007. https://doi.org/10.1016/j.cma.2007.04.007

[20] P. Kagan, A. Fischer, and P. Bar-Yoseph, “Mechanically based models: adaptive refinement for B-spline finite element,” *Int. J. Numer. Methods Eng*., **57**, pp. 1145–1175, 2003. https://doi.org/10.1002/nme.717

[21] C. J. Li, and R. H. Wang, “A new 8-node quadrilateral spline finite element,” *J. Comp. Appl. Math*., vol. 195, pp. 54–65, 2006. https://doi.org/10.1016/j.cam.2005.07.017

[22] M. Kumar, “A second order spline finite difference method for singular two-point boundary value problems,” *Appl. Math. Comp.,* vol. 142, pp. 283–290, 2003. https://doi.org/10.1016/S0096-3003(02)00302-8

[23] M. Kumar, “A fourth-order spline finite difference method for singular two-point boundary value problems,” *Int. J. Comput. Math*., vol. 80, no. 12, pp. 1499–1504, 2003. https://doi.org/10.1080/0020716031000148179

[24] J. Rashidinia, R. Mohammadi, R. Jalilian, and M. Ghasemi, “Convergence of cubic-spline approach to the solution of a system of boundary-value problems,” *Appl. Math. Comp.,* vol. 192, pp. 319–331, 2007. https://doi.org/10.1016/j.amc.2007.03.008

[25] M. Kadalbajoo, and V. Aggarwal, “Fitted mesh B-spline method for solving a class of singular singularly perturbed boundary value problems,” *Int. J. Comput. Math.*, vol. 82, no. 1, pp. 67–76, 2005. https://doi.org/10.1080/00207160412331291080

[26] J. Brown, M. Bloor, M. S. Bloor, and M. Wilson, “The accuracy of B-spline finite element approximations to PDE surfaces,” *Comput. Methods Appl. Mech. Eng.,* no. 158, pp. 221–234, 1998. https://doi.org/10.1016/S0045-7825(98)00252-7

[27] H. Caglar, N. Caglar, and K. Elfaituri, “B-spline interpolation compared with finite difference, finite element and finite volume methods which applied to two-point boundary value problems,” *Appl. Math. Comp.,* vol. 175, pp. 72–79, 2006. https://doi.org/10.1016/j.amc.2005.07.019

[28] D. Pullman, and J. Schaff, “A comparison of 3-D spline variational and finite-element solutions for a cross-ply laminate with a circular hole,” *Mech. Compos. Mater. Struct.*, vol. 5, pp. 309–325, 1998. https://doi.org/10.1080/10759419808945904

[29] T. Mizusawa, “Application of the spline element method to analyze the bending of skew plates,” *Comput. Struct*., vol. 53, no. 2, pp. 439–448, 1994. https://doi.org/10.1016/0045-7949(94)90215-1

[30] A. Leung, and F. Au, “Spline finite elements for beam and plate,” *Comput. Struct*., vol. 37, no. 5, pp. 717–729, 1990. https://doi.org/10.1016/0045-7949(90)90100-G

[31] J. Kong, and Y. Cheung, “Application of the spline finite strip to the analysis of shear-deformable plates,” *Comput. Struct*., vol. 46, no. 6, pp. 985–988, 1993. https://doi.org/10.1016/0045-7949(93)90083-P

[32] S. Pengcheng, H. PeiXing, and L. Yongxia, “Vibration analysis of plates using the multivariable spline element method,” *Int. J. Solids. Struct.*, vol. 29, no. 24, pp. 3289–3295, 1992. https://doi.org/10.1016/0020-7683(92)90041-Q

[33] A. Gupta, J. Kiusalaas, and M. Saraph, “Cubic B-spline for finite element analysis of axisymmetric shells,” *Comput. Struct*., vol. 38, no. 4, pp. 621–632, 1991. https://doi.org/10.1016/0045-7949(91)90042-K

[34] S. Hyun, C. Kim, J. Son, S. Shin, and Y. Kim, “An efficient shape optimization method based on FEM and B-spline curves and shaping a torque converter clutch disk,” *Finite Elem. Anal. Des*., no. 40, pp. 1803–1815, 2004. https://doi.org/10.1016/j.finel.2004.01.005

[35] F. Zhong, and L. Yuqiu, “Linear analysis of tall buildings using spline elements,” *Eng. Struct*., vol. 13, January, pp. 27–33, 1991. https://doi.org/10.1016/0141-0296(91)90005-W

[36] S. Kutluay, and A. Esen, “A B-spline finite element method for the thermistor problem with the modified electrical conductivity,” *Appl. Math. Comp.*, vol. 156, pp. 621–632, 2004. https://doi.org/10.1016/j.amc.2003.08.014

[37] E. Aksan, “Quadratic B-spline finite element method for numerical solution of the Burgers’ equation,” *Appl. Math. Comp.*, vol. 174, pp. 884–896, 2006. https://doi.org/10.1016/j.amc.2005.05.020

[38] A. Ali, G. Gardner, and L. Gardner, “A collocation solution for Burgers’ equation using cubic B-spline finite elements,” *Comput. Methods Appl. Mech. Eng.*, vol. 100, pp. 325–337, 1992. https://doi.org/10.1016/0045-7825(92)90088-2

[39] I. Patlashenko, and T. Weller, “Cubic B-spline collocation method for nonlinear static analysis of panels under mechanical and thermal loadings,” *Comput. Struct*., vol. 49, pp. 89–96, 1993. https://doi.org/10.1016/0045-7949(93)90127-Y

[40] J. H. Ahlberg, E. N. Nilson, and J. L. Walsh, *The theory of splines and their applications. Mathematics in science and engineering,* A series of monographs and textbooks, 1st Edition, Elsevier, vol. 38, 1 January 1967.

[41] R.L. Burden, and J.D. Faires, *Numerical analysis*, 9th Edition, Brookscole, Boston, 2011.

[42] A. Najmuddin, and K.F. Deeba, “Study of polynomial and nonpolynomial spline based approximation”, *Int. J. Curr. Adv. Res*., vol. 8., no.1(E), pp. 16986–16990, 2019.

[43] K. Nishimoto, *An essence of Nishimoto’s fractional calculus (calculus in the 21st century): integrations and differentiations of arbitrary order*, Descartes Press Company, 1991.

[44] K. S. Miller, and B. Ross, *An introduction to the fractional calculus and fractional differential equations*, Wiley, 1993.

[45] M. Francesco, *Fractional calculus and waves in linear viscoelasticity: an introduction to mathematical models*, Imperial College Press, London,

World Scientific, Singapore, May 2010.

[46] W. G. Glöckle, and T. F. Nonnenmacher, "Fractional relaxation and the time-temperature superposition principle," *Rheologica Acta*, vol. 33, no. 4, pp. 337–343, 1994. https://doi.org/10.1007/BF00366960

[47] B. West, M. Bologna, and P. Grigolini, *Physics of fractal operators*, Springer Science & Business Media, 2012.

[48] I. Podlubny, *Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications*, 1st Edition, Elsevier, 1998.

[49] K. Aissani, M. Benchohra, and K. Ezzini "Fractional Integro-differential equations with state-dependent delay," *Discussiones Mathematicae. Differential Inclusions, Control and Optimization*, vol. 34, pp. 153–167, 2014.

[50] W. M. Abd-Elhameed, and Y. H. Youssri, "New ultraspherical wavelets spectral solutions for fractional Riccati differential equations," *Abstr. Appl. Anal.*, vol. 2014, pp. 1–8, 2014. https://doi.org/10.1155/2014/626275

[51] J. Munkhammar, *Riemann–Liouville fractional derivatives and the Taylor-Riemann series*, Project report, U. U. D. M. P. Department of Mathematics, 2004.

[52] J. Kurzweil, *Ordinary differential equations: introduction to the theory of ordinary differential equations in the real domain*, North Holland, Elsevier, 2014.

[53] F. Ikeda, "A numerical algorithm of discrete fractional calculus by using inhomogeneous sampling data," *Trans. SICE*, vol. 42, no. 8, pp. 941–948, 2006. https://doi.org/10.9746/sicetr1965.42.941

[54] E. M. E. Elbarbary, and S. M. El-Sayed, “Higher order pseudospectral differentiation matrices,” *Appl. Numer. Math*., vol. 55, no. 4, pp. 425–438, 2005. https://doi.org/10.1016/j.apnum.2004.12.001

[55] M. S. Taqqu, “Benoit Mandelbrot and fractional Brownian motion,” *Stat. Sci*., vol. 28, no. 1, pp. 131–134, 2013. https://doi.org/10.1214/12-STS389

[56] W. G. Glöckle, and T. F. Nonnenmacher, “A fractional calculus approach to self-similar protein dynamics,” *Biophys. J*, vol. 68, no. 1, pp. 46–53, 1995. https://doi.org/10.1016/S0006-3495(95)80157-8

[57] S. Yousefi, and M. Razzaghi, “Legendre wavelets method for the nonlinear Volterra–Fredholm integral equations,” *Math. Comput. Simul*., vol. 70, no. 1, pp. 1–8, 2005. https://doi.org/10.1016/j.matcom.2005.02.035

[58] A. Arikoglu, and I. Ozkol, “Solution of fractional integra-differential equations by using fractional differential transform method,” *Chaos, Solit. Fractals*, vol. 40, no. 2, pp. 521–529, 2009. https://doi.org/10.1016/j.chaos.2007.08.001

[59] H. H. Sherief, A. M. A. El-Sayed, and A. M. Abd El-Latief, “Fractional order theory of thermo elasticity,” *Int. J. Solids. Struct.,* vol. 47, no. 2, pp. 269–275, 2010. https://doi.org/10.1016/j.ijsolstr.2009.09.034

[60] J. Deng and L. Ma, “Existence and uniqueness of solutions of initial value problems for nonlinear fractional differential equations,” *Appl. Math. Lett*., vol. 23, no. 6, pp. 676–680, 2010. https://doi.org/10.1016/j.aml.2010.02.007

[61] S. Momani, “An explicit and numerical solutions of the fractional KdV equation,” *Math. Comput. Simul.,* vol. 70, no. 2, pp. 110–118, 2005, https://doi.org/10.1016/j.matcom.2005.05.001

[62] I. Podlubny, *Fractional differential equations,* Academic Press, New York, 1999.

[63] K. S. Miller, and B. Ross, *An introduction to the fractional calculus and fractional differential equations,* John Wiley and Sons, Canada, 1993.

[64] A. Y. Luchko, and R. Groreflo, *The initial value problem for some fractional differential equations with the Caputo derivative*, Preprint series A08-98, Fachbreich Mathematik und Informatik, Freic Universitat, Berlin, 1998.

[65] C. Li, D. Qian, and Y. Chen, "On Riemann–Liouville and Caputo derivatives," *Discret. Dyn. Nat. Soc.*, vol. 2011, 2011, https://doi.org/10.1155/2011/562494

[66] K. B. Oldham, and J. Spanier, *The fractional calculus theory and applications of differentiation and integration to arbitrary order*, Academic Press, New York, pp. 219–223, 1974.

[67] G. Adomian, and R. Rach, "On the solution of algebraic equations by the decomposition method," *J. Math. Anal. Appl.*, vol. 105, no. 1, pp. 141–166, 1985, https://doi.org/10.1016/0022- 247X(85)90102-7

[68] G. Adomian, "A review of the decomposition method in applied mathematics," *J. Math. Anal. Appl.*, vol. 135, no. 2, pp. 501–544, 1988, https://doi.org/10.1016/0022-247X(88)90170-9

[69] G. Adomian, "Explicit solutions of nonlinear partial differential equations," vol. 88, pp. 117–126, 1997. https://doi.org/10.1016/S0096-3003(96)00141-5

[70] O. D. Makinde, B. I. Olajuwon, and A. W. Gbolagade, "Adomian decomposition Approach to a boundary layer flow with thermal radiation past a moving vertical porous plate," *Int. J. Appl. Math. Mech.*, vol. 3, no. 3, pp. 62–70, 2007.

[71] Y. Chen, and H. L. An, "Numerical solutions of coupled Burgers equations with time- and space-fractional derivatives," *Appl. Math. Comput.*, vol. 200, no. 1, pp. 87–95, 2008, https://doi.org/10.1016/j.amc.2007.10.050

[72] M. Dehghan, and M. Tatari, "The use of Adomian decomposition method for solving problems in calculus of variations," *Math. Probl. Eng.*, vol. 2006, no. August 2005, pp. 1–12, 2006, https://doi.org/10.1155/MPE/2006/65379

[73] S. Saha Ray, and R. K. Bera, “Solution of an extraordinary differential equation by Adomian decomposition method,” *J. Appl. Math.*, vol. 2004, no. 4, pp. 331–338, 2004, https://doi.org/10.1155/S1110757X04311010

[74] N. H. Sweilam, and M. M. Khader, “Approximate solutions to the nonlinear vibrations of multiwalled carbon nanotubes using Adomian decomposition method,” *Appl. Math. Comput.*, vol. 217, no. 2, pp. 495–505, 2010, https://doi.org/10.1016/j.amc.2010.05.082

[75] Z. Odibat, and S. Momani, “Numerical methods for nonlinear partial differential equations of fractional order,” *Appl. Math. Model.*, vol. 32, no. 1, pp. 28–39, 2008, https://doi.org/10.1016/j.apm.2006.10.025

[76] A. M. Wazwaz, “A new algorithm for calculating adomian polynomials for nonlinear operators,” *Appl. Math. Comput.*, vol. 111, no. 1, pp. 33–51, 2000, https://doi.org/10.1016/s0096-3003(99)00063-6

[77] K. Abbaoui, and Y. Cherruault, “New ideas for proving convergence of decomposition methods,” *Comput. Math. with Appl.*, vol. 29, no. 7, pp. 103–108, 1995, https://doi.org/10.1016/0898-1221(95)00022-Q

[78] G. Adomian, *Solving frontier problems of physics: the decomposition method*, Kluwer Academic Publishers, Boston, 1994. https://doi.org/10.1007/978-94-015-8289-6

[79] G. Adomian, “Nonlinear dissipative wave equations,” *Appl. Math. Lett.*, vol. 11, no. 3, pp. 125–126, 1998, https://doi.org/10.1016/S0893-9659(98)00044-5

[80] I. Dağ, B. Saka, and D. Irk, “Application of cubic B-splines for numerical solution of the RLW equation,” *Appl. Math. Comput.*, vol. 159, no. 2, pp. 373–389, 2004, https://doi.org/10.1016/j.amc.2003.10.020

[81] I. Dağ, D. Irk, and B. Saka, “A numerical solution of the Burgers’ equation using cubic B-splines,” *Appl. Math. Comput.*, vol. 163, no. 1, pp. 199–211, 2005, https://doi.org/10.1016/j.amc.2004.01.028

[82] A. K. Khalifa, K. R. Raslan, and H. M. Alzubaidi, “A collocation method with cubic B-splines for solving the MRLW equation,” *J. Comput. Appl. Math.*, vol. 212, no. 2, pp. 406–418, 2008, https://doi.org/10.1016/j.cam.2006.12.029

[83] T. S. El Danaf, and F. E. I. Abdel Alaal, “Non-polynomial spline method for the solution of the dissipative wave equation,” *Int. J. Numer. Methods Heat Fluid Flow*, vol. 19, no. 8, pp. 950–959, 2009, https://doi.org/10.1108/09615530910994441

[84] R. C. Mittal and R. K. Jain, “Cubic B-splines collocation method for solving nonlinear parabolic partial differential equations with Neumann boundary conditions,” *Commun. Nonlinear Sci. Numer. Simul.*, vol. 17, no. 12, pp. 4616–4625, 2012, https://doi.org/10.1016/j.cnsns.2012.05.007

[85] Z. A. Zaki, “Quadratic non-polynomial spline method for solving the dissipative wave equation,” *International Journal of Innovative Studies in Sciences and Engineering Technology*, vol. 1, no. 1, pp. 1–6, 2015.

[86] T. S. EL-Danaf, K. R. Raslan, and K. K. Ali, “Collocation method with cubic B-splines for solving the generalized regularized long wave equation,” *Int. J. Numer. Methods Appl.*, vol. 15, no. 1, pp. 39–59, 2016. https://doi.org/10.17654/nm015010039

[87] Ö. Ersoy Hepson and İ. Dağ, “The numerical approach to the Fisher’s equation via trigonometric cubic B-spline collocation method,” *Commun. Numer. Anal.*, vol. 2017, no. 2, pp. 91–100, 2017. https://doi.org/10.5899/2017/cna-00293

[88] M. K. Iqbal, M. Abbas, and N. Khalid, “New cubic B-spline approximation for solving non-linear singular boundary value problems arising in physiology,” *Commun. Math. Appl.*, vol. 9, no. 3, pp. 377–392, 2018. https://doi.org/10.26713/cma.v9i3.802

[89] A. Başhan, “An effective application of differential quadrature method based on modified cubic B-splines to numerical solutions of the KdV equation,” *Turkish J. Math.*, vol. 42, no. 1, pp. 373–394, 2018, https://doi.org/10.3906/mat-1609-69

[90] A. Başhan, “An efficient approximation to numerical solutions for the Kawahara equation via modified cubic B-spline differential quadrature method,” *Mediterr. J. Math.*, vol. 16, no. 1, pp. 1–19, 2019, https://doi.org/10.1007/s00009-018-1291-9

[91] A. Iqbal, N. N. Abd Hamid, and A. I. Ahmad, “Cubic B-spline Galerkin method for numerical solution of the coupled nonlinear Schrödinger equation,” *Math. Comput. Simul.*, vol. 174, pp. 32–44, 2020, https://doi.org/10.1016/j.matcom.2020.02.017

[92] Z. Ahmed, N. Ahmad, M. I. Ghonamy, and N. Rashid, “On the numerical solution of the dissipative wave equation at midpoints,” *Journal of Critical Reviews.*, vol. 7, no. 19, pp. 8623–8632, 2020.

[93] M. J. Ablowitz, *Nonlinear dispersive waves: asymptotic analysis and solitons*, Cambridge University Press, 2011.

[94] M. A. Ramadan, T. S. El-Danaf, and F. E. I. Abd Alaal, “Application of the non- polynomial spline approach to the solution of the Burgers equation’,” *Open Appl. Math. J.*, vol. 1, no. 1, pp. 15–20, 2007, https://doi.org/10.2174/1874114207010115O2

[95] V. A. Galaktionov, and S. I. Pohozaev, “Third-order nonlinear dispersive equations: shocks, rarefaction, and blowup waves,” *Comput. Math. Math. Phys.*, vol. 48, no. 10, pp. 1784–1810, 2008, https://doi.org/10.1134/S0965542508100060

[96] R. Comerford, “Computing,” *IEEE Spectr.*, vol. 37, no. 1, pp. 45–50, 2000, https://doi.org/10.1109/6.815438

[97] J. H.Miller, “On the location of zeros of certain classes of polynomials with application to numerical analysis,” *J. Inst. Math. Appl.*, vol. 8, pp. 397–406, 1971. https://doi.org/10.1093/imamat/8.3.397

[98] I. A. Tirmizi, Fazal-i-Haq, and Siraj-ul-Islam, “Non-polynomial spline solution of singularly perturbed boundary-value problems,” *Appl. Math. Comput.*, vol. 196, no. 1, pp. 6–16, 2008, https://doi.org/10.1016/j.amc.2007.05.029

[99] O. A. Taiwo, and O. M. Ogunlaran, “A non-polynomial spline method for solving linear fourth-order boundary-value problems,” *Int. J. Phys. Sci*., vol. 6, no. 13, pp. 3246–3254, 2011, https://doi.org/10.5897/IJPS11.042

[100] B. Lin, “Non-polynomial splines method for numerical solutions of the regularized long wave equation,” *Int. J. Comput. Math*., vol. 92, no. 8, pp. 1591–1607, 2015, https://doi.org/10.1080/00207160.2014.950254

[101] M. Kudu, and I. Amirali, “Method of lines for third order partial differential equations,” *J. Appl. Math. Phys*., vol. 2, pp. 33–36, https://doi.org/10.4236/jamp.2014.22005

[102] T. S. El-danaf, K. R. Raslan, and K. K. Ali, “Non-polynomial spline method for solving the generalized regularized long wave equation,” *Commun. Math. Model. Appl*., vol. 2, no. 2, pp. 1–17, 2017.

[103] M. Li, X. Ding, and Q. Xu, “Non-polynomial spline method for the time-fractional nonlinear Schrödinger equation,” *Adv. Differ. Equations*, vol. 2018, no. 1, pp. 1–15, 2018, https://doi.org/10.1186/s13662-018-1743-3

[104] T. Sultana, A. Khan, and P. Khandelwal, “A new non-polynomial spline method for solution of linear and non-linear third order dispersive equations,” *Adv. Differ. Equations*, vol. 2018, no. 1, pp. 1–14, 2018, https://doi.org/10.1186/s13662-018-1763-z

[105] A. Khan and Shahna, “Non-polynomial quadratic spline method for solving fourth order singularly perturbed boundary value problems,” *J. King Saud Univ. Sci*., vol. 31, no. 4, pp. 479–484, 2019, https://doi.org/10.1016/j.jksus.2017.08.006

[106] A. M. Wazwaz, “An analytic study on the third-order dispersive partial differential equations,” *Appl. Math. Comput*., vol. 142, no. 2–3, pp. 511–520, 2003, https://doi.org/10.1016/S0096-3003(02)00336-3

[107] Siraj-Ul-Islam, M. A. Khan, I. A. Tirmizi, and E. H. Twizell, “Non-polynomial spline approach to the solution of a system of third-order boundary-

value problems," *Appl. Math. Comput.*, vol. 168, no. 1, pp. 152–163, 2005, https://doi.org/10.1016/j.amc.2004.08.044

[108] A. Aminataei, and M. Sarboland, "Numerical solution of two-dimensional shrodinger equation using multivariate quasi-interpolation scheme," *Proceedings of the 44th Annual Iranian Mathemathics Conference*, pp. 27–30, Ferdowsi University of Mashhad, August. 2013.

[109] N. Masmoudi and K. Nakanishi, "From nonlinear Klein–Gordon equation to a system of coupled nonlinear Schrödinger equations," *Math. Ann.*, vol. 324, no. 2, pp. 359– 389, 2002, https://doi.org/10.1007/s00208-002-0342-4

[110] J. N. Homenuke, *Numerical study of the nonlinear Klein–Gordon equation*, PhD thesis, University of British Columbia, 2004.

[111] K. R. Khusnutdinova, "Coupled Klein–Gordon equations and energy exchange in two-component systems," *Eur. Phys. J. Spec. Top.*, vol. 147, no. 1, pp. 45–72, 2007, https://doi.org/10.1140/epjst/e2007-00202-0

[112] M. Dehghan, and A. Shokri, "Numerical solution of the nonlinear Klein–Gordon equation using radial basis functions," *J. Comput. Appl. Math.*, vol. 230, no. 2, pp. 400–410, 2009, https://doi.org/10.1016/j.cam.2008.12.011

[113] R. Sassaman, M. Edwards, F. Majid, and A. Biswas, "1-soliton solution of the coupled nonlinear Klein–Gordon equations," vol. 1, no. 1, pp. 30–37, 2010.

[114] Q. Li, Z. Ji, Z. Zheng, and H. Liu, "Numerical solution of nonlinear Klein–Gordon equation using lattice Boltzmann method," *Appl. Math.*, vol. 2, no. 12, pp. 1479–1485, 2011, https://doi.org/10.4236/am.2011.212210

[115] Y. Wu, and B. Ge, "A multiplicity result for the non-homogeneous Klein–Gordon–Maxwell system in rotationally symmetric bounded domains," *J. Inequalities Appl.*, vol. 2013, no. 1, pp. 1–12, 2013, https://doi.org/10.1186/1029-242x-2013-583

[116] P. Krämer, “The method of multiple scales for nonlinear Klein–Gordon and Schrödinger equations,” Diploma thesis, Karlsruhe Institute of Technology, Germany, 2013.

[117] S. J. Chen, and L. Li, “Multiple solutions for the nonhomogeneous Klein–Gordon equation coupled with Born-Infeld theory on R3,” *J. Math. Anal. Appl.*, vol. 400, no. 2, pp. 517–524, 2013, https://doi.org/10.1016/j.jmaa.2012.10.057

[118] P. F. Guo, K. M. Liew, and P. Zhu, “Numerical solution of nonlinear Klein-Gordon equation using the element-free kp-Ritz method,” Appl. Math. Model., vol. 39, no. 10–11, pp. 2917–2928, 2015, doi: 10.1016/j.apm.2014.11.025.

[119] M. Sarboland, and A. Aminataei, “Numerical solution of the nonlinear Klein–Gordon equation using multiquadric quasi-interpolation scheme,” *Univers. J. Appl. Math.*, vol. 3, no. 3, pp. 40–49, 2015, https://doi.org/10.13189/ujam.2015.030302

[120] N. Raza, A. R. Butt, and A. Javid, “Approximate solution of nonlinear Klein–Gordon equation using Sobolev gradients,” *J. Funct. Spaces*, vol. 2016, pp. 1–8, 2016, https://doi.org/10.1155/2016/1391594

[121] J. Rashidinia, and M. Jokar, “Numerical solution of nonlinear Klein–Gordon equation using polynomial wavelets,” *Adv. Intell. Syst. Comput.*, vol. 441, pp. 199–214, 2016, https://doi.org/10.1007/978-3-319-30322-2_14

[122] H. Shi, and H. Chen, “Multiple positive solutions for nonhomogeneous Klein–Gordon–Maxwell equations,” *Appl. Math. Comput.*, vol. 337, pp. 504–513, 2018, https://doi.org/10.1016/j.amc.2018.05.052

[123] F. F.Ghazi, and L. N. M. Tawfiq, “Coupled Laplace-decomposition method for solving Klein–Gordon equation,” *Int. J. Modern Math. Sci*., vol,. 8, no. 1, pp. 31–41, 2020

[124] R. L. Bagley, and P. J. Torvik, “A theoretical basis for the application of fractional calculus to viscoelasticity,” *J. Rheol*., vol. 27, no. 3, pp. 201–210, 1983. https://doi.org/10.1122/1.549724

[125] L. Debnath, *Nonlinear partial differential equations for scientists and engineers*, Birkhauser, Boston, 1997. https://doi.org/10.1007/978-1-4899-2846-7

[126] A. C. Metaxas, and R. J. Meredith, *Industrial microwave heating*, Peter Peregrinus, London, 1993.

[127] Z. Odibat, and S. Momani, "A generalized differential transform method for linear partial differential equations of fractional order," *Appl. Math. Lett.*, vol. 21, pp. 194–199, 2008. https://doi.org/10.1016/j.aml.2007.02.022

[128] M. Garg, P. Manohar, and S. L. Kalla, "Generalized differential transform method to space-time fractional telegraph equation," *Int. J. Differ. Equations*, vol. 2011, 2011, https://doi.org/10.1155/2011/548982

[129] Z. Zhao and C. Li, "Fractional difference/finite element approximations for the time–space fractional telegraph equation," *Appl. Math. Comput.*, vol. 219, no. 6, pp. 2975–2988, 2012, https://doi.org/10.1016/j.amc.2012.09.022

[130] J. F. Gómez Aguilar, and D. Baleanu, "Solutions of the telegraph equations using a fractional calculus approach," *Proc. Rom. Acad. Ser. A–Math. Phys. Tech. Sci. Inf. Sci.*, vol. 15, no. 1, pp. 27–34, 2014.

[131] M. Abedi-Varaki, S. Rajabi, V. Ghorbani, and F. Hosseinzadeh, "Solution of the space-fractional telegraph equations by using HAM," *Ciência e Nat.*, vol. 37, p. 320, 2015, https://doi.org/10.5902/2179460x20789

[132] H. Lopushanska, and V. Rapita, "Inverse coefficient problem for the semi-linear fractional telegraph equation," *Electron. J. Differ. Equations*, vol. 2015, no. 153, pp. 1–13, 2014.

[133] H. Khan, C. Tunç, and R. A. Khan, "Approximate analytical solutions of space-fractional telegraph equations by Sumudu Adomian decomposition method," Applications and Applied Mathematics, vol. 13, no. 2, pp. 781–802, 2018.

[134] Kamran, M. Uddin, and A. Ali, “On the approximation of time-fractional telegraph equations using localized kernel-based method,” *Adv. Differ. Equations*, vol. 2018, no. 1, 2018, https://doi.org/10.1186/s13662-018-1775-8

[135] L. Wei, L. Liu, and H. Sun, “Numerical methods for solving the time-fractional telegraph equation,” *Taiwan. J. Math.*, vol. 22, no. 6, pp. 1509–1528, 2018, https://doi.org/10.11650/tjm/180503

[136] S. Mohammadian, Y. Mahmoudi, and F. D. Saei, “Solution of fractional telegraph equation with Riesz space-fractional derivative,” *AIMS Math.*, vol. 4, no. 6, pp. 1664–1683, 2019, https://doi.org/10.3934/math.2019.6.1664

[137] T. Akram, M. Abbas, A. I. Ismail, N. H. M. Ali, and D. Baleanu, “Extended cubic B-splines in the numerical solution of time fractional telegraph equation,” *Adv. Differ. Equations*, vol. 2019, no. 1, pp. 1–20, 2019, https://doi.org/10.1186/s13662-019-2296-9

[138] K. Kumar, R. K. Pandey, and S. Yadav, “Finite difference scheme for a fractional telegraph equation with generalized fractional derivative terms,” *Phys. A Stat. Mech. Appl.*, vol. 535, p. 122271, 2019, https://doi.org/10.1016/j.physa.2019.122271

[139] A. Ali, and N. H. M. Ali, “On skewed grid point iterative method for solving 2D hyperbolic telegraph fractional differential equation,” *Adv. Differ. Equations*, vol. 2019, no. 1, 2019, https://doi.org/10.1186/s13662-019-2238-6

[140] M. Hosseininia, and M. H. Heydari, “Meshfree moving least squares method for nonlinear variable-order time fractional 2D telegraph equation involving Mittag–Leffler non-singular kernel,” *Chaos, Solit. Fractals*, vol. 127, pp. 389–399, 2019, https://doi.org/10.1016/j.chaos.2019.07.015

[141] M. Bouaouid, K. Hilal, and S. Melliani, “Nonlocal telegraph equation in frame of the conformable time-fractional derivative,” *Adv. Math. Phys.*, vol. 2019, no. 4, 2019, https://doi.org/10.1155/2019/7528937

[142] S. Mohammadian, Y. Mahmoudi, and F. D. Saei, “Analytical approximation of time-fractional telegraph equation with Riesz space-fractional derivative,” *Differ. Equations Appl.*, vol. 12, no. 3, pp. 243–258, 2020, https://doi.org/10.7153/dea-2020-12-16

[143] L. Wu, Y. Pan, and X. Yang, “An efficient alternating segment parallel finite difference method for multi-term time fractional diffusion-wave equation,” *Comput. Appl. Math.*, vol. 40, no. 2, 2021, https://doi.org/10.1007/s40314-021-01455-0

[144] Y. M. Hamada, “Solution of a new model of fractional telegraph point reactor kinetics using differential transformation method,” *Appl. Math. Model.*, vol. 78, pp. 297–321, 2020, https://doi.org/10.1016/j.apm.2019.10.001

[145] A. Devi, and M. Jakhar, “A new computational approach for solving fractional order telegraph equations,” J. *Sci. Res.*, vol. 13, no. 3, pp. 715–732, 2021, https://doi.org/10.3329/jsr.v13i3.50659

[146] A. E. Hamza, M. Z. Mohamed, E. M. Abd Elmohmoud, and M. Magzoub, “Conformable Sumudu transform of space-time fractional telegraph equation,” *Abstr. Appl. Anal.*, vol. 2021, pp.1-6, 2021, https://doi.org/10.1155/2021/668299.

[147] O. F. Azhar, M. Naeem, F. Mofarreh, and J. Kafle, “Numerical analysis of the fractional-order telegraph equations,” *J. Funct. Spaces*, vol. 2021,pp. 1-14, 2021, https://doi.org/10.1155/2021/2295804

[148] W. Ibrahim, and L. K. Bijiga, “Neural network method for solving time-fractional telegraph equation,” *Math. Probl. Eng.*, vol. 2021, 2021, https://doi.org/10.1155/2021/7167801

[149] N. Vieira, M. M. Rodrigues, and M. Ferreira, “Time-fractional telegraph equation of distributed order in higher dimensions,” *Commun. Nonlinear Sci. Numer. Simul.*, vol. 102, p. 105925, 2021, https://doi.org/10.1016/j.cnsns.2021.105925

[150] M. M. A. Khater, C. Park, J. R. Lee, M. S. Mohamed, and R. A. M. Attia, “Five semi analytical and numerical simulations for the fractional nonlinear space-

time telegraph equation," *Adv. Differ. Equations*, vol. 2021, no. 1, 2021, https://doi.org/10.1186/s13662-021-03387-9

[151] O. Nikan, Z. Avazzadeh, and J. A. T. Machado, "Numerical approximation of the nonlinear time-fractional telegraph equation arising in neutron transport," *Commun. Nonlinear Sci. Numer. Simul*., vol. 99, p. 105755, 2021, https://doi.org/10.1016/j.cnsns.2021.105755

[152] A. C. Loyinmi, and T. K. Akinfe, "An algorithm for solving the Burgers–Huxley equation using the Elzaki transform," *SN Appl. Sci.*, vol. 2, no. 1, pp. 1–17, 2020, https://doi.org/10.1007/s42452-019-1653-3

[153] B. Batiha, M. S. M. Noorani, and I. Hashim, "Application of variational iteration method to the generalized Burgers–Huxley equation," *Chaos, Solit. Fractals*, vol. 36, no. 3, pp. 660–663, 2008, https://doi.org/10.1016/j.chaos.2006.06.080

[154] Q. Wang, Y. Xiong, W. Huang, and V. G. Romanovski, "Isolated periodic wave trains in a generalized Burgers–Huxley equation," *Electronic Journal of Qualitative Theory of Differential Equations*, no. 4, pp. 1–16, 2022, https://doi.org/10.14232/ejqtde.2022.1.4

[155] B. K. Singh, G. Arora, and M. K. Singh, "A numerical scheme for the generalized Burgers–Huxley equation," *J. Egypt. Math. Soc.*, vol. 24, no. 4, pp. 629–637, 2016, https://doi.org/10.1016/j.joems.2015.11.003

[156] A. Khan, M. T. Mohan, and R. Ruiz-Baier, "Conforming, nonconforming and DG methods for the stationary generalized Burgers–Huxley equation," *J. Sci. Comput.*, vol. 88, no. 3, pp. 1–21, 2021, https://doi.org/10.1007/s10915-021-01563-3

[157] O. Y. Yefimova, and N. Kudryashov, "Exact solutions of the Burgers–Huxley equation, " *J. Appl. Math. Mech.*, vol. 3, pp. 413–420, 2004, https://doi.org/10.1016/S0021-8928(04)00055-3

[158] X. Wang, Z. Zhu, and Y. Lu, “Solitary wave solutions of the generalised Burgers–Huxley equation, ” *J. Phys. A: Math. Gen.*, vol. 23, p. 271, 1990. https://doi.org/10.1088/0305-4470/23/3/011

[159] L. Bin Liu, Y. Liang, J. Zhang, and X. Bao, “A robust adaptive grid method for singularly perturbed Burger–Huxley equations,” *Electron. Res. Arch.*, vol. 28, no. 4, pp. 1439–1457, 2020, https://doi.org/10.3934/era.2020076

[160] R. K. Mohanty, W. Dai and D. Liu. “Operator compact method of accuracy two in time and four in space for the solution of time dependent Burgers–Huxley equation,” *Numer. Algorithms*, vol. 70, no. 3, pp. 591–605, 2015, http://works.bepress.com/weizhong-dai/20/

[161] M. Reza, “B-spline collocation algorithm for numerical solution of the generalised Burger’s–Huxley equation,” *Numer. Methods Partial Differ. Equ.*, vol. 29. pp. 1173–1191, 2013. https://doi.org/10.1002/num.21750

[162] A. J. Khattak, “A computational meshless method for the generalized Burger’s–Huxley equation,” *Appl. Math. Model.*, vol. 33, no. 9, pp. 3718–3729, 2009, https://doi.org/10.1016/j.apm.2008.12.010

[163] R. C. Mittal, and A. Tripathi, “Numerical solutions of generalized Burgers–Fisher and generalized Burgers–Huxley equations using collocation of cubic B-splines,” *Int. J. Comput. Math.*, vol. 92, no. 5, pp. 1053–1077, 2015. https://doi.org/10.1080/00207160.2014.920834

[164] A. Kaushik, and M. D. Sharma, “A uniformly convergent numerical method on non-uniform mesh for singularly perturbed unsteady Burger–Huxley equation,” *Appl. Math. Comput.*, vol. 195, no. 2, pp. 688–706, 2008, https://doi.org/10.1016/j.amc.2007.05.067

[165] N. Jha, and M. Wagley, “A family of quasi-variable meshes high-resolution compact operator scheme for Burger’s–Huxley, and Burger’s–Fisher equation,” *Math. Appl. Sci. Eng.*, vol. 1, no. 4, pp. 286–308, 2020, https://doi.org/10.5206/mase/10837

[166] E. S. Fahmy, and I. Bajunaid, “Approximate solution for the generalized time-delayed Burgers–Huxley equation,” *Far East J. Appl. Math*, vol. 28, no.1, pp. 81-94, January 2007.

[167] I. Hashim, M. S. M. Noorani, and M. R. Said Al-Hadidi, “Solving the generalized Burgers–Huxley equation using the Adomian decomposition method,” *Math. Comput. Model.*, vol. 43, no. 11–12, pp. 1404–1411, 2006, https://doi.org/10.1016/j.mcm.2005.08.017

[168] T. A. Abassy, “Improved Adomian decomposition method,” *Comput. Math. with Appl.*, vol. 59, pp. 42–54, 2010. https://doi.org/10.1016/j.camwa.2009.06.009

[169] G. Adomian, *Solving frontier problems of physics: the decomposition method*, Kluwer Academic Publishers, Boston, 1994. https://doi.org/10.1007/978-94-015-8289-6

[170] T. S. El-Danaf, “Solitary wave solutions for the generalised Burgers–Huxley equation,” *Int. J. Nonlinear Sc. Num. Sim.*, vol. 8, no. 3, pp. 315–318, 2007, https://doi.org/10.1515/ijnsns.2007.8.3.315

[171] X. Deng, “Travelling wave solutions for the generalized Burgers–Huxley equation,” *Appl. Math. Comput.*, vol. 204, no. 2, pp. 733–737, 2008, https://doi.org/10.1016/j.amc.2008.07.020

[172] M. Sari, and G. Gürarslan, “Numerical solutions of the generalized Burgers–Huxley equation by a differential quadrature method,” *Math. Probl. Eng.*, vol. 2009, pp. 1–12, 2009, https://doi.org/10.1155/2009/370765

[173] N. Smaoui, A. El-Kadri, and M. Zribi, “Adaptive boundary control of the forced generalized Korteweg–de Vries–Burgers equation,” *Eur. J. Control*, vol. 16, no. 1, pp. 72–84, 2010, https://doi.org/10.3166/ejc.16.72-84

[174] J. Biazar, and F. Mohammadi, “Application of differential transform method to the generalized Burgers–Huxley equation 2. The model problem,” *An. Int. J. Appl. Appl. Math.*, vol. 0, no. 10, pp. 1726–1740, 2010.

[175] A. G. Bratsos, “A fourth order improved numerical scheme for the generalized Burgers–Huxley equation,” *Am. J. Comput. Math.*, vol. 1, no. 3, pp. 152–158, 2011, https://doi.org/10.4236/ajcm.2011.13017

[176] J. E. MacAs-Daz, J. Ruiz-Ramrez, and J. Villa, “The numerical solution of a generalized Burgers–Huxley equation through a conditionally bounded and symmetry-preserving method,” *Comput. Math. with Appl.*, vol. 61, no. 11, pp. 3330–3342, 2011, https://doi.org/10.1016/j.camwa.2011.04.022

[177] M. El-Kady, S. M. El-Sayed, and H. E. Fathy, “Development of Galerkin method for solving the generalized Burger’s–Huxley equation,” *Math. Probl. Eng.*, vol. 2013, 2013, https://doi.org/10.1155/2013/165492

[178] S. Saha Ray, and A. K. Gupta, “On the solution of Burgers–Huxley and Huxley equation using wavelet collocation method,” *C.–Comput. Model. Eng. Sci.*, vol. 91, no. 6, pp. 409–424, 2013, https://doi.org/10.3970/cmes.2013.091.409

[179] J. Liu, H. Y. Luo, G. Mu, Z. Dai, and X. Liu, “New multi-soliton solutions for generalized Burgers–Huxley equation,” *Therm. Sci.*, vol. 17, no. 5, pp. 1486–1489, 2013, https://doi.org/10.2298/TSCI1305486L

[180] E. A. Az-zo, “On the reduced differential transform method and its application to the generalized Burgers-Huxley equation,” *Applied mathematical sciences*, vol. 8, no. 177, pp. 8823–8831, 2014. https://doi.org/10.12988/ams.2014.410835

[181] V. J. Ervin, J. E. Macías-Díaz, and J. Ruiz-Ramírez, “A positive and bounded finite element approximation of the generalized Burgers–Huxley equation,” *J. Math. Anal. Appl.*, vol. 424, no. 2, pp. 1143–1160, 2015, https://doi.org/10.1016/j.jmaa.2014.11.047

[182] B. Inan, “Finite difference methods for the generalized Huxley and Burgers–Huxley equations,” *Kuwait J. Sci.*, vol. 44, no. 3, pp. 20–27, 2017.

[183] N. Kumar, and S. Singh, “Numerical solution of Burgers–Huxley equation using improved nodal integral method,” *Ninth International Conference on*

Computational Fluid Dynamics(*ICCFD), Istanbul, Turkey,* pp. 1–12, 2016.

[184] J. A. T. Machado, A. Babaei, and B. P. Moghaddam, "Highly accurate scheme for the cauchy problem of the generalized Burgers–Huxley equation," *Acta Polytech. Hungarica*, vol. 13, no. 6, pp. 183–195, 2016, https://doi.org/10.12700/aph.13.6.2016.6.10

[185] B. Inan, "A new numerical scheme for the generalized Huxley equation," *Bull. Math. Sci. Appl.*, vol. 16, pp. 105–111, 2016, https://doi.org/10.18052/www.scipress.com/bmsa.16.105

[186] I. Wasim, M. Abbas, and M. Amin, "Hybrid B-spline collocation method for solving the generalized Burgers–Fisher and Burgers–Huxley equations," *Math. Probl. Eng.*, vol. 2018, no. 4, 2018, https://doi.org/10.1155/2018/6143934

[187] A. R. Appadu, B. Inan, and Y. O. Tijani, "Comparative study of some numerical methods for the Burgers–Huxley equation," *Symmetry (Basel).*, vol. 11, no. 1333, 2019, https://doi.org/10.3390/sym11111333

[188] Y. Fu, "Persistence of travelling wavefronts in a generalized Burgers–Huxley equation with long-range diffusion," *J. Appl. Anal. Comput.*, vol. 9, no. 1, pp. 363–372, 2019, https://doi.org/10.11948/2019.363

[189] L. Y. Sun, and C. G. Zhu, "Cubic B-spline quasi-interpolation and an application to numerical solution of generalized Burgers–Huxley equation," *Adv. Mech. Eng.*, vol. 12, no. 11, pp. 1–8, 2020, https://doi.org/10.1177/1687814020971061

[190] M. A. Khan, M. Ali Akbar, N. H. M. Ali, and M. Abbas, "The new auxiliary method in the solution of the Generalized Burgers-Huxley equation," *J. Prime Res. Math.*, vol. 16, no. 2, pp. 16–26, 2020.

[191] A. Kumar, and M. T. Mohan, "Large deviation principle for occupation measures of two dimensional stochastic convective Brinkman–Forchheimer equations," *Stoch. Anal. Appl.*, vol. 40, no. 5, pp. 798-811, 2021, https://doi.org/10.1080/07362994.2021.2005626

[192] A. G. Kushner, and R. I. Matviichuk, “Exact solutions of the Burgers–Huxley equation via dynamics,” *J. Geom. Phys*., vol. 151, p. 103615, 2020, https://doi.org/10.1016/j.geomphys.2020.103615

[193] L. Ebiwareme, “Banach contraction method and tanh–coth approach for the solitary and exact solutions of Burger–Huxley and Kuramoto–Sivashinsky equations,” *Int. J. Math. Trends Technol.*, vol. 67, no. 4, pp. 31–46, 2021, https://doi.org/10.14445/22315373/ijmtt-v67i4p506

[194] M. T. Mohan, and A. Khan, “On the generalized Burgers–Huxley equation: existence, uniqueness, regularity, global attractors and numerical studies,” *Discret. Contin. Dyn. Syst. –Ser. B*, vol. 26, no. 7, pp. 3943–3988, 2021, https://doi.org/10.3934/dcdsb.2020270

[195] F. Abergel, and R. Temam, “On some control problems in fluid mechanics,” *Theor. Comput. Fluid Dyn*., vol. 1, no. 6, pp. 303–325, 1990, https://doi.org/10.1007/BF00271794

[196] A. Balogh, and M. Krstić, “Burgers’ equation with nonlinear boundary feedback: H1 stability, well-posedness and simulation,” *Math. Probl. Eng*., vol. 6, no. 2–3, pp. 189–200, 2000, https://doi.org/10.1155/S1024123X00001320

[197] J. A. Burns, and S. Kang, “A control problem for Burgers’ equation with bounded input/output,” *Nonlinear Dyn*., vol. 2, no. 4, pp. 235–262, 1991, https://doi.org/10.1007/BF00045296

[198] H. Choi, R. Temam, P. Moin, and J. Kim, “Feedback control for unsteady and its application to the stochastic Burgers equation,” *J. Fluid Mech*., vol. 253, pp. 509–543, 1993. https://doi.org/10.1017/S0022112093001880

[199] K. Ito, and S. Kang, “A dissipative feedback control synthesis for systems arising in fluid dynamics,” *SIAM J. Control Optim*., vol. 32, no. 3, pp. 831–854, 1994. https://doi.org/10.1137/S0363012991222619

[200] T. Kobayashi, “Adaptive regulator design of a viscous Burgers system by boundary control,” *IMA J. Math. Control Inform*., vol. 18, no. 3, pp. 427–437, 2001. https://doi.org/10.1093/imamci/18.3.427

[201] M. Krstic, “On global stabilization of Burgers’ equation by boundary control,” *Proc. IEEE Conf. Decis. Control*, vol. 3, pp. 3498–3499, 1998, https://doi.org/10.1109/cdc.1998.758248

[202] N. Smaoui, “Nonlinear boundary control of the generalized Burgers equation,” *Nonlinear Dyn*., vol. 37, no. 1, pp. 75–86, 2004. https://doi.org/10.1023/B:NODY.0000040023.92220.09

[203] N. Smaoui, A. El-Kadri, and M. Zribi, “Nonlinear boundary control of the unforced generalized Korteweg–de Vries–Burgers equation,” *Nonlinear Dyn*., vol. 60, no. 4, pp. 561–574, 2010, https://doi.org/10.1007/s11071-009-9615-8

[204] N. Smaoui, A. El-Kadri, and M. Zribi, “Adaptive boundary control of the unforced generalized Korteweg–de Vries–Burgers equation,” *Nonlinear Dyn*. Vol. 69, pp. 1237–1253, 2012. https://doi.org/10.1007/s11071-012-0343-0

[205] S. Momani, S. Abuasad, and Z. Odibat, “Variational iteration method for solving nonlinear boundary value problems,” *Appl. Math. Comput*., vol. 183, no. 2, pp. 1351–1358, 2006, https://doi.org/10.1016/j.amc.2006.05.13872–84, 2010, https://doi.org/10.3166/ejc.16.72-84

[206] S. Momani, and Z. Odibat, “Numerical approach to differential equations of fractional order,” *J. Comput. Appl. Math*., vol. 207, no. 1, pp. 96–110, 2007, https://doi.org/10.1016/j.cam.2006.07.015

[207] E. Babolian, and J. Saeidian, “Analytic approximate solutions to Burgers, Fisher, Huxley equations and two combined forms of these equations,” *Commun. Nonlinear Sci. Numer. Simul.*, vol. 14, no. 5, pp. 1984–1992, 2009, https://doi.org/10.1016/j.cnsns.2008.07.019

[208] J. Satsuma, “Exact solutions of Burgers’ equation with reaction terms,” *in Topics in soliton theory and exactly solvable nonlinear equations*, World Science Publishing, Singapore, pp. 255–262, 1986.

[209] A. L. Hodgkin, and A. F. Huxley, “A quantitative description of membrane current and its applications to conduction and excitation in nerve,” *J. Physiol*., vol. 117, pp. 500–544, 1952. https://doi.org/10.1113/jphysiol.1952.sp004764

[210] Y. C. Hon, and X. Z. Mao, “An efficient numerical scheme for Burgers’ equation,” *Appl. Math. Comp.*, vol. 95, no. 1, pp 37–50, 1998, https://doi.org/10.1016/S0096-3003(97)10060-1

[211] E. Coddington, and N. Levinson, *Theory of ordinary differential equations, international series in pure and applied mathematics*, McGraw-Hill, the University of Michigan, 10th Edition ,1955.

[212] M. Abramowitz, and I. A. Stegun, *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, Dover, New York, 1964.

[213] G. Acosta, and R. G. Durán, “An optimal Poincaré inequality in L^1 for convex domains,” *Proc. Am. Math. Soc*., vol. 132, no. 1, pp. 195–202, 2003, https://doi.org/10.1090/s0002-9939-03-07004-7

[214] H. K. Khalil, “*Nonlinear systems,” Prentice-Hall*, New Jersey, vol. 2, no. 5, pp. 1–5, 1996.

[215] E. J. Ali, “A new technique of initial boundary value problems using Adomian decomposition method,” *Int. Math. Forum*, vol. 7, no. 17, pp. 799–814, 2012.

[216] Z. M. Alaofi, T. S. El-Danaf, and S. S. Dragomir, “A numerical solution of the dissipative wave equation by means of cubic b-spline method,” *J. Phys. Commun*., vol. 5, no. 10, pp. 105014, 2021, https://doi.org/10.1088/2399-6528/ac2940.

[217] Z. M. Alaofi, T. S. El-Danaf, F. A. Alaal, and S. S. Dragomir, “Quartic non-polynomial spline for solving the third-order dispersive partial differential equation,” *American Journal of Computational Mathematics*, pp. 189–206, 2021, https://doi.org/10.4236/ajcm.2021.113013.

[218] Z. M. Alaofi, T. S. El-Danaf, F. E. I. A. Alaal, and S. S. Dragomir, “Numerical investigations of the coupled nonlinear non-homogeneous partial differential equations,” *J. Appl. Comput. Mech.*, vol. 8, no. 3, pp. 1054–1064, 2022, https://doi.org/10.22055/jacm.2022.39327.3388.

[219] Z. M. Alaofi, T. S. El-Danaf, and S. S. Dragomir, “Comparing solutions to the nonlinear dissipative wave equation,” *J. Appl. Math. Phys.*, vol. 10, no. 4, pp. 1281–1296, 2022, https://doi.org/10.4236/jamp.2022.104040.

[220] Z. M. Alaofi, T. S. El-Danaf, A. Hadhoud, and S. S. Dragomir, “Computational analysis for solving the linear space-fractional telegraph equation,” *Open J. Model. Simul.*, vol. 10, no. 3, pp. 267–282, 2022, https://doi.org/10.4236/ojmsi.2022.103014.

[221] Z. M. Alaofi, T. S. Ali, F. A. Alaal, and S. S. Dragomir, “Adaptive boundary control for the dynamics of the generalized Burgers–Huxley equation,” *Open J. Appl. Sci.*, vol. 12, no. 8, pp. 1416–1438, 2022, https://doi.org/10.4236/ojapps.2022.128098.