NUMERICAL STUDY OF HIGHER ORDER DIFFERENTIAL EQUATIONS USING SOME QUINTIC BASIS FUNCTIONS

Thesis Submitted for the award of the degree of

DOCTOR OF PHILOSOPHY

in

Mathematics

By

Navneet Kaur

Registration Number: 11816322

Supervised by

Dr. Varun Joshi
Associate Professor, Department of Mathematics
LPU, Phagwara

LOVELY PROFESSIONAL UNIVERSITY
PUNJAB
2023

DECLARATION

I, hereby declared that the presented work in the thesis entitled "**NUMERICAL STUDY OF HIGHER ORDER DIFFERENTIAL EQUATIONS USING SOME QUINTIC BASIS FUNCTIONS**" in fulfilment of degree of **Doctor of Philosophy (Ph. D.)** is outcome of research work carried out by me under the supervision Dr. Varun Joshi, working as Associate Professor, in the Department of Mathematics, School of Chemical Engineering and Physical Sciences of Lovely Professional University, Punjab, India. In keeping with general practice of reporting scientific observations, due acknowledgements have been made whenever work described here has been based on findings of other investigator. This work has not been submitted in part or full to any other University or Institute for the award of any degree.

(Signature of Scholar)

Navneet Kaur

11816322

Department of Mathematics

School of Chemical Engineering and Physical Sciences

Lovely Professional University,

Punjab, India

CERTIFICATE

This is to certify that the work reported in the Ph. D. thesis entitled "**Numerical Study of Higher Order Differential Equations Using some Quintic Basis Functions**" submitted in fulfillment of the requirement for the reward of degree of **Doctor of Philosophy (Ph.D.)** in the Department of Mathematics, School of Chemical Engineering and Physical Sciences, is a research work carried out by Navneet Kaur, 11816322, is bonafide record of her original work carried out under my supervision and that no part of thesis has been submitted for any other degree, diploma or equivalent course.

(Signature of Supervisor)

Dr. Varun Joshi

Associate Professor

Department of Mathematics,

School of Chemical Engineering and Physical Sciences

Lovely Professional University,

Punjab, India

ABSTRACT

Recent studies in science and engineering showed that partial differential equations (PDEs) that are nonlinear in nature are used to represent dynamics of many systems, including engineering, diffusion, biology, physics, chaos theory, electromagnetic, and many more. To apprehend or predict the behavior of natural phenomena, mathematical models have huge importance, which made researchers pay attention to the advancement of science and technology. A mathematical model describes the different systems with the usage of formulas, mathematical rules, methods, and theories. Most of the influential mathematical concepts are invented by using differential and integral operators known as calculus, which are considered for simulating and modeling the different mechanisms occurring in the environment. There are numerous engineering problems, which has huge application in our lives, from where some of the problems are so complex that it is quite hard or impossible to find their exact (precise) solution. So, in that scenario, numerical solutions (num. sol.) of these problems or equations get a huge sight and came into knowledge.

This thesis concerns obtaining the numerical solution of a one-dimension third order Korteweg-de Vries (KdV) equation, fourth-order Kuramoto-Sivashinsky (KS) equation, one and two dimension fourth order Extended Fisher Kolmogorav (EFK) with the implementation of three new regimes “quintic Uniform Algebraic Hyperbolic (QUAH) tension B-spline, quintic Hyperbolic B-spline (QHBS), and quintic Uniform Algebraic Trigonometric (QUAT) tension B-spline” with a renowned method known as “Differential Quadrature Method (DQM)”. DQM is used to approximate the derivatives. Then, to obtain the weighting coefficients these quintic splines HBS, QUAH and QUAT are implemented with DQM and the original boundary value problem was then transformed into an ordinary differential equation. The DQ technique has a significant benefit over the previous approaches since it restrain the perturbation from finding improved outcomes for the given nonlinear equations. The Runge-Kutta 43 technique is

then used to resolve the resultant ordinary differential equation (ODE). The precision of the suggested scheme is then illustrated numerically by resolving different test problems and calculating the error norms L_2, L_∞ and GRE. For quick and simple access, calculated results are displayed graphically and in tabular format. Suggested schemes' stability is also examined by using eigenvalues.

In simple language, Numerical solution or approximate solutions are those which are approaching or nearly resembles to the exact solutions. So, in our work, we first solve those problems approximately which already had an exact solution for the purpose of testing or benchmarking, so that applicability and effectiveness of our proposed method can be checked. Consequently, after evaluating the technique's efficiency, proposed technique can be extended to get the solution of more complex and convincing problems for which exact solutions are not available.

There are a variety of techniques for finding numerical answers to nonlinear PDEs, and we employ one of them known as the Differential Quadrature Method (DQM) along with three different quintic B-splines (QBS). Three B-splines used are quintic Hyperbolic B-spline (HBS), QUAH tension BS, and QUAT tension BS.

DQM is a mathematical procedure to solve differential equations. With DQM approach, the weighted average of each functional value at specific places over the entire domain of computation can be applied for the approximation of the spatial derivatives of unknown functions at any grid points. DQM is a numerical approach for solving initial as well as boundary value problems. It was created in the first few years of 1970s by the late Richard Bellman and his colleagues, since then, it has been effectively applied to a range of challenges from a range of sectors, including engineering and also the sciences.

Next, polynomials can be defined piecewise using spline functions. It has been discovered that the interaction flexibility of curve and surface design is improved by these B-spline functions. One major benefit of using a B-spline basis is that the number

of control points in an object's control graph can be changed without affecting the basis function's order.

For instance, in our work, three extensive quintic B-spline (QBS) are adapted for the numerical simulation of three different equations using the DQM. Differential quadrature method (DQM) is appertained for the time discretization of the space discretized system occasioned by B-splines. Moreover, for the easy excess to readers graphical demonstrations of the numerical findings are shown for evaluation with the precise resolution and numerical findings described in the literature. DQM's primary task or purpose is to evaluate weighting coefficients. Consequently, to calculate the weighting coefficients, quintic B-spline function is modified by using DQM and result is reformed in the form of five-band matrix. Therefore, the basis function is utilised to transform the initial boundary value equation into the ODEs, and they are then resolved with the use of the Runge-Kutta method 43 scheme. This is done after the application of DQM and the calculation of the weighting coefficients.

If, we talk about any of the equation, whose numerical solution are obtained, we get so fascinating results of coupled Korteweg-de Vries (CKdV) equation, Kuramoto-sisvashinsky (KS) equations, Extended Fisher Kolmogorov (EFK) equation. This thesis contains 5 chapters. In Chapter 1, brief introduction and literature survey are given for Differential quadrature method (DQM), B-splines, HBS, UAH tension BS, UAT tension BS. Description about the proposed methodology is given. Moreover some conditions as stability condition, different error norms and their formulas are also discussed in this chapter.

In Chapter 2, numerical solution of CKdV equation is obtained. KdV simulates the evolution of plasma waves, internally long waves in the ocean, waves of shallow-water, and crystal lattice waves. KdV system of equation has different type of soliton solutions most commonly named as single soliton solution, double solitons, three solitons interaction, and also birth of solitons. In this chapter, N-soliton where $N \geq 3$ are also

discussed. Hence, CKdV equation is integrable as it has multiple soliton solutions. Also, by definition of soliton, solitons preserve its shape, and amplitude when move with a constant velocity and even after the collision. Same results are obtained by using the proposed technique, and it is seen that with the increasing time, solitons move towards the right and preserve the properties.

In Chapter 3, numerical approximation of KS equation is obtained. In the 1970s, a nonlinear partial differential equation of order four was introduced known as KS equation, also known as the canonical evolution equation and explains the range of physical contexts. Also, it designates about the turbulence in responsive system and model the disseminative uncertainties in a laminar flame front.

In Chapter 4, numerical approximation of EFK equation in one and two dimension is obtained with the implementation of three quintic splines along with DQM. The EFK system of equation has huge applications in various fields as formation of pattern, transmission of domain wall, description about dynamics of brain tumors, mezoscopic model of a phase transition, traveling waves, and extensively used in variety of physical phenomena including plasma physics, population growth, hydrodynamics, spread of infectious diseases, thermonuclear reaction etc.

To assess the applicability of the suggested approach, different equations are resolved numerically and different error norms are calculated for the comparison purposes, which are appeared to be very less. Along with that stability of the technique is also checked by using the method named as matrix stability method and to check the accuracy of our proposed technique error norms L_2 and L_∞ are calculated, which are appeared to be very less and approaching to zero telling us the effectiveness of proposed scheme. Additionally, it is demonstrated that the approach is invariably stable.

The accomplished numerical outcomes are presented with two-dimensional (2D) and three- dimensional (3D) figures, where the comparison with the exact (or precise) solution is also demonstrated. It is clinched that the presented technique is a well-

organized and operative procedure for elucidating the numerical or approximate solution for the complex nonlinear PDEs.

ACKNOWLEDGEMENTS

First and foremost, I praise and thank God, whose support helped me to stay positive in each and every situation and countless blessings have made it possible for me to complete this work. I would like to acknowledge my supervisor, Dr. Varun Joshi, an Associate Professor at Lovely Professional University in Phagwara, whose unwavering support, mentoring, and dedication to research inspired me to finish my research work. His insightful vision, opportune leadership, wise management, immense knowledge, and plentiful experience have assisted me throughout my Ph.D study. I am so grateful to him for giving me invaluable advices, his patience and continuous support, despite his busy schedule, throughout my research work. I am thankful to Dr. Lovi Raj Gupta, Pro Vice-Chancellor at Lovely Professional University, for allowing me to carry out this research work. I would like to thank Dr. Kailash Juglan, Head of School, School of Chemical Engineering and Physical Sciences, Dr. Rekha, Head of RDEC for providing me necessary facilities at Lovely Professional University to continue my work. I would also like to thank Dr. Geeta Arora for her sensible suggestions, and irreplaceable involvement in my research work. I am grateful to all the staff members of the Department of Mathematics, Lovely Professional University, for their time-to-time assistance. I am also thankful to the creators and upholders of MATLAB, as this made this research valuably possible. I am also thankful to Gurpreet Kaur, Preeti Kumari, Bazila Shameem, and Pavas for their friendlier and helping nature. At last, I am whole heartedly thankful to my grandfather Mr. Harbans Singh (who always motivated me to do better, his hand full of prayers on my head was the best thing i had), parents- Mr. Jatinderpal Singh, Mrs. Tajinderpal Kaur, my sister- Navdeep Kaur and brother- Manjodh Singh and my whole family for their love, support and incessant encouragement, who always promote hard work and nurture my dreams with their invaluable understanding of life.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1. Introduction

Differential equations have important implications in every field including economics, physics, engineering, as well as biology. The study of differential equations includes a study of their solutions or the properties of their solutions. There exist different forms of differential equations, which includes ordinary or partial differential equation, homogeneous or heterogeneous equations, and linear or non-linear differential equations. Each of these forms has its own importance or we can say they are significant in their own way. For instance, ordinary differential equations (ODEs) are used to compute electrical flow, illustrate thermodynamics ideas, and motion of an object like a pendulum. Also in medical terminology, they are employed to monitor the progression of disease graphically. Partial differential equations (PDEs) are used to formulate the problem mathematically and aid the solution of problems having more than one variable like heat or sound waves propagating, fluid movement, electrostatics, thermodynamics, electrodynamics, elasticity, etc. Linear Differential equations (LDE) detect the electric circuit's current by determining the motion of an object with air resistance. A large number of fascinating and significant events that are seen in space have been substantially mathematically modeled using Non-linear differential equations (NLDEs). Homogeneous differential equations (HDEs) are employed in the chemical sector as well as in the fields of economics, aerospace, and transportation. Non-homogeneous differential equation of second order predicts the vibrating mass's amplitude in a scenario of near-resonant.

For the modeling of various natural phenomena like sciences, there arise nonlinear partial differential equations. Moreover, Non-linear systems with initial as well

as with boundary value problems have interesting and very useful applicability in engineering and other disciplines, as PDEs are used to represent a large number of engineering mathematical models. Also, these equations have a huge importance in science as various phenomena in science such as the physical laws of structural mechanics, electromagnetic fields, quantum mechanics, and fluid flow diffusion all can be written as initial and boundary value problems.

To apprehend or predict the behavior of natural phenomena, mathematical models have huge importance, which made researchers pay attention to the advancement of science and technology. A mathematical model describes the different systems with the usage of formulas, mathematical rules, methods, and theories. Most of the influential mathematical concepts are invented by using differential and integral operators known as calculus, which are considered for simulating and modeling the different mechanisms occurring in the environment. Prominent mathematicians Leibniz and Newton presented the idea to elucidate physical properties centered on the rate of change.

Nonlinear problems can have two types of solutions, analytical as well as numerical. Numerous researchers solved these nonlinear models, either getting the numerical or analytical solution. Also, in all the thrust areas of engineering, mathematical and physical sciences, the solution of nonlinear problems has huge importance with their solutions. For physical systems, nonlinear systems of equations have an interesting characteristic that can be acknowledged by the solutions of nonlinear problems either numerically or analytically. But due to the complexity of nonlinear PDEs, it becomes difficult to solve them analytically or to have an exact solution to the problem. In such cases, it becomes reasonably important to solve these complex problems numerically. Often when exact solutions of differential equations are not available or there is no way to locate closed-form solutions using an analytical approach, then solutions can be approximated numerically.

The approximate elucidation is formulated by means of functional values at distinct knot values. If we talk about the relationship between the functional values and derivatives in partial differentiation, numerical discretization happens to be the bridge between them. An approximate solution is also referred to as a numerical solution (num. sol.). Currently, numerous numerical discretization methods are accessible to solve initial and boundary value problems. They include finite difference (FD), finite volume (FV), and finite element (FE) which are classified as low-order methods while pseudo spectral and spectral methods are classified as global methods. If we talk about the difference between these methods, that difference is the choice of the base function. For spectral methods, a base function is considered as an infinitely differential function whereas in low-order methods domain is divided into subdomains and hence base function is specified with each element. In other words, Spectral approaches can be seen as a whole-space approximation method and as an extension of FE methods. Lower-order approaches require a lot more grid points to get the required level of accuracy. There may not be a need for a solution at any of the grid points in some differential equation applications. Therefore, while discretizing the domain, a higher-order approach is required that provides precise answers with a less number of grid points.

Therefore, Bellman et al. invented the Differential Quadrature Method (DQM) in the beginning of 1970s which was instigated by the study of Integral quadrature. This technique uses a much fewer grid points to get precise numerical results. The main motive of DQM is to evaluate the weighting factors. Firstly, Bellman et al. introduced two techniques for the calculation of weighting factors of the derivative of the 1^{st} order. Among these methods, usually, the first approach was adopted as it permits choosing arbitrary grid points. But for a large number of the algebraic system of equations, its matrix becomes ill-conditioned because of which grid points used by the early applications of this scheme are not more than 13 [1]. To overcome these drawbacks many efforts are made by researchers. The major revolution was done by Shu and Richard [2]

in which all the techniques were generalized under the high-order approximation of polynomials and analysis of linear vector space.

1.2. Differential Quadrature Method

In DQM, a derivative of a function can be estimated as a linear summation of its value at different node points about the solution dominion of a problem.

$$f_x(x_i) = \frac{df}{dx}\Big| x_i = \sum_{j=1}^{N} a_{ij} \, . f\left(x_j\right) \text{ for } i = 1,2, \dots, N,$$

where a_{ij} signifies weighting coefficients and N is number of grid points in whole domain.

The decision regarding the weighting coefficients is essential. The weighting coefficients are calculated using DQM with a variety of test functions, including the cubic B-splines, Lagrange interpolation polynomial, modified B-splines, trigonometric B-splines, Quartic B-spline, Quintic B-spline, etc. For example Legendre polynomial and spline functions have been used by Bellman et al. [3], Using a Lagrange polynomial, Quan and Chang [4, 5] offered explicit formulation. To compute weighting coefficients, Shu and Richards [2] introduce the notion of linear vector space and polynomial approximation. Başhan and Yağmurlu [6] used cubic B-spline based DQM, Zhong [7], Başhan [8-11] used Quintic B-spline based DQM, Korkmaz and Dag [12] presented polynomial DQM, and Quartic B-spline using DQM, Başhan [13] applied quartic Basis-spline employing DQM, Mittal et al. [14] applied quintic B-spline for the test function with DQM, Arora and Joshi [15] represented the updated version of the trigonometric cubic (order four) B-spline based DQM, Singh and Kumar [16] employed extended cubic B-spline in a modified form based DQM, Mohammed and Saeed [17] used G-spline, Korkmaz et al. [18] used quartic B-spline to solve Burgers' equation, Arora et al. [19] used cubic B-spline with DQM in a modified form for the estimation of Burgers' equation, Korkmaz et al. [20] solved advection-diffusion equation using cubic Basis-spline with DQM, Tamsir et al. [21] used exponential modified cubic B-spline DQM,

Başhan [22] gives solution of Kawahara equation using cubic B-spline in a modified form with DQM, Tamsir et al. [23] solved Fisher's reaction-diffusion system using cubic trigonometric B-spline DQM, Başhan [24] used quintic B-spline with DQM to solve Schrödinger equation numerically, Mittal et al. [25] solved the KS equation with the implementation of quintic B-spline based DQM, Joshi et al.[26] solved 2-D Benjamin-Bona-Mahony-Burgers equation with the use of cubic B-spline in a modified form based DQM, Barrera et al. [27] introduced general spline DQM based on quasi-interpolation for optimal approximation, Joshi and Kapoor [28] used UAH tension BS based DQM for a 2D nonlinear differential equation, Kaur and Kanwar [29] approximate KS equation with the implementation of cubic trigonometric B-spline based DQM, Zhu et al. [30] offered cubic trigonometric B-splines DQM for the solution of fractional advection-diffusion equation numerically, Tamsir et al. [31] proposed cubic hyperbolic B-spline based DQM for the numerical solution of 3D wave equations, Başhan et al. [32] solved Korteweg-de Vries-Burgers' equation for a numerical solution with the implementation of quantic B-spline based DQM, Korkmaz [33] solved advection-diffusion equation numerically using quartic and quintic B-splines with DQM, Thoudam [34] used quintic B-spline based DQM.

Any function $G(x,t)$ is known to be approximated by interpolation at N discrete grid points as:

$$G(x,t) = \sum_{j=1}^{N} d_j G(x_j, t), \qquad (1.1)$$

where d_j represents the basis function, and $G(x_j,t)$ gives the functional value at different grid points x_j for the function $G(x,t)$.

As certain PDEs have a tough time having an analytical solution. Therefore, the functional values at the specified grid point serve as the most common representation of the approximate solution. The dilemma of how to connect the derivatives of PDE and the

functional values at the mesh points, and the numerical discretization approach can be used to close this gap.

1.2.1. Discretization in one dimension

Let us consider the mesh division system $a = x_1 < x_2 < \cdots < x_N = b$ with $h = x_{i+1} - x_i$, where $i = 1,2, \ldots, N$ of a definite period [a, b], with the deliberation that the assumed function $G(x)$ is sufficiently smooth over its solution domain. Then, the Differentiation of the function $G(x)$ concerning $'x'$ for the mesh value x_i can be estimated by a linear addition in the entire domain of all the functional values, can be written as:

$$G_x^{(r)}(x_i) = \frac{d^r G}{dx^r} \, |x_i = \sum_{j=1}^{N} d_{ij}^{(r)} G(x_j), \qquad (1.2)$$

with $i = 1,2,3, \ldots, N \; and \; r = 1,2,3, \ldots, N-1$

where r signifies the derivative's order,r^{th} order weighting coefficients are represented by $d_{ij}^{(r)}$, N provides information on the total number of grid points in the domain., and the index j denotes that $d_{ij}^{(r)}$ represents the conforming weighting factors of the functional value of $G(x_j)$. So, for further calculation, we need to calculate the first, second, and third-order derivative of the function $G(x)$, which can be evaluated by substituting $r = 1,2,3$ in equation *(1.2)*.

Hence, The approximate values of derivatives of 1st, 2nd, 3rd and 4th order can be given as:

$$\begin{aligned} &G_x^{(1)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(1)} G(x_j), \; G_x^{(2)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(2)} G(x_j), \\ &G_x^{(3)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(3)} G(x_j), \; G_x^{(4)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(4)} G(x_j) \end{aligned} \qquad (1.3)$$

where $d_{ij}^{(1)}$, $d_{ij}^{(2)}$, $d_{ij}^{(3)}$, and $d_{ij}^{(4)}$ are the weighting coefficients to be determined using the proposed methodology. In this work, three different quintic splines are used to calculate

the first-order weighting coefficients ($d_{ij}^{(1)}$). After evaluating the weighting coefficients of partial derivatives having first order as explained in section 1.7, their 2^{nd} and higher-order partial derivative can be premeditated by using the formula [1, 35] given below.

$$d_{ij}^{(r)} = r[d_{ij}^{(1)} d_{ii}^{(r-1)} - \frac{d_{ij}^{(r-1)}}{x_i - x_j}], \text{ for } i \neq j \{i, j = 1,2,3, \dots, N\} \text{ and}$$
$$d_{ii}^{(r)} = -\sum_{j=1, j\neq i}^{N} d_{ij}^{(r)}, \text{ for } i = j, \ r = 2,3, \dots, N-1, \tag{1.4}$$

where $d_{ij}^{(r)}$ represents the second, third and fourth-order weighting coefficients for the values $r = 2,3,4$ respectively. So, to evaluate the second order derivative, we get the formula as:

$$d_{ij}^{(2)} = \begin{cases} 2\left[d_{ij}^{(1)} d_{ii}^{(1)} - \dfrac{d_{ii}^{(1)}}{x_i - x_j}\right] & for\ i \neq j \\ -\displaystyle\sum_{i=1, i\neq j}^{N} d_{ij}^{(2)} & for\ i = j \end{cases} \tag{1.5}$$

For the third-order derivative,

$$d_{ij}^{(3)} = \begin{cases} 3\left[d_{ij}^{(1)} d_{ii}^{(2)} - \dfrac{d_{ii}^{(2)}}{x_i - x_j}\right] & for\ i \neq j \\ -\displaystyle\sum_{i=1, i\neq j}^{N} d_{ij}^{(3)} & for\ i = j \end{cases} \tag{1.6}$$

and, fourth-order derivative can be obtained as:

$$d_{ij}^{(4)} = \begin{cases} 4\left[d_{ij}^{(1)} d_{ii}^{(3)} - \dfrac{d_{ii}^{(3)}}{x_i - x_j}\right] & for\ i \neq j \\ -\displaystyle\sum_{i=1, i\neq j}^{N} d_{ij}^{(4)} & for\ i = j \end{cases} \tag{1.7}$$

1.2.2. Discretization in two dimension

Let us consider the mesh division system $a = x_1 < x_2 < \cdots < x_N = b$ and $c = y_1 < y_2 < \cdots < y_M = d$ with a uniform step size $h_1 = x_{i+1} - x_i$, and $h_2 = y_{j+1} - y_j$ respectively in x and y directions, where $i = 1,2, \ldots, N$, $j = 1,2, \ldots, M$ of a definite period $[a, b] \times [c, d]$. With the deliberation that the assumed function $G(x, y, t)$ is sufficiently smooth over its solution domain. Then, the Differentiation of $G(x, y, t)$ concerning (x, y) for mesh value (x_i, y_j) can be estimated by a linear addition in the entire domain of all the functional values. Using the idea of the equation (1.2) spatial derivative of r^{th} order of $G(x, y, t)$ w.r.t. x (preserving y_j fixed) and w.r.t. y (preserving x_i fixed) can be obtained. So, the r^{th} order derivative for the function $G(x, y, t)$ at a point $x = x_i$ along the line $y = y_j$ provided by:

$$G_x^{(r)}(x_i, y_j, t) = \frac{\partial^r G(x_i, y_j, t)}{\partial x^r} = \sum_{k=1}^{N} d_{ik}^{(r)} G(x_k, y_j, t), \qquad (1.8)$$

and, the r^{th} order derivative for the function $G(x, y, t)$ at a point $y = y_j$ along any line $x = x_i$ is provided by:

$$G_y^{(r)}(x_i, y_j, t) = \frac{\partial^r G(x_i, y_j, t)}{\partial y^r} = \sum_{k=1}^{N} d_{jk}^{(r)} G(x_j, y_k, t), \qquad (1.9)$$

where, r is the positive integer, $1 \le i \le N$, $1 \le j \le M$, and $d_{ik}^{(r)}$ and $d_{jk}^{(r)}$ are the respective weighting coefficients, which give the r^{th} derivative at the knots. Further, first-order weighting coefficients can be evaluated by the same procedure as explained in section 1.7, and for higher order weighting coefficients, we have the formula:

$$d_{ik}^{(r)} = \begin{cases} r\left[d_{ik}^{(r)} d_{ii}^{(r-1)} - \dfrac{d_{ii}^{(r-1)}}{x_i - x_j}\right] & for\ i \neq k \\ -\sum\limits_{i=1, i\neq k}^{N} d_{ik}^{(2)} & for\ i = k \end{cases} \qquad (1.10)$$

and

$$d_{jk}^{(r)} = \begin{cases} r\left[d_{jk}^{(r)} d_{jj}^{(r-1)} - \dfrac{d_{jj}^{(r-1)}}{x_i - x_j}\right] & for\ j \neq k \\ -\sum_{j=1, j\neq k}^{M} d_{jk}^{(2)} & for\ j = k \end{cases} \qquad (1.11)$$

After obtaining the weighting factors for both the space variables x and y, the space derivative of respective order (r) can be evaluated as in equations (1.8) and (1.9). After that by using discretization, one can find the necessary differential equation's solution.

1.3. B-spline

B-spline is the abbreviation for the word Basis spline, which was initially used by Isaac Jacob Schoenberg [36] in 1946. In the numerical analysis field of mathematics, a spline function known as B-spline is a function with marginal support for a particular degree, level of smoothness, and domain partition [37]. If we talk about the word spline, it is a device used by architects, shipbuilding industries, or other sectors to draw a curve through the given points so that the curve, its slope, and its curvature are all continuous. Carl de Boor [38] and Prenter [39] provide a comprehensive explanation of the spline theory.

A spline function having order n is known as a piecewise polynomial of degree $n-1$, in a variable x. The points, where these pieces meet are known as knots or grid points. The concept of spline can also be recognized by the problem of fitting a polynomial that passes through the points whose functional values are there. For instance, a linear polynomial can be fitted, if two points are given. In case three such points are given, a quadratic polynomial will be plotted. Also if four, five, or six such points are given, then cubic, quartic, or quintic polynomials can be plotted respectively. Hence with the increment in the number of points, the degree of polynomial also increases, which is not easy to deal with. The piecewise polynomial came to the aid in order to solve this problem. Piecewise polynomials allow approximating the function using different

polynomials over the sub domains, instead of approximating the function using a single polynomial over the whole domain. Hence the piecewise polynomial can be conceptualized for the higher order approximations. But in some cases, it may happen that the interpolation graph is not smooth, as piecewise polynomials are always continuous but may not be continuously differential on the approximation interval. Splines are employed to tackle this problem. So, spline develops a piecewise polynomial approximation that interpolates the given data ensuring that it is continuously differentiable up to a certain degree. Let us consider a partition $x_0 < x_1 < x_2 < \cdots < x_{n-1} < x_n$ uniformly distributed over the domain $[a = x_0, b = x_N]$, where x_i are known as knots. Any function $S(x)$ is known as a spline having degree k, if $p(x)$ a k^{th} degree polynomial in each interval $[x_i, x_{i+1}], where\ i = 0,1, \dots, n-1$ satisfy the property that $p(x)$ and its 1^{st} $(k-1)$ derivatives are continuous everywhere in the domain $[x_0, x_n]$. So, a spline $S(x)$ in the interval $[x_0, x_n]$ can be expressed as:

$$S(x) = \sum_{i=1}^{n} p(x_i),$$

where $p(x)$ is the polynomial of degree k in every section. Therefore, as each portion has $(k+1)$ coefficients, that results in the collection of $n(k+1)$ coefficients for n sections. So to define a spline, we require $n(k+1)$ number of equations. The fundamental aspect of the basis function is the knot sequence x_i. Let the set X be a collection of $N+1$ real numbers known as knots such that $x_0 \leq x_1 \leq x_2 \leq \cdots \leq x_{N-1} \leq x_N$. The knot vector is referred to as uniform if the knots are evenly distributed. Basis spline function of degree k covers k intervals of $(k+1)$ knots.

The B-spline basis function can vary in degree and shape. Consider $B_{m,k}$ denotes the m^{th} B-spline basis function with degree k. The zero degree Basis-spline, where $k = 0$, is just a step function given as

$$B_{m,0} = \begin{cases} 1 & x \in [x_m, x_{m+1}] \\ 0 & otherwise \end{cases}.$$

The Cox de-Boor recurrence relation for m^{th} B-spline of degree k is given by:

$$B_{m,k}(x) = V_{m,k}B_{m,k-1} + \big(1 - V_{m+1,k}\big)B_{m+1,k-1}(x),$$

where, $V_{m,k} = \left(\frac{x-x_m}{x_{m+k}-x_m}\right)$

To create a wide spectrum of differential quadrature methods, various researchers have chosen a variety of B-spline basis functions as B-spline based DQM has a huge application of getting better results of complex nonlinear PDEs. So, in literature for the numerical solution of different NLEs, a range of B-spline is used depending on their type, degree, or order. For instance, [40] used B-spline method for Bratu's equation, in [41] 6^{th}-degree BS approximation is used for the numerical sol. of 5^{th}-order boundary value problem, [42] apply cubic B-spline scaling functions for the numerical findings of Fokker-Planck equation, [43] introduce Cardinal B-spline wavelet numerical method to solve the Generalized Burgers-Huxley equation, [44] used cubic B-spline with finite difference approach for solution of heat and wave equations, in [45] quartic B-spline collocation technique is constructed, [46] solved generalized Kuramoto-Sivashinsky (GKS) equation numerically using FD and collocation technique along with B-spline functions, [47] apply quadratic (i.e. of order 3) B-spline based FE method for the numerical sol. of 1D NL Burgers' system of equation, [48] cubic trigonometric BS is applied with finite difference scheme for 1D hyperbolic equation, [49] solved Lane-Emden equation using four numerical approaches based on the linear B-spline, [50] used quadratic B-spline with Galerkin method for the numerical solution of Fisher's equation, [51] used quadratic and cubic B-spline based Galerkin approach for the solution of Burgers' equation, cubic B-spline in a modified form with the method DQM for numerical estimation of 2D sine-Gordon equation [52], also for Schrödinger equation [53].

1.4. Hyperbolic B-spline (HBS)

The most important concept of polynomial space is B-spline basis functions, spanned by $\{1, x, x^2, \dots, x^k\}$, where k is any random positive integer. Recently, a number

of new spline curve and surface schemes have been put forth for computer-aided geometric design (CAGD). For instance, Nouisser et al. [54] offered 2π periodic trigonometric, while Maes and Bultheel [55] introduced a standard spherical B-spline, algebraic hyperbolic functions are derived by Eddargani et al. [56], high order discretizations of the differential equation are derived using quintic nonpolynomial spline basis functions by Jha and Mohanty [57], Generalized B-splines are introduced by Roman et al. [58], Changeable degree spline basis function (CD-spline) was represented by Shen et al. [59], CD splines are an extension of iterative integral method, which is made up of piecewise polynomials of distinct degrees, a method for the elevation of the degree of a CD-spline is presented by Shen et al. [60], new kind of uniform splines known as hyperbolic polynomial B-splines are presented by Lü et al. [61], which is spanned over the space $\{sinhx, coshx, x^{k-3}, x^{k-4}, \dots, x, 1\}$, where $k \geq 3$ is a positive arbitrary constant. Patel et al. [62] proposed a robust spline adaptive filter for the modeling of nonlinear systems, Campagna and Conti [63] describe the exceptional type of penalized splines known as Hyperbolic-polynomial (HP) splines. Wang and Fang et al. [64] extended the three types of splines as polynomial, trigonometric, and hyperbolic splines by a brand-new spline named UE-spline. Siddiqi et al. [65] explain the ternary non-stationary subdivision schemes which are based upon the HBS. Siddiqi et al. [66] established non-stationary binary schemes having four as well as five-point subdivision schemes based on HBS. Xumin et al. [67] gave a new method to model parameter curves and surfaces based on hyperbolic B-spline surfaces. Zhang and Krause [68] depicted the idea of an extended cubic B-spline which is uniform in nature using combined hyperbolic and trigonometric basis functions. For polynomial space, hyperbolic B-splines exhibit properties that are remarkably similar to those of basic B-splines. Hyperbolic B-splines schemes have the specialty of precisely replicating the curves like hyperbolas and parabolas. In this work, sixth-order HBS is calculated and obtained quintic HBS is implemented with DQM to attain the numerical solution of higher-order PDEs.

Let $H_m^l(x)$ represents the Hyperbolic B-splines with knots at the points x_i of order $'l'$ where the evenly disseminated N grid points are assumed as $a = x_1 < x_2 < \cdots < x_N = b$ on the usual real axis. The functions defined over [a, b] are built upon the B-splines $\{H_{-1}, H_0, \ldots, H_{N+2}\}$.

1.4.1. Hyperbolic B-spline (HBS) of order one

HBS of order 1 denoted by $H_m^1(x)$ is given by:

$$H_m^1(x) = \begin{cases} 1, & x_m \le x \le x_{m+1} \\ 0, & otherwise \end{cases}. \qquad (1.12)$$

For the second and higher order HBS recurrence relation is given by:

$$H_m^l(x) = \frac{sinh(x - x_m)}{sinh(x_{l+m-1} - x_m)} H_m^{l-1}(x) + \frac{sinh(x_{m+l} - x)}{sinh(x_{m+l} - x_{m+1})} H_{m+1}^{l-1}(x). \qquad (1.13)$$

The aforementioned relationships satisfy the following characteristics,

(a): For $l \ge 2, H_m^l \in C^{l-2}(x)$

(b): $H_m^l(x)$ is a function which is piecewise hyperbolic.

(c): $H_m^l(x) \ge 0$

(d): Domain of $H_m^l(x) = [x_m, x_{m+l}]$

(e): $H_m^l \in \Gamma_l$,

where

$$\Gamma_l = \begin{cases} span\left\{\{\sinh(2Lx), \cosh(2Lx)\}_{L=1}^{\left[\frac{l-1}{2}\right]} \cup \{1\}\right\}, & where\ l\ is\ odd \\ span\left\{\{\sinh\big((2L-1)x\big), \cosh((2L-1)x)\}_{L=1}^{\left[\frac{l}{2}\right]}\right\}, & where\ l\ is\ even \end{cases},$$

and is called the hyperbolic polynomial of order l space.

1.4.2. Hyperbolic B-spline of order two

By using the first-order HBS (1.12) and recurrence relation given in (1.13), second-order HBS $H_m^2(x)$ can be formulated as:

$$H_m^2(x) = \begin{cases} \dfrac{sinh(x - x_m)}{sinh(x_{m+1} - x_m)}, & x \in [x_m, x_{m+1}) \\ \dfrac{sinh(x_{m+2} - x)}{sinh(x_{m+2} - x_{m+1})}, & x \in [x_{m+1}, x_{m+2}) \\ 0, & otherwise \end{cases} \tag{1.14}$$

1.4.3. Hyperbolic B-spline of order three

By using second-order HBS (1.14) and the recurrence relation given in (1.13), third-order HBS $H_m^3(x)$ can be formulated as:

$$H_m^3(x) = \frac{1}{\eta_1} \begin{cases} p^2(x_m), & x \in [x_m, x_{m+1}) \\ p(x_m)g(x_{m+2}) + g(x_{m+3})p(x_{m+1}), & x \in [x_{m+1}, x_{m+2}) \\ g^2(x_{m+3}), & x \in [x_{m+2}, x_{m+3}) \\ 0, & otherwise \end{cases} \tag{1.15}$$

where, $\eta_1 = \sinh(2h)\sinh(h)$, $p(x_m) = \sinh(x - x_m)$, $g(x_m) = \sinh(x_m - x)$

1.4.4. Hyperbolic B-spline of order four

By using third-order HBS (1.15) and the recurrence relation given in (1.13), fourth-order HBS $H_m^4(x)$ can be formulated as:

$$H_m^4(x) = \frac{1}{\eta_2} \begin{cases} p^3(x_m), & x \in [x_m, x_{m+1}) \\ \begin{Bmatrix} p^2(x_m)g(x_{m+2}) + p(x_m)g(x_{m+3}) \\ p(x_{m+1}) + g(x_{m+4})p^2(x_{m+1}) \end{Bmatrix} & x \in [x_{m+1}, x_{m+2}) \\ \begin{Bmatrix} p(x_m)g^2(x_{m+3}) + g(x_{m+4})p(x_{m+1}) \\ g(x_{m+3}) + g^2(x_{m+4})p(x_{m+2}) \end{Bmatrix} & x \in [x_{m+2}, x_{m+3}) \\ g^3(x_{m+4}), & x \in [x_{m+3}, x_{m+4}) \\ 0, & otherwise \end{cases} \tag{1.16}$$

where, $\eta_2 = \sinh(3h)\sinh(2h)\sinh(h)$, $p(x_m) = \sinh(x - x_m)$, $g(x_m) = \sinh(x_m - x)$

1.4.5. Hyperbolic B-spline of order five

By using fourth-order HBS (1.16) and the recurrence relation given in (1.13), fifth-order HBS $H_m^5(x)$ can be formulated as:

$$H_m^5(x) = \frac{1}{\eta_3}\begin{cases} p^4(x_m), & x \in [x_m, x_{m+1}) \\ \begin{Bmatrix} p^3(x_m)g(x_{m+2}) + p^2(x_m)g(x_{m+3}) \\ p(x_{m+1}) + p(x_m)g(x_{m+4})p^2(x_{m+1}) \\ +g(x_{m+5})p^3(x_{m+1}) \end{Bmatrix} & x \in [x_{m+1}, x_{m+2}) \\ \begin{Bmatrix} p^2(x_m)g^2(x_{m+3}) + p(x_m)g(x_{m+4}) \\ p(x_{m+1})g(x_{m+3}) + p(x_m)g^2(x_{m+4}) \\ p(x_{m+2}) + g(x_{m+5})p^2(x_{m+1})g(x_{m+3}) \\ +g(x_{m+5})g(x_{m+4})p(x_{m+1})p(x_{m+2}) \\ +g^2(x_{m+5})p^2(x_{m+2}) \end{Bmatrix} & x \in [x_{m+2}, x_{m+3}) \\ \begin{Bmatrix} p(x_m)g^3(x_{m+4}) + g(x_{m+5})p(x_{m+1}) \\ g^2(x_{m+4}) + p(x_{m+2})g(x_{m+4})g^2(x_{m+5}) \\ +p(x_{m+3})g^3(x_{m+5}) \end{Bmatrix} & x \in [x_{m+3}, x_{m+4}) \\ g^4(x_{m+5}), & x \in [x_{m+4}, x_{m+5}) \\ 0 & otherwise \end{cases}, \quad (1.17)$$

where, $\eta_3 = \sinh(4h)\sinh(3h)\sinh(2h)\sinh(h)$, $p(x_m) = \sinh(x - x_m)$, $g(x_m) = \sinh(x_m - x)$.

1.4.6. Hyperbolic B-spline of order six

By using fifth-order HBS (1.17) and the recurrence relation given in (1.13), sixth-order HBS $H_m^6(x)$ can be formulated as:

$$H_m^6(x)$$

$$= \frac{1}{\eta_4}\begin{cases} p^5(x_m), & x \in [x_m, x_{m+1}) \\ \begin{Bmatrix} p^4(x_m)g(x_{m+2}) + p^3(x_m)p(x_{m+1}) \\ g(x_{m+3}) + p^2(x_m)g(x_{m+4})p^2(x_{m+1})p(x_m) \\ g(x_{m+5})p^3(x_{m+1}) + g(x_{m+6})p^4(x_{m+1}) \end{Bmatrix} & x \in [x_{m+1}, x_{m+2}) \\ \begin{Bmatrix} p^3(x_m)g^2(x_{m+3}) + p^2(x_m)g(x_{m+4})p(x_{m+1}) \\ g(x_{m+3}) + p^2(x_m)g^2(x_{m+4})p(x_{m+2}) + p(x_m) \\ g(x_{m+5})p^2(x_{m+1})g(x_{m+3}) + p(x_m)p(x_{m+1}) \\ p(x_{m+2})g(x_{m+4})g(x_{m+5}) + p(x_m)p^2(x_{m+2}) \\ g^2(x_{m+5}) + p^3(x_{m+1})g(x_{m+3})g(x_{m+6}) \\ +p(x_{m+2})p^2(x_{m+1})g(x_{m+4})g(x_{m+6}) \\ +p(x_{m+1})p^2(x_{m+2})g(x_{m+5})g(x_{m+6}) \\ +g^2(x_{m+6})p^3(x_{m+2}) \end{Bmatrix} & x \in [x_{m+2}, x_{m+3}) \\ \begin{Bmatrix} p^2(x_m)g^3(x_{m+4}) + p(x_m)g(x_{m+5})p(x_{m+1}) \\ g^2(x_{m+4}) + p(x_m)g^2(x_{m+5})p(x_{m+2})g(x_{m+4}) \\ +p(x_m)g^3(x_{m+5})p(x_{m+3}) + g(x_{m+6})p^2(x_{m+1}) \\ g^2(x_{m+4}) + g(x_{m+6})p(x_{m+1})g(x_{m+5})p(x_{m+2}) \\ g(x_{m+4}) + g(x_{m+6})p(x_{m+1})g^2(x_{m+5})p(x_{m+3}) \\ +g^2(x_{m+6})p^2(x_{m+2})g(x_{m+4}) + g^2(x_{m+6}) \\ p(x_{m+2})g(x_{m+5})p(x_{m+3}) + g^3(x_{m+6})p^2(x_{m+3}) \end{Bmatrix} & x \in [x_{m+3}, x_{m+4}) \\ \begin{Bmatrix} p(x_m)g^4(x_{m+5}) + g(x_{m+6})p(x_{m+1})g^3(x_{m+5}) \\ +g^2(x_{m+6})p(x_{m+2})g^2(x_{m+2}) \\ +g^3(x_{m+6})p(x_{m+3})g(x_{m+5}) + g^4(x_{m+6})p(x_{m+4}) \end{Bmatrix} & x \in [x_{m+4}, x_{m+5}) \\ g^5(x_{m+6}), & x \in [x_{m+5}, x_{m+6}) \\ 0, & otherwise \end{cases} \quad (1.18)$$

where, $\eta_4 = \sinh(5h)\sinh(4h)\sinh(3h)\sinh(2h)\sinh(h)$, $p(x_m) = \sinh(x - x_m)$, $g(x_m) = \sinh(x_m - x)$.

Table 1.1: The values of quintic hyperbolic B-spline $H_m^6(x)$ basis functions and derivative functions $(H_m^6)'(x)$ at different grid points is given by:

	x_m	x_{m+1}	x_{m+2}	x_{m+3}	x_{m+4}	x_{m+5}	x_{m+6}
$H_m^6(x)$	0	e_1	e_2	e_3	e_2	e_1	0
$(H_m^6)'(x)$	0	e_4	e_5	e_6	$-e_5$	$-e_4$	0

with the values e_i in the table are given below:

$h = x_m - x_{m-1}$,

$e_1 = \frac{sinh^5(h)}{\eta_4}$, $e_2 = \frac{2sinh^2(h)sinh^3(2h)+sinh^3(h)\sinh(2h)\sinh(3h)+sinh^4(h)\sinh(4h)}{\eta_4}$,

$e_3 = \frac{2sinh^3(h)sinh^2(3h)+4sinh^2(h)sinh^2(2h)\sinh(3h)}{\eta_4}$, $e_4 = \frac{5sinh^4(h)\cosh(h)}{\eta_4}$,

$e_5 = \frac{50sinh^4(h)+130sinh^6(h)+80sinh^8(h)}{\eta_4}$, $e_6 = 0$

1.5. Uniform Algebraic Hyperbolic (UAH) Tension B-spline

In recent years, a number of new splines are suggested by researchers which are defined in non-polynomial spaces. For instance, Zhang [69] suggested a set of curves named C-curves with basis $\{sinx, cosx, x, 1\}$, further Zhang [70] introduced two different forms of Changeable basis splines (CB-splines) on the domain [0,1], Koch and Lyche [71] constructed exponential tension B-splines of any order. However, in high-order instances, these bases do not cross over. Then Lü et al. [61] presented an entirely novel type of uniform splines known as Uniform hyperbolic B-splines spanned over the space $\{sinhx, coshx, x^{k-3}, x^{k-4}, \dots, x, 1\}$, where $k \geq 3$ is a positive arbitrary constant. Mainar and Peña [72] studied about shape preservation and stability properties of CB-splines. Chen and Wang [73] introduced a new basis known as the C-Bezier basis spanned over the space $\{1, x, x^2, \dots, x^{n-2}, sinx, cosx\}$ using integral approach. Wang et al. [74] presented NUAT B-splines (Non-uniform algebraic-trigonometric Basis-splines) spanned over the space $\{1, x, x^2, \dots, x^{k-3}, cosx, sinx\}$ with $k \geq 3$. Jena et al. [75, 76] offered subdivision procedure for trigonometric spline curves. Juan and Wang [77] presented two different type of Bezier-basis for algebraic hyperbolic space. Xu and Wang [78] introduced AHT (Algebraic hyperbolic trigonometric), and non-uniform AHT B-spline curves of order k over the space spanned by $\{sinx, cosx, sinhx, coshx, 1, x, \dots, x^{k-5}\}, k \geq 5$. Alinia and Zarebnia [79] aims to introduce a novel tension B-spline known as the hyperbolic-trigonometric tension B-spline of k^{th} order, formulated across the area

$\{\sin(\tau x), cos(\tau x), sinh(\tau x), cosh(\tau x), 1, x, \dots, x^{k-5}\}, k \geq 5$, where τ represents the tension parameter. Fang et al. [80, 81] presented a generalized curve and surface subdivision technique of random order with a tension parameter upon modification of AT and AH B-splines. There have been so many distinct types of splines proposed over various spaces. Each form has advantages of its own.

Following the concept given in [64, 79, 82] a novel spline known as "Uniform algebraic hyperbolic tension B spline abbreviated as (UAH tension B-spline)" is implemented. In the following work, the UAH tension B-spline is considered along with DQM so that the weighting coefficient can be calculated. Contemplate the region $a \leq x \leq b$ which is subdivided into a mesh of uniform length $h = \frac{b-a}{N}$ by the knots $x_i = a + ih, i = 0,1,2, \dots, N$.

$E_{i,k}(x)$ is the UAH tension B-spline of order k with given grid points x_i which are uniformly distributed at $a = x_0 < x_1 < x_2 < \dots \dots \dots < x_N = b$.

1.5.1. UAH tension B-spline of order two

$$E_{i,2}(x) = \begin{cases} \left\{\dfrac{sinh[\tau(x - x_{i-2})]}{sinh(\tau h)}\right\} & x \in [x_{i-2}, x_{i-1}) \\ \left\{\dfrac{sinh[\tau(x_i - x)]}{sinh(\tau h)}\right\} & x \in [x_{i-1}, x_i) \\ 0 & otherwise \end{cases}, \qquad (1.19)$$

where τ is the tension parameter and is equal to $\sqrt{\eta}$ where η is a real number.

and for $k \geq 3$, the recurrence relation for $E_{i,k}$ is given by:

$$E_{i,k}(x) = \int_{-\infty}^{x} \left(\delta_{i,k-1} E_{i,k-1}(x) - \delta_{i+1,k-1} E_{i+1,k-1}(x)\right) dx\,, \qquad (1.20)$$

here, $\delta_{i,j} = \left(\int_{-\infty}^{\infty} E_{i,j}(x)dx\right)^{-1}, i = 0, \pm 1, \pm 2, \dots$

$\delta_{i,k}, E_{i,k}(x)$ satisfy:

$$\int_{-\infty}^{x} \delta_{i,k} E_{i,k}(x)dx = \begin{cases} 1, & x \geq x_{i+k} \\ 0, & x < x_{i+k} \end{cases} \quad (1.21)$$

In UAH tension B-spline:

$$\delta_{i,2} = \frac{\tau \sinh(\tau h)}{2(\cosh(\tau h)-1)}, \delta_{i,3} = \delta_{i,4} = \delta_{i,5} = \frac{1}{h}, \text{ also}$$
$$\delta_{i,k} = \delta_{i+m,k}, \qquad where\ m = 1,2,3, \ldots . and\ k = 2,3, \ldots \quad (1.22)$$

1.5.2. UAH tension B-spline of order three

By using second-order UAH tension B-spline (1.19) and recurrence relation given in (1.20) , third order UAH tension B-spline $E_{i,3}(x)$ can be formulated as:

$$E_{i,3}(x) = \begin{cases} \left\{\frac{\delta_{i,2}}{\tau sinh(\tau h)}[C_{i-2} - 1]\right\} & x \in [x_{i-2}, x_{i-1}) \\ \left\{1 - \frac{\delta_{i,2}}{\tau sinh(\tau h)}[-2 + C_i + C_{i-1}]\right\} & x \in [x_{i-1}, x_i) \\ \left\{\frac{\delta_{i,2}}{\tau sinh(\tau h)}[C_{i+1} - 1]\right\} & x \in [x_i, x_{i+1}) \\ 0 & otherwise \end{cases}, \quad (1.23)$$

where $C_{i+j} = cosh\left[\tau(x - x_{i+j})\right]$

1.5.3. UAH tension B-spline of order four

By using third-order UAH tension B-spline (1.23) and recurrence relation given in (1.20), fourth-order UAH tension B-spline $E_{i,4}(x)$ formulated as:

$$E_{i,4}(x)$$

$$= \begin{cases} \left\{\frac{\delta_{i,3}\delta_{i,2}}{\tau sinh(\tau h)}\left[-M_{i-2}+\frac{S_{i-2}}{\tau}\right]\right\} & x \in [x_{i-2}, x_{i-1}) \\ \left\{\begin{matrix}\frac{\delta_{i,3}\delta_{i,2}}{\tau sinh(\tau h)}\left[-h+3M_{i-1}-\frac{S_i}{\tau}-\frac{2S_{i-1}}{\tau}\right] \\ +\delta_{i,3}M_{i-1}\end{matrix}\right\} & x \in [x_{i-1}, x_i) \\ \left\{\begin{matrix}1-\frac{\delta_{i,3}\delta_{i,2}}{\tau\, sin\, h(\tau h)}\left[M_{i+1}+2M_i-2\frac{S_{i+1}}{\tau}-\frac{S_i}{\tau}-h\right] \\ -\delta_{i,3}M_i\end{matrix}\right\} & x \in [x_i, x_{i+1}) \\ \left\{\frac{\delta_{i,3}\delta_{i,2}}{\tau sin\, h(\tau h)}\left[M_{i+2}-\frac{S_{i+2}}{\tau}\right]\right\} & x \in [x_{i+1}, x_{i+2}) \\ 0 & otherwise \end{cases}, \quad (1.24)$$

where $M_{i+j} = x - x_{i+j}$, $S_{i+j} = \sinh[\tau(x - x_{i+j})]$

1.5.4. UAH tension B-spline of order five

By using the fourth-order UAH tension B-spline (1.24) and recurrence relation given in (1.20), fifth-order UAH tension B-spline $E_{i,5}(x)$ can be formulated as given below, where, $\delta_1' = \frac{\delta_{i,4}\delta_{i,3}\delta_{i,2}}{2\tau^3 \sinh(\tau h)}$, $\delta_2' = \frac{\delta_{i,4}\delta_{i,3}}{2}$, $\delta_3' = \delta_{i,4}$,

$M_{i+j} = (x - x_{i+j})$, $C_{i+j} = cosh\big[\tau(x - x_{i+j})\big]$

$$E_{i,5}(x)=\begin{cases}\{\delta_1'\times[-\tau^2M_{i-2}^2+2C_{i-2}-2]\} & x\in[x_{i-2},x_{i-1})\\ \left\{\delta_1'\times\begin{bmatrix}4\tau^2M_{i-1}^2-2h\tau^2M_{i-1}-2C_i\\-6C_{i-1}+4\,cosh(\tau h)-h^2\tau^2+4\end{bmatrix}\right\} & x\in[x_{i-1},x_i)\\ \quad +\delta_2'\times[M_{i-1}^2] & \\ \left\{\delta_1'\times\begin{bmatrix}-\tau^2M_{i+1}-5\tau^2M_i^2+4h\tau^2M_i\\+6C_{i+1}+6C_i-8\,cosh(\tau h)\\+2h^2\tau^2-4\end{bmatrix}\right\} & x\in[x_i,x_{i+1})\\ \quad +\delta_2'\times[h^2-2M_i^2] & \\ \quad +\delta_3'\times[M_i] & \\ \left\{1-\delta_1'\times\begin{bmatrix}-2\tau^2M_{i+2}^2-2\tau^2M_{i+1}^2\\+2h\tau^2M_{i+1}+6C_{i+2}+2C_{i+1}\\-4\,cosh(\tau h)+h^2\tau^2-4\end{bmatrix}\right\} & x\in[x_{i+1},x_{i+2})\\ \quad -\delta_2'\times[h^2-M_{i+1}^2] & \\ \quad -\delta_3'\times[M_{i+1}] & \\ \{\delta_1'\times[-\tau^2M_{i+3}^2+2C_{i+3}-2]\} & x\in[x_{i+2},x_{i+3})\\ 0 & otherwise\end{cases} \qquad (1.25)$$

1.5.5. UAH tension B-spline of order six

By using fifth-order UAH tension B-spline (1.25) and recurrence relation given in (1.20), sixth-order UAH tension B-spline $E_{i,6}(x)$ can be expressed as below where

$M_{i+j}=\left(x-x_{i+j}\right)$,

$S_{i+j}=\sin\mathrm{h}\left[\tau\left(x-x_{i+j}\right)\right]$,

$\delta_1=\frac{\delta_{i,5}\delta_{i,4}\delta_{i,3}\delta_{i,2}}{6\tau^4\sinh(\tau h)}$,

$\delta_2=\frac{\delta_{i,5}\delta_{i,4}\delta_{i,3}}{6}$,

$\delta_3=\frac{\delta_{i,5}\delta_{i,4}}{2}$, $\delta_4=\delta_{i,5}$

$$E_{i,6}(x)$$

$$= \begin{cases} \{\delta_1 \times [-\tau^3 M_{i-2}^3 + 6S_{i-2} - 6\tau M_{i-2}]\} & x \in [x_{i-2}, x_{i-1}) \\ \left\{\delta_1 \times \begin{bmatrix} -\tau^3 h^3 - 6\tau h \\ +(-3h^2\tau^3 + 12\tau\, cos\, h(\tau h) + 12\tau) M_{i-1} \\ -6S_i - 24S_{i-1} - 3\tau^3 h M_{i-1}^2 \\ +5\tau^3 M_{i-1}^3 + 6\tau M_{i-1} \\ +\delta_2 \times (M_{i-1}^3) \end{bmatrix}\right\} & x \in [x_{i-1}, x_i) \\ \left\{\delta_1 \times \begin{bmatrix} -3\tau^3 h^3 + 12\tau h cosh(\tau h) + 12(\tau h) \\ +24S_{i+1} + 36S_i \\ -\tau^3 M_{i+1}^3 - 9\tau^3 M_i^3 + 9\tau^3 h M_i^2 \\ +(9h^2\tau^3 - 36\tau\, cos\, h(\tau h) - 24\tau) M_i \end{bmatrix}\right\} & x \in [x_i, x_{i+1}) \\ +\delta_2 \times (h^3 + 3h^2 M_i - 3M_i^3) + \delta_3 \times (M_i^2) & \\ \left\{\delta_1 \times \begin{bmatrix} 9\tau^3 h^3 - 24\tau h cosh(\tau h) - 12(\tau h) \\ -36S_{i+2} - 24S_{i+1} \\ +3\tau^3 M_{i+2}^3 + 7\tau^3 M_{i+1}^3 - 9\tau^3 h M_{i+1}^2 + \\ (36\tau\, cos\, h(\tau h) - 9\tau^3 h^2 + 24\tau) M_{i+1} \end{bmatrix}\right\} & x \in [x_{i+1}, x_{i+2}) \\ +\delta_2 \times (h^3 - 6h^2 M_{i+1} + 3M_{i+1}^3) & \\ +\delta_3 \times (h^2 - 2M_{i+1}^2) + \delta_4 \times (M_{i+1}) & \\ \left\{1 - \delta_1 \times \begin{bmatrix} 3\tau^3 M_{i+3}^3 + 6\tau M_{i+3} - 24S_{i+3} \\ +5\tau^3 h^3 - 6\tau h - 12(\tau h)\, cos\, h(\tau h) \\ +2\tau^3 M_{i+2}^3 - 3\tau^3 h M_{i+2}^2 - 6S_{i+2} - \\ (3\tau^3 h^2 - 12\tau\, cos\, h(\tau h) - 12\tau) M_{i+2} \end{bmatrix}\right\} & x \in [x_{i+2}, x_{i+3}) \\ +\delta_2 \times (-2h^3 + 3h^2 M_{i+2} - M_{i+2}^3) & \\ -\delta_3 \times (h^2 - M_{i+2}^2) - \delta_4 \times (M_{i+2}) & \\ \{\delta_1 \times [\tau^3 M_{i+4}^3 + 6\tau M_{i+4} - 6S_{i+4}]\} & x \in [x_{i+3}, x_{i+4}) \\ 0, & otherwise \end{cases} \quad (1.26)$$

Table 1.2: Values of the spline $E_{i,6}(x)$ and its derivative $E'_{i,6}(x)$ are given as:

	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}	x_{i+3}	x_{i+4}
$E_{i,6}(x)$	0	k_1	k_2	k_3	k_2	k_1	0
$E'_{i,6}(x)$	0	k_4	k_5	k_6	$-k_5$	$-k_4$	0

With the values k_i in the table are given below:

$$k_1 = \frac{-\tau^3 h^3 - 6\tau h + 6\sin h(\tau h)}{12(\tau^3 h^3)[\cos h(\tau h) - 1]}$$

$$k_2 = \frac{-4\tau^3 h^3 + 12\tau h - 24\sin h(\tau h) + 12\tau h \cos h(\tau h) + 2\tau^3 h^3 \cosh(\tau h)}{12(\tau^3 h^3)[\cosh(\tau h) - 1]}$$

$$k_3 = \frac{-2\tau^3 h^3 - 12\tau h + 36\sin h(\tau h) - 24\tau h \cos h(\tau h) + 8\tau^3 h^3 \cosh(\tau h)}{12(\tau^3 h^3)[\cosh(\tau h) - 1]}$$

$$k_4 = \frac{-3\tau^3 h^2 - 6\tau + 6\tau \cos h(\tau h)}{12(\tau^3 h^3)[\cosh(\tau h) - 1]}$$

$$k_5 = \frac{-12\tau - 12\tau \cos h(\tau h) + 6\tau^3 h^2 \cos h(\tau h)}{12(\tau^3 h^3)[\cosh(\tau h) - 1]}$$

$$k_6 = 0$$

1.6. Uniform Algebraic Trigonometric (UAT) Tension B-spline

Following the concept discussed in [64, 79, 82] a novel spline known as Uniform algebraic trigonometric tension B spline abbreviated as (UAT tension B-spline) is implemented. UAT tension B-Spline denoted by $A_{i,k}(x)$ of order $'k'$. The region has been designated as $a \leq x \leq b$ subdivided into a mesh of extent $h = \frac{b-a}{N}$ which is uniformly distributed by the knots $x_i = a + ih, i = 0,1,2, \dots, N$. As a result, we obtain $a = x_0 < x_1 < \cdots < x_{N-1} < x_N = b$. Let $A_{i,k}(x)$ symbolizes UAT tension B-spline of order k and knot points x_i which are distributed uniformly at $a = x_0 < x_1 < x_2 < \dots \dots \dots < x_N = b$.

Moreover, Trigonometric Tension B-spline is taken over the space $\{cos\tau x, sin\tau x, x^{k-3}, \dots, x, 1\}$ with $0 < \eta < \frac{\pi}{h}$.

1.6.1. UAT tension B-spline of order two

$$A_{i,2}(x) = \begin{cases} \left\{\dfrac{sin[\tau(x - x_{i-2})]}{sin(\tau h)}\right\}, & x \in [x_{i-2}, x_{i-1}) \\ \left\{\dfrac{sin[\tau(x_i - x)]}{sin(\tau h)}\right\}, & x \in [x_{i-1}, x_i) \\ 0, & otherwise \end{cases}, \quad (1.27)$$

with tension parameter ($\tau = \sqrt{\eta}$) where η is a real number.

and for higher order that is for $k \geq 3$, the recurrence relation for $A_{i,k}$ is given by:

$$A_{i,k}(x) = \int_{-\infty}^{x} \left(\varrho_{i,k-1} A_{i,k-1}(x) - \varrho_{i+1,k-1} A_{i+1,k-1}(x)\right) dx, \quad (1.28)$$

here,

$$\varrho_{i,k} = \left(\int_{-\infty}^{\infty} A_{i,k}(x)dx\right)^{-1}, i = 0, \pm 1, \pm 2, \ldots \quad (1.29)$$

Also, $\varrho_{i,k}, A_{i,k}(x)$ satisfy the condition:

$$\int_{-\infty}^{x} \varrho_{i,k} A_{i,k}(x)dx = \begin{cases} 1, & x \geq x_{i+k} \\ 0, & x < x_{i+k} \end{cases} \quad (1.30)$$

In UAT tension B-spline:

$\varrho_{i,2} = \frac{\tau sin\tau h}{2(1-cos\tau h)}$, $\varrho_{i,3} = \varrho_{i,4} = \varrho_{i,5} = \frac{1}{h}$,

also $\varrho_{i,k} = \varrho_{i+m,k}$, $where\ m = 1,2,3, \ldots . and\ k = 2,3, \ldots$

1.6.2. UAT tension B-spline of order three

By using second-order UAT tension B-spline (1.27) and recurrence relation given in (1.28), third order UAT tension B-spline $A_{i,3}(x)$ can be formulated as:

$$A_{i,3}(x) = \begin{cases} \left\{\frac{\varrho_{i,2}}{\tau sin(\tau h)}[1-\mathbb{C}_{i-2}]\right\} & x \in [x_{i-2}, x_{i-1}) \\ \left\{1-\frac{\varrho_{i,2}}{\tau sin(\tau h)}[2-\mathbb{C}'_{i}-\mathbb{C}_{i-1}]\right\} & x \in [x_{i-1}, x_{i}) \\ \frac{\varrho_{i,2}}{\tau sin(\tau h)}[1-\mathbb{C}'_{i+1}] & x \in [x_{i}, x_{i+1}) \\ 0 & otherwise \end{cases}, \qquad (1.31)$$

where, $\mathbb{C}_{i+j} = cos[\tau(x-x_{i+j})]$, $\mathbb{C}'_{i+j} = cos[\tau(x_{i+j}-x)]$

1.6.3. UAT tension B-spline of order four

By using third-order UAT tension B-spline (1.31) and the recurrence relation given in (1.28), fourth-order UAT tension B-spline $A_{i,4}(x)$ that may be formulated as:

$$A_{i,4}(x) = \begin{cases} \left\{\frac{\varrho_{i,3}\varrho_{i,2}}{\tau sin(\tau h)}\left[\mathcal{M}_{i-2}-\frac{H_{i-2}}{\tau}\right]\right\} & x \in [x_{i-2}, x_{i-1}) \\ \left\{\frac{\varrho_{i,3}\varrho_{i,2}}{\tau sin(\tau h)}\left[h-3\mathcal{M}_{i-1}-\frac{H'_{i}}{\tau}+\frac{2H_{i-1}}{\tau}\right]\right\} & x \in [x_{i-1}, x_{i}) \\ +\varrho_{i,3}\mathcal{M}_{i-1} & \\ \left\{1-\frac{\varrho_{i,3}\varrho_{i,2}}{\tau\, sin(\tau h)}\left[\mathcal{M}'_{i+1}-2\frac{H'_{i+1}}{\tau}+\frac{H_{i}}{\tau}-2\mathcal{M}_{i}+h\right]\right\} & x \in [x_{i}, x_{i+1}) \\ -\varrho_{i,3}\mathcal{M}_{i} & \\ \left\{\frac{\varrho_{i,3}\varrho_{i,2}}{\tau sin(\tau h)}\left[\mathcal{M}'_{i+2}-\frac{H'_{i+2}}{\tau}\right]\right\} & x \in [x_{i+1}, x_{i+2}) \\ 0 & otherwise \end{cases} \qquad (1.32)$$

where, $\mathcal{M}_{i+j} = x - x_{i+j}$, $\mathcal{M}'_{i+j} = x_{i+j} - x$, $H_{i+j} = sin[\tau(x-x_{i+j})]$, $H'_{i+j} = sin[\tau(x_{i+j}-x)]$

1.6.4. UAT tension B-spline of order five

By using the 4[th]-order UAT tension BS (1.32) and recurrence relation given in (1.28), fifth order UAT tension B-spline $A_{i,5}(x)$ that may be formulated as:

$$A_{i,5}(x)=\begin{cases}\{\varrho_1'\times[\tau^2\mathcal{M}_{i-2}^2+2\mathbb{C}_{i-2}-2]\} & x\in[x_{i-2},x_{i-1})\\ \left\{\varrho_1'\times\begin{bmatrix}-4\tau^2\mathcal{M}_{i-1}^2+2h\tau^2\mathcal{M}_{i-1}-2\mathbb{C}_i\\-6\mathbb{C}_{i-1}+4\,cos(\tau h)+h^2\tau^2+4\end{bmatrix}\right\} & x\in[x_{i-1},x_i)\\ \qquad+\varrho_2'\times[\mathcal{M}_{i-1}^2] & \\ \left\{\varrho_1'\times\begin{bmatrix}\tau^2\mathcal{M}_{i+1}^2+5\tau^2\mathcal{M}_i^2-4h\tau^2\mathcal{M}_i\\+6\mathbb{C}_{i+1}+6\mathbb{C}_i-8\,cos(\tau h)\\-2h^2\tau^2-4\end{bmatrix}\right\} & x\in[x_i,x_{i+1})\\ \qquad+\varrho_2'\times[h^2-2\mathcal{M}_i^2] & \\ \qquad+\varrho_3'\times[\mathcal{M}_i] & \\ \left\{1-\varrho_1'\times\begin{bmatrix}2\tau^2\mathcal{M}_{i+2}^2+2\tau^2\mathcal{M}_{i+1}^2\\-2h\tau^2\mathcal{M}_{i+1}+6\mathbb{C}_{i+2}+2\mathbb{C}_{i+1}\\-4\,cos(\tau h)-h^2\tau^2-4\end{bmatrix}\right\} & x\in[x_{i+1},x_{i+2})\\ \qquad-\varrho_2'\times[h^2-\mathcal{M}_{i+1}^2] & \\ \qquad-\varrho_3'\times[\mathcal{M}_{i+1}] & \\ \{\varrho_1'[\tau^2(\mathcal{M}_{i+3}')^2+2\mathbb{C}_{i+3}'-2]\} & x\in[x_{i+2},x_{i+3})\\ 0 & otherwise\end{cases}, \qquad (1.33)$$

where, $\varrho_1'=\frac{\varrho_{i,4}\varrho_{i,3}\varrho_{i,2}}{2\tau^3\sin(\tau h)}$, $\varrho_2'=\frac{\varrho_{i,4}\varrho_{i,3}}{2}$, $\mathcal{M}_{i+j}=x-x_{i+j}$, $\mathcal{M}_{i+j}'=x_{i+j}-x$, $\mathbb{C}_{i+j}=cos\big[\tau(x-x_{i+j})\big]$, $\mathbb{C}_{i+j}'=cos\big[\tau(x_{i+j}-x)\big]$

1.6.5. UAT tension B-spline of order six

By using the fifth-order UAT tension B-spline (1.33) and recurrence relation given in (1.28), sixth-order UAT tension B-spline $A_{i,6}(x)$ can be formulated as:

$$A_{i,6}(x)$$

$$= \begin{cases} \{\varrho_1 \times [\tau^3 \mathcal{M}_{i-2}^3 + 6H_{i-2} - 6\tau \mathcal{M}_{i-2}]\} & x \in [x_{i-2}, x_{i-1}) \\ \left\{ \varrho_1 \times \begin{bmatrix} \tau^3 h^3 - 6\tau h \\ +(3h^2\tau^3 + 12\tau\, cos(\tau h) + 12\tau)\mathcal{M}_{i-1} \\ -6H_i - 24H_{i-1} + 3\tau^3 h \mathcal{M}_{i-1}^2 \\ -5\tau^3 \mathcal{M}_{i-1}^3 + 6\tau \mathcal{M}_{i-1} \end{bmatrix} \right\} & x \in [x_{i-1}, x_i) \\ \left\{ \varrho_1 \times \begin{bmatrix} 3\tau^3 h^3 + 12\tau h cos(\tau h) \\ +12(\tau h) + 24H_{i+1} + 36H_i + \\ \tau^3 \mathcal{M}_{i+1}^3 + 9\tau^3 \mathcal{M}_i^3 - 9\tau^3 h \mathcal{M}_i^2 \\ -(9h^2\tau^3 + 36\tau\, cos(\tau h) + 24\tau)\mathcal{M}_i \end{bmatrix} \right\} & x \in [x_i, x_{i+1}) \\ +\varrho_2 \times [h^3 + 3h^2 \mathcal{M}_i - 3\mathcal{M}_i^3] & \\ +\varrho_3 \times [\mathcal{M}_i^2] & \\ \left\{ \varrho_1 \times \begin{bmatrix} -9\tau^3 h^3 - 24\tau h cos(\tau h) \\ -12(\tau h) - 36H_{i+2} - 24H_{i+1} \\ -3\tau^3 \mathcal{M}_{i+2}^3 - 7\tau^3 \mathcal{M}_{i+1}^3 + 9\tau^3 h \mathcal{M}_{i+1}^2 \\ +(9\tau^3 h^2 + 36\tau\, cos(\tau h) + 24\tau)\mathcal{M}_{i+1} \end{bmatrix} \right\} & x \in [x_{i+1}, x_{i+2}) \\ +\varrho_2 \times [h^3 - 6h^2 \mathcal{M}_{i+1} + 3\mathcal{M}_{i+1}^3] & \\ +\varrho_3 \times [h^2 - 2\mathcal{M}_{i+1}^2] & \\ +\varrho_4 \times [M_{i+1}] & \\ \left\{ 1 - \varrho_1 \times \begin{bmatrix} -3\tau^3 \mathcal{M}_{i+3}^3 + 6\tau \mathcal{M}_{i+3} - 24H_{i+3} \\ -5\tau^3 h^3 - 6\tau h - 12(\tau h)\, cos(\tau h) \\ -6H_{i+2} - 2\tau^3 \mathcal{M}_{i+2}^3 + 3\tau^3 h \mathcal{M}_{i+2}^2 \\ +(3\tau^3 h^2 + 12\tau\, cos(\tau h) + 12\tau)\mathcal{M}_{i+2} \end{bmatrix} \right\} & x \in [x_{i+2}, x_{i+3}) \\ -\varrho_2 \times [2h^3 - 3h^2 \mathcal{M}_{i+2} + \mathcal{M}_{i+2}^3] & \\ -\varrho_3 \times [h^2 - \mathcal{M}_{i+2}^2] & \\ -\varrho_4 \times [\mathcal{M}_{i+2}] & \\ \{\varrho_1[-\tau^3 \mathcal{M}_{i+4}^3 + 6\tau \mathcal{M}_{i+4} - 6H_{i+4}]\} & x \in [x_{i+3}, x_{i+4}) \\ 0 & otherwise \end{cases} \quad (1.34)$$

where, $\varrho_1 = \frac{\varrho_{i,5}\varrho_{i,4}\varrho_{i,3}\varrho_{i,2}}{6\tau^4 \sin(\tau h)}$, $\varrho_2 = \frac{\varrho_{i,5}\varrho_{i,4}\varrho_{i,3}}{6}$, $\varrho_3 = \frac{\varrho_{i,5}\varrho_{i,4}}{2}$, $\varrho_4 = \varrho_{i,5}$, $\mathcal{M}_{i+j} = x - x_{i+j}$, $H_{i+j} = \sin[\tau(x - x_{i+j})]$.

Table 1.3: At different knot points values of spline $A_{i,k}(x)$ and its derivative.

	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}	x_{i+3}	x_{i+4}
$A_{i,k}(x)$	0	w_1	w_2	w_3	w_2	w_1	0
$A'_{i,k}(x)$	0	w_4	w_5	w_6	$-w_5$	$-w_4$	0

with the values w_i in the table are given below:

$$w_1 = \frac{\tau^3h^3-6\tau h+6\sin(\tau h)}{12(\tau^3h^3)[1-\cos(\tau h)]},\ w_2 = \frac{4\tau^3h^3+12\tau h-24\sin(\tau h)+12\tau h\cos(\tau h)-2\tau^3h^3\cos(\tau h)}{12(\tau^3h^3)[1-\cos(\tau h)]},$$

$$w_3 = \frac{2\tau^3h^3-12\tau h+36\sin(\tau h)-24\tau h\cos(\tau h)-8\tau^3h^3\cos(\tau h)}{12(\tau^3h^3)[1-\cos(\tau h)]},\quad w_4 = \frac{3\tau^3h^2-6\tau+6\tau\cos(\tau h)}{12(\tau^3h^3)[1-\cos(\tau h)]},$$

$$w_5 = \frac{12\tau-12\tau\cos(\tau h)-6\tau^3h^2\cos(\tau h)}{12(\tau^3h^3)[1-\cos(\tau h)]},\ w_6 = 0$$

The above spline functions HBS, UAH tension BS, and UAT tension BS will be modified by which the resultant matrix get transform to a diagonally dominant matrix [21], giving us an improved outcome. The following formulas can be used for reformation. In this work modified HBS basis function is provided as:

$$\begin{aligned} MH_1(x) &= H_1(x) + 2H_0(x) \\ M\mathrm{H}_2(x) &= H_2(x) - H_0(x) \\ M\mathrm{H}_j(x) &= H_j(x), \qquad (\mathrm{j} = 3,4,5,\dots,N-2) \\ M\mathrm{H}_{N-1}(x) &= H_{N-1}(x) - H_{N+1}(x) \\ M\mathrm{H}_N(x) &= H_N(x) + 2H_{N+1}(x) \end{aligned} \tag{1.35}$$

And following formulas can be used for the reformation of UAH tension B-spline

$$\begin{aligned} \mathrm{ME}_{1,k}(x) &= E_{1,k}(x) + 2E_{0,k}(x) \\ \mathrm{ME}_{2,k}(x) &= E_{2,k}(x) - E_{0,k}(x) \\ \mathrm{ME}_{i,k}(x) &= E_{i,k}(x), \qquad (i = 3,4,5,\dots,N-2) \end{aligned} \tag{1.36}$$

$$\mathrm{ME}_{(N-1),k}(x) = E_{(N-1),k}(x) - E_{(N+1),k}(x)$$

$$ME_{N,k}(x) = E_{N,k}(x) + 2E_{(N+1),k}(x)$$

The below given formulas can be employed for improvised UAT tension B-spline:

$$\mathrm{MA}_{1,k}(x) = A_{1,k}(x) + 2A_{0,k}(x)$$

$$\mathrm{MA}_{2,k}(x) = A_{2,k}(x) - A_{0,k}(x)$$

$$\mathrm{MA}_{i,k}(x) = A_{i,k}(x), \qquad (i = 3,4,5, \dots, N-2) \tag{1.37}$$

$$\mathrm{MA}_{(N-1),k}(x) = A_{(N-1),k}(x) - A_{(N+1),k}(x)$$

$$MA_{N,k}(x) = A_{N,k}(x) + 2A_{(N+1),k}(x)$$

1.7. Determination of weighting coefficients

1.7.1. Weighting coefficients for quintic HBS

The 1^{st} order derivative for the modified basis function HBS (1.35) may be formulated with the use of equation (1.2) for grid point x_i as:

$$MH'_m(x_i) = \sum_{j=1}^{N} d_{ij}^{(1)} MH_m(x_j), \tag{1.38}$$

for $i = 1, 2, 3, \dots, N$ and $m = 1, 2, 3, \dots, N$

That results in the following system of matrix

$$B\vec{d}^{(1)}[i] = \vec{S}[i], \tag{1.39}$$

Where coefficient matrix B is given by:

$$\begin{bmatrix} e_3 & e_2 & e_1 & 0 & 0 & 0 & \cdots & 0 \\ e_2 & e_3 & e_2 & e_1 & 0 & 0 & \cdots & 0 \\ e_1 & e_2 & e_3 & e_2 & e_1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & e_1 & e_2 & e_3 & e_2 & e_1 \\ 0 & \cdots & 0 & 0 & e_1 & e_2 & e_3 & e_2 \\ 0 & \cdots & 0 & 0 & 0 & e_1 & e_2 & e_3 \end{bmatrix}$$

$\vec{d}^{(1)}$ [i] is the weighting coefficient vector corresponding to the grid point x_i, which is given by:

$$\vec{d}^{(1)}[i] = \begin{bmatrix} d_{i1}^{(1)} \\ d_{i2}^{(1)} \\ \vdots \\ \vdots \\ d_{iN}^{(1)} \end{bmatrix},$$

and $\vec{S}[i] = [s_{i1}, s_{i2}, s_{i3}, \dots, s_{iN}]^T$ is the vector corresponding to x_i, where $i = 1,2, \dots, N$ which can be calculated as:

$$\vec{S}[1] = \begin{bmatrix} e_6 \\ e_5 \\ e_4 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \vec{S}[2] = \begin{bmatrix} -e_5 \\ e_6 \\ e_5 \\ e_4 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \vec{S}[3] = \begin{bmatrix} -e_4 \\ -e_5 \\ e_6 \\ e_5 \\ e_4 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \dots\dots, \vec{S}[N-2] = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ -e_4 \\ -e_5 \\ e_6 \\ e_5 \\ e_4 \end{bmatrix},$$

$$\vec{S}[N-1] = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ -e_4 \\ -e_5 \\ e_6 \\ e_5 \end{bmatrix}, \vec{S}[N] = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ -e_4 \\ -e_5 \\ e_6 \end{bmatrix}$$

The above five-band matrix system is resolved by the encoding in MATLAB, by which $\vec{d}^{(1)}[i]$ will be obtained that is 1^{st}order weighting factors, $d_{i1}^{(1)}$, $d_{i2}^{(1)}$, $d_{i3}^{(1)}$,..., $d_{i(N-1)}^{(1)}$, $d_{iN}^{(1)}$. After evaluating these coefficients of partial derivatives having 1^{st} order, their 2^{nd} and higher-order partial derivative can be premeditated by implementing the formula [1, 35] given below.

$$d_{ij}^{(r)} = r[d_{ij}^{(1)} d_{ii}^{(r-1)} - \frac{d_{ij}^{(r-1)}}{x_i - x_j}], \text{ for } i \neq j \ \{i, j = 1,2,3, \dots, N\} \text{ and}$$

$$r = 2,3, \dots, N-1 \qquad (1.40)$$

$$d_{ii}^{(r)} = -\sum_{j=1, j\neq i}^{N} d_{ij}^{(r)}, \text{ for } i = j$$

1.7.2. Weighting coefficients for quintic UAH tension B-spline

The 1^{st} order derivative for the modified basis function UAH tension B-spline (1.36) can be formulated by using equation (1.2) for the grid point x_i as:

$$ME_{1,k}^{(1)}(x_i) = \sum_{j=1}^{N} n_{ij}^{(1)} ME_{1,k}(x_j), \qquad (1.41)$$

for $i = 1, 2, 3, \dots, N$ and $k = 1, 2, 3, \dots, N$

That results in the resulting system of matrix

$$B''\vec{n}^{(1)}[i] = \vec{K}[i], \qquad (1.42)$$

where coefficient matrix B'' is given by:

$$\begin{bmatrix} k_3 & k_2 & k_1 & 0 & 0 & 0 & \cdots & 0 \\ k_2 & k_3 & k_2 & k_1 & 0 & 0 & \cdots & 0 \\ k_1 & k_2 & k_3 & k_2 & k_1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & k_1 & k_2 & k_3 & k_2 & k_1 \\ 0 & \cdots & 0 & 0 & k_1 & k_2 & k_3 & k_2 \\ 0 & \cdots & 0 & 0 & 0 & k_1 & k_2 & k_3 \end{bmatrix}$$

$\vec{n}^{(1)}$ [i] is vector for the grid point x_i, and is given by:

$$\vec{n}^{(1)}[i] = \begin{bmatrix} n_{i1}^{(1)} \\ n_{i2}^{(1)} \\ \vdots \\ \vdots \\ n_{iN}^{(1)} \end{bmatrix},$$

and $\vec{K}[i] = [s_{i1}, s_{i2}, s_{i3}, \dots, s_{iN}]^T$ is the vector corresponding to x_i, where $i = 1,2, \dots, N$ which can be calculated as:

$$\vec{K}[1] = \begin{bmatrix} k_6 \\ k_5 \\ k_4 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \vec{K}[2] = \begin{bmatrix} -k_5 \\ k_6 \\ k_5 \\ k_4 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \vec{K}[3] = \begin{bmatrix} -k_4 \\ -k_5 \\ k_6 \\ k_5 \\ k_4 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \dots\dots, \vec{K}[N-2] = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ -k_4 \\ -k_5 \\ k_6 \\ k_5 \\ k_4 \end{bmatrix},$$

$$\vec{K}[N-1]=\begin{bmatrix}0\\:\\:\\0\\-k_4\\-k_5\\k_6\\k_5\end{bmatrix},\ \vec{K}[N]=\begin{bmatrix}0\\:\\:\\0\\-k_4\\-k_5\\k_6\end{bmatrix}$$

The above system of five-band matrix system is resolute by encrypting in MATLAB, by which $\vec{n}^{(1)}[i]$ will be attained, or weighting coefficients of order one, $n_{i1}^{(1)}$, $n_{i2}^{(1)}$, $n_{i3}^{(1)}$,..., $n_{i(N-1)}^{(1)}$, $n_{iN}^{(1)}$. After evaluating these coefficients of partial derivatives having 1^{st} order, their 2^{nd} and higher-order derivative can be computed partially with the formulation given below.

$$n_{ij}^{(r)}=r[n_{ij}^{(1)}n_{ii}^{(r-1)}-\frac{n_{ij}^{(r-1)}}{x_i-x_j}],\ \text{for “}i\neq j\ \{i,j=1,2,3,\dots,N\}\text{” and}$$
$$n_{ii}^{(r)}=-\sum_{j=1,j\neq i}^{N}n_{ij}^{(r)},\ \text{for “}i=j,r=2,3,\dots,N-1\text{”} \qquad (1.43)$$

1.7.3. Weighting coefficients for quintic UAT tension B-spline

The 1^{st} order derivative for the modified basis function UAT tension B-spline (1.37) can be formulated by using equation (1.2) for the grid point x_i as:

$$MA_{1,k}^{(1)}(x_i)=\sum_{j=1}^{N}v_{ij}^{(1)}MA_{1,k}(x_j), \qquad (1.44)$$

for “$i=1,2,3,\dots,N$ and $k=1,2,3,\dots,N$”

Utilising quintic UAT tension B-spline at knot points in the formulae of modified basis and eliminating weighting coefficients from the system, “a 5-banded coefficient matrix” will be obtained as follows:

$$T'\vec{v}^{(1)}[i] = \vec{D}[i], \qquad (1.45)$$

where

$$\vec{v}^{(1)}[i] = \begin{bmatrix} v_{i1}^{(1)} \\ v_{i2}^{(1)} \\ v_{i3}^{(1)} \\ \vdots \\ \vdots \\ v_{iN}^{(1)} \end{bmatrix}$$

referred to as a vector of weighting factors for x_i with $i = 1,2,\dots,N$, and $\vec{D}[i] = [d_{i1}, d_{i2}, d_{i3}, \dots, d_{iN}]^T$ is the coefficient vector corresponding to x_i, where $i = 1,2,\dots,N$ which can be calculated as:

$$\vec{D}[1] = \begin{bmatrix} w_6 \\ w_5 \\ w_4 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \vec{D}[2] = \begin{bmatrix} -w_5 \\ w_6 \\ w_5 \\ w_4 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \vec{D}[3] = \begin{bmatrix} -w_4 \\ -w_5 \\ w_6 \\ w_5 \\ w_4 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}, \dots\dots, \vec{D}[N-2] = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ -w_4 \\ -w_5 \\ w_6 \\ w_5 \\ w_4 \end{bmatrix},$$

$$\vec{D}[N-1] = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ -w_4 \\ -w_5 \\ w_6 \\ w_5 \end{bmatrix}, \vec{D}[N] = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ -w_4 \\ -w_5 \\ w_6 \end{bmatrix},$$

and coefficient matrix T' is given by:

$$\begin{bmatrix} w_3 & w_2 & w_1 & 0 & 0 & 0 & \cdots & 0 \\ w_2 & w_3 & w_2 & w_1 & 0 & 0 & \cdots & 0 \\ w_1 & w_2 & w_3 & w_2 & w_1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & w_1 & w_2 & w_3 & w_2 & w_1 \\ 0 & \cdots & 0 & 0 & w_1 & w_2 & w_3 & w_2 \\ 0 & \cdots & 0 & 0 & 0 & w_1 & w_2 & w_3 \end{bmatrix}$$

The above five-band matrix system is resolute by scrambling in MATLAB, by which $\vec{v}^{(1)}[i]$ will be attained.

After evaluating the weighting factors of partial derivatives having 1^{st} order, their 2^{nd} and higher-order partial derivative can be premeditated by the formula [1, 35] as:

$$v_{ij}^{(k)} = k[v_{ij}^{(1)} v_{ii}^{(k-1)} - \frac{v_{ij}^{(k-1)}}{t_i - t_j}], \text{ for } "i \neq j \ \{i, j = 1,2,3, \dots, N\}" \text{ and}$$
$$v_{ii}^{(k)} = -\sum_{j=1, j\neq i}^{N} v_{ij}^{(k)}, \text{ for } "i = j, k = 2,3, \dots, N-1," \qquad (1.46)$$

where $v_{ij}^{(k)}$ represents the second, third, and fourth-order weighting factors with $k = 2,3,4$ respectively.

1.8. Strong stability-preserving time-stepping Runge-Kutta (SSP-RK43) technique

The high-order time discretization method named Strong stability preserving (SSP) was established for the time integration of hyperbolic conservation laws with semi-discretization. To ensure convergence while using numerical methods to approximate PDE solutions, we often rely on linear stability theory. These time discretization techniques maintain the strong stability characteristics of the first-order Euler time stepping in combination with spatial discretization, in any norm or seminorm [83].

According to the well-known Lax equivalence theorem, stability is both necessary as well as sufficient for convergence for a linear approach to be consistent with a linear problem [84]. Strang [85] expanded on this finding by demonstrating that, for sufficiently smooth nonlinear situations, an approximation is convergent if it is consistent and its linearized form is L_2 stable. But in case, the PDEs have discontinuous solution, the linear stability theory no longer ensures convergence as in the case of hyperbolic PDEs. As a result, high-order spatial discretizations are created, and when used in conjunction with the Euler time stepping approach, they provide the required nonlinear stability qualities that allow for the approximate approximation of discontinuous solutions. The SSP time discretization method is the most feasible method for a semi-discrete regime of PDE line approximation which was initially established in [86, 87], also named total variation diminishing (TVD). Further research on this group of approaches was done in [88-93]. The fundamental concept was to suppose that the forward Euler time discretization of first-order of the regime of ODE is highly stable for a specified norm where Δt (step size) is sufficiently constrained so that higher-order time discretization which preserves strong stability for a given norm or for a different time step can be obtained. SSP time discretization is the most appropriate form and worth achieving strong stability of specified norm. After discretizing the space derivatives of time-dependent PDEs, a system of ODEs is attained as:

$$u_t = A(u) \tag{1.47}$$

The Runge-Kutta technique, also known as a strong stability-preserving time-stepping Runge-Kutta (SSP-RK43) regime [94], will next be used to better clarify this scheme of ODEs, represented as follows:

$$u^{(1)} = u^{(m)} + \frac{\Delta t}{2} A(u^{(m)})$$

$$u^{(2)} = u^{(1)} + \frac{\Delta t}{2} A(u^{(1)})$$

$$u^{(3)} = \frac{2}{3} u^{(m)} + \frac{u^{(2)}}{3} + \frac{\Delta t}{6} A(u^{(2)}) \qquad (1.48)$$

$$u^{(m+1)} = u^{(3)} + \frac{\Delta t}{2} A(u^{(3)})$$

1.9. Stability of the proposed scheme

We used matrix stability method to show the stability of proposed method [15, 95, 96]. After the implementation of DQM, the respective PDE equation will get converted into an ODEs, which resolved further using the RK method. With the use of the matrix technique, it is possible to analyse the method's stability, which is dependent on the eigenvalues of the coefficient matrix of this system. After using the proposed scheme particular equation will be converted into:

$$u_t(x,t) = B'u + f\big(u(x,t)\big), \qquad (1.49)$$

here B' stands for the coefficient matrix obtained from the particular equation, $f(u(x,t))$ indicates the non-homogeneous part with boundary conditions. Stability conditions for computational methods that solve equation (1.47) depends on the stability of the ODE (1.49). The matrix B' plays a crucial role since its eigenvalues have a direct impact on the precise solutions. Consequently, the coefficient matrix's eigenvalues can be used to accurately characterize the entire scenario. The stable exact solutions are displayed as $t \to \infty$ when all the eigenvalues of B' have real component, that is $R(\lambda_i) \leq 0$. The conditions [97] for the stable solution as $t \to \infty$ requires:

a) $-2.78 < \Delta t.\lambda_i < 0$, if eigenvalues are real.

b) $-2\sqrt{2} < \Delta t.\lambda_i < 2\sqrt{2}$, if only complex eigenvalues exist.

c) $\Delta t.\lambda_i$ should be in that area as in Figure 1.1, if complex eigenvalues exist.

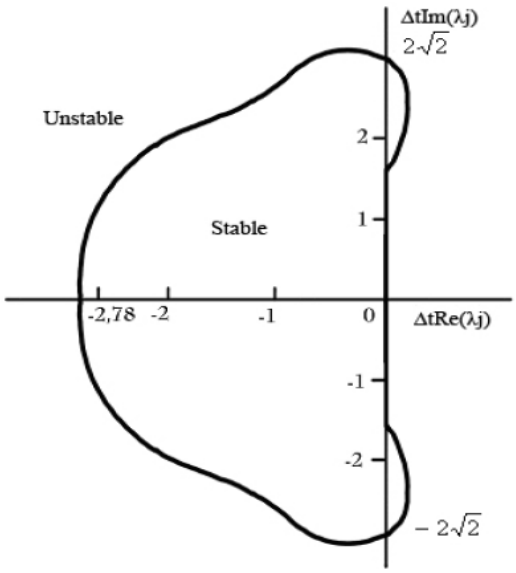

Figure 1.1: Stability Region

If real component of the complex eigenvalues are tiny positive integers, there will be some tolerance [97]. The eigenvalues of the matrix B' are identified for the problems due to the coarse linearization. Also, as the prior explained conditions depend on the time step Δt. So, to attain the stable numerical solution of a particular problem, time step Δt can be selected accordingly.

1.10. Error norms

The acquired findings are compared with precise as well as with numerical answers found in the literature by other researchers in order to assess the suggested method's correctness. For expressing the versatility of proposed numerical algorithm, different numerical examples are presented. For efficiency and accuracy check three error norms named Euclidian, maximum, and global relative error (GRE) are calculated, which are formulated as:

$$L_2 = \sqrt{h\sum_{i=0}^{n}(p_i - P_i)^2} \tag{1.50}$$

$$L_\infty = \max_{0\le i\le n}|p_i - P_i| \tag{1.51}$$

$$GRE = \frac{\sum_{i=0}^{n}|p_i - P_i|}{\sum_{i=0}^{n}|p_i|}, \tag{1.52}$$

$$Average\ error = \frac{1}{N}\sum_{i=0}^{n}|p_i - P_i| \tag{1.53}$$

where p_i symbolizes the exact and P_i symbolizes the numerical solutions at ith grid point.

Additionally, to assess the convergence rate of the suggested technique the order of convergence is obtained, and the formula that has been employed is given:

$$Order\ of\ convergence\ (O) = \frac{log\left(\frac{E(N_2)}{E(N_1)}\right)}{log\left(\frac{N_1}{N_2}\right)} \tag{1.54}$$

The numerical solutions N_1 and N_2 each has N grid points and 2N grid points, respectively, $E(N_i)$ symbolizes the error with N_i grid points.

1.11. Motivation and Objectives of the work

1.11.1. Research motivation

The present work is motivated by knowing the fact that PDEs have huge application in various fields, such as fluid mechanics, electromagnetism, heat transfer, and quantum mechanics. Numerical solutions enable the study and prediction of these phenomena. As numerous PDEs have no analytical solutions because of their complex nature, numerical methods provide a practical way to approximate solutions in such

cases. This motivated and forced me to explore numerical solution for some problems. We have used quintic B-splines with DQM. Quintic B-splines are higher-order interpolation method compared to lower-degree B-splines, which results in more accurate approximations, and provides smooth and continuous representations of functions. Moreover, DQM when combined with quintic B-splines, can help reduce numerical dispersion, and can be applied to a wide range of problems.

1.11.2. Research Objectives

1) Numerical solution of higher order differential equations will be obtained by using quintic Hyperbolic B-spline basis functions.
2) Numerical solution of third or fourth order differential equations will be obtained by using Uniform Algebraic trigonometric [UAT] tension B-spline as basis function.
3) To obtain the numerical solution, we will use differential quadrature method (DQM) along with quintic Uniform Algebraic Hyperbolic [UAH] tension B-spline.
4) The hyperbolic B-spline basis functions, UAT tension B-spline and UAH tension B-spline will be used to solve equations with two space dimensions.

Chapter 2

Numerical Solution of Coupled Korteweg-de Vries Equation using quintic B-spline

In this Chapter, numerical solution of Coupled Korteweg-de Vries equation is obtained using above mentioned three quintic B-splines.

Research Objectives

1) Numerical solution of higher order differential equations will be obtained by using quintic Hyperbolic B-spline basis functions.
2) Numerical solution of third or fourth order differential equations will be obtained by using Uniform Algebraic trigonometric [UAT] tension B-spline as basis function.
3) To obtain the numerical solution, we will use differential quadrature method (DQM) along with quintic Uniform Algebraic Hyperbolic [UAH] tension B-spline.

2.1. Introduction

To apprehend or predict the behavior of natural phenomena, mathematical models have huge importance, which made researchers pay attention to the advancement of science and technology. For the modeling of various natural phenomena like sciences, there arise nonlinear partial differential equations. During the past few years, in order to get the solutions of models with higher dimensions having soliton-like structures are also considered [98]. Solitons are found with solutions for many equations including the Schrödinger equation [13, 99, 100], CKdV equation [8, 13], sine-Gordon equation [101]. For reference, description of long-wave in deep oceans and shallow seas [102, 103], solutions of ion-acoustic [104-106]. Also to designate the concept of tidal waves, the

nonlinear Schrödinger equation describes waves occurring in deep water, whereas the KdV system of equation designates the special effects in surface water. Even the collaboration of two extended waves having unlike dispersion affairs is defined with the coupled KdV equations. However, in physical situations, the idealization of KdV equations arises highly due to the supposition of persistent coefficients.

Originally Korteweg-de Vries equation was obtained through a long process and has taken near about sixty years of period [107], which was begun by the trials of Scott-Russell [108], the research of Boussinesq and Rayleigh nearby 1870 [109-113] then lastly concluding by the study of Korteweg and De Vries [114].

Korteweg-de Vries equation in simplified form is given by:

$$\frac{\partial W}{\partial t} - 6W\frac{\partial W}{\partial x} + \frac{\partial^3 W}{\partial x^3} = 0. \qquad (2.1)$$

This is the outcome of research about extensive waves in water having little depth, where position, time is signified by x, t respectively, and $w = w(x,t)$ represents the wave surface.

Coupled Korteweg-de Vries equation, abbreviated as (CKdV), was given by Hirota and Satsuma in 1981 [115]. They hosted the CKdV equation as:

$$\begin{aligned} \frac{\partial W}{\partial t} &= r\left(\frac{\partial^3 W}{\partial x^3} + 6W\frac{\partial W}{\partial x}\right) + 2sZ\frac{\partial Z}{\partial x}, \\ \frac{\partial Z}{\partial t} &= -\frac{\partial^3 Z}{\partial x^3} - 3W\frac{\partial Z}{\partial x}, \boldsymbol{x_L \le x \le x_R}. \end{aligned} \qquad (2.2)$$

with the initial conditions as

$$W(x,0) = f'(x), Z(x,0) = g'(x), x \in [x_L, x_R], \qquad (2.3)$$

and the boundary conditions as:

$$\begin{cases} W(x_L,t)=\vartheta_1(x_L,t), & W(x_R,t)=\vartheta_2(x_R,t) \\ Z(x_L,t)=\omega_1(x_L,t), & Z(x_R,t)=\omega_2(x_R,t), \quad t\in[0,T] \\ W_x(x_L,t)=\vartheta_3(x_L,t), & W_x(x_R,t)=\vartheta_4(x_R,t), \\ Z_x(x_L,t)=\omega_3(x_L,t), & Z_x(x_R,t)=\omega_4(x_R,t), \quad t\in[0,T] \end{cases}, \tag{2.4}$$

where r,s are random parameters [115], $f'(x)$, $g'(x)$ are given continuous functions, $W=W(x,t)$ defines the field of horizontal velocity, $Z=Z(x,t)$ defines the deviation of height from an ambient position of liquid and (Z,Z_x) defines the part of force on the KdV wave scheme along with the linear dispersion relation ($\omega=\frac{k^2}{2}$) respectively.

The interaction between the long waves having different dispersive relations is described by these equations, which are first presented in 1981 [115]. Also, weak dispersive waves like acoustic, internal and planetary waves, etc. are allied with the CKdV system [116, 117]. CKdV has great importance in every field, because of which numerous researchers studied this system analytically as well as numerically. Jager [118] discuss the origin of the KdV equation and compare it with the Boussinesq equation. Zhu [107] gives a difference scheme for coupled KdV. Many researchers investigate this nonlinear CKdV system analytically [119-125]. Kaya and Inan [112] used the decomposition method along with symbolic computation to solve CKdV numerically. Fan [113] used symbolic computation with the Tanh method for the solutions of this nonlinear system. Ma et al. [126] obtained some analytical solutions of this system by using Jacobian elliptic function expansion. Assas [127] solve CKdV by using the iteration method. Halim et al. [116] solve it numerically by using the finite-difference method. Tam et al. [111] created a novel kind of soliton solution. The authors employed a variety of techniques, taking into consideration the significance of this system, to come up with solutions for the CKdV system, including the Hybrid method with QBS [8], pseudospectral method [130], Galerkin method [129], modified form of Khater method with BS [133], Collocation method along with QBS [142], perturbation approach [128], Masaaki presents the symmetry and law of conservation of CKdV [145], finite difference technique [131], (NIT) New Iterative method [136], Galerkin method with quadratic BS

[143], the Petrov-Galerkin approach and the cubic Bs approximation methodology [137], QBS along with collocation method [132], homotopy analysis scheme [144], sine-cosine method, tanh-expension method, Kudryashov-expansion method [134], q-HATM and Laplace transform method [139], Lagrange polynomial with DQM [146], Haar Wavelet scheme [141], (DTM) Differential transformation method [135], FD method and DQM with modified cubic BS [140], exp-function method [138], Guo et al. present an enhanced Physics Informed Neural Network (PINN) [147].

2.2. Integrability

In recent years, study about soliton solutions of nonlocal integrable equations attracts great attention, as a result of which, much work has been done in this field by researchers. For instance, soliton solution for nonlocal reverse-time nonlinear Schr*ö*dinger equation is obtained by establishing Riemann-Hilbert problems [148], nonlocal integrable equations are constructed for modified Korteweg-de Vries equation [149, 150], and Frobenius integrable decompositions are introduced for generalization of integrable time-space decompositions for partial differential equations [151]. Many dependable methods are engaged in the previous work to inspect the complete integrability of nonlinear coupled KdV equations, and it is very fascinating to note that there are such powerful methods for the integrability of coupled equations. In the theory of solitary waves, some of the nonlinear equations give just one soliton solution for example modified KdV equation having time-dependent coefficients and some equations have only two soliton solutions but not N-soliton solutions with N>2 for example KdV equation of order nine and Boussinesq equation of order six [152]. Most significantly, those solitonic equations are called integrable which have 3 soliton solutions, and more than 3 solitons soliton solutions. In other words, equations with N-soliton are integrable if they have soliton solution for $N \geq 3$. Moreover in literature, for the construction of the N-soliton of the equations, firstly equations are transformed into the bilinear form [153]. So, for the conversion of CKdV system (2.2) let us consider the following transformations:

$$W = -2(lnf)_{xx},\ \ Z = \frac{\sqrt{3}}{3}\frac{g}{f}. \tag{2.5}$$

Substituting the above transformation (2.5) into (2.2) with the constants $r = \frac{1}{2}, s = 3$, and integrating the obtained equation with respect to x by taking constant of integration as $zero$, we obtain the equation as

$$\frac{-D_x D_t f.f}{f^2} = \frac{1}{2}\left[\frac{-D_x^4 f.f}{f^2} + 3\left(\frac{D_x^2 f.f}{f^2}\right)^2 - 3\left(\frac{D_x^2 f.f}{f^2}\right)^2\right] + \left(\frac{g}{f}\right)^2 \tag{2.6}$$

$$\frac{D_t g.f}{f^2} = -\left[\frac{D_x^3 g.f}{f^2} - 3\frac{D_x g.f}{f^2}\frac{D_x^2 f.f}{f^2}\right] - 3\frac{D_x^2 f.f}{f^2}\frac{D_x g.f}{f^2}, \tag{2.7}$$

where $D_x\ and\ D_t$ are Hirota bilinear operators. Concisely CKdV equation (2.2) can be written in a bilinear form as

$$\begin{gathered}\left(\frac{1}{2}D_x^4 - D_x D_t\right) f.f = g^2 \\ (D_t + D_x^3) g.f = 0.\end{gathered} \tag{2.8}$$

Thus N-soliton of CKdV system (2.2) can be constructed with the implementation of the perturbation method [153]. For further calculation, expanding f, g as the power series with a small parameter ε as

$$\begin{aligned} f &= 1 + f^{(2)}\varepsilon^2 + f^{(4)}\varepsilon^4 + \cdots + f^{(2p)}\varepsilon^{2p} + \cdots \\ g &= g^{(1)}\varepsilon + g^{(3)}\varepsilon^3 + \cdots + g^{(2p+1)}\varepsilon^{2p+1} + \cdots \end{aligned} \tag{2.9}$$

Substituting equation (2.9) into (2.8) and collecting the like powers of ε we get,

$$2\left(\frac{1}{2}f_{xxx}^{(2)} - f_{xt}^{(2)}\right) = g^{(1)}g^{(1)} \tag{2.10}$$

$$2\left(\frac{1}{2}f_{xxxx}^{(4)} - f_{xt}^{(4)}\right) = -\left(\frac{1}{2}D_x^4 - D_x D_t\right)\left(f^{(2)}.f^{(2)} + g^{(1)}.g^{(3)} + g^{(3)}.g^{(1)}\right) \tag{2.11}$$

$$2\left(\frac{1}{2}f_{xxxx}^{(6)} - f_{xt}^{(6)}\right) = -\left(\frac{1}{2}D_x^4 - D_x D_t\right)\left(f^{(2)}.f^{(4)} + g^{(1)}.g^{(5)} + g^{(3)}.g^{(3)} + \right. \tag{2.12}$$

$$g^{(5)}.g^{(1)}\Big)$$

… … … … … … … … … … … ….

$$g_t^{(1)} + g_{xxx}^{(1)} = 0 \qquad (2.13)$$

$$g_t^{(3)} + g_{xxx}^{(3)} = -(D_t + D_x^3)g^{(1)}.f^{(2)} \qquad (2.14)$$

$$g_t^{(5)} + g_{xxx}^{(5)} = -(D_t + D_x^3)(g^{(1)}.f^{(4)} + g^{(3)}.f^{(2)}) \qquad (2.15)$$

… … … … … … … … … … … ….

1) *One-soliton solution*. The solution of the equation (2.13) can be taken as [154]

$$g^{(1)} = 1 + e^{\eta_1}, \qquad (2.16)$$

with the dispersal relation as

$$\eta_1 = w_1 t + c_1 x + \eta_1^0, \text{ and } w_1 + c_1^3 = 0.$$

2) *Two-soliton solution*. For two-solitons equation (2.13) retain solution [153] as

$$g^{(1)} = e^{\eta_1} + e^{\eta_2}, \qquad (2.17)$$

With the values $\eta_i = w_i t + c_i x + \eta_i^0$, and $w_i + c_i^3 = 0, (i = 1,2)$

Substituting equation (2.17) into (2.10), we obtain

$$2\left(\frac{1}{2}f_{xxx}^{(2)} - f_{xt}^{(2)}\right) = e^{\eta_1+\eta_1} + e^{\eta_1+\eta_2} + e^{\eta_2+\eta_1} + e^{\eta_2+\eta_2} \qquad (2.18)$$

$$f^{(2)} = e^{\eta_1+\eta_1+\theta_{13}} + e^{\eta_1+\eta_2+\theta_{14}} + e^{\eta_2+\eta_1+\theta_{23}} + e^{\eta_2+\eta_2+\theta_{24}} \qquad (2.19)$$

Where, $e^{\theta_{i(2+l)}} = \frac{1}{3(\eta_i+\eta_l)^2(\eta_i^2+\eta_l^2)}, (i, l = 1,2)$

Using equation (2.17) and (2.19) in equation (2.11), and with the further calculation for equation (2.12) and (2.15), and taking $g^5 = f^6 = \cdots = 0, \varepsilon = 1$, the functions f, g will reduced to

$$\boldsymbol{f_2(t,x) = 1 + e^{\eta_1+\eta_1+\theta_{13}} + e^{\eta_1+\eta_2+\theta_{14}} + e^{\eta_2+\eta_1+\theta_{23}} + e^{\eta_2+\eta_2+\theta_{24}} + e^{\eta_1+\eta_1+\eta_2+\theta_{13}+\theta_{14}+\theta_{23}+\theta_{24}}}$$

$$g_2(t,x) = e^{\eta_1} + e^{\eta_2} + e^{\eta_1+\eta_2+\eta_1+\theta_{12}+\theta_{13}+\theta_{23}} + e^{\eta_1+\eta_2+\eta_2+\theta_{12}+\theta_{14}+\theta_{24}} \quad (2.20)$$

Substituting equation (2.20) into equation (2.5), two-soliton solution of CKdV system (2.2) can be worked out.

3) *N*-Soliton

In general, for N-soliton we have

$$f_N(t,x) = \sum\nolimits_{\zeta=0,1} F_1(\zeta) e^{\sum_{i=1}^{2N}\zeta_i\eta_i+\sum_{1\le i<l}^{2N}\zeta_i\zeta_l\theta_{il}}, \quad (2.21)$$

$$g_N(t,x) = \sum\nolimits_{\zeta=0,1} F_2(\zeta) e^{\sum_{i=1}^{2N}\zeta_i\eta_i+\sum_{1\le i<l}^{2N}\zeta_i\zeta_l\theta_{il}}, \quad (2.22)$$

Where $\sum_{\zeta=0,1}$.represents the summation over all conceivable combination of $\zeta_i = 0,1$ $(i = 1,2,\dots,2N)$. The summation $\sum_{1\le i<l}^{2N}$. means summation over all possible pairs (i,l) picked from the set $\{1,2,\dots,2N\}$ with the situation that $i < l$. The symbol $F_1(\zeta)$ stands for $\sum_{i=1}^{N}\zeta_i = \sum_{i=1}^{N}\zeta_{N+i}$ and $F_2(\zeta)$ stands for $\sum_{i=1}^{N}\zeta_i = 1 + \sum_{i=1}^{N}\zeta_{N+i}$, and

$$e^{\theta_{i(N+l)}} = \frac{1}{3(\eta_i+\eta_l)^2(\eta_i^2+\eta_l^2)}, (i,l = 1,2,\dots,N),$$

$$\eta^{\theta_{il}} = 3(\eta_i-\eta_l)^2(\eta_i^2+\eta_l^2), (i < l = 2,3,\dots,N)$$

Substituting (2.21) and (2.22) in (2.5) leads to the general form of N-soliton solutions. Moreover, as in this manuscript soliton solution of CKdV system is presented numerically containg single soliton solution, double soliton and three-soliton interaction also, hence we can say that CKdV equation (2.2) is completely integrable. Other equations such as the modified KdV equation, the Gardner equation, and the sine-Gordon equation, also give multiple soliton solutions, hence known as completely integrable equations.

This Chapter aims to explore the applicability of "quintic HBS, quintic UAH tension BS, and quintic UAT tension BS with DQM" to solve coupled KdV systems of equations in one dimension numerically.

2.3. Impementation of quintic HBS with DQM

Let's consider the mesh division system $a = x_1 < x_2 < \cdots < x_n = b$ of a closed interval [a,b], which is evenly distributed and has a step size $h = x_{i+1} - x_i$. To calculate numerical solution of CKdV system (2.2), we need to estimate the derivatives. So, for the further computation, we must compute the 1^{st}, 2^{nd}, and 3^{rd} order derivative for functions W, Z. By DQM r^{th} order numerical derivative of W, Z will be estimated as:

$$W_x^{(r)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(r)} W(x_j)$$
$$Z_x^{(r)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(r)} Z(x_j) \qquad (2.23)$$

By using the above equations, the approximate values of derivatives of 1st, 2nd and 3rd order can be given as:

$$W_x^{(1)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(1)} W(x_j), Z_x^{(1)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(1)} Z(x_j)$$
$$W_x^{(2)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(2)} W(x_j), Z_x^{(2)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(2)} Z(x_j) \qquad (2.24)$$
$$W_x^{(3)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(3)} W(x_j), Z_x^{(3)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(3)} Z(x_j)$$

So to obtain the following system of ODEs, we must apply the formulas from equation (2.24) to equation (2.2):

$$W_t = r\left[\sum_{j=1}^{N} d_{ij}^{(3)} W(x_j) + 6W \sum_{j=1}^{N} d_{ij}^{(1)} W(x_j)\right] + 2sZ \sum_{j=1}^{N} d_{ij}^{(1)} Z(x_j)$$

$$Z_t = -\sum_{j=1}^{N} d_{ij}^{(3)} Z(x_j) - 3W \sum_{j=1}^{N} d_{ij}^{(1)} Z(x_j) \qquad (2.25)$$

The first order weighting factors $d_{ij}^{(1)}$ can be determined by using the quintic HBS, as explained in the section (1.7.1), and second or higher order weighting coefficient can be computed with the use of formulae given in the equation (1.40). The Runge-Kutta technique, also known as a (SSP-RK43) regime [155], will next be used to better clarify this scheme of ordinary differential equations.

2.4. Impementation of quintic UAH tension B-spline with DQM

Let's consider the mesh division system $a = x_1 < x_2 < \cdots < x_n = b$ of a closed interval [a,b], which is evenly distributed and has a step size $h = x_{i+1} - x_i$. To calculate the numerical solution of CKdV system (2.2), we need to estimate the derivatives. So, for the further computation, we must compute the needed derivative of the functions W, Z. So, by DQM, the approximate values of derivatives of 1st, 2nd and 3rd order can be given as:

$$W_x^{(1)}(x_i) = \sum_{j=1}^{N} n_{ij}^{(1)} W(x_j), Z_x^{(1)}(x_i) = \sum_{j=1}^{N} n_{ij}^{(1)} Z(x_j)$$

$$W_x^{(2)}(x_i) = \sum_{j=1}^{N} n_{ij}^{(2)} W(x_j), Z_x^{(2)}(x_i) = \sum_{j=1}^{N} n_{ij}^{(2)} Z(x_j) \qquad (2.26)$$

$$W_x^{(3)}(x_i) = \sum_{j=1}^{N} n_{ij}^{(3)} W(x_j), Z_x^{(3)}(x_i) = \sum_{j=1}^{N} n_{ij}^{(3)} Z(x_j)$$

So to obtain the following system of ODEs, we must apply the formulas from equation (2.26) to equation (2.2):

$$W_t = r\left[\sum_{j=1}^{N} n_{ij}^{(3)} W(x_j) + 6W\sum_{j=1}^{N} n_{ij}^{(1)} W(x_j)\right] + 2sZ\sum_{j=1}^{N} n_{ij}^{(1)} Z(x_j)$$

$$Z_t = -\sum_{j=1}^{N} n_{ij}^{(3)} Z(x_j) - 3W\sum_{j=1}^{N} n_{ij}^{(1)} Z(x_j) \qquad (2.27)$$

The first order weighting factors $n_{ij}^{(1)}$ can be determined by using the quintic UAH, as explained in the section (1.7.2), and second or higher order weighting coefficient can be computed with the use of formulae given in the equation (1.40). The Runge-Kutta technique, also known as a SSP-RK43 regime [155], will next be used to better clarify this scheme of ordinary differential equations.

2.5. Impementation of quintic UAT tension B-spline with DQM

Let's consider the mesh division system $a = x_1 < x_2 < \cdots < x_n = b$ of a closed interval [a,b], which is evenly distributed and has a step size $h = x_{i+1} - x_i$. To calculate the numerical solution of CKdV system (2.2), we need to estimate the derivatives. So, for the further computation, we must compute the needed derivative of the function W, Z. So, by DQM, the approximate values of derivatives of 1st, 2nd and 3rd order can be given as:

$$W_x^{(1)}(x_i) = \sum_{j=1}^{N} v_{ij}^{(1)} W(x_j), Z_x^{(1)}(x_i) = \sum_{j=1}^{N} v_{ij}^{(1)} Z(x_j)$$

$$W_x^{(2)}(x_i) = \sum_{j=1}^{N} v_{ij}^{(2)} W(x_j), Z_x^{(2)}(x_i) = \sum_{j=1}^{N} v_{ij}^{(2)} Z(x_j) \qquad (2.28)$$

$$W_x^{(3)}(x_i) = \sum_{j=1}^{N} v_{ij}^{(3)} W(x_j), Z_x^{(3)}(x_i) = \sum_{j=1}^{N} v_{ij}^{(3)} Z(x_j)$$

So to obtain the following system of ODEs, we must apply the formulas from equation (2.28) to equation (2.2):

$$W_t = r\left[\sum_{j=1}^{N} v_{ij}^{(3)} W(x_j) + 6W \sum_{j=1}^{N} v_{ij}^{(1)} W(x_j)\right] + 2sZ \sum_{j=1}^{N} v_{ij}^{(1)} Z(x_j)$$

$$Z_t = -\sum_{j=1}^{N} v_{ij}^{(3)} Z(x_j) - 3W \sum_{j=1}^{N} v_{ij}^{(1)} Z(x_j) \qquad (2.29)$$

The first order weighting factors $v_{ij}^{(1)}$ can be calculated with the implementation of the quintic UAT as explained in the section (1.7.3), and second or higher order weighting coefficient can be computed with the use of formulae given in the equation (1.40). The Runge-Kutta technique, also known as a SSP-RK43 regime [155], will next be used to better clarify this scheme of ordinary differential equations.

2.6. Numerical solution of CKdV employing three B-splines

In this module different examples are solved to check the preciseness and accurateness of the proposed method. Two test problems N-soliton solution and Birth of soliton is represented for numerical simulations. To achieve this, different numerical tests are executed like the single soliton test, double soliton, three soliton interaction and the

birth of solitons. To know more about the reliability of the used method, the errors are evaluated using formulas:

$$L_2(W) = \sqrt{h\sum_{j=0}^{N}\left|W_j^{exact} - W_j^{numerical}\right|^2} \qquad (2.30)$$

$$L_\infty(W) = \max_{0\le j\le N}\left|W_j^{exact} - W_j^{numerical}\right|$$

In the same way,

$$L_2(Z) = \sqrt{h\sum_{j=0}^{N}\left|Z_j^{exact} - Z_j^{numerical}\right|^2} \qquad (2.31)$$

$$L_\infty(Z) = \max_{0\le j\le N}\left|Z_j^{exact} - Z_j^{numerical}\right|$$

Also, the conservation of the method is checked by evaluating the conserved values I_1, I_2, I_3 that is

$$I_1 = \int_{-\infty}^{\infty} W\,dx \text{ and } I_2 = \int_{-\infty}^{\infty}\left(W^2 + \frac{2}{3}sZ^2\right)dx,$$

$$I_3 = \int_{-\infty}^{\infty}\left[(1+r)\left(W^3 - \frac{1}{2}W_x^2\right) + s(WZ^2 - Z_x^2)\right]dx$$

Example 1: N-Soliton Solution

N-soliton colabration of the CKdV can be disintegrated into N single soliton elements. The construction of N-Soliton solution of CKdV can generalised. Accordingly, generalised initial condition for N-Soliton solution can be framed as:

$$W(x,0) = \sum_{e=1}^{N} W_e(x,0), \qquad Z(x,0) = \sum_{e=1}^{N} Z_e(x,0) \qquad (2.32)$$

where, $W_e(x,0) = 2\rho_e^2 sech^2(\mu_e)$, $Z_e(x,0) = \frac{1}{2\sqrt{\phi_e}} sech(\mu_e)$

with $\mu_e = \rho_e(x - y_e) + \frac{1}{2\log(\phi_e)}$, $\phi_e = \frac{-s}{8(4r+1)\rho_e^4}$, $e = 1,2,3,\dots,N$

where by assigning the finite value to N, the respective soliton interaction can be evaluated. Here, N-Soliton interaction are shown with the examples for the cases N=1, 2, 3 named as single soliton, double soliton, three soliton interaction. Moreover birth of solitons with the progress in time is also formulated. As we are computing the numerical solution of CKdV equation, hence limiting the value of N-Soliton to a finite number because of the limitation in programming. So, N-soliton interaction is subdivided into three parts as 1(A) Single Soliton, 1(B) Double Solitons and 1(C) Three soliton interactions. Also, single soliton is represented in three different cases with different set of parameters.

Example 1(A): Single Soliton

For Single soliton with N=1 in equation (2.32), Coupled KdV equation has the exact solution as

$$W(x,t) = 2\rho^2 sech^2(\mu) \quad and \quad Z(x,t) = \frac{1}{2\sqrt{\phi}} sech(\mu), \tag{2.33}$$

where $\mu = \rho(x - \rho^2 t) + \frac{1}{2\log(\phi)}$, $\phi = \frac{-s}{8(4r+1)\rho^4}$

and initial condition as:

$$W(x,0) = 2\rho^2 sech^2(\mu) \; and \; Z(x,0) = \frac{1}{2\sqrt{\phi}} sech(\mu), \tag{2.34}$$

where $\mu = \rho x + \frac{1}{2\log(\phi)}$, $\phi = \frac{-s}{8(4r+1)\rho^4}$

and boundary condition

$$W(x_1,t) = W(x_2,t) = Z(x_1,t) = Z(x_2,t) = 0 \qquad (2.35)$$

To express a single soliton's numerically we have three sets of parameters [132] as:

(a): $r = 0.5,\ s = -3, \rho = 0.5$

(b): $r = -0.125, s = -3, \rho = 0.5$

(c): $r = -0.5,\ s = 3, \rho = 0.5$

A soliton is a name given to the solitary wave that behaves like a particle satisfying the condition that when it propagates with the constant velocity it must maintain its shape and also a soliton conserves its amplitude, speed, and shape after a collision with a different soliton. For single solitons numerical findings are acquired with $N = 421, \Delta t = 0.0001, t = 5$ to $t = 20$, in domain [-25, 25] The graphical depiction of analytical and numerical findings for $N = 421$ at various times are shown in the figures using three splines HBS, UAH tension BS and UAT tension BS.

(a) Let's take the CKdV equation (2.2) with values of parameters $r = 0.5, s = -3, \rho = 0.5$ with precise elucidation (2.33) and initial boundary conditions (2.34). In Figure 2.1 and Figure 2.2 compatibility of numerical findings with exact sol. is presented using HBS. Comparison for the same is depicted in Figure 2.3, Figure 2.4 with applicability of UAH tension BS and in Figure 2.5, Figure 2.6 using UAT tension BS. In Table 2.1 for soliton W, and in Table 2.2 for soliton Z error norms L_2 and L_∞ are calculated, which are reasonably small and are also superior comparable to the earlier findings reported in the literature until $t = 5$. Likewise conserved quantities I_1, I_2 and I_3 for CKdV are also calculated and represented in Table 2.3. From the tabulated values of the invariants I_1, I_2 and I_3 it is very evident that they are almost constant and are congruent with the prior work.

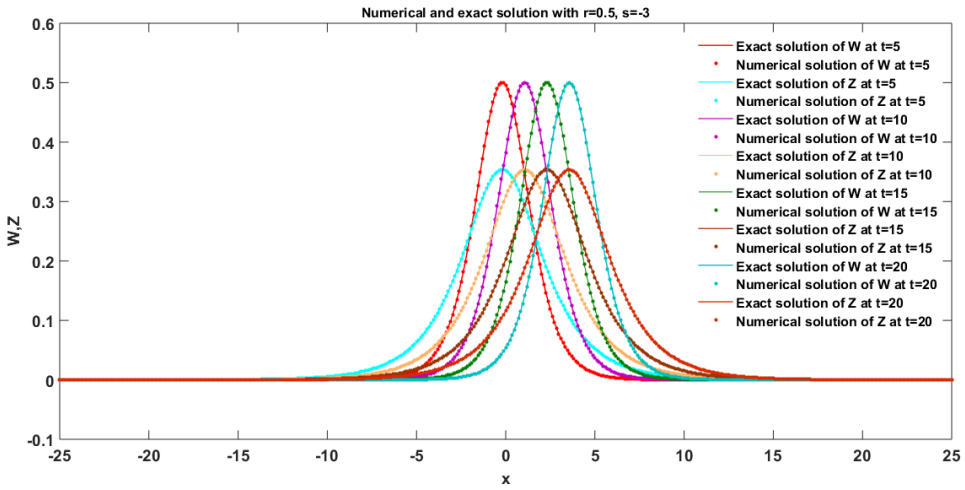

Figure 2.1: Comparability of exact and num. sol. at several time values for case (a) of example 1(A) with HBS.

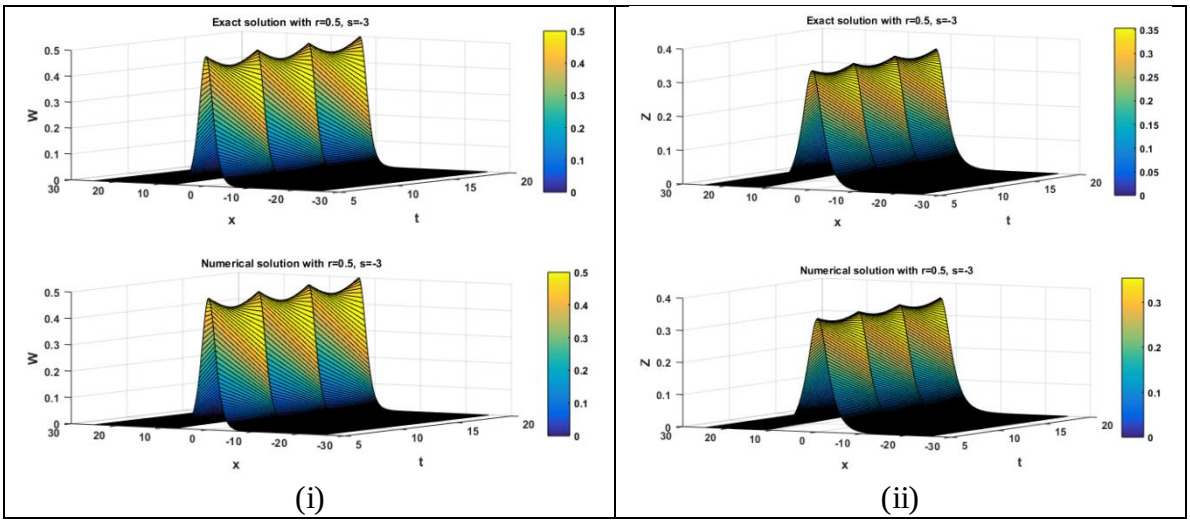

Figure 2.2: Comparability of exact and num. solutions for $(i)\ W(x,t)$, $(ii)\ Z(x,t)$ at various time values for case (a) of example 1(A) with HBS.

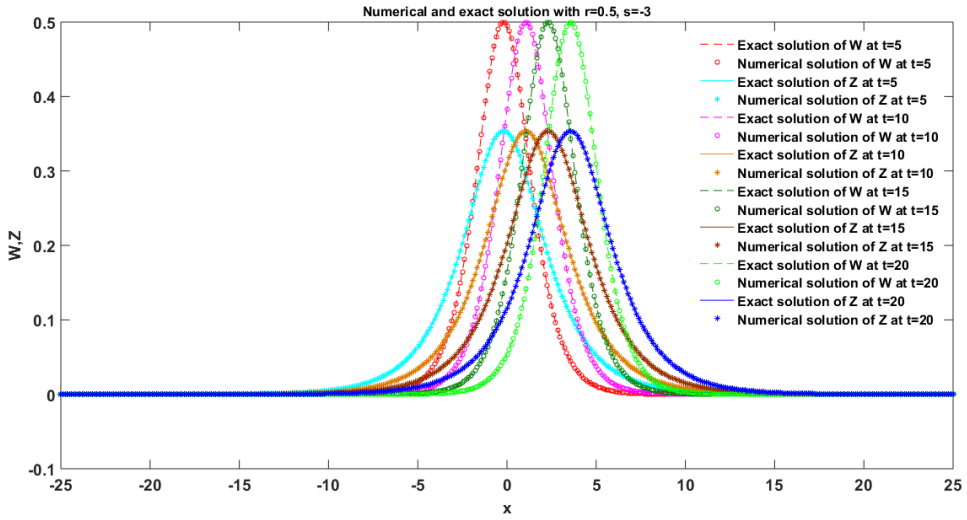

Figure 2.3: Comparability of exact and num. sol. at several time values for case (a) of example 1(A) with UAH tension B-spline.

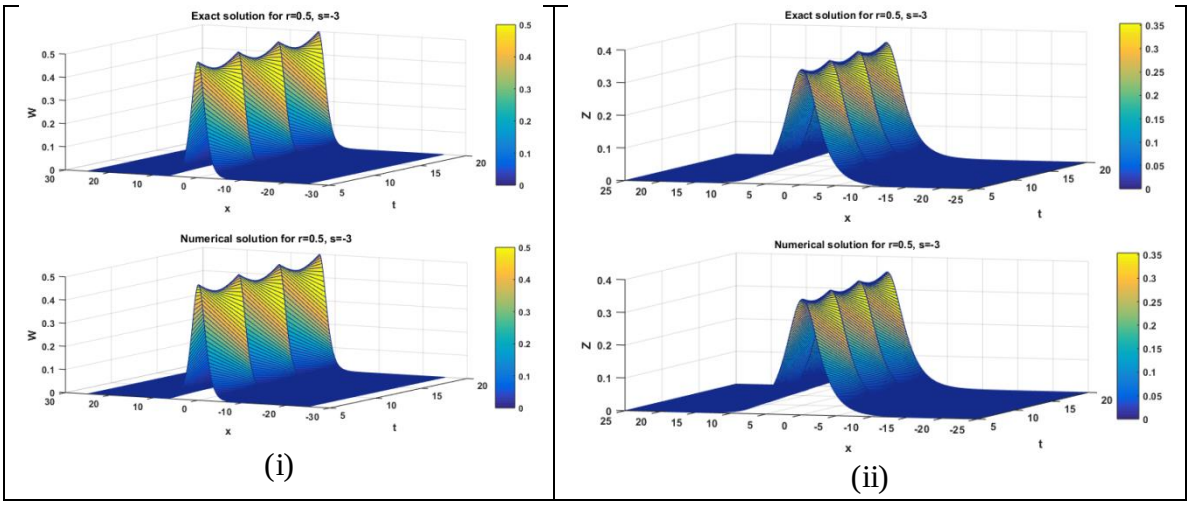

Figure 2.4: Comparison of exact and num. solutions of $(i)\ W(x,t)$, $(ii)\ Z(x,t)$ at various time values for case (a) of example 1(A) with UAH tension B-spline.

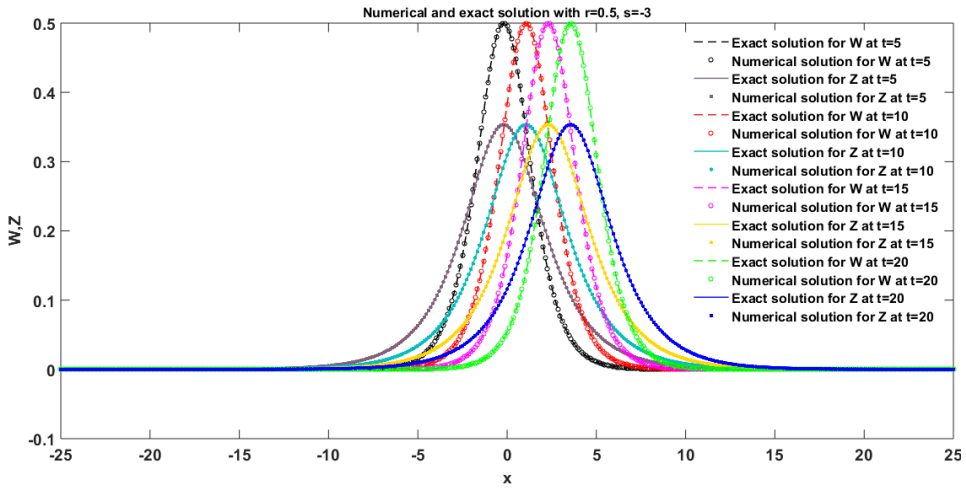

Figure 2.5: Comparability of exact and num. sol. at several time values for case (a) of example 1(A) with UAT tension B-spline.

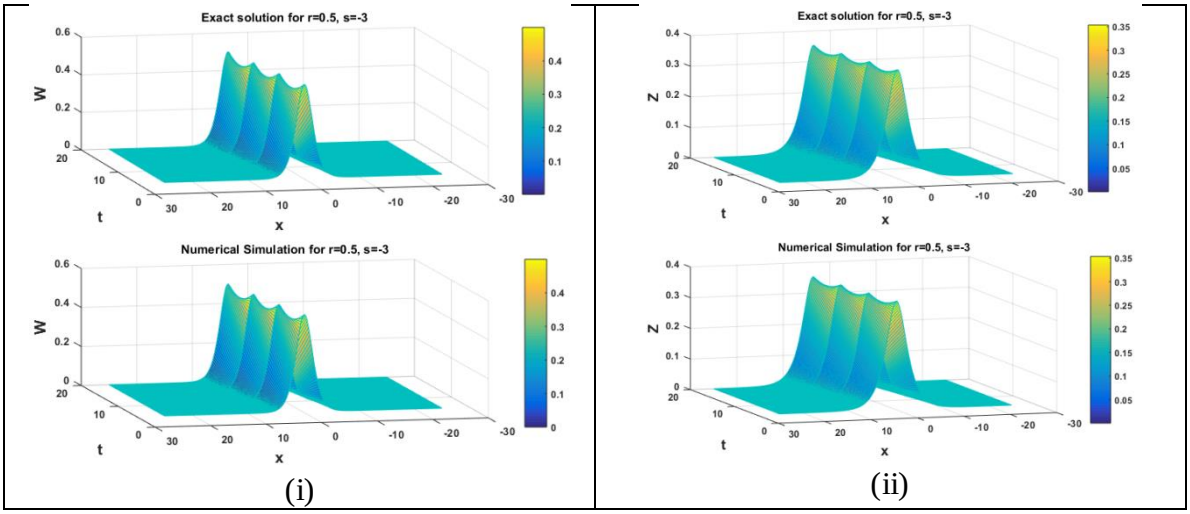

Figure 2.6: Comparison of exact and num. solutions of $(i)\ W(x,t)$, $(ii)\ Z(x,t)$ at various time values for case (a) of example 1(A) with UAT tension B-spline.

Table 2.1: Comparability of error norms for single soliton W(x,t) with parameters $\Delta t = 0.0001, r = 0.5, s = -3, \rho = 0.5$ (single soliton case (a)).

t	$L_2(W)$					$L_\infty(W)$				
	Başhan [140]	Raslan et al. [142]	HBS	UAH tension B-spline	UAT tension B-spline	Başhan [140]	Raslan et al. [142]	HBS	UAH tension B-spline	UAT tension B-spline
1	1.7000	6.06802E	2.5678	1.8347	1.8364	1.1000	3.97188	1.4709	1.0594	1.0597

	E-06	-07	E-07	E-07	E-07	E-06	E-07	E-07	E-07	E-07
2	2.4000 E-06	8.99862E -07	1.3685 E-06	1.3695 E-06	1.3696 E-06	1.5000 E-06	5.47651 E-07	5.4758 E-07	5.0850 E-07	5.0868 E-07
3	3.3000 E-06	1.18161E -06	2.7038 E-06	2.6507 E-06	2.6506 E-06	1.9000 E-06	6.89839 E-07	9.6544 E-07	8.4958 E-07	8.4987 E-07
4	3.5000 E-06	1.28135E -06	4.3591 E-06	4.2195 E-06	4.2191 E-06	1.9000 E-06	6.98286 E-06	1.5686 E-06	1.6331 E-06	1.6333 E-06
5	4.1000 E-06	1.88599E -06	6.1971 E-06	5.8549 E-06	5.8538 E-06	2.3000 E-06	1.05770 E-06	2.2274 E-06	2.1317 E-06	2.1312 E-06

Table 2.2: Comparability of error norms for single soliton Z(x,t) with parameters $\Delta t = 0.0001, r = 0.5, s = -3, \rho = 0.5$ (single soliton case (a)).

t	$L_2(Z)$					$L_\infty(Z)$				
	Başh-an [140]	Raslan et al. [142]	HBS	UAH tension B-spline	UAT tension B-spline	Başh-an [140]	Raslan et al. [142]	HBS	UAH tension B-spline	UAT tension B-spline
1	7.3000 E-06	4.43377 E-06	2.2590 E-06	2.2720 E-06	2.2721 E-06	4.8000 E-06	3.02357 E-06	1.2714 E-06	1.1286 E-06	1.1303 E-06
2	7.6000 E-06	6.08938 E-06	2.5222 E-06	2.5327 E-06	2.5330 E-06	4.4000 E-06	3.83209 E-06	1.4262 E-06	1.2151 E-06	1.2154 E-06
3	7.9000 E-06	7.21810 E-06	2.7308 E-06	2.7536 E-06	2.7539 E-06	3.7000 E-06	4.73962 E-06	1.3928 E-06	1.3561 E-06	1.3614 E-06
4	8.3000 E-06	1.06873 E-05	3.0432 E-06	3.0406 E-06	3.0408 E-06	3.9000 E-06	2.21990 E-06	1.3106 E-06	1.3269 E-06	1.3253 E-06
5	8.4000 E-06	2.95667 E-05	3.4302 E-06	3.4479 E-06	3.4479 E-06	3.3000 E-06	8.57852 E-05	1.7440 E-06	1.9081 E-06	1.9036 E-06

Table 2.3: Conserved quantities for single soliton with $r = 0.5, s = -3, \rho = 0.5, \Delta t = 0.0001$ *and for the interval [-25, 25] till t=5.*

t	I_1				I_2				I_3		
	Başhan [140]	HBS	UAH tension B-spline	UAT tension B-spline	Başhan [140]	HBS	UAH tension B-spline	UAT tension B-spline	HBS	UAH tension B-spline	UAT tension B-spline
1	1.999998	2.0000	2.0000	2.0000	-0.333333	-0.3333	-0.3333	-0.3333	-0.0750	-0.0750	-0.0750
2	2.0000000	2.0000	2.0000	2.0000	-0.333333	-0.3333	-0.3333	-0.3333	-0.0750	-0.0750	-0.0750
3	1.999999	2.0000	2.0000	2.0000	-0.33333	-0.333	-0.333	-0.333	-0.075	-0.075	-0.075

4	2.0000000	2.0000	2.0000	2.0000	3 - 0.333335	3 - 0.3333	3 - 0.3333	3 - 0.3333	0 - 0.075	0 - 0.075	0 - 0.075
5	2.0000000	2.0000	2.0000	2.0000	3 - 0.333333	3 - 0.3333	3 - 0.3333	3 - 0.3333	0 - 0.075	0 - 0.075	0 - 0.075

(b) Let's take the CKdV equation (2.2) with values of parameters $r = -0.125, s = -3, \rho = 0.5$ having precise elucidation (2.33) and initial boundary conditions (2.34). In Figure 2.7 and Figure 2.8 compatibility of numerical findings with precise sol. is presented using HBS. Comparison for the same is depicted in Figure 2.9, Figure 2.10 with applicability of UAH tension BS and in Figure 2.11, Figure 2.12 using UAT tension BS. In Table 2.4 for soliton W, and in Table 2.5 for soliton Z error norms L_2 and L_∞ are calculated, which are reasonably small and are also superior comparable to the earlier findings reported in literature until $t = 20$. Likewise conserved quantities I_1, I_2 and I_3 for CKdV are also calculated and represented in Table 2.6. From the tabulated values of the invariants I_1, I_2 and I_3 it is very evident that they are almost constant and are congruent with the prior work.

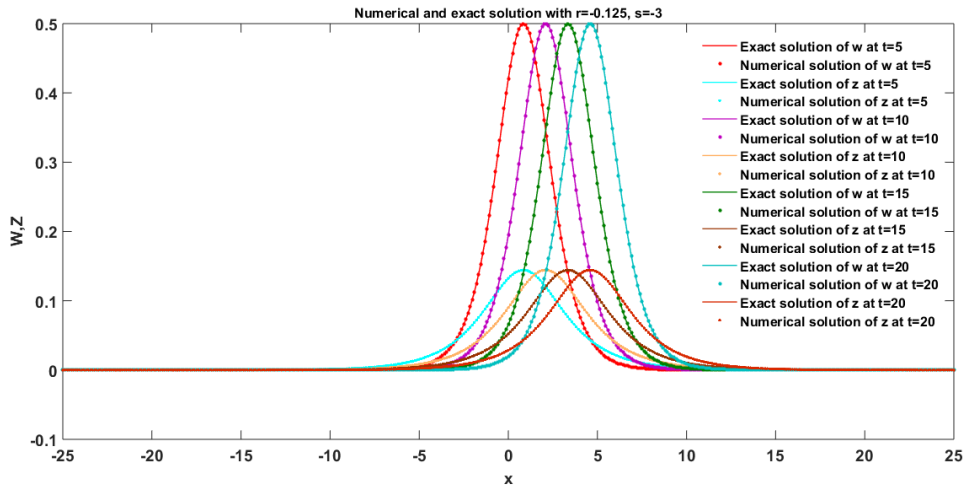

Figure 2.7: Comparison of exact and numerical solutions at different values of time for case (b) of example 1(A) with HBS.

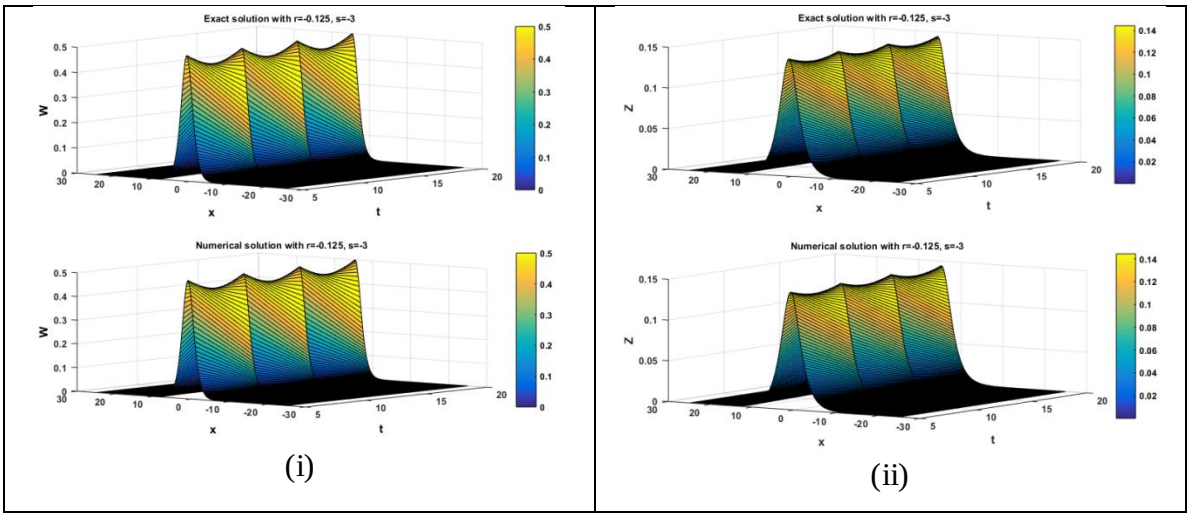

Figure 2.8: Comparison of exact and num. sol. of $(i)\ W(x,t)$, $(ii)\ Z(x,t)$ at various time values of time for case (b) of example 1(A) with HBS.

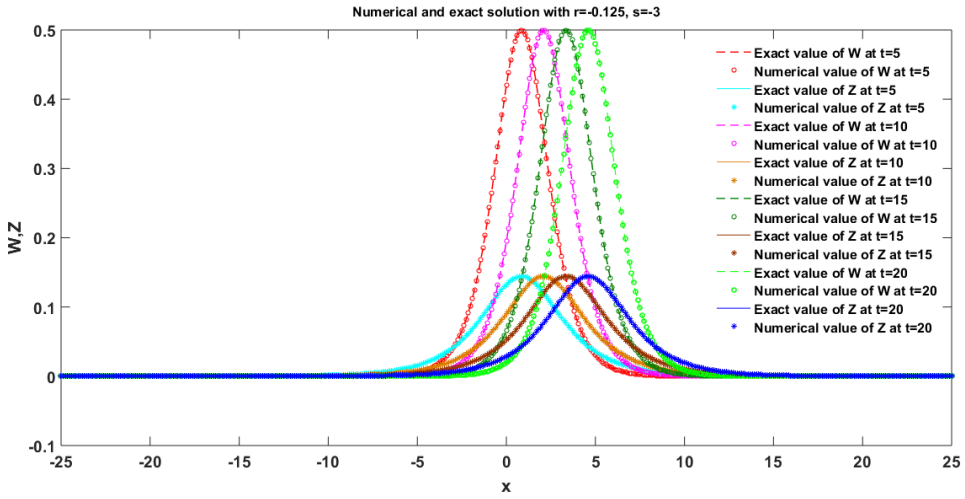

Figure 2.9: Comparability of exact and num. sol. at several time values for case (b) of example 1(A) with UAH tension B-spline.

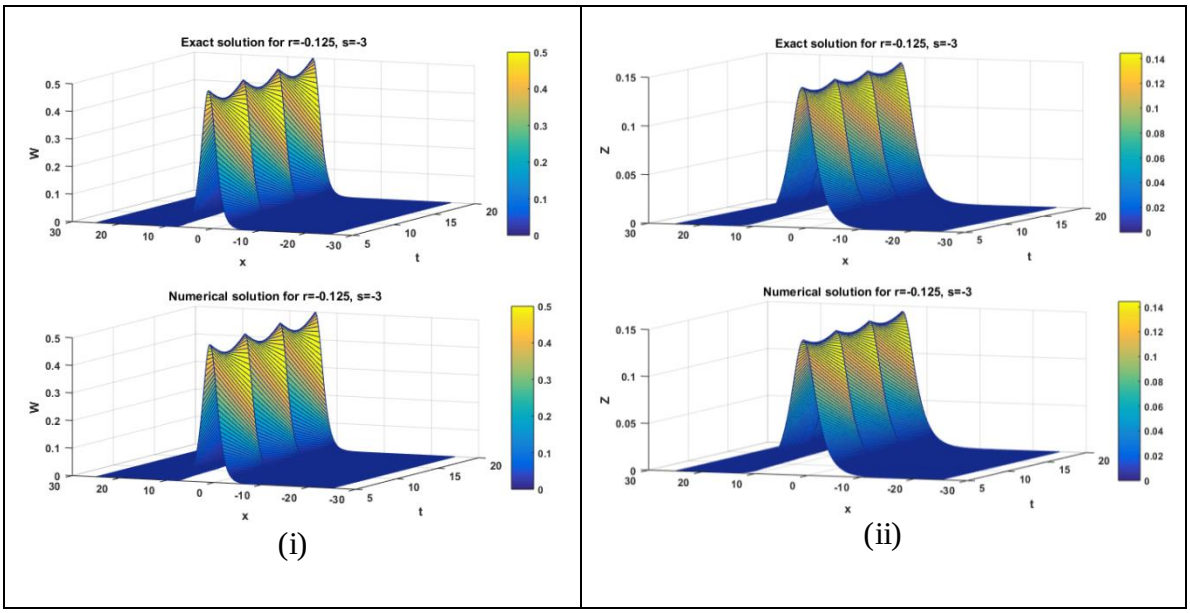

Figure 2.10: Comparison of exact and num. solutions of $(i)\ W(x,t)$, $(ii)\ Z(x,t)$ at various time values for case (b) of example 1(A) with UAH tension B-spline.

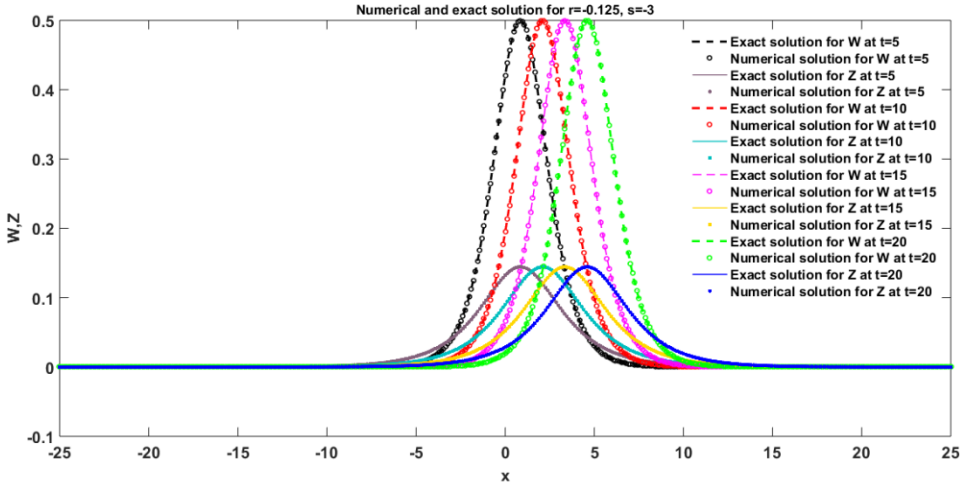

Figure 2.11: Comparability of exact and num. solutions at various time values for case (b) of example 1(A) with UAT tension B-spline.

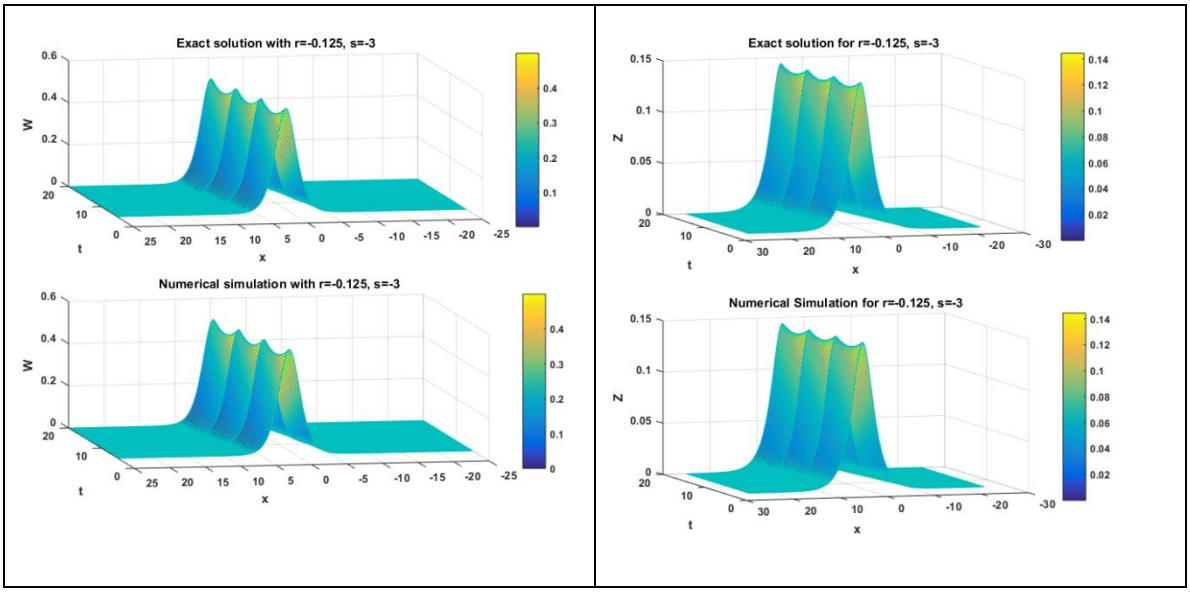

Figure 2.12: Comparability of exact and num. solutions of $(i)\ W(x,t)$, $(ii)\ Z(x,t)$ at various time values for case (b) of example 1(A) with UAT tension B-spline.

Table 2.4: Comparability of error norms for single soliton W(x,t) with parameters $\Delta t = 0.0001, r = -0.125, s = -3, \rho = 0.5$ (single soliton case (b)).

t	$L_2(W)$				$L_\infty(W)$			
	Başhan [140]	HBS	UAH tension BS	UAT tension BS	Başhan [140]	HBS	UAH tension BS	UAT tension BS
2	8.0000E-07	1.7410 E-07	1.1941E-07	1.1942E-07	5.0000E-07	1.3247E-07	7.0362E-08	7.0419E-08
4	1.1000E-06	4.0045 E-07	3.2216E-07	3.2210E-07	7.0000E-07	2.4682E-07	1.5463E-07	1.5458E-07
6	9.0000E-07	6.1505 E-07	5.3366E-07	5.3354E-07	5.0000E-07	2.7127E-07	2.0175E-07	2.0163E-07
8	1.6000E-06	8.5473 E-07	7.6789E-07	7.6774E-07	1.0000E-06	3.7777E-07	2.9167E-07	2.9140E-07
10	1.3000E-06	1.1889 E-06	1.1245E-06	1.1243E-06	8.0000E-07	5.0059E-07	4.5458E-07	4.5469E-07
15	2.2000E-06	2.8773 E-06	2.6593E-06	2.6587E-06	1.3000E-06	1.875E-06	1.0184E-06	1.0181E-06
20	4.1000E-06	6.2955 E-06	5.8143E-06	5.8128E-06	2.7000E-06	2.3245E-06	1.9279E-06	1.9271E-06

Table 2.5: Comparability of error norms for single soliton Z(x,t) with parameters $\Delta t = 0.0001, r = -0.125, s = -3, \rho = 0.5$ (single soliton case (b)).

$L_2(Z)$	$L_\infty(Z)$

t	Başhan [140]	HBS	UAH tension BS	UAT tension BS	Başhan [140]	HBS	UAH tension BS	UAT tension BS
2	2.5000E-06	7.4631E-07	7.5720E-07	7.5728E-07	1.2000E-06	3.5384E-07	3.6552E-07	3.6532E-07
4	3.0000E-06	9.8153E-07	9.7824E-07	9.7823E-07	1.4000E-06	5.6291E-07	4.9960E-07	4.9884E-07
6	3.3000E-06	1.2688E-06	1.2494E-06	1.2491E-06	1.5000E-06	7.4539E-07	7.1415E-07	7.1599E-07
8	3.9000E-06	1.5424E-06	1.4879E-06	1.4875E-06	2.1000E-06	8.9148E-07	8.3063E-07	8.3237E-07
10	4.5000E-06	1.9168E-06	1.8894E-06	1.8891E-06	2.7000E-06	1.0857E-06	1.1851E-06	1.1888E-06
15	7.5000E-06	3.6618E-06	3.5386E-06	3.5384E-06	5.0000E-06	2.2573E-06	2.2127E-06	2.2129E-06
20	1.4000E-05	7.6306E-06	7.4228E-06	7.4221E-06	9.5000E-06	4.7940E-06	4.7203E-06	4.7046E-06

Table 2.6: Conserved quantities for single soliton with $r = -0.125, s = -3, \rho = 0.5, \Delta t = 0.0001$ *and for the interval [-25, 25] till t=20.*

t	I_1				I_2				I_3		
	Başh-an [140]	HBS	UAH tension BS	UAT tension BS	Başh-an [140]	HBS	UAH tension BS	UAT tension BS	HBS	UAH tension BS	UAT tensio n BS
2	1.999999	2.0000	2.0000	2.0000	0.500000	0.5000	0.5000	0.5000	0.1125	0.1125	0.1125
4	1.999999	2.0000	2.0000	2.0000	0.500001	0.5000	0.5000	0.5000	0.1125	0.1125	0.1125
6	1.999999	2.0000	2.0000	2.0000	0.500000	0.5000	0.5000	0.5000	0.1125	0.1125	0.1125
8	1.999999	2.0000	2.0000	2.0000	0.500000	0.5000	0.5000	0.5000	0.1125	0.1125	0.1125
10	2.000000	2.0000	2.0000	2.0000	0.500000	0.5000	0.5000	0.5000	0.1125	0.1125	0.1125
15	1.999999	2.0000	2.0000	2.0000	0.500000	0.5000	0.5000	0.5000	0.1125	0.1125	0.1125
20	2.000000	2.0000	2.0000	2.0000	0.500000	0.5000	0.5000	0.5000	0.1125	0.1125	0.1125

(c) Let's take the CKdV equation (2.2) with values of parameters $r = -0.5, s = 3, \rho = 0.5$ with precise elucidation (2.33) and initial boundary conditions (2.34). In Figure 2.13 and Figure 2.14 compatibility of numerical findings with precise sol. is presented using HBS. Comparison for the same is depicted in Figure 2.15, Figure 2.16 with applicability of UAH tension B-spline and in Figure 2.17, Figure 2.18 using UAT

tension BS. In Table 2.7 for soliton W, and in Table 2.8 for soliton Z error norms L_2 and L_∞ are calculated, which are reasonably small and are also superior comparable to the earlier findings reported in the literature until $t = 20$. Likewise conserved quantities I_1, I_2 and I_3 for CKdV are also calculated and represented in Table 2.9. From the tabulated values of the invariants I_1, I_2 and I_3 it is very evident that they are almost constant and are congruent with the prior work.

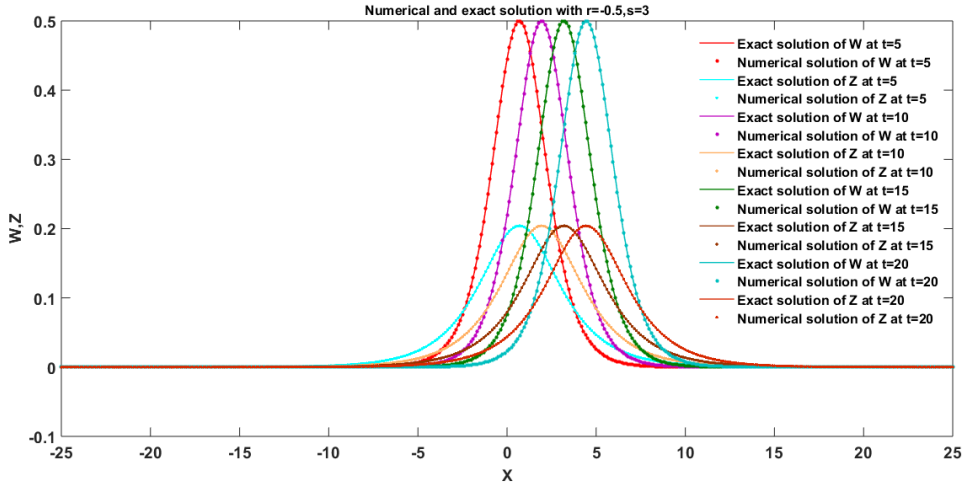

Figure 2.13: Comparability of exact and num. sol. at several time values for case (c) of example 1(A) with HBS.

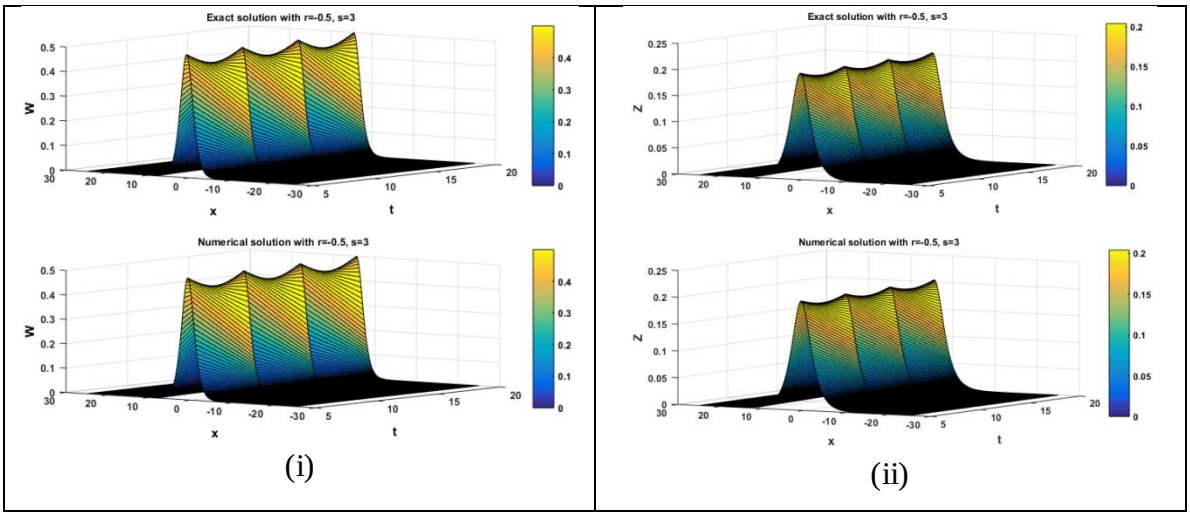

Figure 2.14: Comparability of exact and num. sol. of $(i)\ W(x,t)$, $(ii)\ Z(x,t)$ at various time values for case (c) of example 1(A) with HBS.

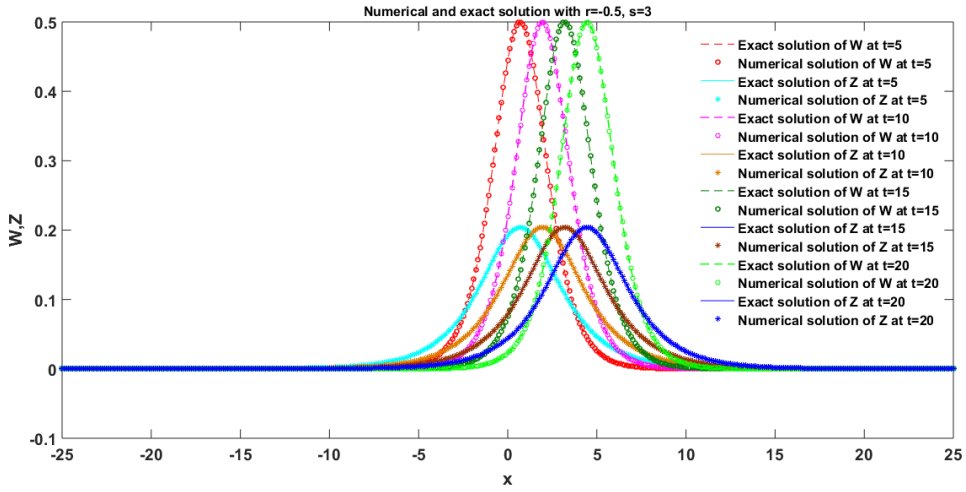

Figure 2.15: Comparability of exact and num. sol. at various time values for case (c) of example 1(A) with UAH tension B-spline.

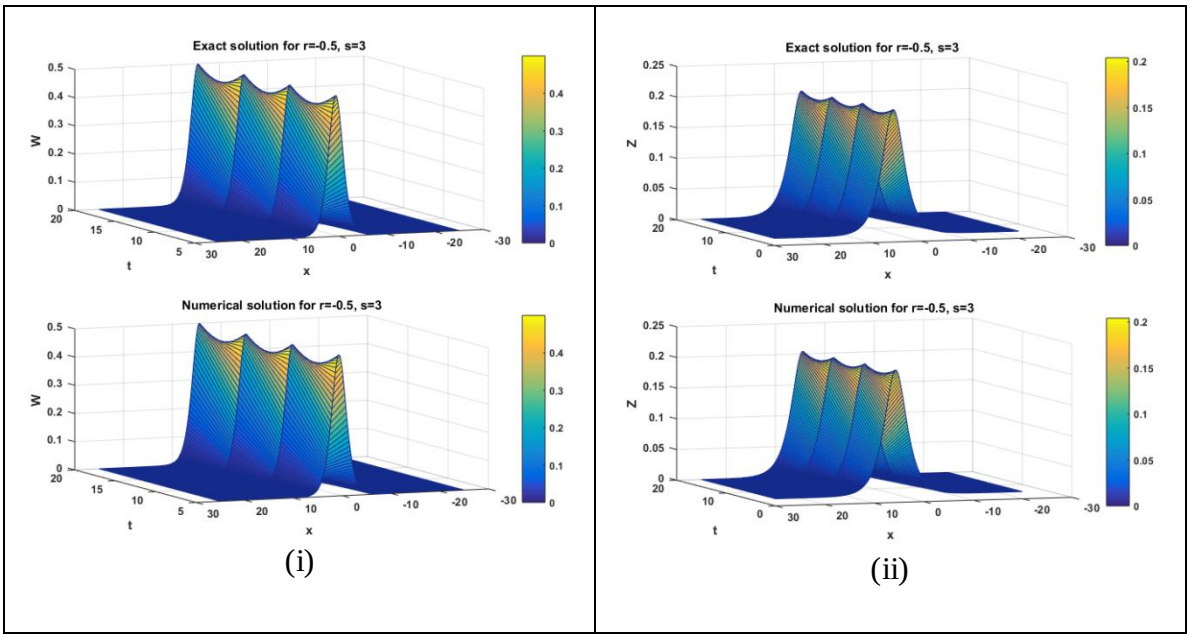

Figure 2.16: Comparison of exact and num. sol. of $(i)\ W(x,t)$, $(ii)\ Z(x,t)$ at different values of time for case (c) of example 1(A) with UAH tension B-spline.

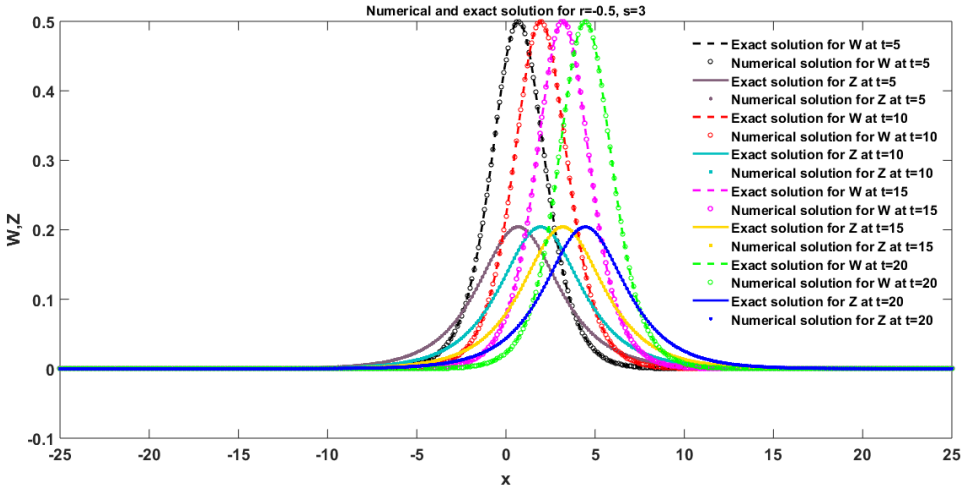

Figure 2.17: Comparability of exact and num. sol. at several time values for case (c) of example 1(A) with UAT tension B-spline.

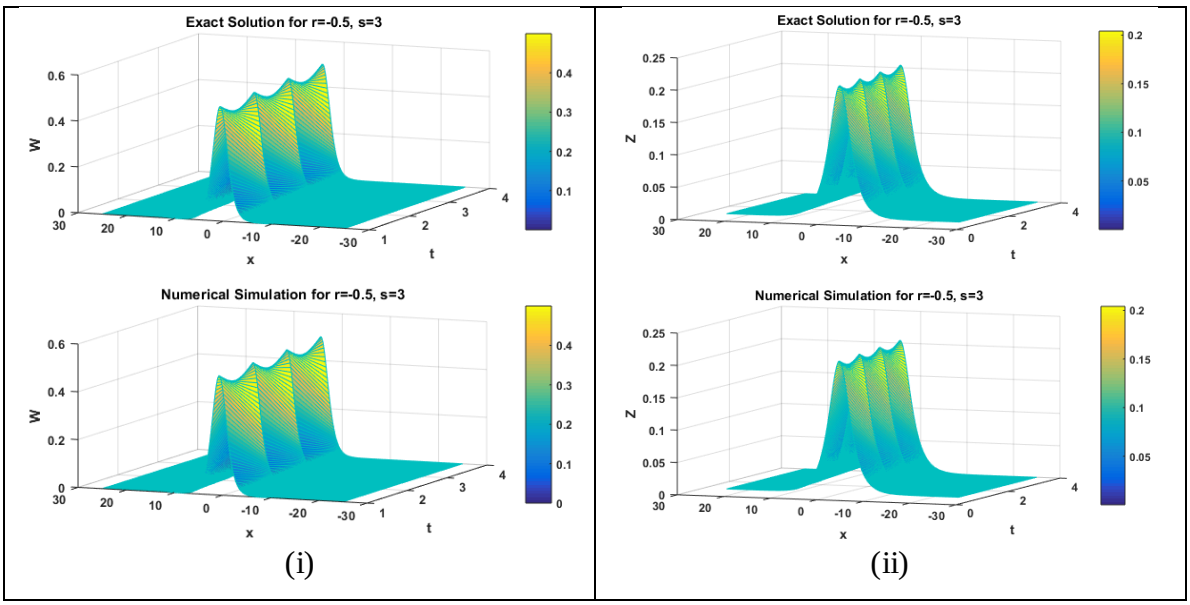

Figure 2.18: Comparison of exact and num. sol. of $(i)\ W(x,t)$, $(ii)\ Z(x,t)$ at various time values for case (c) of example 1(A) with UAT tension B-spline.

Table 2.7: Comparability of error norms for single soliton W(x,t) with parameters $\Delta t = 0.0001, r = -0.5, s = 3, \rho = 0.5$ (single soliton case (c)).

t	$L_2(W)$					$L_\infty(W)$				
	Raslan et al. [142]	Başha-n [140]	HBS	UAH tension BS	UAT tension BS	Raslan et al. [142]	Başhan [140]	HBS	UAH tension BS	UAT tension BS
0.5	4.1592 3E-07	-	1.4567 E-07	7.4375 E-08	7.4360 E-08	2.9676 1E-07	-	1.0745 E-07	3.2508 E-08	3.2519 E-08
1	5.5971 2E-07	-	2.2510 E-07	1.2296 E-07	1.2294 E-07	3.4258 8E-07	-	1.4681 E-07	5.9093 E-08	5.9040 E-08
1.5	6.4125 5E-07	-	3.1563 E-07	2.0746 E-07	2.0743 E-07	3.7505 8E-07	-	1.9969 E-07	8.7385 E-08	8.7288 E-08
2	7.0035 5E-07	-	3.9617 E-07	3.3488 E-07	3.3496 E-07	4.0978 4E-07	-	2.6674 E-07	1.7076 E-07	1.7098 E-07
5	-	3.0000 E-06	1.4712 E-06	1.3742 E-06	1.3741 E-06	-	2.0000E -06	6.6186 E-07	6.1263 E-07	6.1173 E-07
10	-	4.0000 E-06	4.0015 E-06	3.6695 E-06	3.6684 E-06	-	2.1000 E-06	1.2940 E-06	1.2265 E-06	1.2253 E-06
15	-	5.3000 E-06	9.7901 E-06	9.0598 E-06	9.0573 E-06	-	2.7000 E-06	3.0917 E-06	2.6594 E-06	2.6597 E-06
20	-	6.7000 E-06	2.3964 E-05	2.2217 E-05	2.2211 E-05	-	3.1000 E-06	6.8767 E-06	7.2341 E-06	7.2337 E-06

Table 2.8: Comparability of error norms for single soliton Z(x,t) with parameters $\Delta t = 0.0001, r = -0.5, s = 3, \rho = 0.5$ (single soliton case (c)).

t	$L_2(Z)$					$L_\infty(Z)$				
	Raslan et al. [142]	Başha-n [140]	HBS	UAH tension BS	UAT tension BS	Raslan at al. [142]	Başhan [140]	HBS	UAH tension BS	UAT tension BS
0.5	9.6420 9E-07	-	8.9124 E-07	8.7019 E-07	8.7013 E-07	7.9101 9E-07	-	5.8314 E-07	5.5990 E-07	5.5993 E-07
1	1.3432 9E-06	-	1.0314 E-06	1.0317 E-06	1.0318 E-06	1.0382 1E-06	-	6.1459 E-07	6.3185 E-07	6.3283 E-07
1.5	1.5795 7E-06	-	1.0391 E-06	1.0290 E-06	1.0289 E-06	1.1012 2E-06	-	5.5395 E-07	5.1346 E-07	5.1210 E-07
2	1.7764 7E-06	-	1.0756 E-06	1.0908 E-06	1.0909 E-06	1.3066 0E-06	-	5.3859 E-07	5.3237 E-07	5.3198 E-07
5	-	4.3000 E-06	1.5746 E-06	1.5715 E-06	1.5714 E-06	-	2.0000E -06	1.0087 E-06	1.0153 E-06	1.0148 E-06
10	-	6.0000 E-06	2.1652 E-06	2.1387 E-06	2.1382 E-06	-	3.9000 E-06	1.2671 E-06	1.3922 E-06	1.3983 E-06
15	-	9.6000 E-06	2.6512 E-06	2.5946 E-06	2.5948 E-06	-	6.4000 E-06	2.0182 E-06	2.0387 E-06	2.0562 E-06
20	-	1.7900 E-05	3.5967 E-06	3.5488 E-06	3.5493 E-06	-	1.2000 E-05	3.6167 E-06	3.5783 E-06	3.5955 E-06

Table 2.9: Conserved quantities for single soliton with $r = -0.5, s = 3, \rho = 0.5, \Delta t = 0.0001$ and for the interval [-25, 25] till t=20.

t	I_1				I_2				I_3		
	Başh-an [140]	HBS	UAH	UAT	Başhan [140]	HBS	UAH	UAT	HBS	UAH	UAT
5	2.000 001	2.0000	2.000 0	2.000 0	1.00000 0	1.000 0	1.0000	1.000 0	0.225 0	0.2250	0.225 0
10	2.000 00	2.0000	2.000 0	2.000 0	1.0000 01	1.000 0	1.0000	1.000 0	0.225 0	0.2250	0.225 0
15	2.000 00	2.0000	2.000 0	2.000 0	1.0000 01	1.000 0	1.0000	1.000 0	0.225 0	0.2250	0.225 0
20	2.000 00	2.0000	2.000 0	2.000 0	1.0000 00	1.000 0	1.0000	1.000 0	0.225 0	0.2250	0.225 0

Example 1(B): Double Solitons

Double solitons are the interaction of two different solitons in another way we can say the sum of two single solitons. To study double solitons, initial conditions [132] for $N = 2$ in equation (2.32) can be signified as:

$$W(x,0) = \sum_{e=1}^{2} W_e(x,0), \qquad Z(x,0) = \sum_{e=1}^{2} Z_e(x,0) \qquad (2.36)$$

where, $W_e(x,0) = 2\rho_e^2 sech^2(\mu_e)$, $\qquad Z_e(x,0) = \frac{1}{2\sqrt{\phi_e}}\text{sech}(\mu_e)$

with $\mu_e = \rho_e(x - y_e) + \frac{1}{2\log(\phi_e)}$, $\qquad \phi_e = \frac{-s}{8(4r+1)\rho_e^4}$, $e = 1,2$

The exact solution for double solitons for the system of CKdV equation is specified by

$$W(x,t) = \sum_{e=1}^{2} W_e(x,t), \qquad Z(x,t) = \sum_{e=1}^{2} Z_e(x,t) \qquad (2.37)$$

where,

$W_e(x,t) = 2\rho_e^2 sech^2(\mu_e)$, $\qquad Z_e(x,t) = \frac{1}{2\sqrt{\phi_e}}\text{sech}(\mu_e)$

with $\mu_e = \rho_e(x - y_e - \rho_e^2 t) + \frac{1}{2\log(\phi_e)}$, $\qquad \phi_e = \frac{-s}{8(4r+1)\rho_e^4}$, $e = 1,2$

To acquire the numerical solutions for double solitons of both solitons W and Z, parameters [137] are chosen as: $r = 0.5, s = -3, \rho_1 = 1.0, \rho_2 = 0.6, y_1 = 10, y_2 = 30, N = 421\ and\ \Delta t = 0.0001\ for\ \ 0 \leq x \leq 70$. For double solitons graphs are pictured for numerical and exact solution for above parameters using the three splines named HBS, UAH tension BS, UAT tension BS with differential quadrature method. In Figure 2.19, Figure 2.20, and Figure 2.21 precise and numerical findings are compared of the double solitons, and are represented using HBS, UAH tension BS, and UAT tension BS respectively. From the figures, it is very evident that numerical solutions attained with the current scheme are congruent with the exact solutions. Furthermore the conserved quantities for double solitons of CKdV I_1, I_2 and I_3 are computed and given in the Table 2.12 and also error norms in Table 2.10 for $W(x,t)$, and in Table 2.11 for $Z(x,t)$.

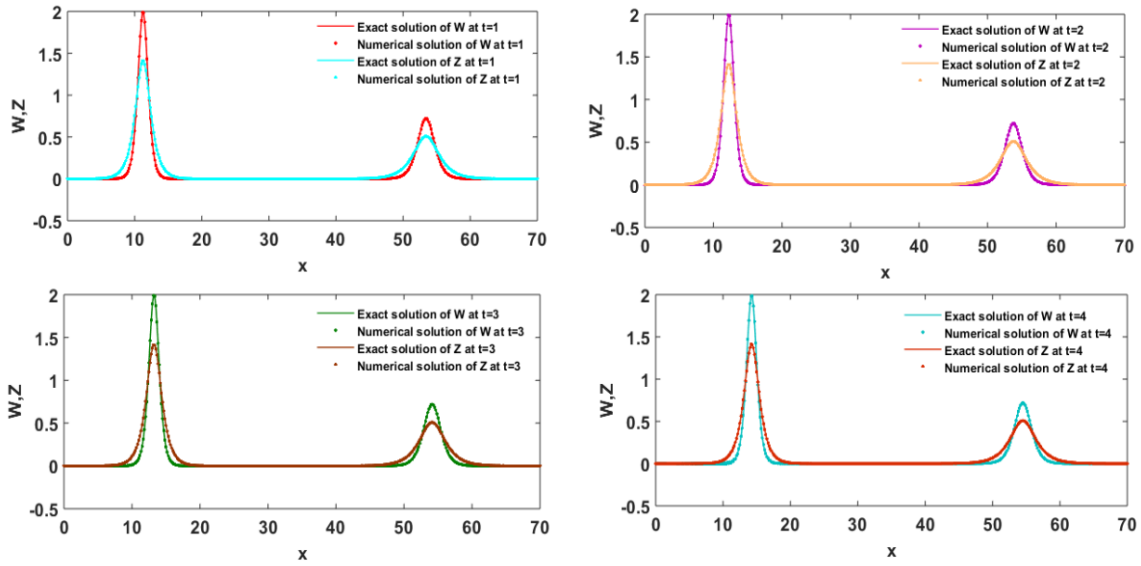

Figure 2.19: Graphical comparison of analytical and num. sol. for the simulation of double solitons at distinct values of time $t = 1, 2, 3, 4$ for the system of CKdV. With HBS.

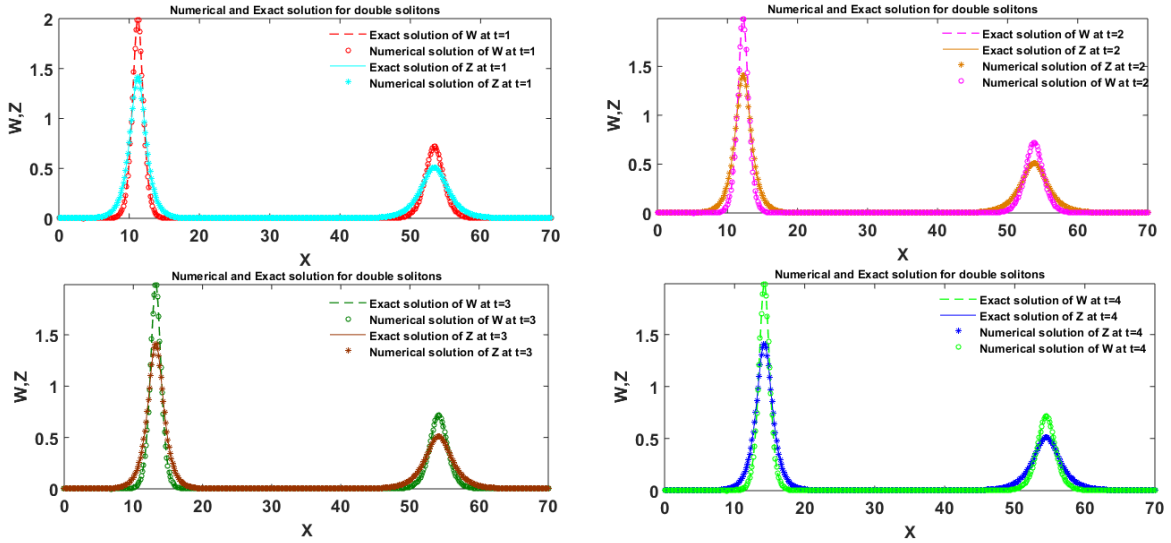

Figure 2.20: Graphical comparison of analytical and num. sol. for the simulation of double solitons at distinct values of time $t = 1, 2, 3, 4$ for the system of CKdV. With UAH tension B-spline.

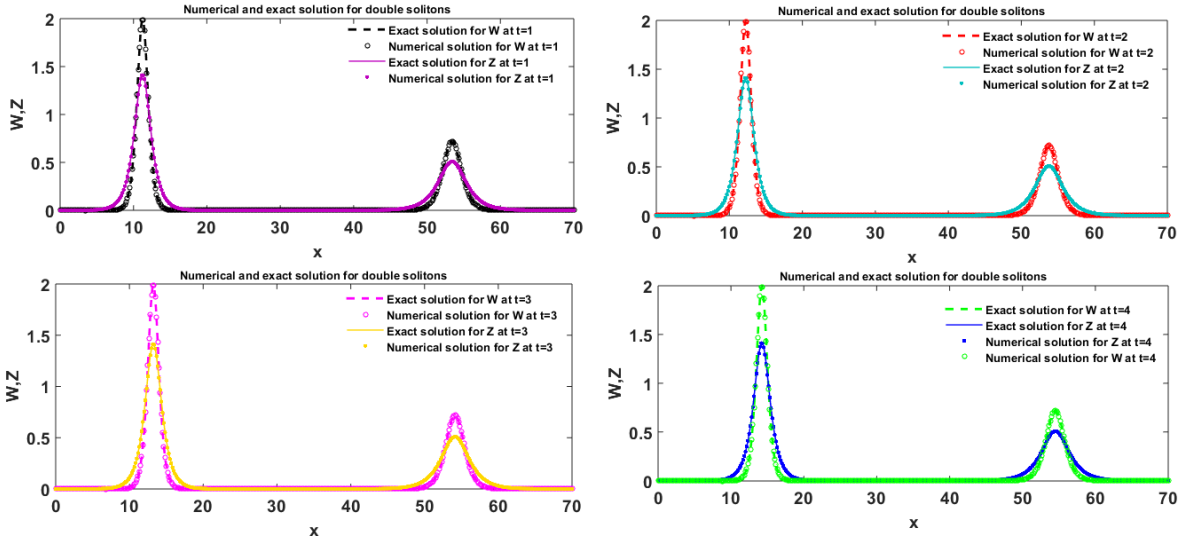

Figure 2.21: Graphical comparison of analytical and num. sol. for the simulation of double solitons at distinct values of time $t = 1, 2, 3, 4$ for the system of CKdV with UAT tension B-spline.

Table 2.10: The error norms for double solitons of soliton $W(x,t)$ at distinct time levels with $N = 421, \Delta t = 0.0001, r = 0.5, s = -3, \rho_1 = 1.0, \rho_2 = 0.6, y_1 = 10, y_2 = 30, [0,70]$.

t	$L_2(W)$			$L_\infty(W)$		
	HBS	UAH tension BS	UAT tension BS	HBS	UAH tension BS	UAT tension BS
0.2	5.5747E-05	2.0309E-05	2.0246E-05	5.1281E-05	1.5947E-05	1.5872E-05
0.4	8.7932E-05	4.0937E-05	4.0870E-05	7.4169E-05	2.5713E-05	2.5599E-05
0.6	1.2653E-04	6.2282E-05	6.2183E-05	9.1608E-05	3.2219E-05	3.2080E-05
0.8	1.6996E-04	8.5567E-05	8.5428E-05	1.0821E-04	4.2205E-05	4.2042E-05
1	2.2368E-04	1.1672E-04	1.1652E-04	1.5463E-04	5.1418E-05	5.1314E-05
2	8.1000E-04	4.8276E-04	4.8182E-04	4.6454E-04	1.9900E-04	1.9853E-04

Table 2.11: The error norms for double solitons of soliton $Z(x,t)$ at distinct time levels with $N = 421, \Delta t = 0.0001, r = 0.5, s = -3, \rho_1 = 1.0, \rho_2 = 0.6, y_1 = 10, y_2 = 30, [0,70]$.

t	$L_2(Z)$			$L_\infty(Z)$		
	HBS	UAH tension BS	UAT tension BS	HBS	UAH tension BS	UAT tension BS
0.2	2.6682E-05	2.3542E-05	2.3536E-05	1.6759E-05	1.6267E-05	1.6265E-05
0.4	3.4074E-05	2.6318E-05	2.6317E-05	2.0034E-05	1.5489E-05	1.5458E-05
0.6	4.3950E-05	2.8666E-05	2.8659E-05	3.5558E-05	1.3724E-05	1.3727E-05
0.8	5.8332E-05	3.2787E-05	3.2756E-05	4.7768E-05	1.5459E-05	1.5434E-05
1	7.1582E-05	3.6115E-05	3.6078E-05	6.2229E-05	1.9597E-05	1.9441E-05
2	1.6244E-04	8.8871E-05	8.8701E-05	1.2836E-04	5.6726E-05	5.6409E-05

Table 2.12: The invariants for double solitons at distinct time levels with $N = 421, \Delta t = 0.0001, r = 0.5, s = -3, \rho_1 = 1.0, \rho_2 = 0.6, y_1 = 10, y_2 = 30, [0,70]$.

t	I_1			I_2			I_3		
	HBS	UAH tension B-spline	UAT tension B-spline	HBS	UAH tension B-spline	UAT tension b-spline	HBS	UAH tension B-spline	UAT tension B-spline
0.2	6.4000	6.4000	6.4000	-3.2427	-3.2427	-3.2427	-2.5866	-2.5866	-2.5866
0.4	6.4000	6.4000	6.4000	-3.2427	-3.2427	-3.2427	-2.5866	-2.5866	-2.5866
0.6	6.4000	6.4000	6.4000	-3.2427	-3.2427	-3.2427	-2.5866	-2.5866	-2.5866
0.8	6.4000	6.4000	6.4000	-3.2427	-3.2427	-3.2427	-2.5866	-2.5866	-2.5866
1	6.4000	6.4000	6.4000	-3.2427	-3.2427	-3.2427	-2.5866	-2.5866	-2.5866
2	6.4001	6.3999	6.3999	-3.2426	-3.2427	-3.2427	-2.5866	-2.5866	-2.5866

Example 1(C): Three Soliton interaction

As the interaction of double solitons having same direction and different amplitude is illustrated in the above example. In the same way, the interaction of three solitons is presented. To study three soliton interaction, initial conditions is represented as addition of three well alienated single solitons formed by taking N=3 in equation (2.32) in the form given below:

$$W(x,0) = \sum_{e=1}^{3} W_e(x,0), \qquad Z(x,0) = \sum_{e=1}^{3} Z_e(x,0) \qquad (2.38)$$

where, $W_e(x,0) = 2\rho_e^2 sech^2(\mu_e)$, $\qquad Z_e(x,0) = \frac{1}{2\sqrt{\phi_e}}\text{sech}(\mu_e)$

with $\mu_e = \rho_e(x - y_e) + \frac{1}{2\log(\phi_e)}$, $\qquad \phi_e = \frac{-s}{8(4r+1)\rho_e^4}$, $e = 1,2,3$

For the numerical simulation of three soliton interaction, the set parameters are choosen as: $r = -0.5, s = 3, \rho_1 = 1.0, \rho_2 = 0.9, \rho_3 = 0.8, y_1 = 10, y_2 = 30, y_3 = 50, N = 421$ and $\Delta t = 0.0001$ for $0 \leq x \leq 80$. With the use of the above parameters and conditions for a sum of three single solitons, both solitons W and Z of the CKdV system are represented graphically in the three figures mentioned as Figure 2.22, Figure 2.23, and Figure 2.24 using HBS, UAH tension B-spline, and UAT tension B-psline respectively. Also, conserved quantities I_1, I_2 and I_3 for three soliton interaction of CKdV are also calculated and represented in Table 2.13. From the tabulated values of the invariants I_1, I_2 and I_3 it is very evident that they are almost constant.

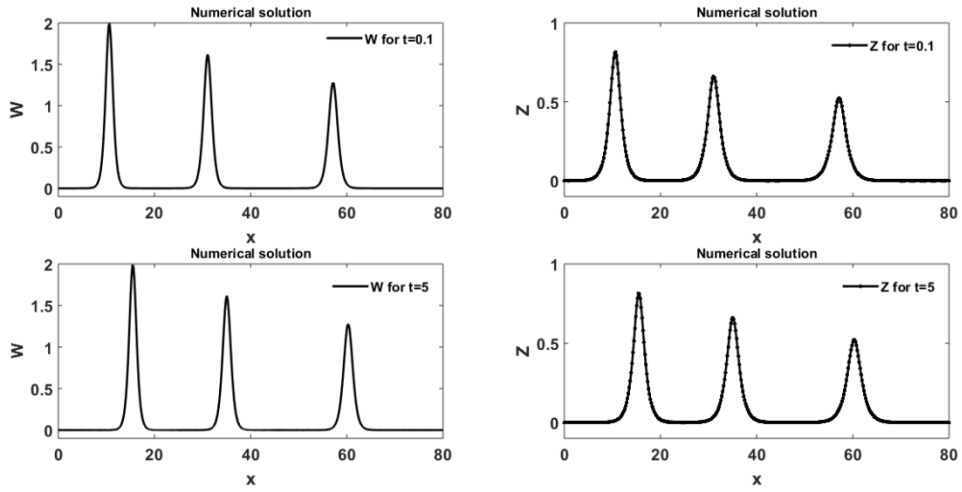

Figure 2.22: : Graphical illustration of numerical solution for the simulation of three solitons at distinct values of time $t = 0.1,5$ *with HBS.*

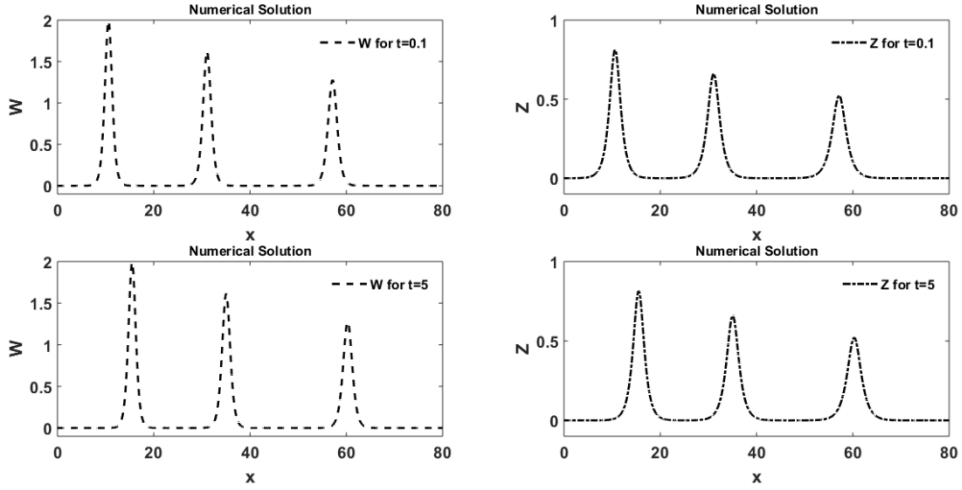

Figure 2.23: Graphical illustration of numerical solution for the simulation of three solitons at distinct values of time $t = 0.1,5$ *with UAH tension B-spline.*

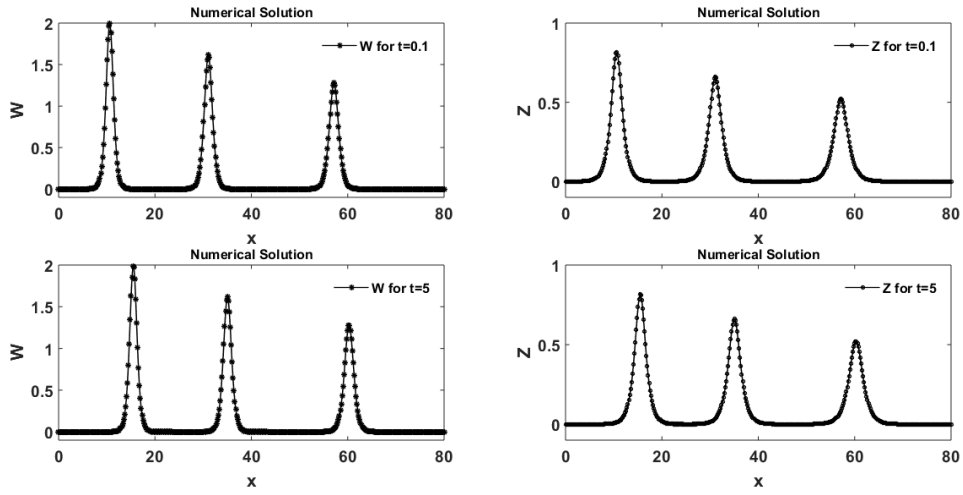

Figure 2.24: Graphical illustration of numerical solution for the simulation of three solitons at distinct values of time $t = 0.1,5$ *with UAT tension B-spline.*

Table 2.13: The conserved quantities for three soliton interaction for various time values with the parameters $r = -0.5, s = 3, grid\ points = 421, [0,80]$.

t	I_1			I_2			I_3		
	HBS	UAH tension BS	UAT tension BS	HBS	UAH tension BS	UAT tension BS	HBS	UAH tension BS	UAT tension BS
0.1	10.8000	10.8000	10.8000	17.9280	17.9280	17.9280	13.8109	13.8108	13.8108
1	10.8000	10.8000	10.8000	17.9280	17.9280	17.9280	13.8109	13.8108	13.8108
2	10.8000	10.8000	10.8000	17.9280	17.9280	17.9280	13.8109	13.8108	13.8108
3	10.8000	10.8000	10.8000	17.9280	17.9280	17.9280	13.8109	13.8108	13.8108
4	10.8001	10.8001	10.8001	17.9281	17.9280	17.9280	13.8110	13.8109	13.8109
5	10.8004	10.8001	10.8001	17.9285	17.9283	17.9283	13.8113	13.8110	13.8110

Example 2: Birth of Solitons

In this example CKdV equation will be considered with the initial condition taken as Gaussian initial condition [137] as:

$$W(x,0) = \exp(-0.01x^2)$$
$$Z(x,0) = \exp(-0.01x^2) \qquad (2.39)$$

and boundary conditions

$$W(x_L, t) = W(x_R, t) = 0$$
$$Z(x_L, t) = Z(x_R, t) = 0 \qquad (2.40)$$

with the parameters $r = 0.5, s = -3, -50 \leq x \leq 150, N = 421, \Delta t = 0.0001$

With the help of these conditions and parameters birth of solitons for the solitons, W and Z are represented graphically from Figure 2.25 to Figure 2.27 for distinct values of time ($t = 1\ to\ 5$). Using three B-splines. From the below graphical representation, it can be observed that for $t = 1$ that is with the initial value of time, we are getting a single wave for both the solitons W, Z at the same amplitude near about 1 having location $x = 0$ as well. And as we increase the time, there is increase in simulation also or we can conclude that there is an occurrence of new solitons with the increase of time. In other words, with the progress in time, there is an occurance of new solitons with different amplitudes which can lead to increase in number of soliton with the further increase in the value of time. Likewise conserved quantities I_1, I_2 and I_3 for birth of solitons of CKdV are also calculated and represented in Table 2.14. From the tabulated values of the invariants I_1, I_2 and I_3 it is very evident that they are almost constant.

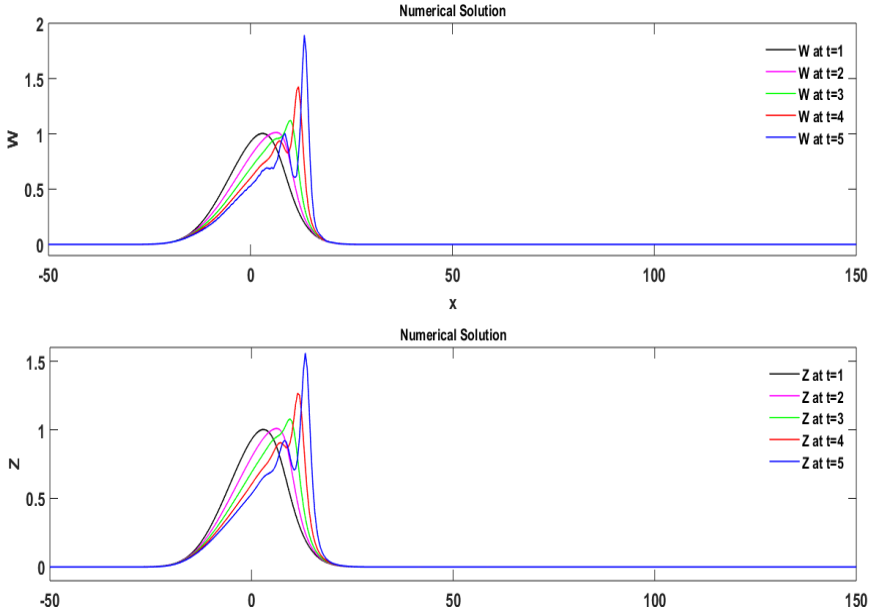

Figure 2.25: Simulation of birth of solitons for both the solitons W and Z of coupled KdV equations at t=1 to t=5 with HBS.

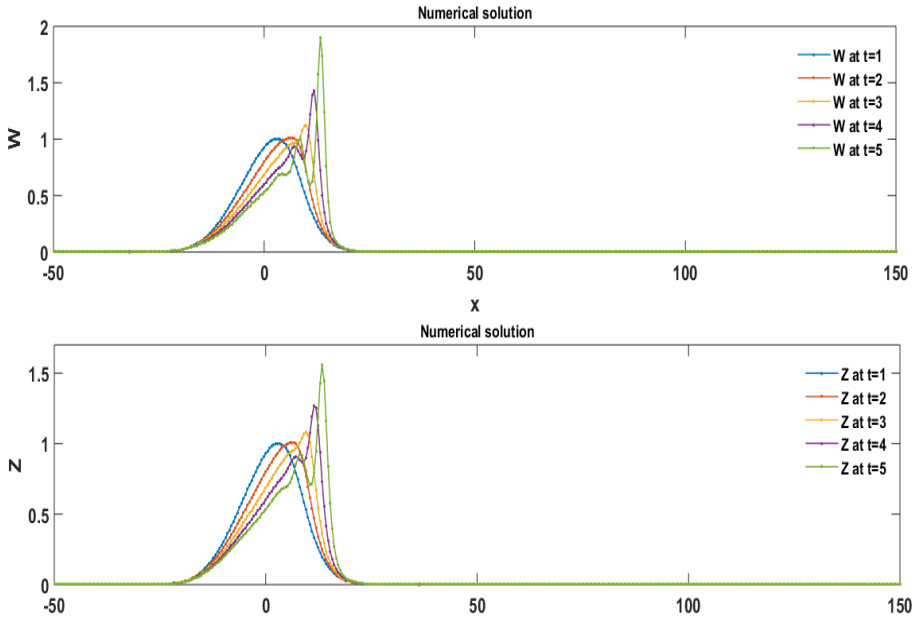

Figure 2.26: Simulation of birth of solitons for both the solitons W and Z of coupled KdV equations at t=1 to t=5 with UAH tension B-spline.

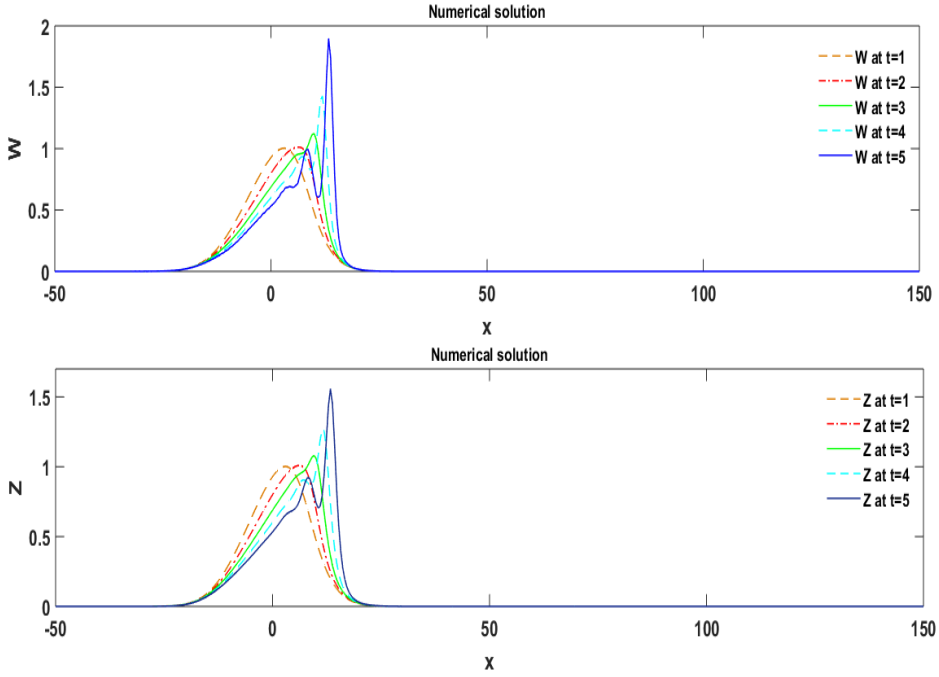

Figure 2.27: Simulation of birth of solitons for both the solitons W and Z of coupled KdV equations at t=1 to t=5 with UAT tension B-spline.

Table 2.14: The conserved quantities for the birth of solitons for different time levels with the parameters $r = 0.5, s = -3, \rho = 0.5, N = 421, \Delta t = 0.0001, [-50,\ 150]$.

t	I_1			I_2			I_3		
	HBS	UAH tension B-spline	UAT tension BS	HBS	UAH tension BS	UAT tension BS	HBS	UAH tension BS	UAT tension BS
1	17.7245	17.7245	17.7245	-12.5331	-12.5331	-12.5331	-15.0682	-15.0679	-15.0679
2	17.7245	17.7245	17.7245	-12.5332	-12.5331	-12.5331	-15.0683	-15.0679	-15.0679
3	17.7245	17.7245	17.7245	-12.5333	-12.5331	-12.5331	-15.0686	-15.0679	-15.0679
4	17.7245	17.7245	17.7245	-12.5337	-12.5332	-12.5332	-15.0693	-15.0680	-15.0680
5	17.7245	17.7246	17.7246	-12.5339	-12.5327	-12.5327	-15.0732	-15.0691	-15.0691

2.7. Stability of the Scheme

We used matrix stability method to show the stability of proposed method [15, 95, 96] After discretization of equations (2.2), they will be transformed into the system of ODEs equation (2.25) in case of HBS. So, after converting the CKdV equation into

ODEs (2.25) and assuming the nonlinear term as constant the system (2.2) can be inscribed as:

$$\frac{\partial}{\partial t}\begin{bmatrix} W \\ Z \end{bmatrix} = B'[W \quad Z]^T + f(W,Z) \tag{2.41}$$

where $B' = \begin{bmatrix} rd_{ij}^{(3)} & O \\ O & -d_{ij}^{(3)} \end{bmatrix}$ and $f(W,Z)$ is nonlinear term of the equation, O is the null matrix. In a similar manner, this formation can be made with UAH tension BS, and with UAT tension BS. For the constancy of the proposed technique, eigen values will be computed for different grid points as shown in the figure from Figure 2.28 - Figure 2.30 of the matrix B' with different splines. It is found that all the eigenvalues of B' are complex and lie within the stability region as explained in the section 1.9, so the proposed scheme is stable.

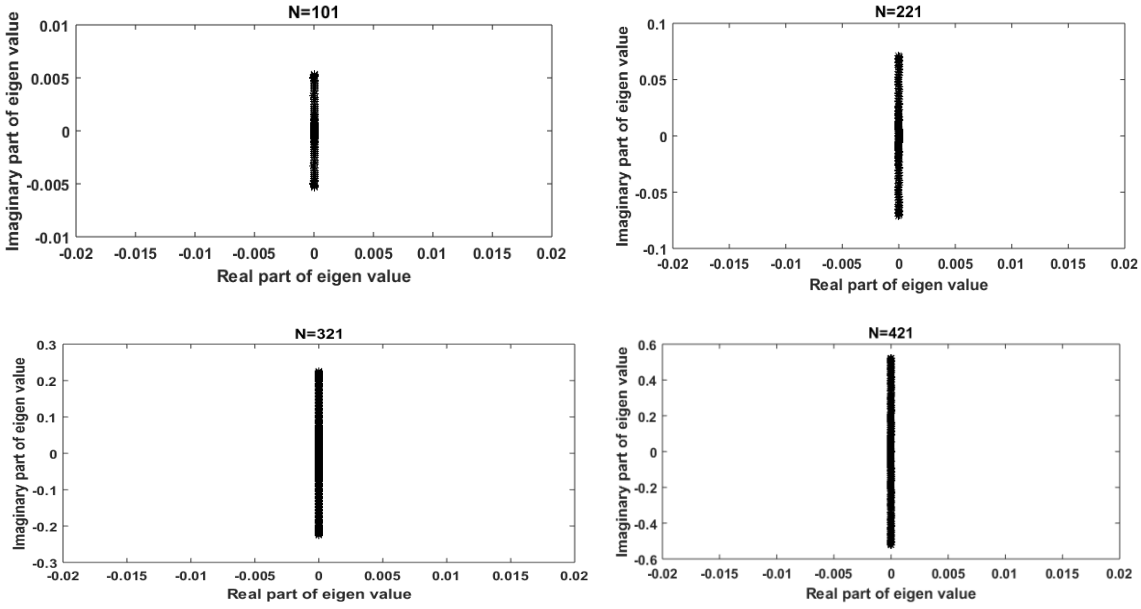

Figure 2.28: Plot of eigen values with HBS.

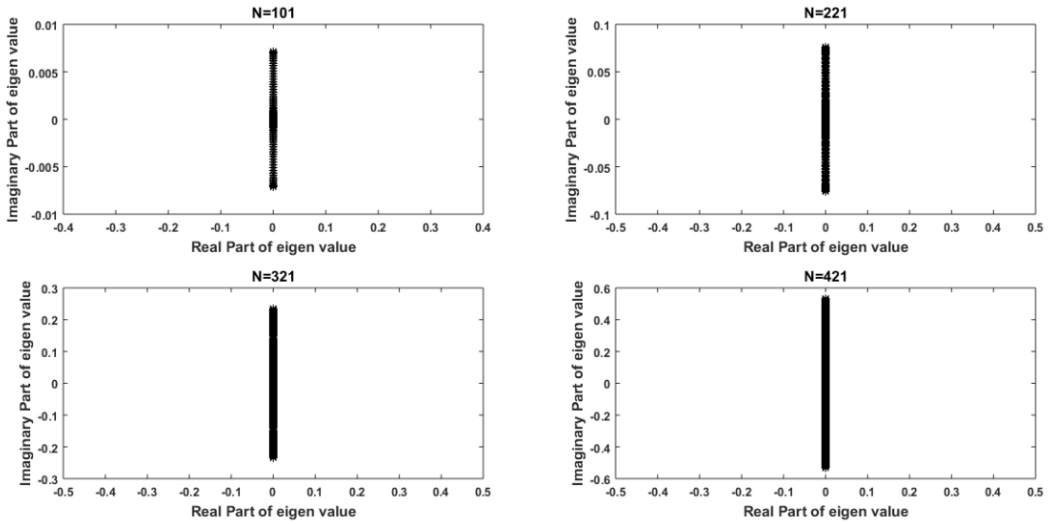

Figure 2.29: Plot of eigen values with UAH tension B-spline.

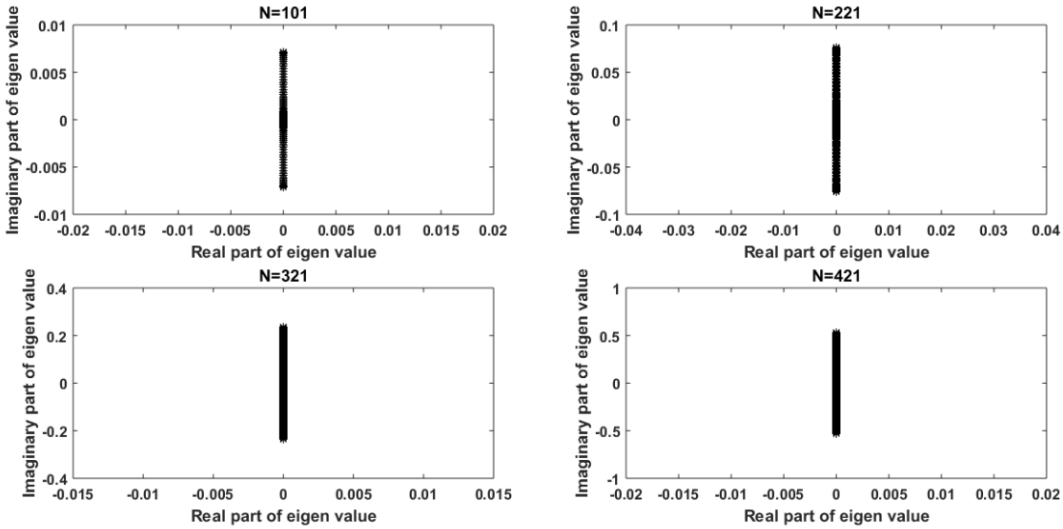

Figure 2.30: Plot of eigenvalues with UAT tension B-spline.

2.8. Summary

In this Chapter, numerical solution of the equation known as CKdV is provided, with the implementation of three B-splines of order six that is

(a) Quintic Hyperbolic B-spline with DQM

(b) Quintic Uniform Algebraic Hyperbolic tension B-spline with DQM

(c) Quintic Uniform Algebraic trigonometric tension B-spline with DQM

With the implementation of DQM, the system of equation got converted into ODEs, and then calculated by using the SSP-Runge-Kutta method. Also, by calculating the error norms L_2 and L_∞, and conserved quantities I_1, I_2, and I_3 the method's performance and accuracy are measured. It has been perceived that conserved quantities almost remain constant during the interaction and error norms are reasonably small. Attained results are signified in figures, tabular forms, and findings are compared with precise solution where applicable. In conclusion, the CKdV system of equations as well as a variety of PDEs may be elucidate using the current approach, which has been proven to be efficient and successful.

Chapter 3

Numerical Solution of Kuramoto-Sivashinsky Equation using quintic B-splines

In this Chapter, numerical solution of Kuramoto-Sivashinsky equation is obtained using above mentioned three quintic B-splines.

Research Objectives

1) Numerical solution of higher order differential equations will be obtained by using quintic Hyperbolic B-spline basis functions.
2) Numerical solution of third or fourth order differential equations will be obtained by using Uniform Algebraic trigonometric [UAT] tension B-spline as basis function.
3) To obtain the numerical solution, we will use differential quadrature method (DQM) along with quintic Uniform Algebraic Hyperbolic [UAH] tension B-spline.

3.1. Introduction

In a variety of science and engineering fields, the behaviour and effects of many phenomena are modelled by mathematical concepts, for which nonlinear partial differential equations are introduced, which play a huge role in describing such occurrences. In many cases, these PDEs are so complex, that their exact solutions are not available or it is impossible to evaluate exact solutions for these problems. So, in such cases, the evaluation of numerical solutions becomes important and profound to explain them. In the current study, the fourth-order nonlinear Kuramoto-Sivashinsky (KS) problem is solved numerically. In the 1970s, a nonlinear partial differential equation of order four was introduced known as Kuramoto-Sivashinsky (KS) equation, also known as the canonical evolution equation and explains the range of physical contexts. Also, it

designates about the turbulence in responsive system and model the disseminative uncertainties in a laminar flame front [156, 157].

It reveals several novel solutions in a variety of circumstances, such as phase turbulence, plasma instabilities, flame front propagation in reaction-diffusion (RD) system [158], chaos and travelling waves of permanent shape [159, 160], mathematical modelling of various physical occurrences specifying reaction-diffusion systems, in plasma insecure drift waves, flame front instability, fluid flow on a verticle plate, longwaves among two viscous fluids, and in some homogeneous medium uniform oscillating chemical reaction [161, 162], plasmas drift waves, RD system [163], modelling of solitary pulses [164]. The KS equation with a muddled behaviour describes solutions like travelling waves having unchangeable finite spatial domains shape [165]. The KS equation is precisely defined mathematically and also has solution with rich dynamical characteristics [166-168] that have drawn the interest of numerous researchers. For numerical purpose, the traveling wave solutions of KS equation were classified in three different classes namely solitary waves, oscillatory and regular shocks [169], which further studied by Yang [170]. The KS model is one of the PDEs that reveal spatiotemporally chaotic behavior [164], this equation has also been investigated as a model for systems classified by clear organized patterns with spontaneous appearance [171], where the thermodynamic equilibrium of an unstructured system is driven away [172, 173]. Moreover, this equation has a great application in plasma physics [174], ion sputtering [175], and describes many other phenomena [163, 157, 176].

The equation for the generalised Kuramoto-Sivashinsky (GKS) system is provided by:

$$P_t + PP_x + \alpha P_{xx} + \beta P_{xxx} + \gamma P_{xxxx} = 0, \tag{3.1}$$

with an initial condition as:

$$P(x, 0) = \omega(x), \tag{3.2}$$

plus having boundary conditions as:

$$\begin{cases} P(x_L,t) = \omega_1(t), & P(x_R,t) = \omega_2(t), \\ P_x(x_L,t) = \vartheta_1(t), & P_x(x_R,t) = \vartheta_2(t), \end{cases} \tag{3.3}$$

where, $x \in [x_L, x_R]$, and α, β, γ are nonzero constants, $\omega(x), \omega_i(t), \vartheta_i(t)$ are the specified functions. The linear terms of the above equation (3.1) with positive values of coefficients describes the equilibrium between long-wave instability and short-wave stability. The term P_{xx} describes the instability at large scales and P_{xxxx} is responsible for the small scale damping, and the term PP_x which is nonlinear in nature has the ability to filter energy by transferring it between tiny and large sizes [25, 177].

Because of the huge application of the KS equation, phenomenal research work has been done in this field, and the KS equation in one dimension is solved numerically by numerous researchers [46, 129, 168, 178-184]. Over the past years, a computational approach seeking an approximate solution to the KS equation has become quite popular. Initially, to solve the KS problem numerically, Manickam et al. [166] first utilised the cubic spline collocation approach, followed by the quintic B-spline collocation method [185], the septic B-spline collocation method [186], the quintic B-spline based DQM [25], and the quintic trigonometric based DQM [187]. In the past several years, different methods are used to achieve the numerical solution to the KS problem including the B-spline function based numerical method [46], the Galerkin method [129], the orthogonal spline of order four with method of collocation [166], the BDF approach [167], method which is mesh-free along with the function named radial basis function [184], the tanh-function method [188], the Chebyshev spectral collocation method [189], Lattice Boltzmann Method [190], three classes of the wave solution and the posedness of the KS problem were quantitatively explained [169, 191], kernel Hilbert space method [192], variational iteration method [193], linear feedback controls and approach [194], polynomial scaling functions [195], Taylor approximation approach [196], Crank-Nicolson approach with quintic B-spline [185], exponential cubic B-spline technique [197].

3.2. Impementation of method

Let's consider the mesh division system $a = x_1 < x_2 < \cdots < x_n = b$ of a closed interval [a,b], which is evenly transmitted and has a step size $h = x_{i+1} - x_i$. To calculate the numerical solution of KS system (3.1), we need to approximate the derivatives. So, for the further computation, we must compute the 1^{st}, 2^{nd}, 3^{rd}, 4^{th}order derivative of the function P. By DQM r^{th} order numerical derivative of P will be estimated as:

$$P_x^{(r)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(r)} P(x_j). \qquad (3.4)$$

By using the above equations, the approximate values of derivatives of 1st, 2nd, 3rd and 4th order can be given as:

$$P_x^{(1)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(1)} P(x_j)\,, P_x^{(2)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(2)} P(x_j),$$
$$P_x^{(3)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(3)} P(x_j)\,, P_x^{(4)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(4)} P(x_j), \qquad (3.5)$$

where, $d_{ij}^{(1)}$, $d_{ij}^{(2)}$, $d_{ij}^{(3)}$, and $d_{ij}^{(4)}$ are the weighting coefficients of respective order as the 'power' of these values . Consequently, system of equations as given below can be created with the applicability of the formulas from equation (3.5) to equation (3.1):

$$P_t = -\left\{ P \sum_{j=1}^{N} d_{ij}^{1} P(x_j) + \alpha \sum_{j=1}^{N} d_{ij}^{2} P(x_j) + \beta \sum_{j=1}^{N} d_{ij}^{3} P(x_j) + \gamma \sum_{j=1}^{N} d_{ij}^{4} P(x_j) \right\} \qquad (3.6)$$

The first order weighting coefficients can be calculated by using the quintic HBS, quintic UAH tension BS, and quintic UAT tension BS as explained in the section (1.7.1), (1.7.2), and (1.7.3) respectively and second or higher order weighting coefficient can be computed with the use of the formulae given in the equation (1.40). The SSP-RK43

regime [155] is then utilized to calculate the resulting ordinary differential equations further.

3.3. Numerical solution of KS employing three quintic B-splines

For expressing the versatility of proposed numerical algorithm, four different numerical examples are presented. For efficiency and accuracy check three error norms named Euclidian, maximum and global relative error (GRE) are calculated, formulated as:

$$L_2(P) = \sqrt{h\sum_{j=0}^{N}\left|P_j^{exact} - P_j^{numerical}\right|^2}$$

$$L_\infty(P) = \max_{0\le j\le N}\left|P_j^{exact} - P_j^{numerical}\right| \qquad (3.7)$$

$$GRE = \frac{\sum_{i=0}^{n}\left|P_j^{exact} - P_j^{numerical}\right|}{\sum_{i=0}^{n}\left|P_j^{numerical}\right|}$$

Additionally, the order of convergence is obtained for the purpose to assess the rate of convergence of the suggested technique and the formula that has been employed is given by:

$$Order\ of\ convergence\ (O) = \frac{log\left(\frac{E(N_2)}{E(N_1)}\right)}{log\left(\frac{N_1}{N_2}\right)} \qquad (3.8)$$

The numerical solutions N_1 and N_2 each has N grid points and 2N grid points, respectively, $E(N_i)$ symbolizes the error with N_i grid points.

Example 1:

Consider the KS equation below that is obtained by inserting the parameters [25, 129, 184, 187, 190, 198] $\alpha = 1, \beta = 4, \gamma = 1$ in equation (3.1):

$$P_t + PP_x + P_{xx} + 4P_{xxx} + P_{xxxx} = 0,\ x \in [a,b] = [-30,30],\ t = [1,2,3,4] \qquad (3.9)$$

having initial condition

$$P(x,0) = 9 + \mathbb{d} - 15\{tanh[\mathfrak{g}(x-\nu)] + tanh^2[\mathfrak{g}(x-\nu)] - tanh^3[\mathfrak{g}(x-\nu)]\} \qquad (3.10)$$

The analytical solution is given by

$$P(x,t) = 9 + \mathbb{d} - 15\begin{Bmatrix} tanh[\mathfrak{g}(x-\mathbb{d}t-\nu)] + \\ tanh^2[\mathfrak{g}(x-\mathbb{d}t-\nu)] - tanh^3[\mathfrak{g}(x-\mathbb{d}t-\nu)] \end{Bmatrix} \qquad (3.11)$$

where $\mathbb{d} = 6, \mathfrak{g} = \frac{1}{2}, \nu = -10, x\epsilon[-30,30]$.

For the numerical solution, the analytical solution (3.11) is used to obtain boundary conditions as:

$$\begin{aligned} P(a,t) &= 9 + \mathbb{d} - 15\begin{Bmatrix} tanh[\mathfrak{g}(a-\mathbb{d}t-\nu)] + \\ tanh^2[\mathfrak{g}(a-\mathbb{d}t-\nu)] - tanh^3[\mathfrak{g}(a-\mathbb{d}t-\nu)] \end{Bmatrix} \\ P(b,t) &= 9 + \mathbb{d} - 15\begin{Bmatrix} tanh[\mathfrak{g}(b-\mathbb{d}t-\nu)] + \\ tanh^2[\mathfrak{g}(b-\mathbb{d}t-\nu)] - tanh^3[\mathfrak{g}(b-\mathbb{d}t-\nu)] \end{Bmatrix} \end{aligned} \qquad (3.12)$$

In example 1, equation (3.1) is approximated numerically by assigning the values to the parameters as $\alpha = 1, \beta = 4, \gamma = 1$, and $a = -30, b = 30$ i.e. $x\epsilon[-30,30]$ at different values of time, $t = 1,2,3,4$. Figures are used to display the outcomes that were discovered from Figure 3.1 to Figure 3.3 for the equation (3.9) with the implementation of three splines. The figures represent the solution numerically comparing with the exact solution for equation (3.9) and are presented graphically both in 2D and 3D form. From the graphical representation of approximate solution, it is very evident that acquired numerical findings are well-aligned with the exact solution. Global Relative Error has been shown for a range of node points and comparison is made with [25, 190, 198] at time levels $t = 1,2,3,4$ with $\Delta t = 0.0001$ in Table 3.1. In Table 3.2, norms L_2 and L_∞ are calculated with grid points $N = 401$ and $\Delta t = 0.0001$, along with that comparison is made with [187]. And in Table 3.3 computaional time is given.

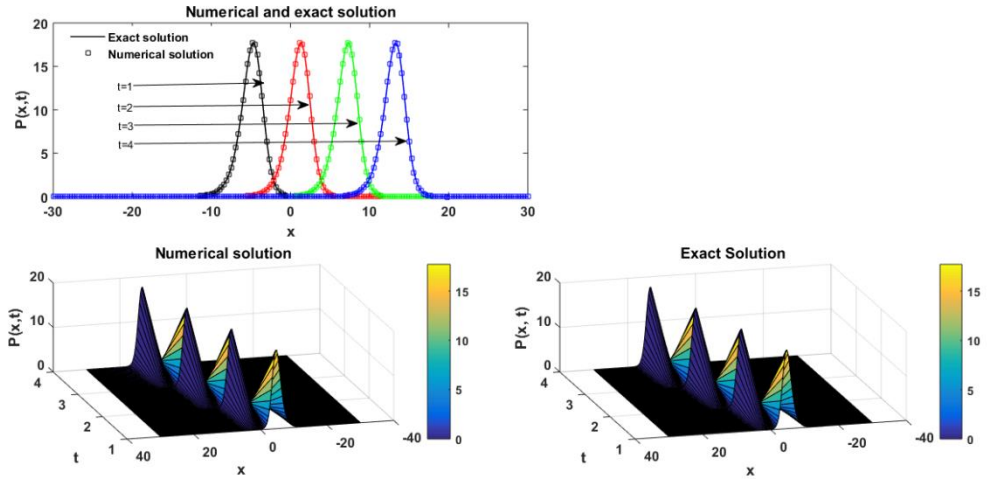

Figure 3.1: Comparability of exact and num. sol. for N=201,$t\epsilon[1,4]$, $\Delta t = 0.0001$ of example 1 with HBS spline.

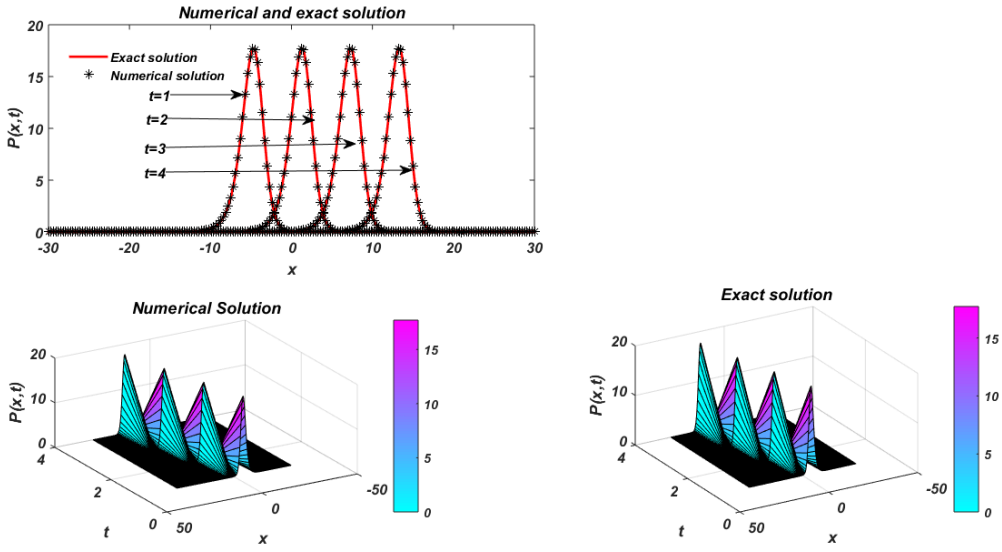

Figure 3.2: Comparability of exact and num. sol. for N=201,$t\epsilon[1,4]$, $\Delta t = 0.0001$ of example 1 with UAH tension B-spline.

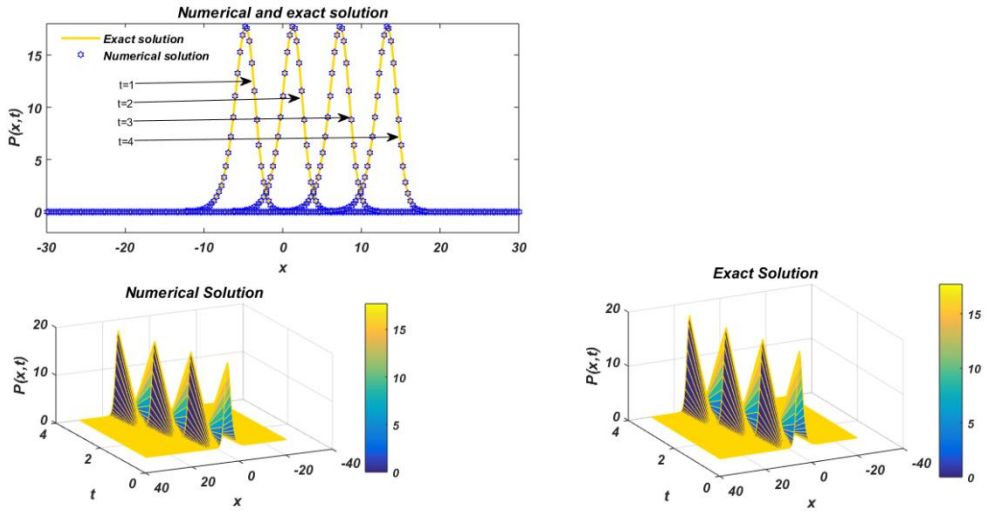

Figure 3.3: Comparability of exact and num. sol. for N=201,$t\epsilon[1,4]$*,* $\Delta t = 0.0001$ *of example 1 with UAT tension B-spline.*

Table 3.1: Comparability of GRE norm in the num. sol. of Example 1 at various time values with $\Delta t = 0.0001$.

t	HBS		UAH tension B-spline		UAT tension B-spline		Iqbal et al. [198]	Mittal & Dahiya [25]	Lai & Ma [190]
	N=201	N=401	N=201	N=401	N=201	N=401	N=150	N=200	N=600
1	2.1652E-04	3.5346E-06	6.3245E-06	1.1086E-07	6.1706E-06	1.0862E-07	5.1400E-04	5.0414E-04	2.5945E-02
2	2.7129E-04	4.5140E-06	9.6672E-06	5.4581E-07	9.4902E-06	5.4402E-07	1.3900E-03	1.0017E-03	2.7959E-02
3	5.9746E-04	1.5727E-05	2.7462E-05	6.9284E-06	2.7152E-05	6.9170E-06	3.0200E-03	2.3313E-03	2.6701E-02
4	8.0000E-03	1.7965E-04	2.6545E-04	9.3505E-05	2.6341e-04	9.3335E-05	5.0300E-03	4.0021E-03	3.5172E-02

Table 3.2: Evaluation of L_2 *and* L_∞ *Error norms for example 1 with* $N = 401, \Delta t = 0.0001$.

t	$L_2(P)$				$L_\infty(P)$			
	Arora & Joshi [187]	HBS	UAH tension B-spline	UAT tension B-spline	Arora & Joshi [187]	HBS	UAH tension B-spline	UAT tension B-spline
1	3.1300E-05	5.2927E-05	1.7468E-06	1.7099E-06	2.6200E-05	2.9878E-05	1.3698E-06	1.3513E-06
2	3.8200E-05	6.8336E-05	6.2587E-06	6.2411E-06	3.7400E-05	5.5721E-05	3.9348E-06	3.8986E-06

3	1.1900E-04	1.8697E-04	8.0912E-05	8.0780E-05	7.5100E-05	1.3027E-04	2.6002E-05	2.5903E-05
4	1.5400E-03	2.300E-03	1.2000E-03	1.2000E-03	4.6900E-04	7.9316E-04	3.4890E-04	3.4819E-04

Table 3.3: Computational time

t	**HBS**	**UAH**	**UAT**
	CPU	CPU	CPU
1	3.763978 seconds	2.975470 seconds	3.206847seconds
2	4.97164 seconds	4.111641 seconds	4.545863 seconds
3	6.579897 seconds	6.533581 seconds	6.841112 seconds
4	8.065664 seconds	7.528215 seconds	7.666945seconds

Example 2:

Consider the KS equation below that is obtained by implanting the parameters [187] $\alpha = 1, \beta = -4, \gamma = 1$ in equation (3.1):

$$P_t + PP_x + P_{xx} - 4P_{xxx} + P_{xxxx} = 0, x \in [a, b][-30, 30],\ t = [1, 2, 3, 4] \qquad (3.13)$$

with the initial condition as:

$$P(x, 0) = -9 + \mathbb{d} - 15\{tanh[\mathfrak{g}(x - v)] - tanh^2[\mathfrak{g}(x - v)] - tanh^3[\mathfrak{g}(x - v)]\} \qquad (3.14)$$

The analytical solution is given by

$$P(x, t) = -9 + \mathbb{d} - 15\begin{Bmatrix} tanh[\mathfrak{g}(x - \mathbb{d}t - v)] - \\ tanh^2[\mathfrak{g}(x - \mathbb{d}t - v)] - tanh^3[\mathfrak{g}(x - \mathbb{d}t - v)] \end{Bmatrix} \qquad (3.15)$$

where $\mathbb{d} = -6, \mathfrak{g} = \frac{1}{2}, v = -10, x\epsilon[-30, 30]$.

With the help of an analytical solution (3.15) boundary conditions for the equation (3.13) can be extracted as:

$$P(a,t) = -9 + \mathbb{d} - 15\left\{\begin{matrix} tanh[\mathfrak{g}(a - \mathbb{d}t - \nu)] - \\ tanh^2[\mathfrak{g}(a - \mathbb{d}t - \nu)] - tanh^3[\mathfrak{g}(a - \mathbb{d}t - \nu)] \end{matrix}\right\}$$
$$P(b,t) = -9 + \mathbb{d} - 15\left\{\begin{matrix} tanh[\mathfrak{g}(b - \mathbb{d}t - \nu)] - \\ tanh^2[\mathfrak{g}(b - \mathbb{d}t - \nu)] - tanh^3[\mathfrak{g}(b - \mathbb{d}t - \nu)] \end{matrix}\right\} \quad (3.16)$$

In example 2, equation (3.1) is estimated numerically by assigning the values to the parameters as $\alpha = 1, \beta = -4, \gamma = 1$, and $a = -30, b = 30$ i.e. $x\epsilon[-30,30]$ at different values of time, $t = 0.5, 1, 1.5, 2$. Figures are used to display the outcomes that were discovered from Figure 3.4 to Figure 3.6 for the equation (3.13) with the implementation of three splines. The figures represent the solution numerically comparing with the exact solution for equation (3.13) and are presented graphically both in 2D and 3D form. From the graphical representation of approximate solution, it is very evident that acquired numerical findings are well-aligned with the exact solution. Global Relative Error has been shown for a range of node points in Table 3.4, and comparison is made with [187] at time levels $t = 0.5, 1, 1.5, 2$ with $\Delta t = 0.0001$. In Table 3.5, error norms L_2 and L_∞ are calculated for grid points $N = 401$ and $\Delta t = 0.0001$, along with that comparison is made with [187]. In Table 3.6 computational time is given.

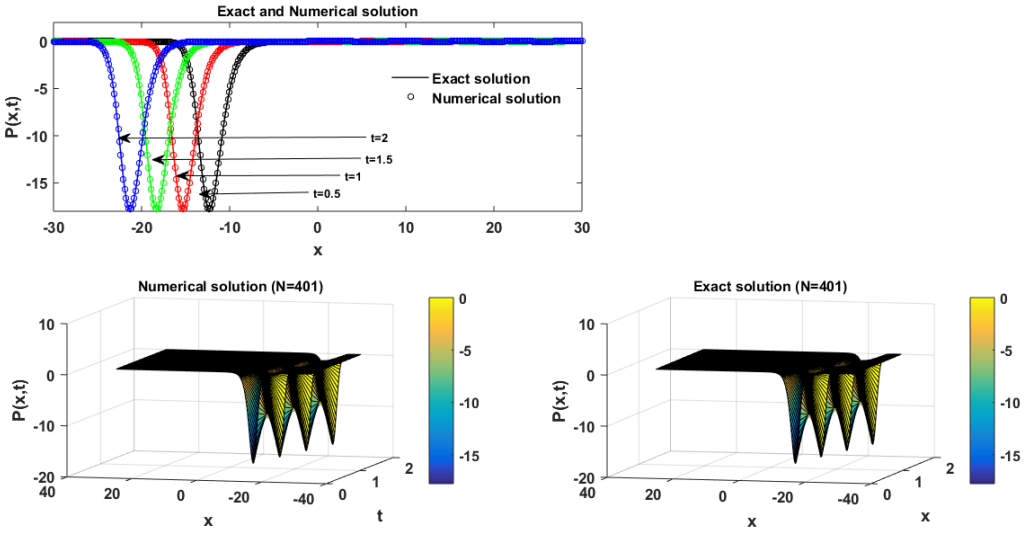

Figure 3.4: Comparison of exact and numerical solutions of example 2 for N=401,$t\epsilon[0.5, 2]$, $\Delta t = 0.0001$ with HBS spline.

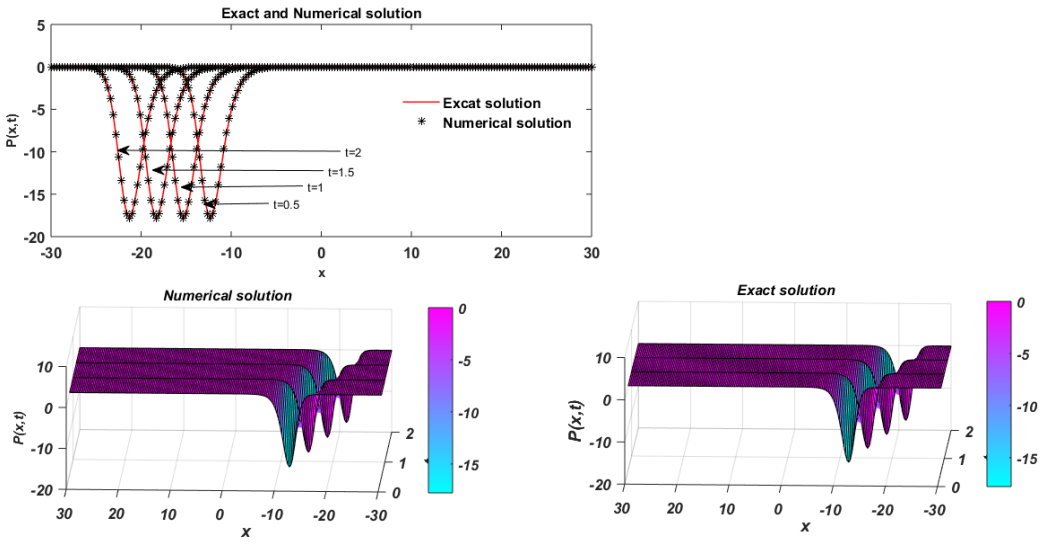

Figure 3.5: Comparability of exact and num. sol. for N=201,$t\epsilon[0.5,2]$, $\Delta t = 0.0001$ of example 2 with UAH tension B-spline.

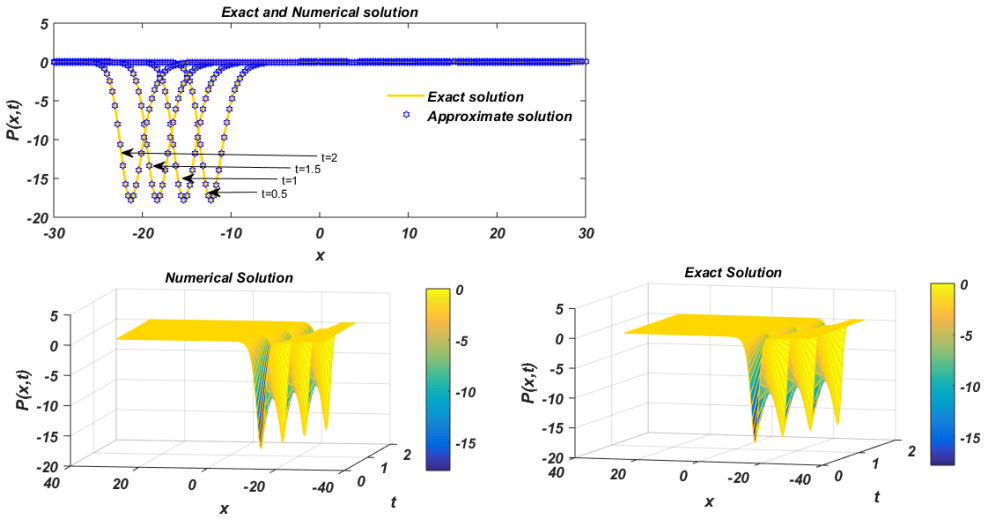

Figure 3.6: Comparison of exact and numerical solutions of example 2 for N=201,$t\epsilon[0.5,2]$, $\Delta t = 0.0001$ with UAT tension B-spline.

Table 3.4: Comparison of GRE norm in the num. sol. of Example 2 at various time values with $\Delta t = 0.0001$.

t	HBS		UAH tension BS		UAT tension BS		Arora & Joshi [187]
	N=201	N=401	N=201	N=401	N=201	N=401	N=201
0.5	1.7572E-04	2.8573E-06	5.5927E-06	7.8917E-08	5.4556E-06	7.6904E-08	1.5800E-06

1	2.2177E-04	3.6069E-06	6.5605E-06	9.1809E-08	6.4001E-06	8.9415E-08	1.7500E-06
1.5	2.5451E-04	4.1303E-06	7.5900E-06	1.0623E-07	7.4186E-06	1.0374E-07	1.9100E-06
2	2.9297E-04	4.8067E-06	9.8116E-06	1.8741E-07	9.6213E-06	1.8466E-07	2.0900E-06

Table 3.5: Evaluation of L_2 and L_∞ Error norms for example 2 with $N = 401, \Delta t = 0.0001$.

t	$L_2(P)$				$L_\infty(P)$			
	Arora & Joshi [187]	HBS	UAH tension BS	UAT tension BS	Arora & Joshi [187]	HBS	UAH tension BS	UAT tension BS
0.5	3.0300E-05	5.0139E-05	1.6976E-06	1.6595E-06	2.3600E-05	3.1509E-05	1.4919E-06	1.4604E-06
1	3.1300E-05	5.2972E-05	1.7024E-06	1.6646E-06	2.4100E-05	2.9614E-05	1.2958E-06	1.2667E-06
1.5	3.3600E-05	5.7130E-05	1.8099E-06	1.7748E-06	2.8800E-05	4.0067E-05	1.7108E-06	1.6875E-06
2	3.7600E-05	6.8753E-05	3.1344E-06	3.1072E-06	3.4000E-05	5.3711E-05	2.6903E-06	2.6900E-06

Table 3.6:Computational time

t	**HBS**	**UAH**	**UAT**
	CPU	CPU	CPU
0.5	2.343775 seconds	2.047269 seconds	2.044348 seconds
1	2.986916 seconds	3.053471 seconds	3.039848 seconds
1.5	3.763580 seconds	3.747244 seconds	3.720020 seconds
2	5.494263 seconds	4.625891 seconds	4.326531 seconds

Example 3:

Consider KS equation attained by implanting $\alpha = 1, \beta = 0, \gamma = 1$ in (3.1):

$$P_t + PP_x + P_{xx} + P_{xxxx} = 0 \qquad (3.17)$$

Because it has an advantage in displaying chaotic behaviour over a confined area with periodic boundary conditions, this particular KS equation has fascinating behaviour that attracts a lot of interest. Taking a Gaussian condition into account now [25, 187] as:

$$P(x,0) = exp(-x^2),\ x \in [-30,30] \qquad (3.18)$$

and with the boundary conditions as:

$$P(-30,t) = 0,\ and\ P(30,t) = 0 \qquad (3.19)$$

In example 3, the numerical solution of eq. (3.17) is approximately represented by setting the parameters of eq. (3.1) to the values $\alpha = 1, \beta = 0, \gamma = 1$, and $a = -30, b = 30$, i.e. $x \in [-30,30]$, $N = 201$ for various time values, $t = 1,5,10,20$, and time-discretization as $\Delta t = 0.001$. With the aid of Figure 3.7 to Figure 3.10 the solutions are displayed for the equation (3.17). It is evident from the approximate solution's graphical representation that the numerical findings obtained are well-aligned with results found in the previous work [25, 185, 187]. In Table 3.7, and Table 3.8 the convergence rate of example 3 with the implementation of the present method is portrayed for $t = 1,3,5,10$.

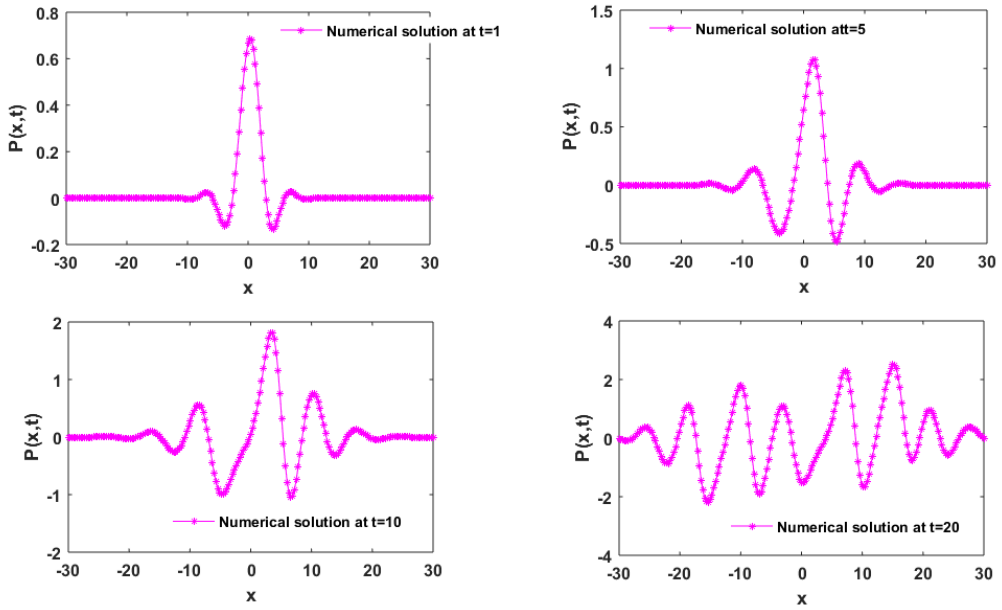

Figure 3.7: KS equation having Chaotic behaviour with "Gaussian condition" of example 3 for "N=201, $\Delta t = 0.001$ *at t=1, 5, 10, 20" with HBS.*

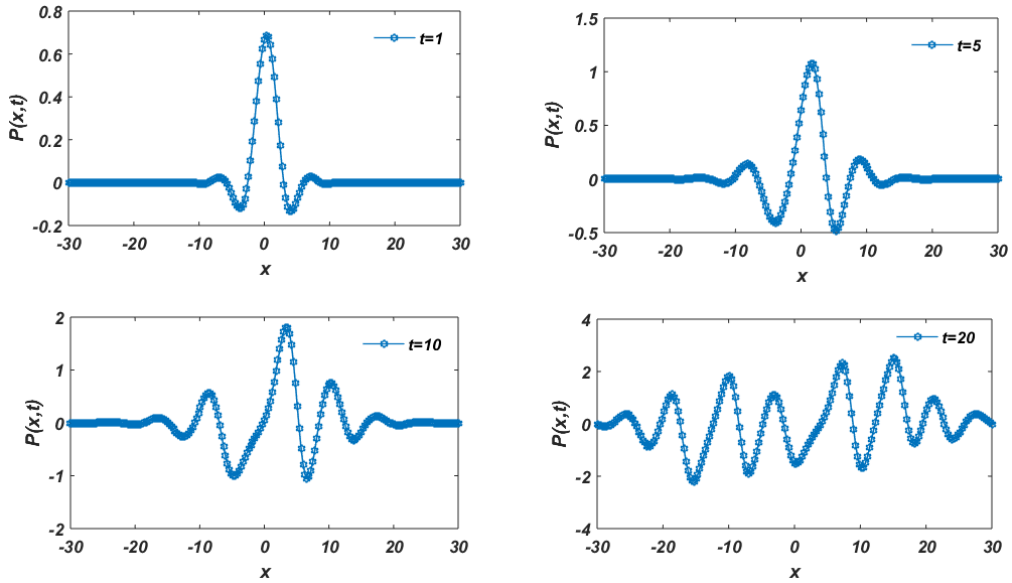

Figure 3.8: KS equation having Chaotic behaviour with "Gaussian condition" of example 3 for "N=201, $\Delta t = 0.001$ *at t=1, 5, 10, 20" with UAH B-spline.*

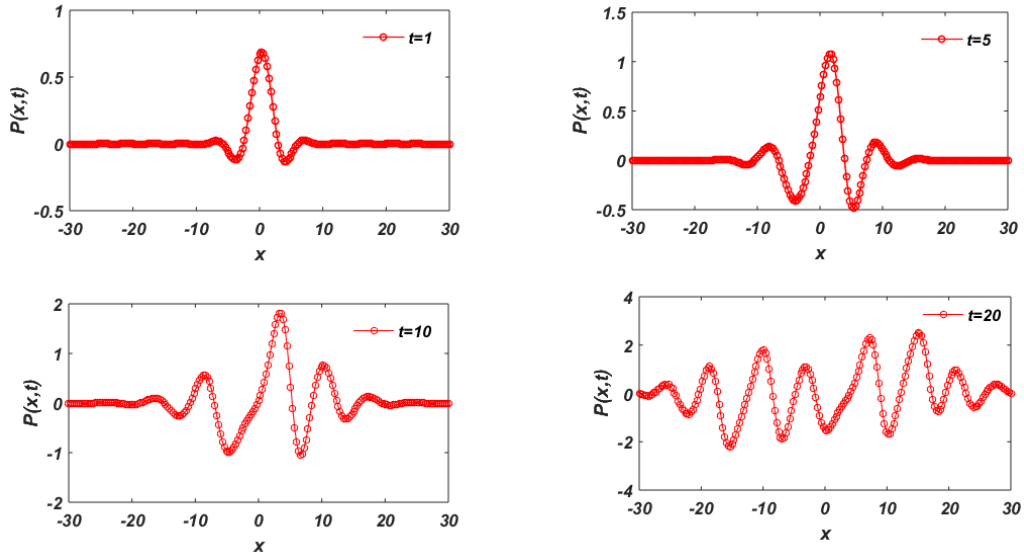

Figure 3.9: KS equation having Chaotic behaviour with "Gaussian condition" of example 3 for "N=201, $\Delta t = 0.001$ *at t=1, 5, 10, 20" with UAT B-spline.*

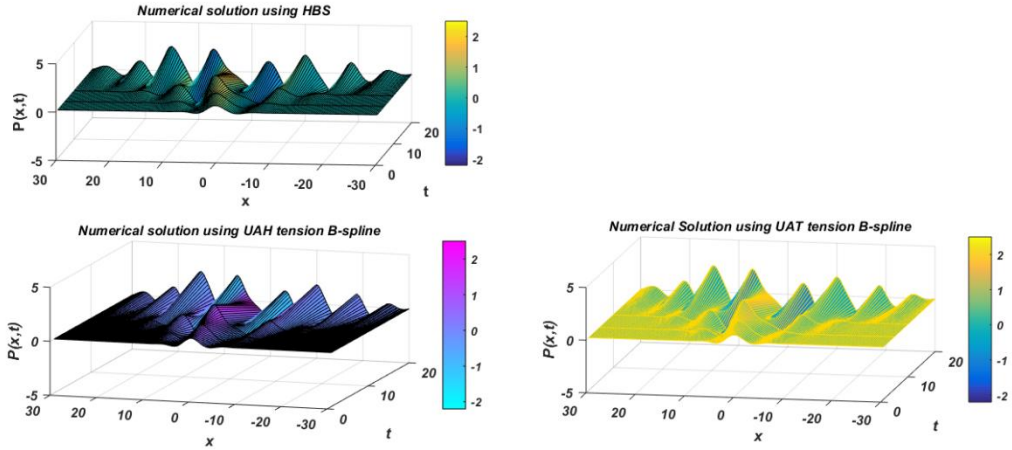

Figure 3.10: The 3d view of the erratic behavior of the KS equation of example 3 for N=201, $\Delta t = 0.001$ *at t=1, 5, 10, 20*

Table 3.7: Calculation of order of convergence (O) of $P(x,t)$ *in example 3 at different values of t, and for* $\Delta t = 0.001$*.with HBS.*

t	N	L_2	O	L_∞	O	CPU(s)
0.5	20	4.07E-03	-	7.33E-02	-	0.109434
	40	1.27E-02	1.642968	3.79E-01	2.371256	0.113129
1	20	8.16E-03	-	1.43E-01	-	0.128449
	40	1.57E-02	0.939795	4.76E-01	1.730729	0.136115

1.5	20	1.22E-02	-	2.05E-01	-	0.139052
	40	1.78E-02	0.554219	5.44E-01	1.404565	0.156126
2	20	1.58E-02	-	2.52E-01	-	0.161702
	40	1.96E-02	0.303673	5.91E-01	1.231399	0.192165
2.5	20	1.91E-02	-	2.71E-01	-	0.179398
	40	2.08E-02	0.124938	6.12E-02	1.173179	0.209498
3	20	2.19E-02	-	2.77E-01	-	0.190595
	40	2.17E-02	0.01224	5.95E-01	1.104771	0.227749

Table 3.8: Calculation of order of convergence (O) of $P(x,t)$ in example 3 at different values of t, and for $\Delta t = 0.001$.

t	**N**	**UAH tension B-spline**				**UAT tension B-spline**			
		L_2	O	L_∞	O	L_2	O	L_∞	O
0.5	20	9.70E-03	-	1.63E-01	-	9.86E-03	-	1.66E-01	-
	40	8.51E-03	0.18942	2.21E-01	0.437736	8.44E-03	0.22312	2.19E-01	0.393545
1	20	1.69E-02	-	2.90E-01	-	1.71E-02	-	2.95E-01	-
	40	7.98E-03	1.07844	1.93E-01	0.59246	7.92E-03	1.10871	1.90E-01	0.6370
1.5	20	2.36E-02	-	4.10E-01	-	2.39E-02	-	4.16E-01	-
	40	7.56E-03	1.64387	1.65E-01	1.31581	7.51E-03	1.67162	1.62E-01	1.35863
2	20	3.15E-02	-	5.41E-01	-	3.19E-02	-	5.49E-01	-
	40	7.24E-03	2.11952	1.41E-01	1.9451	7.20E-03	2.14664	1.39E-01	1.98535
2.5	20	4.17E-02	-	7.01E-01	-	4.23E-02	-	7.11E-01	-
	40	7.02E-03	2.56981	1.19E-01	2.55374	6.99E-03	2.59896	1.18E-01	2.59097
3	20	5.59E-02	-	9.11E-01	-	5.70E-02	-	9.27E-01	-
	40	6.96E-03	3.00762	1.17E-01	2.96413	6.94E-03	3.03855	1.16E-01	2.99289

Example 4:

Consider the following KS equation obtained by substituting $\alpha = 1, \beta = 0$ with different values of γ as $\gamma = \frac{0.4}{\pi^2}, \frac{0.2}{\pi^2}$.

$$P_t + PP_x + P_{xx} + \gamma P_{xxxx} = 0 \qquad (3.20)$$

and initial and boundary [185, 25] conditions as:

$$P(x,0) = -\sin(\pi x), x \in [-1,1] \quad (3.21)$$

$$P_{xx}(-1,t) = 0, z_{xx}(1,t) = 0 \ and \quad (3.22)$$

$$P(-1,t) = 0, P(1,t) = 0 \quad (3.23)$$

To perceive the approximate solution, values of parameters are taken as $\Delta t = 0.00001, N(number\ of\ grid\ points) = 41, \gamma = \frac{0.4}{\pi^2}, \frac{0.2}{\pi^2}$.

In example 4, equation (3.1) is estimated numerically by setting the parameters values to $\alpha = 1, \beta = 0$ with varying values of 'γ', and for $a = -1,\ b = 1$ that is $x \in [-1,1]$ at varying $t \in [0.1,2]$ and time-discretization as $\Delta t = 0.00001$. In Figure 3.11, the numerical output of equation (3.20) is displayed for $\gamma = 0.2/\pi^2$ and in Figure 3.12 for $\gamma = 0.4/\pi^2$ with $\Delta t = 0.00001,\ N = 41,$ at various values of time $t = 0.1, 0.2, 0.3, 0.4, 0.5, 0.7$ with the implementation of the splines named HBS, UAH tension BS and quintic UAT tension BS. It is extremely obvious from the graphical depiction that the obtained numerical findings are well-aligned with the outcomes reported in the previous work [25, 185]. From Table 3.9 to Table 3.12, the convergence rate of example 4 with the implementation of the present method is portrayed for $t = 0.5, 1, 1.5, 2$ with $\gamma = 0.2/\pi^2$, and $\gamma = 0.4/\pi^2$.

Table 3.9: Calculation of order of convergence with L_2 error $P(x,t)$ in example 4 at different values of t with $\gamma = 0.2/\pi^2$.

		HBS		UAH tension BS		UAT tension BS	
t	N	L_2	Order of Convergence	L_2	Order of Convergence	L_2	Order of Convergence
0.5	20	4.02E-01	-	3.91E-01	-	3.91E-01	-
	40	1.06E-01	1.92749	1.05E-01	1.90013	1.05E-01	1.90008
1	20	6.80E-01	-	6.65E-01	-	6.65E-01	-
	40	1.74E-01	1.96657	1.72E-01	1.94966	1.72E-01	1.94965
1.5	20	6.84E-01	-	6.70E-01	-	6.70E-01	-
	40	1.79E-01	1.93554	1.77E-01	1.91977	1.77E-01	1.91977
2	20	6.84E-01	-	6.74E-01	-	6.74E-01	-
	40	2.96E-01	1.20999	1.97E-01	1.77594	2.02E-01	1.73675

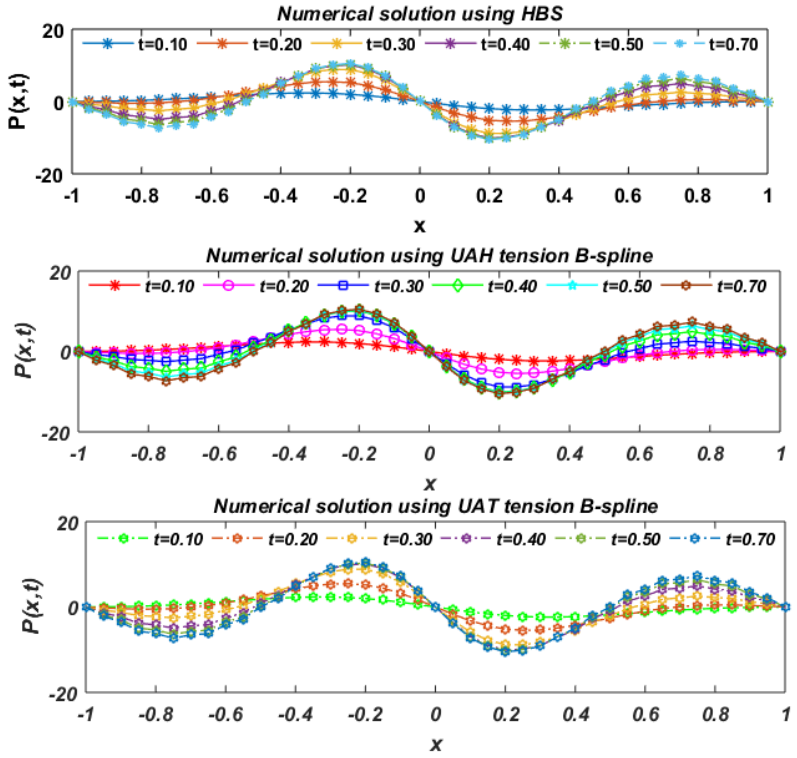

Figure 3.11: The time-dependent profile of $P(x,t)$ of KS equation versus x for $\gamma = 0.2/\pi^2$

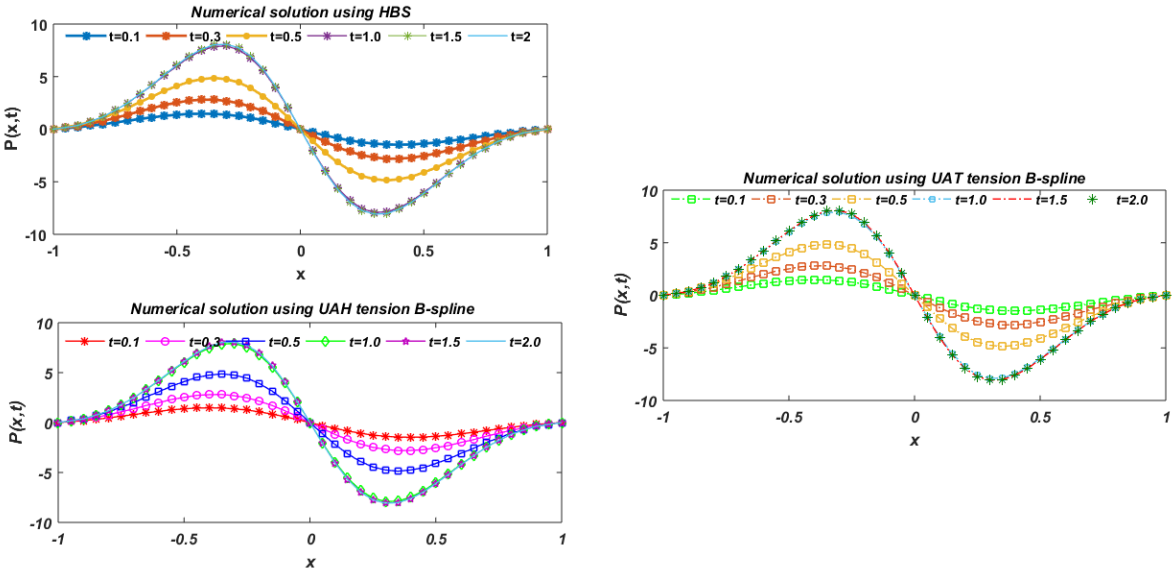

Figure 3.12: The time-dependent profile of $P(x,t)$ of KS equation versus x for $\gamma = \frac{0.4}{\pi^2}$.

Table 3.10: Calculation of order of convergence with L_∞ error $P(x,t)$ in example 4 at different values of t with $\gamma = 0.2/\pi^2$.

		HBS		UAH tension BS		UAT tension BS	
t	N	L_∞	Order of Convergence	L_∞	Order of Convergence	L_∞	Order of Convergence
0.5	20	3.52E-00	-	3.40E+00	-	3.40E+00	-
	40	1.42E-00	1.30602	1.41E+00	1.27241	1.41E+00	1.27233
1	20	5.68E-00	-	5.52E+00	-	5.52E+00	-
	40	2.19E-00	1.37591	2.17E+00	1.34778	2.17E+00	1.34772
1.5	20	5.73E-00	-	5.57E+00	-	5.57E+00	-
	40	2.25E-00	1.35001	2.22E+00	1.32314	2.22E+00	1.32307
2	20	5.74E-00	-	5.63E+00	-	5.63E+00	-
	40	3.99E-00	0.52496	2.54E+00	1.1519	2.57E+00	1.1301

Table 3.11: Calculation of order of convergence with L_2 error $P(x,t)$ in example 4 at different values of t with $\gamma = 0.4/\pi^2$.

		HBS		UAH tension BS		UAT tension BS	
t	N	L_2	Order of Convergence	L_2	Order of Convergence	L_2	Order of Convergence
0.5	10	1.02E-01	-	1.06E-01	-	1.06E-01	-
	20	5.95E-02	0.89311	5.79E-02	0.87309	5.79E-02	0.873
	40	2.60E-02	1.19549	2.57E-02	1.17381	2.57E-02	1.1738
1	10	8.05E-02	-	8.04E-02	-	8.04E-02	-
	20	1.58E-02	2.3455	1.54E-02	2.38555	1.54E-02	2.38499
	40	1.08E-02	0.54221	1.07E-02	0.52349	1.07E-02	0.52349
1.5	10	9.45E-02	-	9.32E-02	-	9.31E-02	-
	20	7.19E-03	3.4523	6.97E-03	3.74147	6.97E-03	3.74105
	40	5.65E-03	0.3495	5.55E-03	0.32781	5.55E-03	0.32772
2	10	9.54E-02	-	9.33E-02	-	9.33E-02	-
	20	7.33E-03	3.6732	6.76E-03	3.78809	6.76E-03	3.78768
	40	5.60E-03	0.38884	5.36E-03	0.33306	5.36E-03	0.33295

Table 3.12: Calculation of order of convergence with L_∞ error $P(x,t)$ in example 4 at different values of t with $\gamma = 0.4/\pi^2$.

		HBS		UAH tension BS		UAT tension BS	
t	N	L_∞	Order of Convergence	L_∞	Order of Convergence	L_∞	Order of Convergence
0.5	10	5.75E-01	-	5.65E-01	-	5.65E-01	-

	20	4.48E-01	0.3045	4.35E-01	0.37574	4.35E-01	0.37554
	40	2.70E-01	0.7277	2.67E-01	0.7051	2.67E-01	0.70508
	10	4.74E-02	-	4.72E-01	-	4.72E-01	-
1	20	1.26E-01	1.94323	1.23E-01	1.93868	1.23E-01	1.93805
	40	1.26E-01	0.09851	1.16E-01	0.08365	1.16E-01	0.0837
	10	5.62E-02	-	5.44E-01	-	5.43E-01	-
1.5	20	4.82E-02	3.44342	4.64E-02	3.55095	4.64E-01	3.55048
	40	5.76E-02	0.259253	5.66E-02	0.285684	5.65E-01	0.285735
	10	5.54E-02	-	5.45E-01	-	5.44E-01	-
2	20	4.64E-02	3.5821	4.51E-02	3.59421	4.51E-01	3.59371
	40	5.52E-02	0.249642	5.41E-02	0.263287	5.41E-01	0.263306

3.4. Stability of the Scheme

The literature has covered both matrix stability and energy stability approaches extensively for stability analysis purposes [15, 199-201]. Take into account the time-dependent issue listed below using a collection of appropriate beginning and boundary values.

$$\frac{\partial P}{\partial t} = q'(P). \tag{3.24}$$

Here, q' is the spatial operator, with the accompanying initial and boundary value conditions. With the process of discretization, equation (3.24) gets condensed into a set of ODEs:

$$\frac{dP}{dt} = [Q]\mathbb{P} + F(\mathbb{b}), \tag{3.25}$$

where $F(\mathbb{b})$ indicates the non-homogeneous section with boundary conditions, Q denotes the coefficient matrix, and $\mathbb{P}$ symbolizes the unknown vector at the knot locations excluding the border, here Q takes the values as:

$$Q = -k\left(\gamma \sum_{j=1}^{N} d_{ij}^{4} + \beta \sum_{j=1}^{N} d_{ij}^{3} + \alpha \sum_{j=1}^{N} d_{ij}^{2}\right). \tag{3.26}$$

As exact solution of equation (3.25) is defined by the eigenvalues of the matrix Q, the stability of equation (3.25) is dependent on these values. The matrix Q plays a crucial role since its eigenvalues have a direct impact on the precise solutions. Consequently, the coefficient matrix's eigenvalues can be used to accurately characterize the entire scenario. The stable exact solutions are displayed as $t \to \infty$ when all the eigenvalues of Q have real component, that is $R(\lambda_i) \leq 0$. If the eigenvalues of the aforementioned matrix Q meet the criteria outlined in [15, 97].

It is shown that the suggested approach is reliable because the eigenvalues of matrix Q at various node positions occupy the stability region, the region that is explained in section 1.9. For various grid sizes, eigenvalues have been determined and are shown in the figures from Figure 3.13 to Figure 3.15.

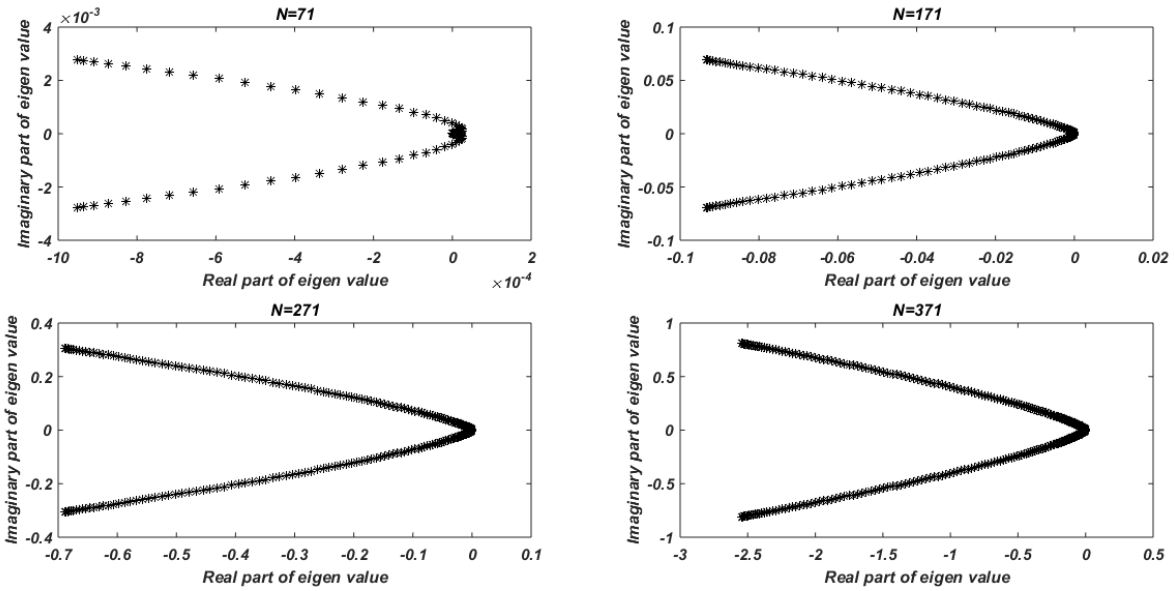

Figure 3.13: Plot of eigen values with HBS.

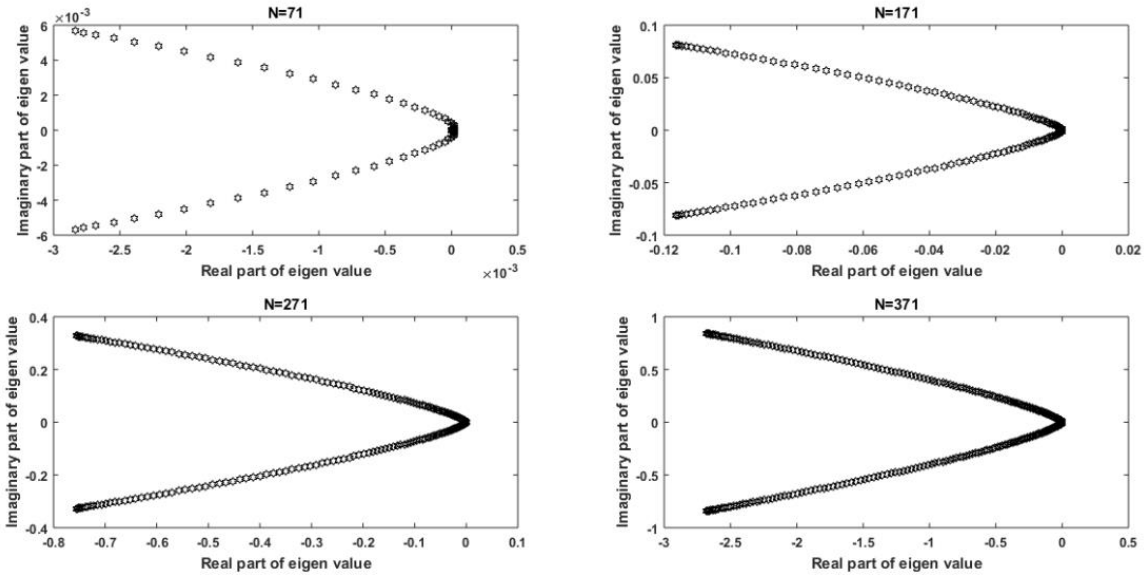

Figure 3.14: Plot of eigen values with UAH tension B-spline.

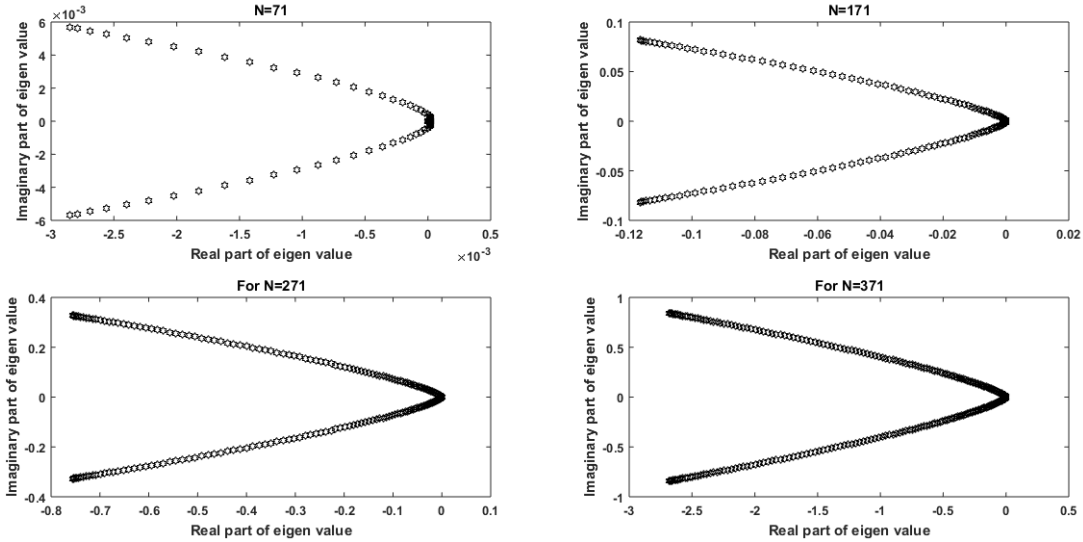

Figure 3.15: Plot of eigen values with UAT tension B-spline.

3.5. Summary

In this Chapter, numerical solution of the equation known as Kuramoto-Sivashinsky (KS) equation is provided, with implementation of three B-splines of order six that is

(d) Quintic Hyperbolic B-spline with DQM

(e) Quintic Uniform Algebraic Hyperbolic tension B-spline with DQM

(f) Quintic Uniform Algebraic trigonometric tension B-spline with DQM

With the implementation of DQM, the system of equation got converted into ODEs, then calculated by using the SSP Runge-Kutta method. It is witnessed that with the rise of time, the accuracy of solution is getting reduced which is because of the time truncation issues in the time derivative term. Also obtained approximations exhibit the same behavior as in the literature and are graphically displayed at different time intervals. Moreover, solutions that are obtained numerically are matched with the precise solution and are in excellent agreement with them. To assess the applicability of the suggested approach, four different test problems are resolved and different error norms L_2, L_∞, and GRE are calculated for the comparison purposes, which are appeared to be very less. Additionally, it is demonstrated that the approach is invariably stable.

Chapter 4

Numerical Solution of Extended Fisher-Kolmogorov Equation using quintic B-splines

In this Chapter, numerical solution of Extended Fisher-Kolmogorov equation in one and two dimension is obtained using above mentioned three quintic B-splines.

Research Objectives

1) Numerical solution of higher order differential equations will be obtained by using quintic Hyperbolic B-spline basis functions.
2) Numerical solution of third or fourth order differential equations will be obtained by using Uniform Algebraic trigonometric [UAT] tension B-spline as basis function.
3) To obtain the numerical solution, we will use differential quadrature method (DQM) along with quintic Uniform Algebraic Hyperbolic [UAH] tension B-spline.
4) The hyperbolic B-spline basis functions, UAT tension B-spline and UAH tension B-spline will be used to solve equations with two space dimensions.

4.1. Introduction

Many problems in different fields are demonstrated by nonlinear partial differential equations. Nonlinear equations can be shown by various renowned examples, which includes in quantum mechanics Schrödinger and Klein-Gordon equations, in biology nonlinear reaction-diffusion system, moreover generalized Zakharov system, and Gross-Pitaevskii equation. In fluid mechanics the Euler and Navier-Stokes equations, the Cahn-Hilliard equation from material sciences [202, 203]. In order to find solutions to PDEs that are nonlinear, researchers have consistently sought out effective techniques. In this

chapter, fourth order one and two dimension Extended Fisher-Kolmogorov (EFK) equation is examined by implementing the quintic splines based DQM.

The one dimension (1D) EFK equation can be given in the following form

$$U_t - U_{xx} + \Phi U_{xxxx} + (U^3 - U) = 0,\ x\epsilon[x_L, x_R] \text{ and } t\epsilon[0, t], \tag{4.1}$$

with drichlet boundary conditions as

$$U(x_L, t) = g_1(t), U(x_R, t) = g_2(t), U_{xx}(x_L, t) = 0, U_{xx}(x_R, t) = 0, \tag{4.2}$$

and initial condition as

$$U(x, 0) = U_0(x), \tag{4.3}$$

where, $U \in [a, b] \times [0, T]$, and $\Phi > 0$.

The two dimension (2D) EFK equation is given by:

$$U_t + \Phi \Delta^2 U - \Delta U + q(U) = g(x, y, t), \tag{4.4}$$

or this can be written as:

$$U_t + \Phi\left(U_{xxxx} + 2U_{xx}U_{yy} + U_{yyyy}\right) - \left(U_{xx} + U_{yy}\right) + q(U) = g(x, y, t), \tag{4.5}$$

with initial and boundary problems,

$$\begin{gathered} U(x, y, t) = U_0(x, y), \\ U = 0, \Delta U = 0, \textit{ on boundary}, \end{gathered} \tag{4.6}$$

where. $q(U) = U^3 - U$, the functions g_1 and g_2 are well-known and Φ is a constant which has positive values.

The EFK equation given in (4.1), and (4.4), that was proposed by Coullet et al. [204], also by Dee and Van Saarlos [205-207]. By keeping $\Phi = 0$ in equation (4.1) the standard Fisher-Kolmogorov (FK) equation can be obtained. Or, we can say that EFK equation can be obtained by adding forth-order stabilizing derivative to the FK equation [208, 209].

The equation EFK contains huge significant applications in various fields including in physics formation of pattern for bi-stable systems [206], transmission of domain wall in

liquid crystals [210], description about dynamics of brain tumors [211], around Lipschitz points phase transition mezoscopic model [212], for reaction-diffusion systems, travelling waves [213, 214], EFK extensively used in variety of physical phenomena including plasma physics, population growth, hydrodynamics, spread of infectious diseases, thermonuclear reaction etc. [213].

Recently, the steady state of this equation has received some attention, which exhibits periodic, heteroclinic or homoclinic solution depending on Φ. Numerous numerical approaches have been applied to analyse the 1D as well as 2D EFK problem numerically including orthogonal cubic spline based collocation method [215], Crank-Nicolson based finite difference method [208], quintic B-spline collocation technique [216, 14], C^1-conforming finite element method [217], Fourier pseudo-spectral technique, finite difference method for 2D EFK [218, 209], Four meshless techniques [219], finite element Galerkin method [217].

4.2. Implementation of method

4.2.1. Implementation of method for 1D EFK equation

Let's consider the mesh division system $a = x_1 < x_2 < \cdots < x_n = b$ of a closed interval [a,b], which is evenly transmitted and has a step size $h = x_{i+1} - x_i$. To calculate the numerical solution of EFK system (4.1), we need to approximate the derivatives. So, for the further computation, we must compute the 1^{st}, 2^{nd}, 3^{rd}, 4^{th} order derivative of the function U. By DQM r^{th} order numerical derivative of U will be estimated as:

$$U_x^{(r)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(r)} U(x_j) \qquad (4.7)$$

By using the above equations, the approximate values of derivatives of 1^{st}, 2^{nd}, 3^{rd} and 4^{th} order can be given as:

$$U_x^{(1)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(1)} U(x_j)\,, U_x^{(2)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(2)} U(x_j),$$
$$U_x^{(3)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(3)} U(x_j)\,, U_x^{(4)}(x_i) = \sum_{j=1}^{N} d_{ij}^{(4)} U(x_j), \quad (4.8)$$

where, $d_{ij}^{(1)}$, $d_{ij}^{(2)}$, $d_{ij}^{(3)}$, and $d_{ij}^{(4)}$ are the weighting coefficients of respective order as the 'power' of these values. Consequently, system of equations as given below can be created with the applicability of the formulas from equation (4.8) to equation (4.1):

$$U_t = -\left\{ -\sum_{j=1}^{N} d_{ij}^2 U(x_j) + \Phi \sum_{j=1}^{N} d_{ij}^4 U(x_j) + ((U(x_i))^3 - U(x_i)) \right\} \quad (4.9)$$

The weighting factors of order 1^{st} may be determined with the use of the quintic HBS, quintic UAH tension BS, and quintic UAT tension BS as explained in the section (1.7.1), (1.7.2), and (1.7.3) respectively and $2nd$ or higher order weighting coefficient may be computed with the use of the formulae given in the equation (1.40). The SSP-RK43 regime [155] is then utilized to calculate the resulting ordinary differential equations further.

4.2.2. Implementation of method for 2D EFK equation

Let us consider the mesh division system $a = x_1 < x_2 < \cdots < x_N = b$ and $c = y_1 < y_2 < \cdots < y_M = d$ with a uniform step size $h_1 = x_{i+1} - x_i$, and $h_2 = y_{j+1} - y_j$ respectively in x and y directions, where $i = 1,2, \ldots, N$, $j = 1,2, \ldots, M$ of a definite period $[a, b] \times [c, d]$. By DQM r^{th} order derivative of U w.r.t x and w.r.t y can be approximated as:

$$\frac{\partial^r U(x_i, y_j, t)}{\partial x^r} = \sum_{k=1}^{N} d_{ik}^{(r)} U(x_k, y_j, t)$$
$$\frac{\partial^r U(x_i, y_j, t)}{\partial y^r} = \sum_{k=1}^{N} b_{jk}^{(r)} U(x_i, y_k, t) \quad (4.10)$$

where, $d_{ij}^{(r)}, b_{ij}^{(r)}$ are weighting coefficeients of r^{th} order derivatives w.r.t x and y respectively. By using the above equations, the approximate values of derivatives of 1st, 2nd, 3rd and 4th order w.r.t. x can be given as:

$$\frac{\partial U(x_i, y_j, t)}{\partial x} = \sum_{k=1}^{N} d_{ik}^{(1)} U(x_k, y_j, t), \frac{\partial^2 U(x_i, y_j, t)}{\partial x^2} = \sum_{k=1}^{N} d_{ik}^{(2)} U(x_k, y_j, t)$$
$$\frac{\partial^3 U(x_i, y_j, t)}{\partial x^3} = \sum_{k=1}^{N} d_{ik}^{(3)} U(x_k, y_j, t), \frac{\partial^4 U(x_i, y_j, t)}{\partial x^4} = \sum_{k=1}^{N} d_{ik}^{(4)} U(x_k, y_j, t) \quad (4.11)$$

In a similar manner, the approximate values of derivatives of 1st, 2nd, 3rd and 4th order w.rt. y can be given as:

$$\frac{\partial U(x_i, y_j, t)}{\partial y} = \sum_{k=1}^{N} b_{jk}^{(1)} U(x_i, y_k, t), \frac{\partial^2 U(x_i, y_j, t)}{\partial y^2} = \sum_{k=1}^{N} b_{jk}^{(2)} U(x_i, y_k, t)$$
$$\frac{\partial^3 U(x_i, y_j, t)}{\partial y^3} = \sum_{k=1}^{N} b_{jk}^{(3)} U(x_i, y_k, t), \frac{\partial^4 U(x_i, y_j, t)}{\partial y^4} = \sum_{k=1}^{N} b_{jk}^{(4)} U(x_i, y_k, t) \quad (4.12)$$

Consequently, system of equations as given below can be created with the applicability of formulas from equation (4.18), and (4.19) to equation (4.5):

$$U_t(x_i, y_j, t) = -\Bigg\{\Phi\left(\sum_{k=1}^{N} d_{ik}^{(4)} U(x_k, y_j) + 2 \times \sum_{k=1}^{N} d_{ik}^{(2)} U(x_k, y_j) \times \sum_{k=1}^{N} d_{ik}^{(2)} U(x_i, y_k)\right) - \left(\sum_{k=1}^{N} d_{ik}^{(2)} U(x_k, y_j) + \sum_{k=1}^{N} d_{ik}^{(2)} U(x_i, y_k)\right) + \left(\left(U(x_i, y_j)\right)^3 - U(x_i, y_j)\right)\Bigg\} + g(x, y, t) \quad (4.13)$$

The weighting factors of order 1^{st} may be determined with the use of the quintic HBS, quintic UAH tension BS, and quintic UAT tension BS as explained in the section (1.7.1), (1.7.2), and (1.7.3) respectively and $2nd$ or higher order weighting coefficient can be computed with the use of the formulae given in the equation (1.10), (1.11). The SSP-RK43 regime [155] is then utilized to calculate the resulting ordinary differential equations further.

4.3. Numerical solution of EFK employing three quintic B-splines

4.3.1. Numerical Solution one-dimension EFK equation

Example 1:

Consider one dimension EFK equation (4.1) with initial condition [14, 187, 220] as

$$U(x, 0) = -10^{-3} exp(-x^2), \quad (4.14)$$

and boundary condition as:

$$U(-4, t) = -1, U(4, t) = -1, U_{xx}(-4, t) = 0, U_{xx}(4, t) = 0. \quad (4.15)$$

To perceive the approximate solution, parameters are taken as $\Delta t = 0.001, N(number\ of\ grid\ points) = 101$. The profiles of $U(x, t)$ that depend on time at $\Phi = 0, 0.0001$ are figured for various time values.

In example 1, equation (4.1) is approximated numerically with varying values of Φ , and with a domain $x \in [-4, 4]$ at varying $t \in [0.25, 5]$ and time-discretization as $\Delta t =$

0.001. From Figure 4.1 to Figure 4.3, the numerical output of equation (4.1) with initial condition (4.14) is displayed for $\Phi = 0, 0.0001$ at various values of time with the implementation of the splines named quintic HBS, quintic UAH tension BS and quintic UAT tension BS. It is extremely obvious from the graphical depiction that the numerical findings obtained are well-aligned with findings found in the previous work [14, 187, 219]. From the figures, it is evident that as time goes on solution decays and finally approaches to -1.

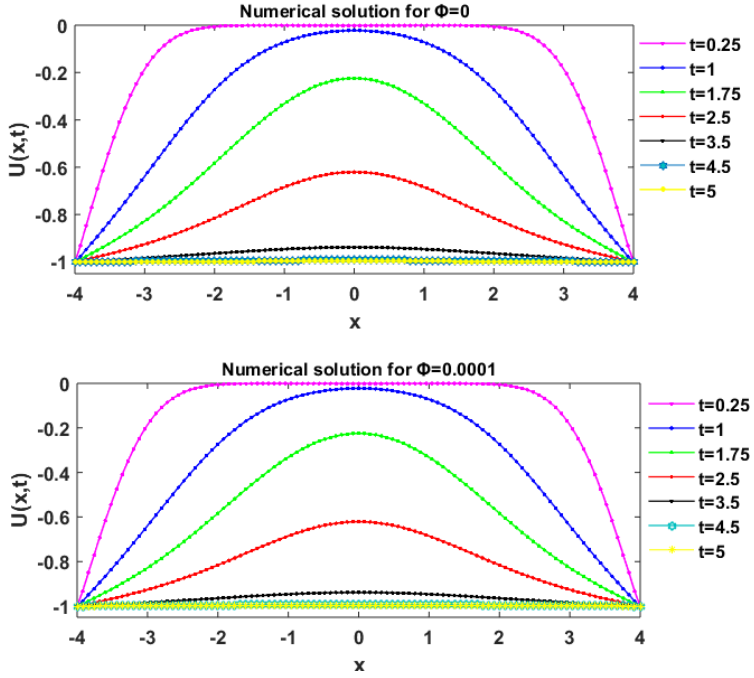

Figure 4.1: For various values of time, the num. sol. to Example 1 with $\Delta t = 0.001$, *N=101, and* $\Phi = 0, \Phi = 0.0001$ *with HBS.*

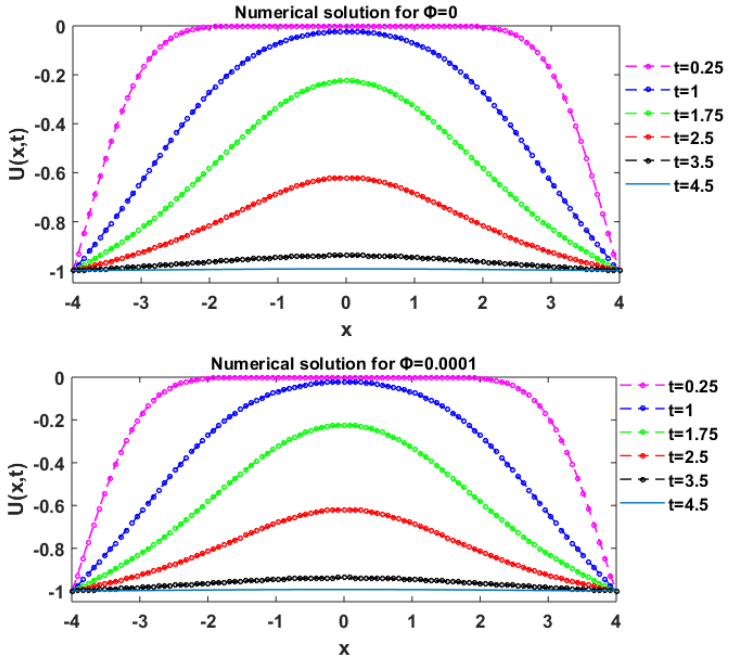

Figure 4.2: For various values of time, the num. sol. to Example 1 with $\Delta t = 0.001$, *N=101, and* $\Phi = 0, \Phi = 0.0001$ *with UAH tension B-spline.*

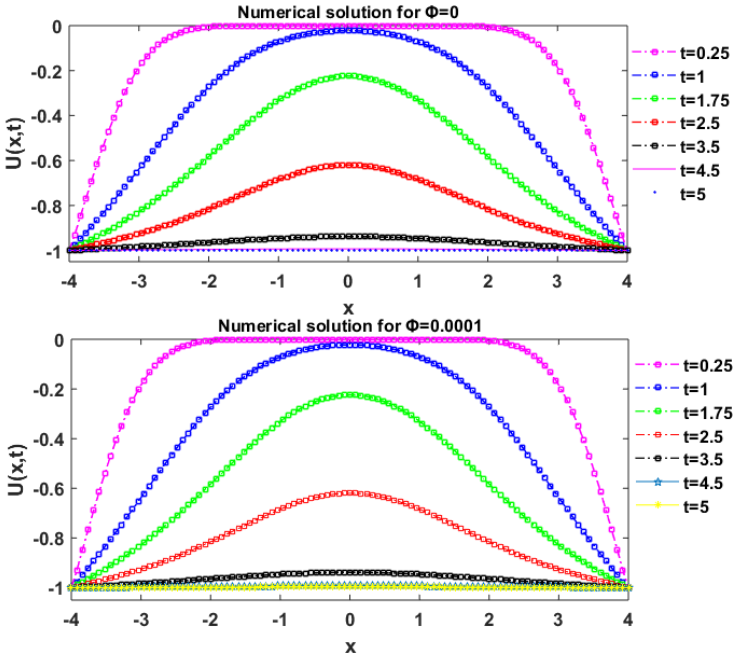

Figure 4.3: For various values of time, the num. sol. to Example 1 with $\Delta t = 0.001$, *N=101, and* $\Phi = 0, \Phi = 0.0001$ *with UAT tension B-spline.*

Example 2:

Consider one dimension EFK equation (4.1) with initial condition [14, 187, 220] as:

$$U(x,0) = 10^{-3}exp(-x^2), \tag{4.16}$$

and boundary condition as:

$$U(-4,t) = 1, U(4,t) = 1, U_{xx}(-4,t) = 0, U_{xx}(4,t) = 0. \tag{4.17}$$

To perceive the approximate solution, parameters are taken as $\Delta t = 0.001, N(number\ of\ grid\ points) = 51$. The profiles of $U(x,t)$ that depend on time at $\Phi = 0.0001$ are figured for various time values.

In example 2, equation (4.1) is approximated numerically for $\Phi = 0.0001$, and with a domain $x \in [-4,4]$ at varying $t \in [0.25,5]$ and time-discretization as $\Delta t = 0.001$. From Figure 4.4 to Figure 4.6, the numerical output of (4.1) with initial condition (4.16) is displayed for $\Phi = 0.0001$ at various values of time with the implementation of the splines named quintic HBS, quintic UAH tension BS and quintic UAT tension BS. It is extremely obvious from the graphical depiction that the numerical findings obtained are well-aligned with findings found in the previous work [14, 187, 219]. From the figures, it is evident that as time goes solution decays and finally approaches to 1.

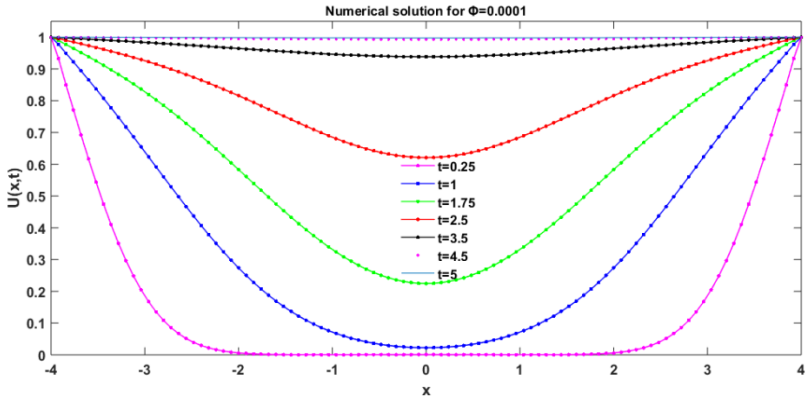

Figure 4.4: For various values of time, the num. sol. to Example 2 with $\Delta t = 0.001$, *N=51, and* $\Phi = 0.0001$ *with HBS.*

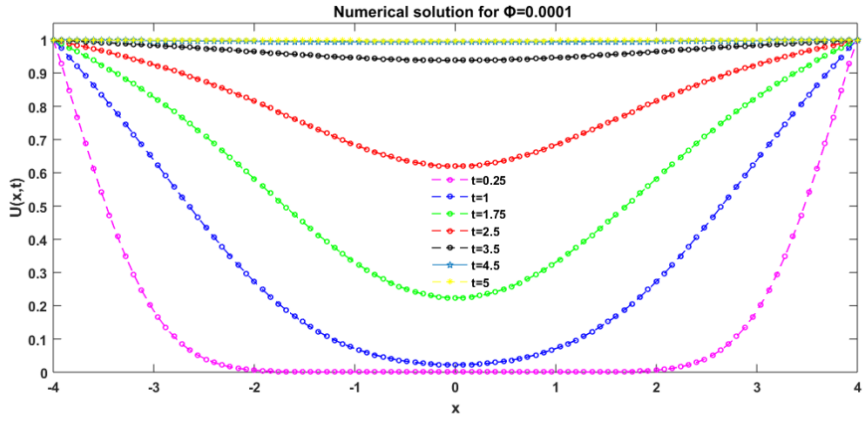

Figure 4.5: For various values of time, the num. sol. to Example 2 with $\Delta t = 0.001$, *N=51, and* $\Phi = 0.0001$ *with UAH tension B-spline.*

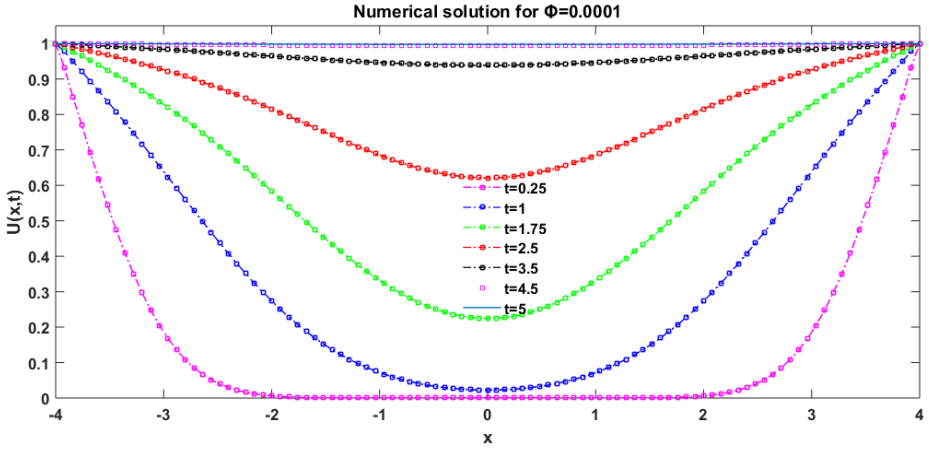

Figure 4.6: For various values of time, the num. sol. to Example 2 with $\Delta t = 0.001$, *N=51, and* $\Phi = 0.0001$ *with UAT tension B-spline.*

Example 3:

Consider one dimension EFK equation (4.1) with initial [14, 187, 220, 221] and boundary condition as:

$$U(x,0) = -sin(\pi x), \qquad (4.18)$$

$$U(-4,t) = 0, U(4,t) = 0, U_{xx}(-4,t) = 0, U_{xx}(4,t) = 0$$

To perceive the approximate solution, values of parameters are taken as $\Delta t = 0.0001, N(number\ of\ grid\ points) = 421$. The profiles of $U(x,t)$ that depends on time at $\Phi = 0, 0.0001$ are figured for various time values.

In example 3, equation (4.1) is approximated numerically for varying values of Φ , and with a domain $x \in [-4,4]$ at varying $t \in [0.01, 0.20]$ and time-discretization as $\Delta t = 0.0001$. From Figure 4.7 to Figure 4.9, the numerical output of equation (4.1) with initial condition (4.18) is displayed for $\Phi = 0, 0.0001$ at various values of time with the implementation of the splines named quintic HBS, quintic UAH tension BS and quintic UAT tension BS. It is extremely obvious from the graphical depiction that the numerical findings obtained are well-aligned with findings found in the previous work [14, 187, 219, 221].

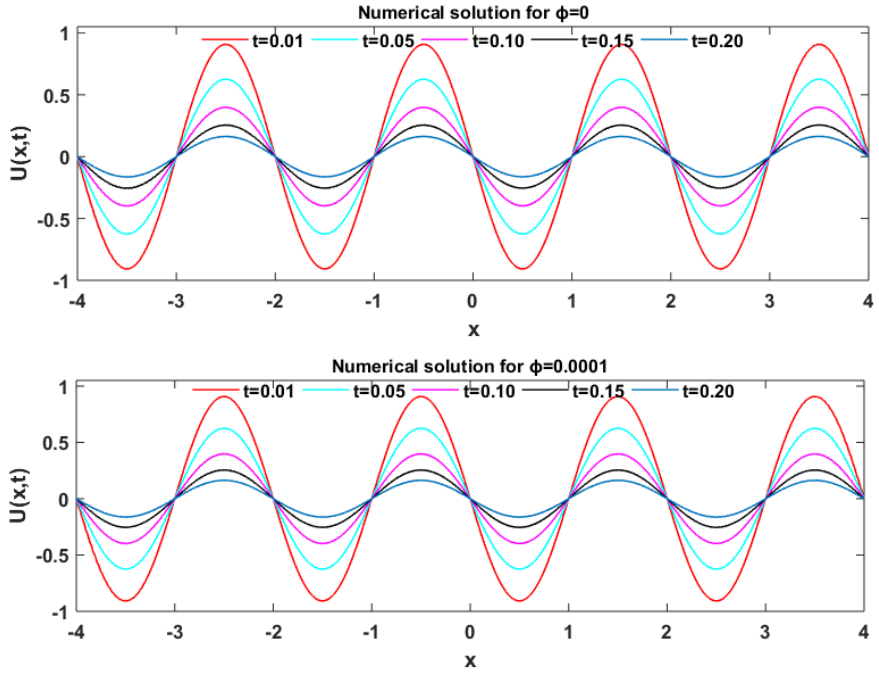

Figure 4.7: Num. sol. of example 3 having distinct time values, $\Delta t = 0.0001$, *N=421, and* $\Phi = 0, \Phi = 0.0001$ *with HBS.*

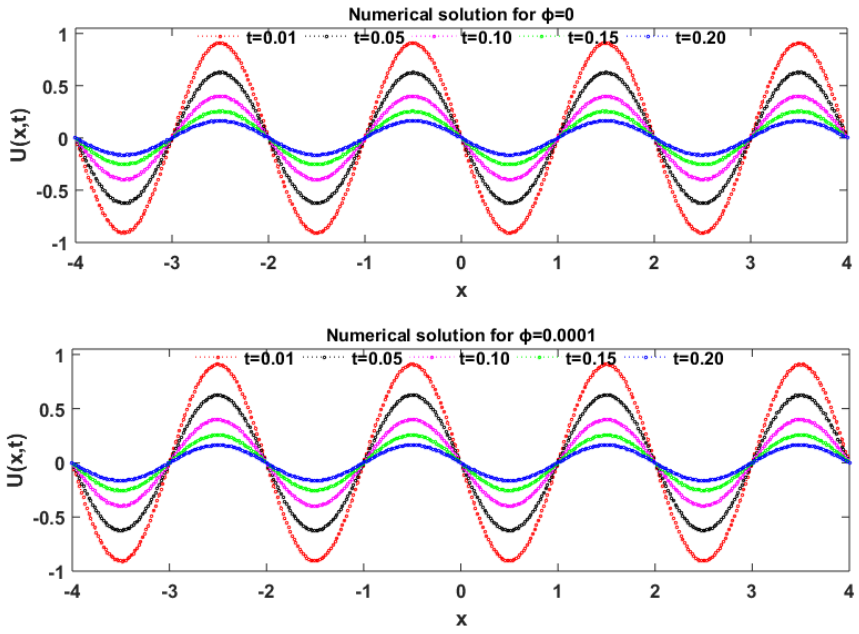

Figure 4.8: Num. sol. of example 3 having distinct time values, $\Delta t = 0.0001$, *N=421, and* $\Phi = 0, \Phi = 0.0001$ *with UAH tension B-spline.*

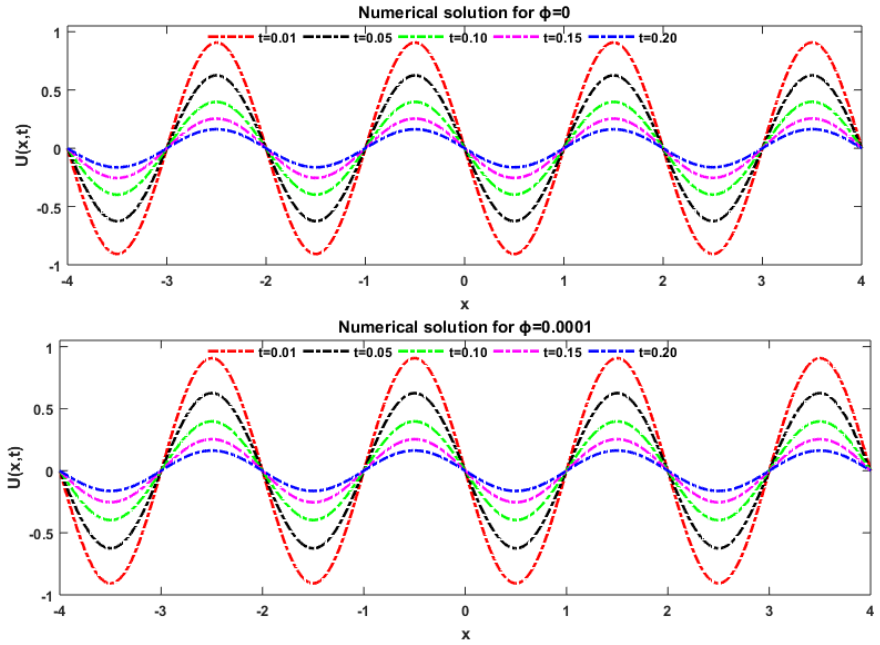

Figure 4.9: Num. sol. of example 3 having distinct time values, $\Delta t = 0.0001$, *N=421, and* $\Phi = 0, \Phi = 0.0001$ *with UAT tension B-spline.*

4.3.2. Numerical Solution two-dimension EFK equation

Example 4:

Consider two dimension EFK equation (4.4) for $g(x, y, t) = 0$ with initial condition [220, 209] as:

$$U(x, y, 0) = x^3(1 - x)^3 y^3(1 - y)^3 \text{ with the domain } (x, y)\epsilon[0,1] \times [0,1], \quad (4.19)$$

and boundary condition as: $U = 0, \Delta U = 0$

To perceive the approximate solution with $q(U) = U^3 - U$, values of parameters are taken as $\Delta t = 0.0001, N(number\ of\ grid\ points) = 31$. The profiles of $U(x, t)$ that depends on time at $\Phi = 0.0001$ are figured for various time values.

In example 4, the 2D EFK equation (4.5) is approximated numerically for $\Phi = 0.0001$, and with a domain $(x, y)\epsilon[0,1] \times [0,1]$ at varying $t \in [0.1,1]$ and time-discretization as $\Delta t = 0.0001$. From Figure 4.10 to Figure 4.15, the numerical output of equation (4.5) with initial condition (4.19) is displayed for $\Phi = 0.0001$ at various values of time with the implementation of the splines named quintic HBS, quintic UAH tension BS and quintic UAT tension BS. It is extremely obvious from the graphical depiction that the numerical findings obtained are well-aligned with findings found in the previous work.

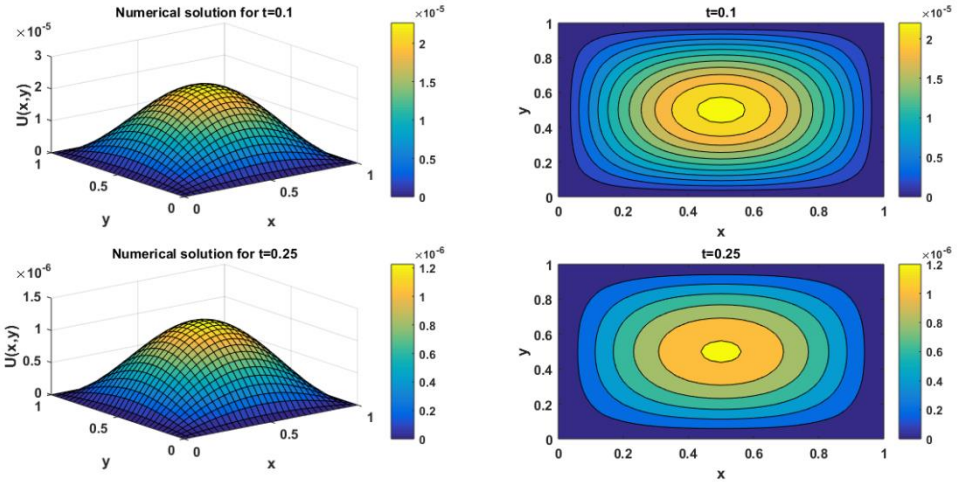

Figure 4.10: Num. sol. of EFK equation (example 4) at time $t = 0.1, 0.25$ *and* $\Delta t = 0.0001, N = 31$ *with* $\Phi = 0.0001$ *with HBS.*

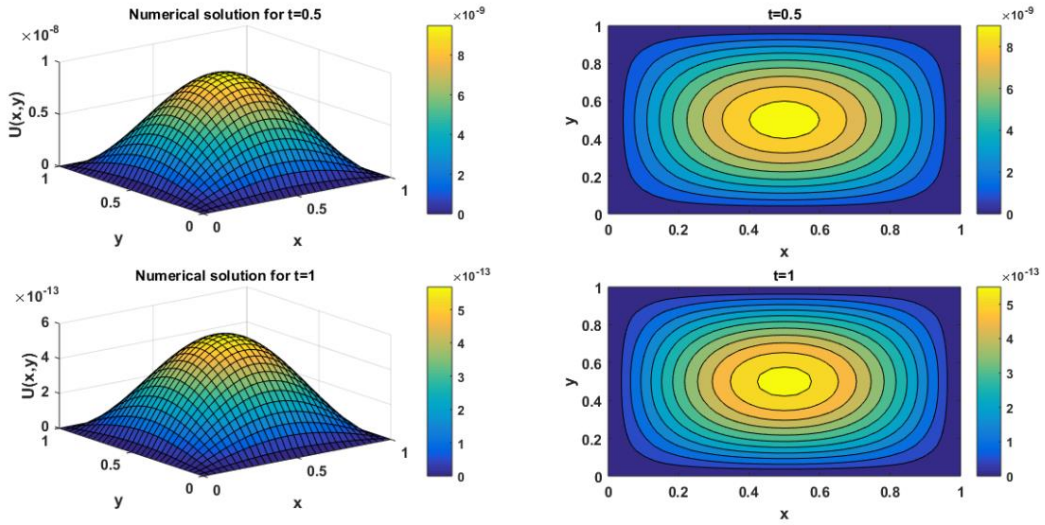

Figure 4.11: Num. sol. of EFK equation (example 4) at time $t = 0.5, 1$ *and* $\Delta t = 0.0001, N = 31$ *with* $\Phi = 0.0001$ *with HBS.*

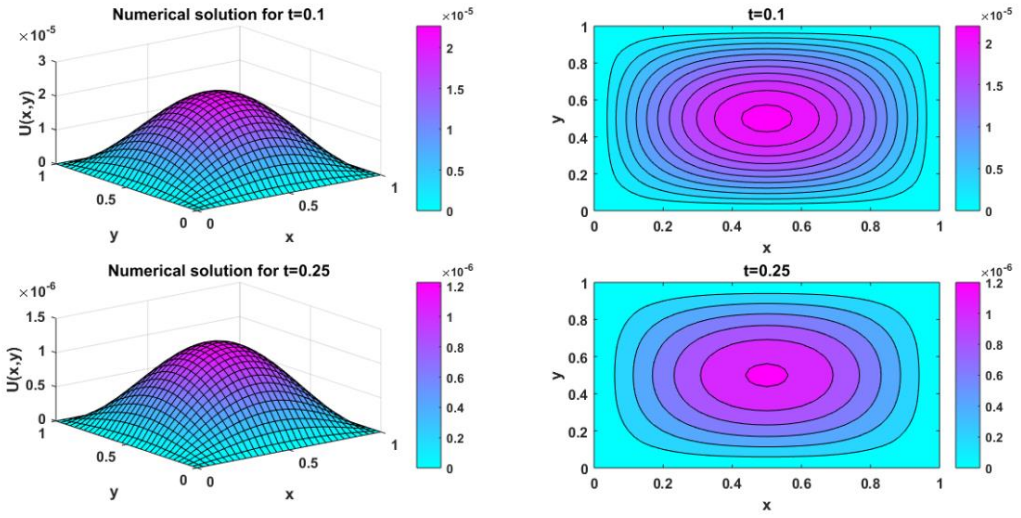

Figure 4.12: Num. sol. of EFK equation (example 4) at time $t = 0.1, 0.25$ *and* $\Delta t = 0.0001, N = 31$ *with* $\Phi = 0.0001$ *with UAH tension B-spline.*

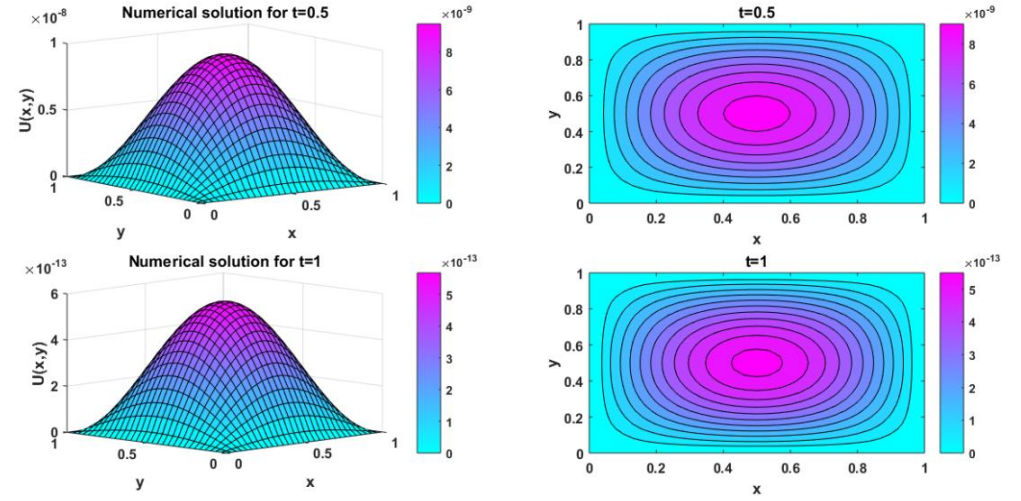

Figure 4.13: Num. sol. of EFK equation (example 4) at time $t = 0.5, 1$ *and* $\Delta t = 0.0001, N = 31$ *with* $\Phi = 0.0001$ *with UAH tension B-spline.*

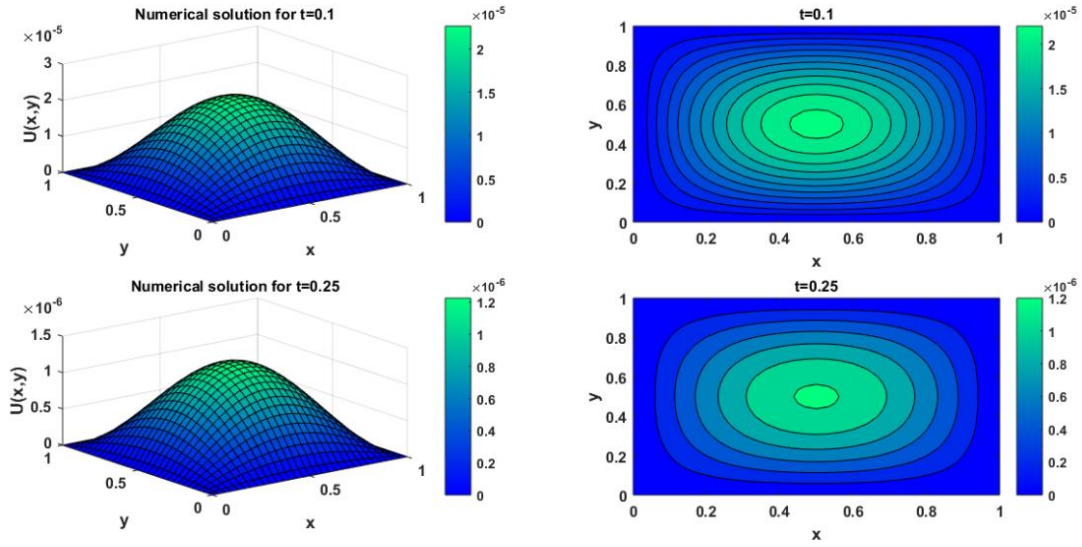

Figure 4.14: Num. sol. of EFK equation (example 4) at time $t = 0.1, 0.25$ *and* $\Delta t = 0.0001, N = 31$ *with* $\Phi = 0.0001$ *with UAT tension B-spline.*

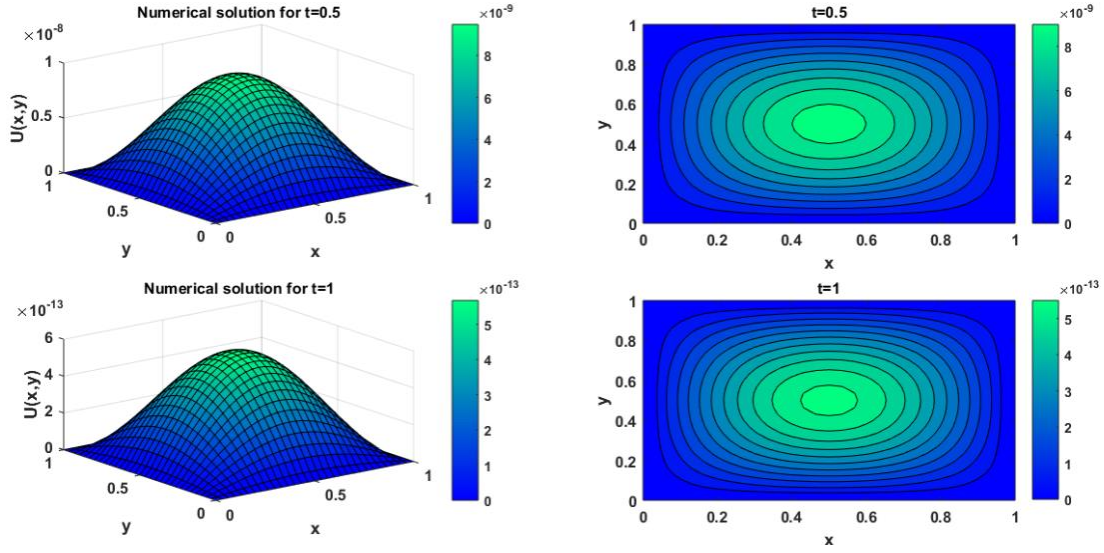

Figure 4.15: Num. sol. of EFK equation (example 4) at time $t = 0.5, 1$ *and* $\Delta t = 0.0001, N = 31$ *with* $\Phi = 0.0001$ *with UAT tension B-spline.*

Example 5:

Consider two dimension EFK equation (4.4) with

$$g(x,y,t) = sin(2\pi x)sin(2\pi y)e^{-t}\left(sin(2\pi x)^2 sin(2\pi y)^2 e^{(-t)^2} + 64\Phi(\pi)^4 + 8(\pi)^2\right) \quad (4.20)$$

Analytical solution [209] is defined by:

$$U(x,y,t) = sin(2\pi x)sin(2\pi y)e^{-t}. \quad (4.21)$$

From the analytical solution initial and boundary conditions can be extracted for the problem.

To perceive the approximate solution with $q(U) = U^3 - U$, values of parameters are taken as $\Delta t = 0.00001, N(number\ of\ grid\ points) = 41$. The profiles of $U(x,t)$ that depens on time at $\Phi = 0.0001$ are figured for various time values.

In example 5, the 2D EFK equation (4.5) is approximated numerically for varying values of $\Phi = 0.0001$, and with a domain $(x,y)\epsilon[0,1] \times [0,1]$ at varying $t \in [0.1,1]$ and time-discretization as $\Delta t = 0.0001$. Numerical solution of equation (4.5) with exact solution (4.21) is displayed for $\Phi = 0.0001$ at various values of time with the implementation of the splines named quintic HBS, quintic UAH tension BS and quintic UAT tension BS. From Figure 4.16 to Figure 4.18, the evaluation of numerical and precise solutions of the EFK, are represented using HBS, UAH tension BS, and UAT tension BS. It is extremely obvious from the graphical depiction that the numerical findings obtained are well-aligned with findings found in the previous work and with exact solutions. From Table 4.1 to Table 4.3 norms L_2, L_∞ and average norm are calculated with grid points $N = 41$ and $\Delta t = 0.00001$, along with that comparison is made with [222].

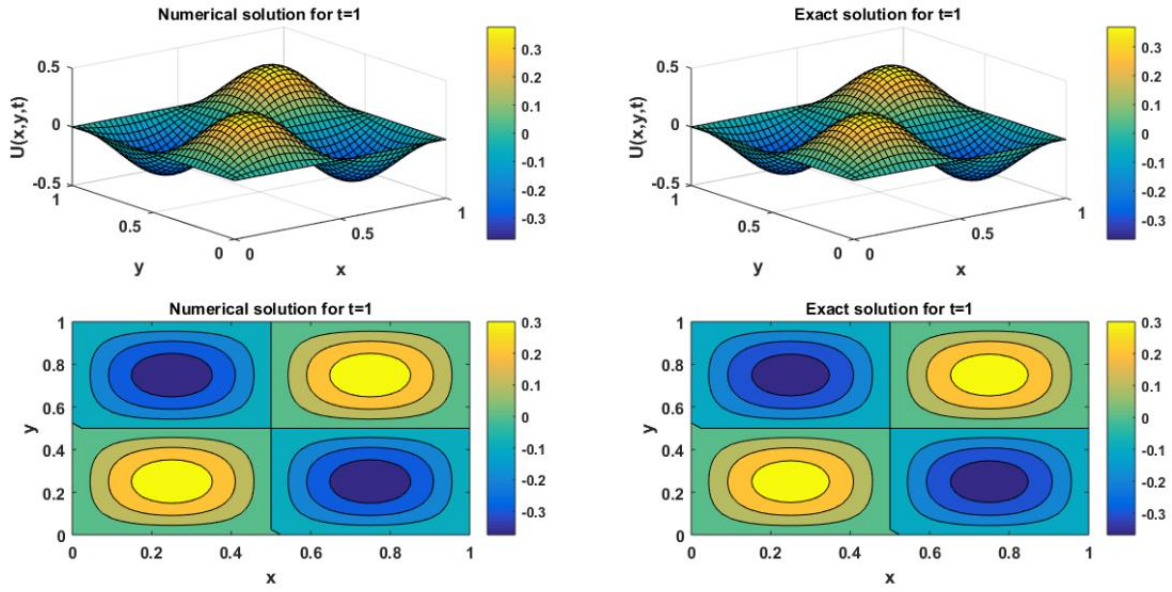

Figure 4.16: Evaluation of numerical and precise solutions of EFK equation (example 5) for time $t = 1$ *and* $\Delta t = 0.00001, N = 41$ *with* $\Phi = 0.0001$ *with HBS.*

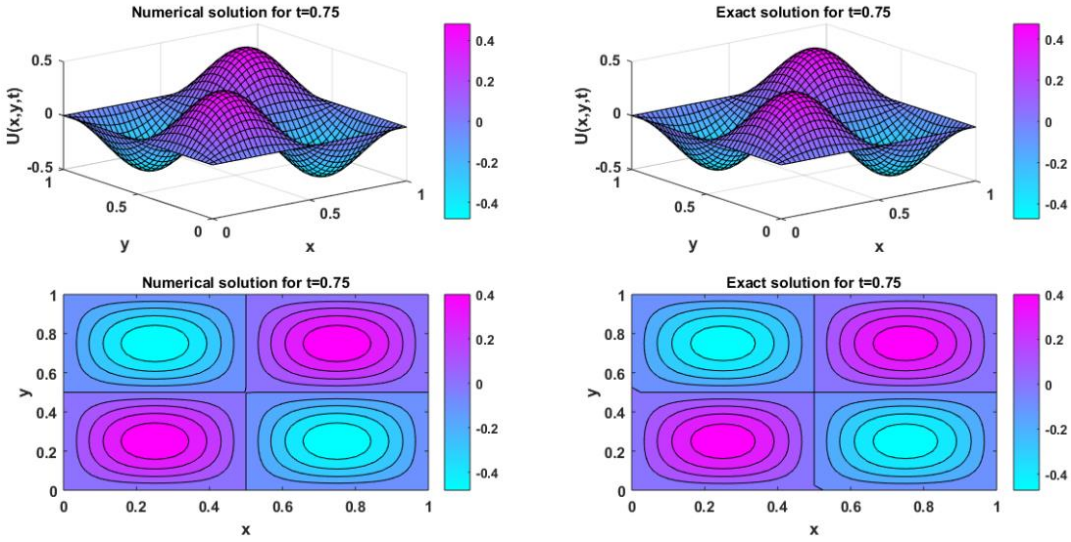

Figure 4.17: Evaluation of numerical and precise solutions of EFK equation (example 5) for time $t = 0.75$ *and* $\Delta t = 0.00001, N = 41$ *with* $\Phi = 0.0001$ *with UAH tension B-spline.*

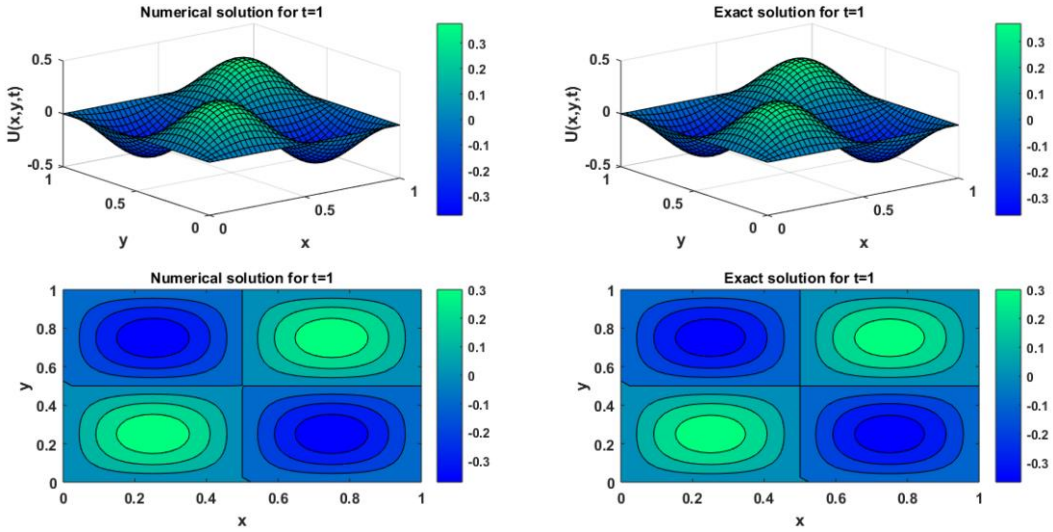

Figure 4.18: Evaluation of numerical and precise solutions of EFK equation (example 5) for time $t = 1$ *and* $\Delta t = 0.00001, N = 41$ *with* $\Phi = 0.0001$ *with UAT tension B-spline.*

Table 4.1: Comparison of Maximum error norm L_∞ *with parameters* $\Delta t = 0.00001, N = 41$, *and* $\Phi = 0.0001$ *of example 5 for different time values.*

t	HBS	UAH tension B-spline	UAT tension B-spline	Joshi [222]
0.1	1.51E-02	1.51E-02	1.43E-02	-
0.25	1.30E-02	1.31E-02	1.21E-02	1.60E-02
0.75	7.90E-03	8.00E-03	9.40E-03	9.72E-03
1	6.20E-03	7.40E-03	7.40E-03	7.58E-03

Table 4.2: Comparison of Average error norm with parameters $\Delta t = 0.00001, N = 41$, *and* $\Phi = 0.0001$ *of example 5 for different time values.*

t	HBS	UAH tension B-spline	UAT tension B-spline	Joshi [222]
0.1	6.50E-03	6.50E-03	4.80E-03	-
0.25	5.60E-03	5.60E-03	4.50E-03	4.64E-03
0.75	3.40E-03	3.40E-03	3.10E-03	2.81E-03
1	2.70E-03	2.60E-03	2.60E-03	2.19E-03

Table 4.3: Comparison of error norm L_2 *with parameters* $\Delta t = 0.00001, N = 41$, *and* $\Phi = 0.0001$ *of example 5 for different time values.*

t	HBS	UAH tension B-spline	UAT tension B-spline	Joshi [222]
0.1	1.16E-02	1.15E-02	1.43E-02	-
0.25	9.90E-03	9.90E-03	1.21E-02	6.70E-03

0.75	6.00E-03	6.00E-03	9.40E-03	4.07E-03
1	4.70E-03	7.40E-03	7.40E-03	3.17E-03

4.4. Stability of the Scheme

We used matrix stability method to show the stability of proposed method [15, 95, 96]. Discretizing the equation (4.1), that will be transformed into the system of ODEs equation (4.9) in case of HBS. So, after converting the EFK equation into a system of ODEs (4.9) and assuming the nonlinear term as constant the equation (4.1) can be inscribed as:

$$\frac{dP}{dt} = [Q]\mathbb{P} + F(\mathbb{b}) \tag{4.22}$$

where $F(\mathbb{b})$ indicates the non-homogeneous section of equation (4.1) with boundary conditions, Q denotes the coefficient matrix, and $\mathbb{P}$ symbolizes the unknown vector at the knot locations excluding the border, here Q takes the values as::

$$Q = -k\left(\Phi \sum_{j=1}^{N} d_{ij}^{4} - \sum_{j=1}^{N} d_{ij}^{2}\right). \tag{4.23}$$

In a similar manner, this formation can be made with UAH, and with UAT tension B-spline. For the constancy of the proposed technique, eigen values of the matrix Q are calculated for various grid points as shown in the figure from Figure 4.19-Figure 4.21 with different splines in 1D It is found that all the eigenvalues of Q are complex and lie within the stability region as explained in the section 1.9, hence it is possible to conclude that the suggested approach is stable.. In 2D it is found that all the eigenvalues of Q are real and lie within the stability region as explained in the section 1.9, which are shown from Figure 4.22 to Figure 4.24.

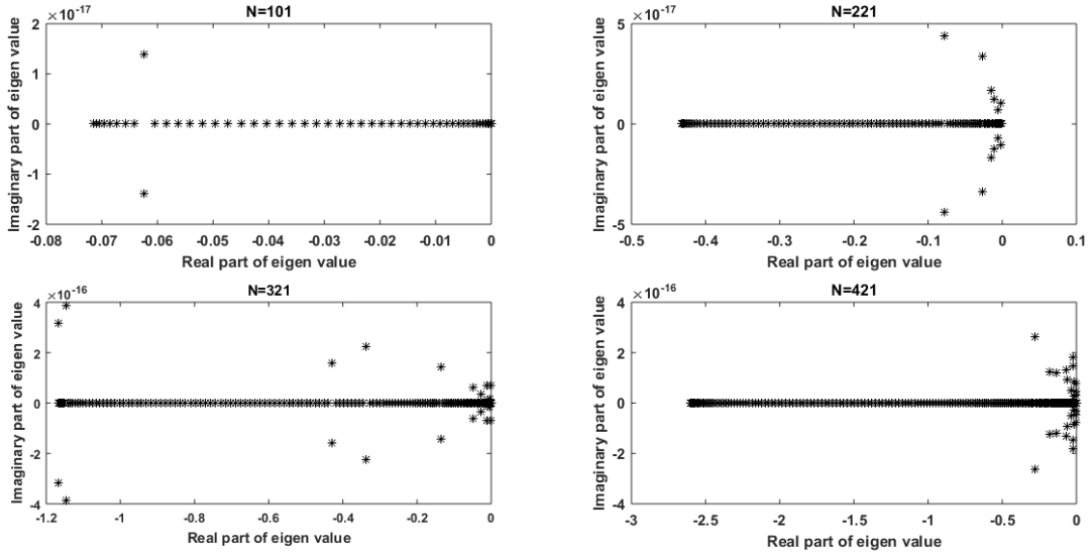

Figure 4.19: Plot of eigen values of EFK with HBS in one dimension.

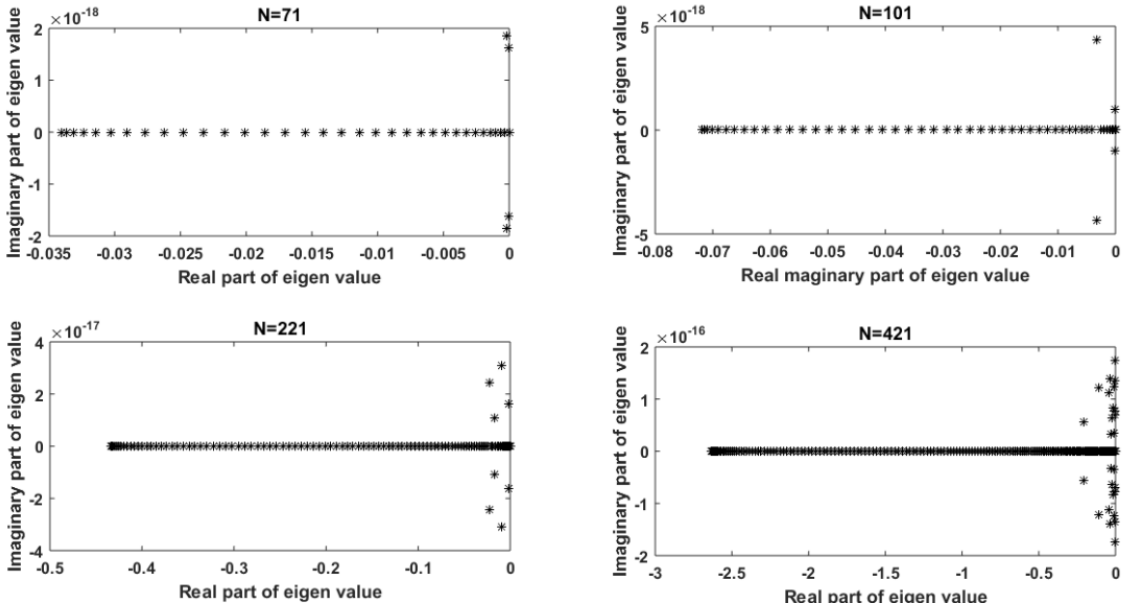

Figure 4.20: Plot of eigen values of EFK with UAH tension B-spline in one dimension.

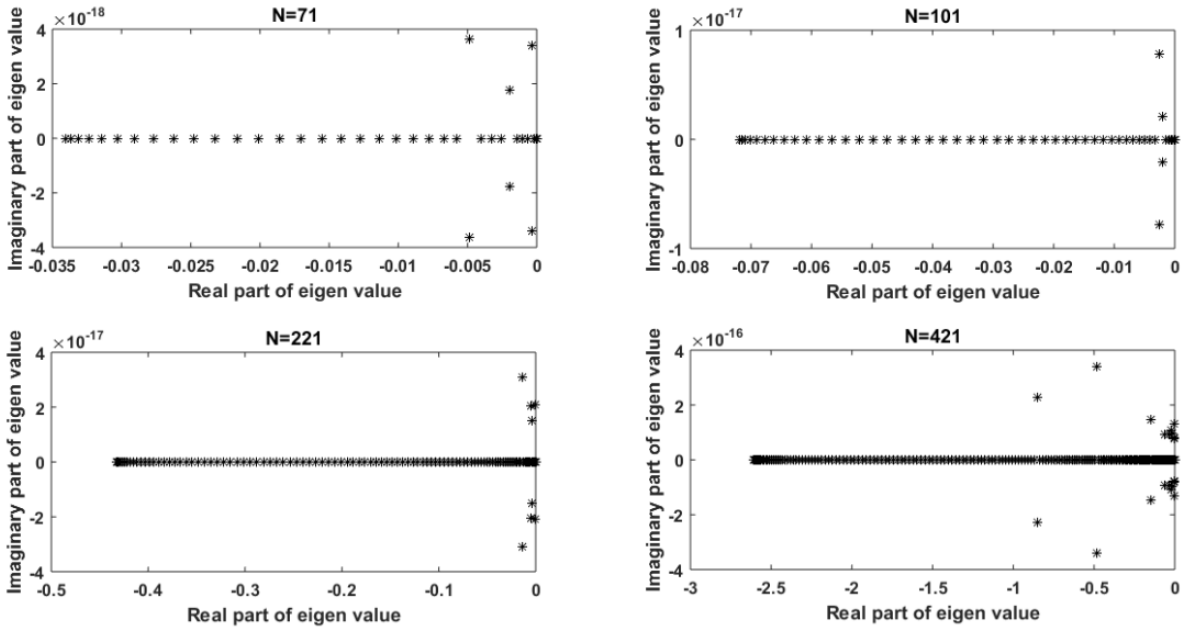

Figure 4.21: Plot of eigen values of EFK with UAT tension B-spline in one dimension.

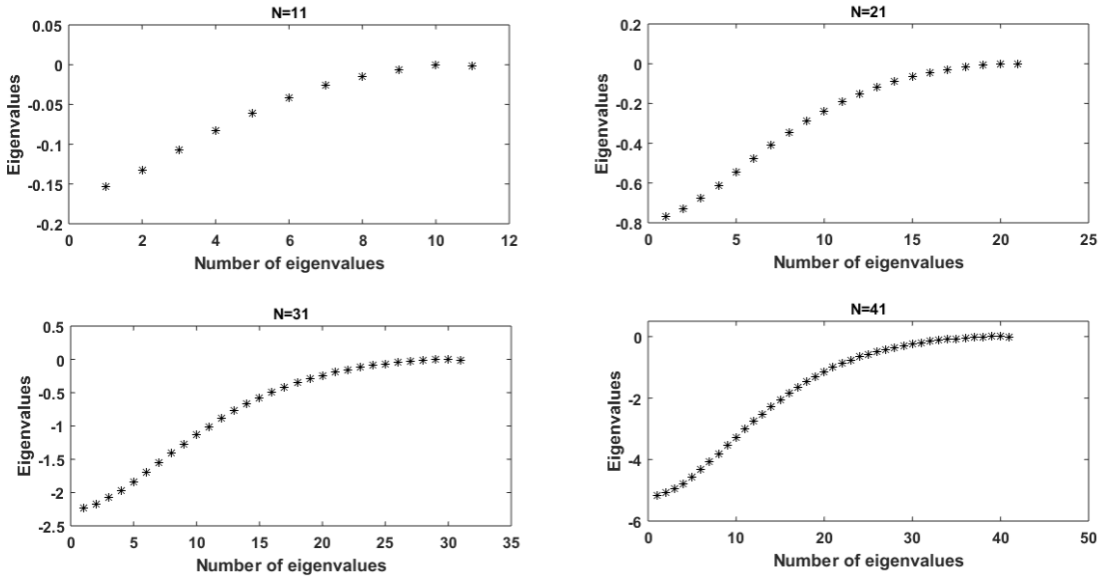

Figure 4.22: Plot of eigen values of EFK with HBS in two dimension.

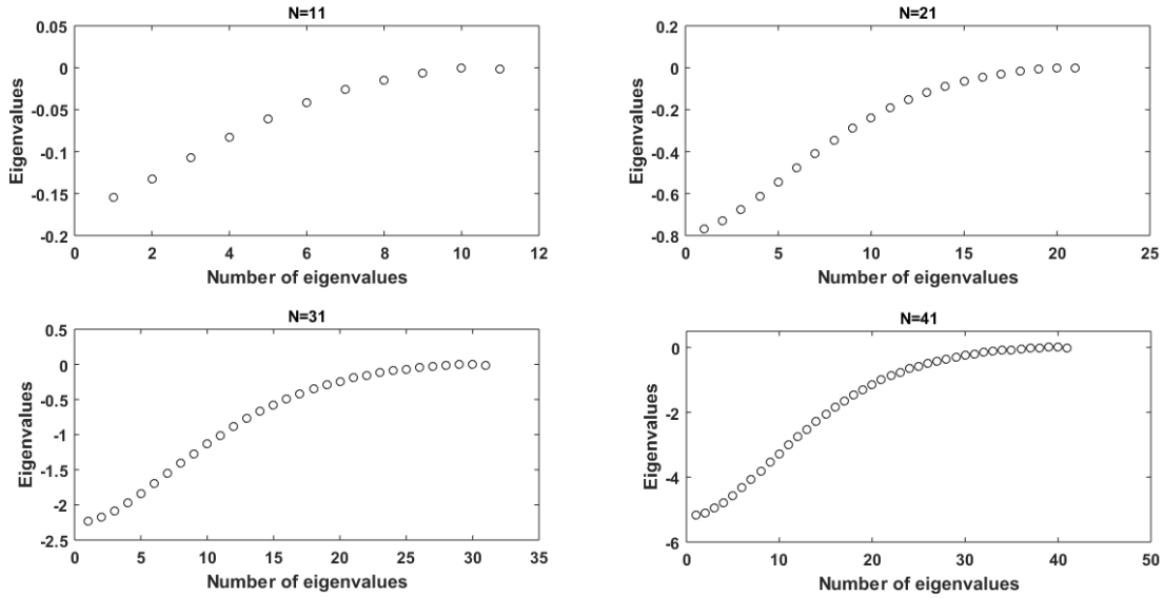

Figure 4.23: Plot of eigen values of EFK with UAH tension B-spline in two dimension.

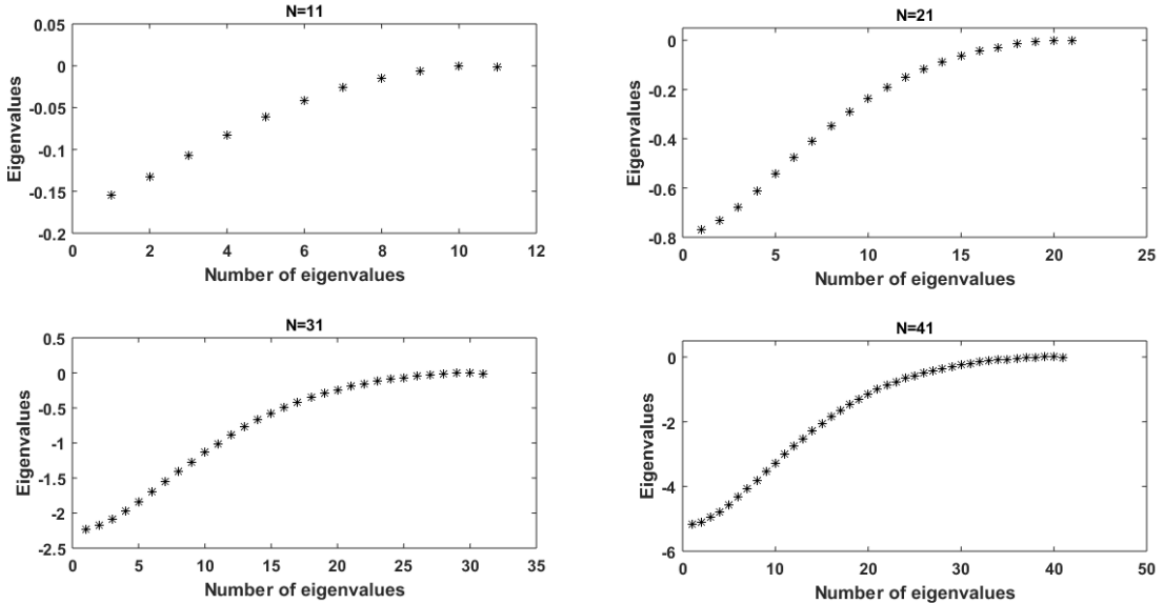

Figure 4.24: Plot of eigen values of EFK with UAT tension B-spline in two dimension.

4.5. Summary

In this Chapter, numerical solution of equation known as the Extended Fisher-Kolmogorov (EFK) equation for 1D and 2D is provided, with the implementation of three B-splines of order six that is

(a) Quintic Hyperbolic B-spline with DQM
(b) Quintic Uniform Algebraic Hyperbolic tension B-spline with DQM
(c) Quintic Uniform Algebraic trigonometric tension B-spline with DQM

With the implementation of DQM, the system of equation got converted into ODEs, then calculated by using the SSP Runge-Kutta method. Also obtained approximations exhibit the same behavior as in the literature and are graphically displayed at different time intervals. Moreover, solutions that are obtained numerically are matched with the precise solution and are in excellent agreement with them. To assess the applicability of the suggested approach, five different test problems in one as well as in two dimensions are resolved. Additionally, it is demonstrated that the approach is invariably stable.

Chapter 5

Conclusion

This thesis emphasizes depicting the computational numerical elucidation of partial differential equations that are nonlinear in nature by evolving the new regime of sixth-order with differential quadrature method. In this thesis, three novel techniques which are developed to get the numerical approximation of PDEs are:

(a) Quintic Hyperbolic B-spline with DQM
(b) Quintic Uniform Algebraic Hyperbolic tension B-spline with DQM
(c) Quintic Uniform Algebraic Trigonometric tension B-spline with DQM

Firstly, the above mentioned B-splines with DQM are utilised to compute the weighting factors. The basis function then transforms the initial boundary value issue into an ODEs after computing the weighting coefficients. This equation is then solved using the strong preserving Runge-Kutta 43 regime. Exactness and modification for suggested approach are then verified by numerically resolving a variety of test problems and calculating the L_2 and L_∞ error norms. For quick and simple access, calculated results are shown visually and in tabular format.

Quintic B-spline curves are commonly employed in computer graphics and CAD applications because they create smooth and continuous curves. Additionally, they calculate quickly and are simple to modify using computer techniques.

Hyperbolic B-spline functions produce numerical approximations that are piecewise continuous and in closed form. UAT and UAH tension B-splines are created over a space covered by hybrid functions like algebraic and trigonometric or hyperbolic and algebraic. According to the comparison of numerical findings, each of these strategies reduces the computational complexity.

The equations which are solved for the numerical solution have huge application in several scientific and technical domains. The proposed approaches are used to elucidate three PDEs that are nonlinear in nature named Coupled Korteweg-de Vries equation (CKdV), Kuramoto-Sivashinsky (KS) equation, and Extended Fisher-Kolmogorov (EFK) equation.

Chapter 1 contains, brief introduction about the proposed work including literature survey, and the formulas of used splines, errors, introduction about stability.

Chapter 2 demonstrated the attained solution of CKdV equation numerically using the three proposed B-spline functions with DQM. Method's efficiency is demonstrated by applying it on N-soliton solution of CKdV equation which are further subdivided into three sections (single soliton, double solitons, and three soliton interaction). And in other example birth of solitons of CKdV is discussed. The numerical solutions are successfully obtained and are represented graphically in the figures for all the test problems, which are quite accurate, and in good agreement with the exact solution or with the results present in the literature. Also for the accuracy of the proposed scheme error norms L_2 and L_∞ and preserved quantities are calculated which are coming out to be reasonably small In conclusion, the CKdV system of equations as well as a variety of PDEs may be elucidate using the current approach, which has been proven to be efficient and successful.

Chapter 3 is devoted to solve KS equation numerically by employing the proposed method. All three splines with DQM are applied to solve KS equation numerically. For efficiency check of the suggested approach, four different examples are solved to obtain KS equation's numerical solution. The derived approximations demonstrate the same behaviour as described in the literature and are perfectly matched to the exact solution where available. To evaluate the applicability of the recommended approach, different error norms, convergence order and also global relative error is calculated.

Chapter 4 is devoted to solve EFK equation numerically in 1D and 2D with the implementation of three different quintic B-splines with DQM as test function. There are five separate test problems that are solved numerically for efficiency and accuracy check of the presented approach. The derived approximations demonstrate the same behaviour as described in the previous work by another researchers and are perfectly matched with the precise solution where available. Furthermore, it is shown that the method is consistently stable.

However, it is important to note that the choice of numerical method and basis function depends on the specific characteristic of the problem we are trying to solve, so careful consideration of the problem is essential. Also, to solve the PDEs of sixth or higher order, quantic B-spline may not be suitable, we need higher order B-spline for this purpose. While DQM with quintic has its own advantages as it give us more accurate approximations than lower degree B-splines, provide smooth and continuous representation of functions, reduces numerical dispersion, use of higher order B-splines leads to faster convergence of numerical methods, DQM with presented quintic B-splines can be applied to a wide range of problems, and is relatively easy to implement compared to some other numerical techniques for solving PDEs, making it aacessible to a wider range of researchers and engineers.

In conclusion, it is envisaged that the current numerical approach will be useful for researchers in the field in order to discover highly accurate numerical solutions to additional significant partial differential equations that arise in numerous scientific branches.

Future Scope

- The suggested approach can also be applicable to obtain the solution of fractional order PDEs.
- With the help of suggested scheme solution of integro-differential equations can also be approximated.

- The proposed scheme can also be applicable for the solution of higher order 3D partial differential equations.

REFERENCES

1. Shu, C. (2000). Differential quadrature and its application in engineering. Springer Science & Business Media.
2. Shu, C., & Richards, B. E. (1990). High resolution of natural convection in a square cavity by generalized differential quadrature. In *Proceedings of the 3rd international conference on advances in numeric methods in engineering: theory and application, Swansea, UK*, *2*, 978-985.
3. Bellman, R., Kashef, B. G., & Casti, J. (1972). Differential quadrature: a technique for the rapid solution of nonlinear partial differential equations. *Journal of computational physics*, *10*(1), 40-52.
4. Quan, J. R., & Chang, C. T. (1989). New insights in solving distributed system equations by the quadrature method—I. Analysis. *Computers & Chemical Engineering*, *13*(7), 779-788.
5. Quan, J. R., & Chang, C. T. (1989). New insights in solving distributed system equations by the quadrature method—II. Numerical experiments. *Computers & Chemical Engineering*, *13*(9), 1017-1024.
6. Başhan, A., & Yağmurlu, N. M. (2022). A mixed method approach to the solitary wave, undular bore and boundary-forced solutions of the Regularized Long Wave equation. *Computational and Applied Mathematics*, *41*(4), 169.
7. Zhong, H. (2004). Spline-based differential quadrature for fourth order differential equations and its application to Kirchhoff plates. *Applied Mathematical Modelling*, *28*(4), 353-366.
8. Başhan, A. (2020). An effective approximation to the dispersive soliton solutions of the coupled KdV equation via combination of two efficient methods. *Computational and Applied Mathematics*, *39*(2), 80.
9. Başhan, A., Murat Yağmurlu, N., Uçar, Y., & Esen, A. (2021). A new perspective for the numerical solution of the modified equal width wave equation. *Mathematical Methods in the Applied Sciences*, *44*(11), 8925-8939.

10. Başhan, A. (2021). Modification of quintic B-spline differential quadrature method to nonlinear Korteweg-de Vries equation and numerical experiments. *Applied Numerical Mathematics, 167*, 356-374.
11. Başhan, A. (2022). Nonlinear dynamics of the Burgers' equation and numerical experiments. *Mathematical Sciences, 16*(2), 183-205.
12. Korkmaz, A., & Dag, I. (2011). Polynomial based differential quadrature method for numerical solution of nonlinear Burgers' equation. *Journal of the Franklin Institute, 348*(10), 2863-2875.
13. Başhan, A. (2019). A mixed methods approach to Schrödinger equation: Finite difference method and quartic B-spline based differential quadrature method. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA), 9*(2), 223-235.
14. Mittal, R. C., & Dahiya, S. (2016). A study of quintic B-spline based differential quadrature method for a class of semi-linear Fisher-Kolmogorov equations. *Alexandria Engineering Journal, 55*(3), 2893-2899.
15. Arora, G., & Joshi, V. (2018). A computational approach using modified trigonometric cubic B-spline for numerical solution of Burgers' equation in one and two dimensions. *Alexandria engineering journal, 57*(2), 1087-1098.
16. Singh, B. K., & Kumar, P. (2018). An algorithm based on a new DQM with modified extended cubic B-splines for numerical study of two dimensional hyperbolic telegraph equation. *Alexandria engineering journal*, *57*(1), 175-191.
17. Mohammed, O. H., & Saeed, M. A. (2019). Numerical solution of thin plates problem via differential quadrature method using G-spline. *Journal of King Saud University-Science, 31*(2), 209-214.
18. Korkmaz, A., Aksoy, A. M., & Dag, I. (2011). Quartic B-spline differential quadrature method. *Int. J. Nonlinear Sci, 11*(4), 403-411.
19. Arora, G., & Singh, B. K. (2013). Numerical solution of Burgers' equation with modified cubic B-spline differential quadrature method. *Applied Mathematics and Computation, 224*, 166-177.

20. Korkmaz, A., & Dağ, İ. (2012). Cubic B-spline differential quadrature methods for the advection-diffusion equation. *International Journal of Numerical Methods for Heat & Fluid Flow*, *22*(8), 1021-1036.
21. Tamsir, M., Srivastava, V. K., & Jiwari, R. (2016). An algorithm based on exponential modified cubic B-spline differential quadrature method for nonlinear Burgers' equation. *Applied Mathematics and Computation*, *290*, 111-124.
22. Başhan, A. (2019). An efficient approximation to numerical solutions for the Kawahara equation via modified cubic B-spline differential quadrature method. *Mediterranean Journal of Mathematics*, *16*(1), 14.
23. Tamsir, M., Dhiman, N., & Srivastava, V. K. (2018). Cubic trigonometric B-spline differential quadrature method for numerical treatment of Fisher's reaction-diffusion equations. *Alexandria engineering journal*, *57*(3), 2019-2026.
24. Başhan, A., Uçar, Y., Murat Yağmurlu, N., & Esen, A. (2018). A new perspective for quintic B-spline based Crank-Nicolson-differential quadrature method algorithm for numerical solutions of the nonlinear Schrödinger equation. *The European Physical Journal Plus*, *133*(1), 12.
25. Mittal, R. C., & Dahiya, S. (2017). A quintic B-spline based differential quadrature method for numerical solution of Kuramoto-Sivashinsky equation. *International Journal of Nonlinear Sciences and Numerical Simulation*, *18*(2), 103-114.
26. Joshi, P., Pathak, M., & Lin, J. (2023). Numerical study of generalized 2-D nonlinear Benjamin–Bona–Mahony–Burgers equation using modified cubic B-spline differential quadrature method. *Alexandria Engineering Journal*, *67*, 409-424.
27. Barrera, D., González, P., Ibáñez, F., & Ibáñez, M. J. (2015). A general spline differential quadrature method based on quasi-interpolation. *Journal of Computational and Applied Mathematics*, *275*, 465-479.
28. Joshi, V., & Kapoor, M. (2021). A Novel Technique for Numerical Approximation of 2 Dimensional Non-Linear Coupled Burgers' Equations using Uniform Algebraic Hyperbolic (UAH) Tension B-Spline based Differential Quadrature Method. *Appl. Math. Inf. Sci*, *15*, 217-239.

29. Kaur, A., & Kanwar, V. (2022). Numerical Solution of Generalized Kuramoto–Sivashinsky Equation Using Cubic Trigonometric B-Spline Based Differential Quadrature Method and One-Step Optimized Hybrid Block Method. *International Journal of Applied and Computational Mathematics*, *8*, 1-19.
30. Zhu, X. G., Nie, Y. F., & Zhang, W. (2017). An efficient differential quadrature method for fractional advection–diffusion equation. *Nonlinear Dynamics*, *90*, 1807-1827.
31. Tamsir, M., Meetei, M. Z., & Msmali, A. H. (2022). Hyperbolic B-Spline Function-Based Differential Quadrature Method for the Approximation of 3D Wave Equations. *Axioms*, *11*(11), 597.
32. Başhan, A., Karakoç, S. B. G., & Geyikli, T. (2015). Approximation of the KdVB equation by the quintic B-spline differential quadrature method.
33. Korkmaz, A., & Dağ, I. (2016). Quartic and quintic B-spline methods for advection–diffusion equation. *Applied Mathematics and Computation*, *274*, 208-219.
34. Thoudam, R. (2017). Numerical solutions of coupled Klein–Gordon–Zakharov equations by quintic B-spline differential quadrature method. *Applied Mathematics and Computation*, *307*, 50-61.
35. Wang, X. (2015). Differential quadrature and differential quadrature based element methods: theory and applications. Butterworth-Heinemann.
36. Schoenberg, I. J. (1946). Contributions to the problem of approximation of equidistant data by analytic functions. *Quarterly of Applied Mathematics*, *4*(2), 45-99.
37. Schoenberg, I. J. (1964). On trigonometric spline interpolation. *Journal of mathematics and mechanics*, 795-825.
38. De Boor, C., & De Boor, C. (1978). *A practical guide to splines* (Vol. 27, p. 325). New York: springer-verlag.
39. Prenter, P. M. (2008). *Splines and variational methods*. Courier Corporation.
40. Caglar, H., Caglar, N., Özer, M., Valarıstos, A., & Anagnostopoulos, A. N. (2010). B-spline method for solving Bratu's problem. *International Journal of Computer Mathematics*, *87*(8), 1885-1891.

41. Caglar, H. N., Caglar, S. H., & Twizell, E. H. (1999). The numerical solution of fifth-order boundary value problems with sixth-degree B-spline functions. *Applied Mathematics Letters*, *12*(5), 25-30.
42. Lakestani, M., & Dehghan, M. (2009). Numerical solution of Fokker-Planck equation using the cubic B-spline scaling functions. *Numerical Methods for Partial Differential Equations: An International Journal*, *25*(2), 418-429.
43. Shiralashetti, S. C., & Kumbinarasaiah, S. (2018). Cardinal b-spline wavelet based numerical method for the solution of generalized Burgers–Huxley equation. *International journal of applied and computational mathematics*, *4*, 1-13.
44. Goh, J., Majid, A. A., & Ismail, A. I. M. (2011). Numerical method using cubic B-spline for the heat and wave equation. *Computers & Mathematics with Applications*, *62*(12), 4492-4498.
45. Roul, P. (2020). A fourth order numerical method based on B-spline functions for pricing Asian options. *Computers & Mathematics with Applications*, *80*(3), 504-521.
46. Lakestani, M., & Dehghan, M. (2012). Numerical solutions of the generalized Kuramoto–Sivashinsky equation using B-spline functions. *Applied Mathematical Modelling*, *36*(2), 605-617.
47. Aksan, E. N. (2006). Quadratic B-spline finite element method for numerical solution of the Burgers' equation. *Applied Mathematics and Computation*, *174*(2), 884-896.
48. Abbas, M., Majid, A. A., Ismail, A. I. M., & Rashid, A. (2014). The application of cubic trigonometric B-spline to the numerical solution of the hyperbolic problems. *Applied Mathematics and Computation*, *239*, 74-88.
49. Lakestani, M., & Dehghan, M. (2013). Four techniques based on the B-spline expansion and the collocation approach for the numerical solution of the Lane–Emden equation. *Mathematical Methods in the Applied Sciences*, *36*(16), 2243-2253.
50. Dağ, İ., Şahin, A., & Korkmaz, A. (2010). Numerical investigation of the solution of Fisher's equation via the B-spline Galerkin method. *Numerical Methods for Partial Differential Equations*, *26*(6), 1483-1503.

51. Dağ, İ., Saka, B., & Boz, A. (2005). B-spline Galerkin methods for numerical solutions of the Burgers' equation. *Applied Mathematics and Computation*, *166*(3), 506-522.
52. Shukla, H. S., Tamsir, M., & Srivastava, V. K. (2015). Numerical simulation of two dimensional sine-Gordon solitons using modified cubic B-spline differential quadrature method. *AIP Advances*, *5*(1), 017121.
53. Bashan, A., Yagmurlu, N. M., Ucar, Y., & Esen, A. (2017). An effective approach to numerical soliton solutions for the Schrödinger equation via modified cubic B-spline differential quadrature method. *Chaos, Solitons & Fractals*, *100*, 45-56.
54. Nouisser, O., Driss Sbibih, and Paul Sablonnière. "A family of spline quasi-interpolants on the sphere." *Numerical Algorithms* 33 (2003): 399-413.
55. Maes, J., & Bultheel, A. (2007). Modeling sphere-like manifolds with spherical Powell–Sabin B-splines. *Computer aided geometric design*, *24*(2), 79-89.
56. Eddargani, S., Lamnii, A., Lamnii, M., Sbibih, D., & Zidna, A. (2019). Algebraic hyperbolic spline quasi-interpolants and applications. *Journal of Computational and Applied Mathematics*, *347*, 196-209.
57. Jha, N., & Mohanty, R. K. (2014). Quintic hyperbolic nonpolynomial spline and finite difference method for nonlinear second order differential equations and its application. *Journal of the Egyptian Mathematical Society*, *22*(1), 115-122.
58. Roman, F., Manni, C., & Speleers, H. (2017). Numerical approximation of GB-splines by a convolutional approach. *Applied Numerical Mathematics*, *116*, 273-285.
59. Shen, W., Wang, G., & Yin, P. (2013). Explicit representations of changeable degree spline basis functions. *Journal of computational and applied mathematics*, *238*, 39-50.
60. Shen, W., Yin, P., & Tan, C. (2016). Degree elevation of changeable degree spline. *Journal of Computational and Applied Mathematics*, *300*, 56-67.
61. Lü, Y., Wang, G., & Yang, X. (2002). Uniform hyperbolic polynomial B-spline curves. *Computer Aided Geometric Design*, *19*(6), 379-393.

62. Patel, V., Bhattacharjee, S. S., & George, N. V. (2021). A family of logarithmic hyperbolic cosine spline nonlinear adaptive filters. *Applied Acoustics*, *178*, 107973.
63. Campagna, R., & Conti, C. (2021). Penalized hyperbolic-polynomial splines. *Applied Mathematics Letters*, *118*, 107159.
64. Wang, G. (2008). Unified and extended form of three types of splines. *Journal of computational and applied mathematics*, *216*(2), 498-508.
65. Siddiqi, S. S., us Salam, W., & Rehan, K. (2016). Hyperbolic forms of ternary non-stationary subdivision schemes originated from hyperbolic B-splines. *Journal of Computational and Applied Mathematics*, *301*, 16-27.
66. Siddiqi, S. S., us Salam, W., & Rehan, K. (2016). Construction of binary four and five point non-stationary subdivision schemes from hyperbolic B-splines. *Applied Mathematics and Computation*, *280*, 30-38.
67. Xumin, L., Weixiang, X., Yong, G., & Yuanyuan, S. (2010). Hyperbolic polynomial uniform B-spline curves and surfaces with shape parameter. *Graphical Models*, *72*(1), 1-6.
68. Zhang, J., & Krause, F. L. (2005). Extending cubic uniform B-splines by unified trigonometric and hyperbolic basis. *Graphical Models*, *67*(2), 100-119.
69. Zhang, J. (1996). C-curves: an extension of cubic curves. *Computer Aided Geometric Design*, *13*(3), 199-217.
70. Zhang, J. (1997). Two different forms of CB-splines. *Computer aided geometric design*, *14*(1), 31-41.
71. Koch, P. E., & Lyche, T. (1991). Construction of exponential tension B-splines of arbitrary order. In *Curves and surfaces* (pp. 255-258). Academic Press.
72. Mainar, E., & Peña, J. M. (2002). A basis of C-Bézier splines with optimal properties. *Computer Aided Geometric Design*, *19*(4), 291-295.
73. Chen, Q., & Wang, G. (2003). A class of Bézier-like curves. *Computer aided geometric design*, *20*(1), 29-39.
74. Wang, G., Chen, Q., & Zhou, M. (2004). NUAT B-spline curves. *Computer Aided Geometric Design*, *21*(2), 193-205.

75. Jena, M. K., Shunmugaraj, P., & Das, P. C. (2002). A subdivision algorithm for trigonometric spline curves. *Computer Aided Geometric Design*, *19*(1), 71-88.
76. Jena, M. K., Shunmugaraj, P., & Das, P. C. (2003). A non-stationary subdivision scheme for generalizing trigonometric spline surfaces to arbitrary meshes. *Computer aided geometric design*, *20*(2), 61-77.
77. Ya-Juan, L., & Guo-Zhao, W. (2005). Two kinds of B-basis of the algebraic hyperbolic space. *Journal of Zhejiang University-SCIENCE A*, *6*(7), 750-759.
78. Xu, G., & Wang, G. Z. (2007). AHT Bézier curves and NUAHT B-spline curves. *Journal of Computer Science and Technology*, *22*(4), 597-607.
79. Alinia, N., & Zarebnia, M. (2018). A new tension B-spline method for third-order self-adjoint singularly perturbed boundary value problems. *Journal of Computational and Applied Mathematics*, *342*, 521-533.
80. Fang, M. E., Ma, W., & Wang, G. (2010). A generalized curve subdivision scheme of arbitrary order with a tension parameter. *Computer Aided Geometric Design*, *27*(9), 720-733.
81. Fang, M. E., Ma, W., & Wang, G. (2014). A generalized surface subdivision scheme of arbitrary order with a tension parameter. *Computer-Aided Design*, *49*, 8-17.
82. Alinia, N., & Zarebnia, M. (2019). A numerical algorithm based on a new kind of tension B-spline function for solving Burgers-Huxley equation. *Numerical Algorithms*, *82*, 1121-1142.
83. Gottlieb, S. (2005). On high order strong stability preserving Runge-Kutta and multi step time discretizations. *Journal of scientific computing*, *25*, 105-128.
84. Strikwerda, J. C. (2004). *Finite difference schemes and partial differential equations*. Society for Industrial and Applied Mathematics.
85. Strang, G. (1964). Accurate partial difference methods: II. Non-linear problems. *Numerische Mathematik*, *6*(1), 37-46.
86. Shu, C. W. (1988). Total-variation-diminishing time discretizations. *SIAM Journal on Scientific and Statistical Computing*, *9*(6), 1073-1084.

87. Shu, C. W., & Osher, S. (1988). Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of computational physics*, *77*(2), 439-471.
88. Gottlieb, S., & Gottlieb, L. A. J. (2003). Strong stability preserving properties of Runge–Kutta time discretization methods for linear constant coefficient operators. Journal of Scientific Computing, 18, 83-109.
89. Gottlieb, S., & Shu, C. W. (1998). Total variation diminishing Runge-Kutta schemes. Mathematics of computation, 67(221), 73-85.
90. Hundsdorfer, W., Ruuth, S. J., & Spiteri, R. J. (2003). Monotonicity-preserving linear multistep methods. SIAM Journal on Numerical Analysis, 41(2), 605-623.
91. Ruuth, S. J., & Spiteri, R. J. (2002). Two barriers on strong-stability-preserving time discretization methods. Journal of Scientific Computing, 17, 211-220.
92. Ruuth, S. J., & Spiteri, R. J. (2004). High-order strong-stability-preserving Runge--Kutta methods with downwind-biased spatial discretizations. SIAM Journal on Numerical Analysis, 42(3), 974-996.
93. Ruuth, S. (2006). Global optimization of explicit strong-stability-preserving Runge-Kutta methods. Mathematics of Computation, 75(253), 183-207.
94. Spiteri, R. J., & Ruuth, S. J. (2002). A new class of optimal high-order strong-stability-preserving time discretization methods. *SIAM Journal on Numerical Analysis*, *40*(2), 469-491.
95. Singh, B. K., & Kumar, P. (2016). A novel approach for numerical computation of Burgers' equation in (1+ 1) and (2+ 1) dimensions. *Alexandria Engineering Journal*, *55*(4), 3331-3344.
96. Iserles, A. (1985). Numerical solution of differential equations , by MK Jain. Pp 698.£ 17· 95. 1984. ISBN 0-85226-432-1 (Wiley Eastern). *The Mathematical Gazette*, *69*(449), 236-237.
97. Jain, M. K. (1983). Numerical solution of differential equations, 2nd ed., *Wiley, New York*, NY.

98. Djidjeli, K., Price, W. G., & Twizell, E. H. (1995). Numerical solutions of a damped sine-Gordon equation in two space variables. *Journal of Engineering Mathematics*, *29*(4), 347-369.
99. Natiq, H., Said, M. R. M., Ariffin, M. R. K., He, S., Rondoni, L., & Banerjee, S. (2018). Self-excited and hidden attractors in a novel chaotic system with complicated multistability. *The European Physical Journal Plus*, *133*, 1-12.
100. Bashan, A., Yagmurlu, N. M., Ucar, Y., & Esen, A. (2017). An effective approach to numerical soliton solutions for the Schrödinger equation via modified cubic B-spline differential quadrature method. *Chaos, Solitons & Fractals*, *100*, 45-56.
101. Argyris, J., Haase, M., & Heinrich, J. C. (1991). Finite element approximation to two-dimensional sine-Gordon solitons. *Computer methods in applied mechanics and engineering*, *86*(1), 1-26.
102. Osborne, A. R. (1995). The inverse scattering transform: tools for the nonlinear Fourier analysis and filtering of ocean surface waves. *Chaos, Solitons & Fractals*, *5*(12), 2623-2637.
103. Ostrovsky, L. A., & Stepanyants, Y. A. (1989). Do internal solitions exist in the ocean?. *Reviews of Geophysics*, *27*(3), 293-310.
104. Das, G. C., & Sarma, J. (1999). Response to "Comment on 'A new mathematical approach for finding the solitary waves in dusty plasma'"[Phys. Plasmas 6, 4392 (1999)]. *Physics of Plasmas*, *6*(11), 4394-4397.
105. Gao, Y. T., & Tian, B. (2001). Ion-acoustic shocks in space and laboratory dusty plasmas: Two-dimensional and non-traveling-wave observable effects. *Physics of Plasmas*, *8*(7), 3146-3149.
106. Chen, C. O. K., & Ho, S. H. (1999). Solving partial differential equations by two-dimensional differential transform method. *Applied Mathematics and computation*, *106*(2-3), 171-179.
107. Zhu, S. (1999). A difference scheme for the coupled KdV equation. *Communications in Nonlinear Science and Numerical Simulation*, *4*(1), 60-63.

108. Boussinesq, J. (1872). Théorie des ondes et des remous qui se propagent le long d'un canal rectangulaire horizontal, en communiquant au liquide contenu dans ce canal des vitesses sensiblement pareilles de la surface au fond. *Journal de mathématiques pures et appliquées*, *17*, 55-108.
109. Boussinesq, J. (1877). *Essai sur la théorie des eaux courantes*. Impr. Nationale, 1-680.
110. Rayleigh, L. (1876). On waves. *Phil. Mag.*, *1*, 257-259.
111. Tam, H. W., Ma, W. X., Hu, X. B., & Wang, D. L. (2000). The Hirota-Satsuma coupled KdV equation and a coupled Ito system revisited. *Journal of the Physical Society of Japan*, *69*(1), 45-52.
112. Kaya, D., & Inan, I. E. (2004). Exact and numerical traveling wave solutions for nonlinear coupled equations using symbolic computation. *Applied mathematics and computation*, *151*(3), 775-787.
113. Fan, E. G. (2002). Traveling wave solutions for nonlinear equations using symbolic computation. *Computers & Mathematics with Applications*, *43*(6-7), 671-680.
114. Korteweg, D. J., & De Vries, G. (1895). XLI. On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, *39*(240), 422-443.
115. Hirota, R., & Satsuma, J. (1981). Soliton solutions of a coupled Korteweg-de Vries equation. *Physics Letters A*, *85*(8-9), 407-408.
116. Halim, A. A., Kshevetskii, S. P., & Leble, S. B. (2003). Numerical integration of a coupled Korteweg-de Vries system. *Computers & Mathematics with Applications*, *45*(4-5), 581-591.
117. Halim, A. A., & Leble, S. B. (2004). Analytical and numerical solution of a coupled KdV–MKdV system. *Chaos, Solitons & Fractals*, *19*(1), 99-108.
118. De Jager, E. M. (2006). On the origin of the Korteweg-de Vries equation. *arXiv preprint math/0602661*.

119. Lu, H., & Wang, M. (1999). Exact soliton solutions of some nonlinear physical models. *Physics Letters A*, *255*(4-6), 249-252.
120. Tian, B., & Gao, Y. T. (2001). A System of the Coupled Korteweg-de Vries Equations and Computerized Symbolic Computation. *International Journal of Modern Physics C*, *12*(03), 361-366.
121. Cao, D. B., Yan, J. R., & Zhang, Y. (2002). Exact solutions for a new coupled MKdV equations and a coupled KdV equations. *Physics Letters A*, *297*(1-2), 68-74.
122. Zayed, E. M. E., Zedan, H. A., & Gepreel, K. A. (2004). On the solitary wave solutions for nonlinear Hirota–Satsuma coupled KdV of equations. *Chaos, Solitons & Fractals*, *22*(2), 285-303.
123. Zhang, J. L., Wang, M. L., Wang, Y. M., & Fang, Z. D. (2006). The improved F-expansion method and its applications. *Physics Letters A*, *350*(1-2), 103-109.
124. Inan, I. E. (2007). Exact solutions for coupled KdV equation and KdV equations. *Physics Letters A*, *371*(1-2), 90-95.
125. Ghany, H. A. (2011). Exact solutions for stochastic generalized Hirota-Satsuma coupled KdV equations. *Chinese Journal of Physics*, *49*(4), 926-940.
126. Ma, Z. Y., & Zhu, J. M. (2007). Jacobian elliptic function expansion solutions for the Wick-type stochastic coupled KdV equations. *Chaos, Solitons & Fractals*, *32*(5), 1679-1685.
127. Assas, L. M. (2008). Variational iteration method for solving coupled-KdV equations. *Chaos, Solitons & Fractals*, *38*(4), 1225-1228.
128. Ganji, D. D., & Rafei, M. (2006). Solitary wave solutions for a generalized Hirota–Satsuma coupled KdV equation by homotopy perturbation method. *Physics Letters A*, *356*(2), 131-137.
129. Xu, Y., & Shu, C. W. (2006). Local discontinuous Galerkin methods for the Kuramoto–Sivashinsky equations and the Ito-type coupled KdV equations. *Computer methods in applied mechanics and engineering*, *195*(25-28), 3430-3447.

130. Rashid, A. (2007). Numerical solution of Korteweg-de Vries equation by the Fourier pseudospectral method. *Bulletin of the Belgian Mathematical Society-Simon Stevin*, *14*(4), 709-721.
131. Soliman, A. A., Ali, A. H. A., & Raslan, K. R. (2009). Numerical solution for the KdV equation based on similarity reductions. *Applied Mathematical Modelling*, *33*(2), 1107-1115.
132. Ismail, M. S. (2009). Numerical solution of a coupled Korteweg–de Vries equations by collocation method. *Numerical Methods for Partial Differential Equations: An International Journal*, *25*(2), 275-291.
133. Ali, A. T., Khater, M. M., Attia, R. A., Abdel-Aty, A. H., & Lu, D. (2020). Abundant numerical and analytical solutions of the generalized formula of Hirota-Satsuma coupled KdV system. *Chaos, Solitons & Fractals*, *131*, 109473.
134. Alquran, M., Jaradat, I., & Baleanu, D. (2019). Shapes and dynamics of dual-mode Hirota–Satsuma coupled KdV equations: exact traveling wave solutions and analysis. *Chinese Journal of Physics*, *58*, 49-56.
135. Gokdogan, A., Yildirim, A., & Merdan, M. (2012). Solving coupled-KdV equations by differential transformation method. *World Appl. Sci. J*, *19*(12), 1823-1828.
136. Jibran, M., Nawaz, R., Khan, A., & Afzal, S. (2018). Iterative solutions of Hirota Satsuma coupled KDV and modified coupled KDV systems. *Mathematical Problems in Engineering*, *2018*, 1-18.
137. Ismail, M. S., & Ashi, H. A. (2014). A numerical solution for Hirota-Satsuma coupled KdV equation. *Abstract and Applied Analysis*, *2014*, 1-9.
138. Abdelrahman, M. A., Almatrafi, M. B., & Alharbi, A. (2020). Fundamental solutions for the coupled KdV system and its stability. *Symmetry*, *12*(3), 429.
139. Akinyemi, L., & Huseen, S. N. (2020). A powerful approach to study the new modified coupled Korteweg–de Vries system. *Mathematics and Computers in Simulation*, *177*, 556-567.

140. Başhan, A. (2019). A mixed algorithm for numerical computation of soliton solutions of the coupled KdV equation: Finite difference method and differential quadrature method. *Applied Mathematics and Computation*, *360*, 42-57.
141. Oruç, Ö., Bulut, F., & Esen, A. (2017). A numerical treatment based on Haar wavelets for coupled KdV equation. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, *7*(2), 195-204.
142. Raslan, K. R., El-Danaf, T. S., & Ali, K. K. (2016). Collocation method with quintic B-spline method for solving Hirota–Satsuma coupled KDV equation. *International Journal of Applied Mathematical Research*, *5*(2), 123-131.
143. Kutluay, S., & Ucar, Y. (2013). A quadratic B-spline Galerkin approach for solving a coupled KdV equation. *Mathematical Modelling and Analysis*, *18*(1), 103-121.
144. Abbasbandy, S. (2007). The application of homotopy analysis method to solve a generalized Hirota–Satsuma coupled KdV equation. *Physics Letters A*, *361*(6), 478-483.
145. Ito, M. (1982). Symmetries and conservation laws of a coupled nonlinear wave equation. *Physics Letters A*, *91*(7), 335-338.
146. Kaur, N., & Joshi, V. (2020). Numerical Solution of Gray Scott Reaction-Diffusion Equation using LagrangePolynomial. *Journal of Physics: Conference Series*, *1531(1)*, 012058.
147. Guo, Y., Cao, X., Liu, B., & Gao, M. (2020). Solving partial differential equations using deep learning and physical constraints. *Applied Sciences*, *10*(17), 5917.
148. Ma, W. (2022). Riemann-Hilbert problems and soliton solutions of nonlocal reverse-time NLS hierarchies. *Acta Mathematica Scientia*, *42*(1), 127-140.
149. Ma, W. X. (2021). Nonlocal PT-symmetric integrable equations and related Riemann–Hilbert problems. *Partial Differential Equations in Applied Mathematics*, *4*, 100190.
150. Ma, W. X. (2022). Riemann–Hilbert problems and soliton solutions of type $(\lambda*,-\lambda*)$ reduced nonlocal integrable mKdV hierarchies. *Mathematics*, *10*(6), 870.

151. Ma, W. X., Wu, H., & He, J. (2007). Partial differential equations possessing Frobenius integrable decompositions. *Physics Letters A*, *364*(1), 29-32.
152. Wazwaz, A. M. (2011). Integrability of coupled KdV equations. *Central European Journal of Physics*, *9*, 835-840.
153. Jian-Ping, W., Xian-Guo, G., & Xiao-Lin, Z. (2009). N-soliton solution of a generalized hirota–satsuma coupled KdV equation and its reduction. *Chinese Physics Letters*, *26*(2), 020202.
154. Capetillo, P., & Hornewall, J. (2021). Introduction to the Hirota Direct Method.
155. Jiwari, R., Pandit, S., & Mittal, R. C. (2012). Numerical simulation of two-dimensional sine-Gordon solitons by differential quadrature method. *Computer Physics Communications*, *183*(3), 600-616.
156. Sivashinsky, G. I. (1977). Nonlinear analysis of hydrodynamic instability in laminar flames—I. Derivation of basic equations. *Acta astronautica*, *4*(11), 1177-1206.
157. Kuramoto, Y. (1978). Diffusion-induced chaos in reaction systems. *Progress of Theoretical Physics Supplement*, *64*, 346-367.
158. Rademacher, J. D., & Wittenberg, R. W. (2006). Viscous shocks in the destabilized Kuramoto-Sivashinsky equation. *Journal of computational and Nonlinear Dynamics*, *1*, 336-347.
159. Hyman, J. M., & Nicolaenko, B. (1986). The Kuramoto-Sivashinsky equation: a bridge between PDE's and dynamical systems. *Physica D: Nonlinear Phenomena*, *18*(1-3), 113-126.
160. Michelson, D. (1986). Steady solutions of the Kuramoto-Sivashinsky equation. *Physica D: Nonlinear Phenomena*, *19*(1), 89-111.
161. Hooper, A. P., & Grimshaw, R. (1985). Nonlinear instability at the interface between two viscous fluids. *The Physics of fluids*, *28*(1), 37-45.
162. Sivashinsky, G. I. (1983). Instabilities, pattern formation, and turbulence in flames. *Annual Review of Fluid Mechanics*, *15*(1), 179-199.

163. Kuramoto, Y., & Tsuzuki, T. (1976). Persistent propagation of concentration waves in dissipative media far from thermal equilibrium. *Progress of theoretical physics*, *55*(2), 356-369.
164. Saprykin, S., Demekhin, E. A., & Kalliadasis, S. (2005). Two-dimensional wave dynamics in thin films. I. Stationary solitary pulses. *Physics of fluids*, *17*(11), 117105.
165. Conte, R. (2003). Exact solutions of nonlinear partial differential equations by singularity analysis. In *Direct and inverse methods in nonlinear evolution equations*, *632*, 1-83. Springer, Berlin, Heidelberg.
166. Manickam, A. V., Moudgalya, K. M., & Pani, A. K. (1998). Second-order splitting combined with orthogonal cubic spline collocation method for the Kuramoto-Sivashinsky equation. *Computers & Mathematics with Applications*, *35*(6), 5-25.
167. Akrivis, G., & Smyrlis, Y. S. (2004). Implicit–explicit BDF methods for the Kuramoto–Sivashinsky equation. *Applied numerical mathematics*, *51*(2-3), 151-169.
168. Lopez-Marcos, M. A. (1994). Numerical analysis of pseudospectral methods for the Kuramoto-Sivashinsky equation. *IMA Journal of Numerical Analysis*, *14*(2), 233-242.
169. Hooper, A. P., & Grimshaw, R. (1988). Travelling wave solutions of the Kuramoto-Sivashinsky equation. *Wave motion*, *10*(5), 405-420.
170. Yang, T. S. (1997). On traveling-wave solutions of the Kuramoto-Sivashinsky equation. *Physica D: Nonlinear Phenomena*, *110*(1-2), 25-42.
171. Cueto-Felgueroso, L., & Peraire, J. (2008). A time-adaptive finite volume method for the Cahn–Hilliard and Kuramoto–Sivashinsky equations. *Journal of Computational Physics*, *227*(24), 9985-10017.
172. Daumont, I., Kassner, K., Misbah, C., & Valance, A. (1997). Cellular self-propulsion of two-dimensional dissipative structuresand spatial-period tripling Hopf bifurcation. *Physical Review E*, *55*(6), 6902.
173. Paniconi, M., & Elder, K. R. (1997). Stationary, dynamical, and chaotic states of the two-dimensional damped Kuramoto-Sivashinsky equation. *Physical Review E*, *56*(3), 2713-2721.

174. Cohen, B. I., Krommes, J. A., Tang, W. M., & Rosenbluth, M. N. (1976). Non-linear saturation of the dissipative trapped-ion mode by mode coupling. *Nuclear fusion*, *16*(6), 971-972.
175. Cuerno, R., & Barabási, A. L. (1995). Dynamic scaling of ion-sputtered surfaces. *Physical review letters*, *74*(23), 4746-47469.
176. Sivashinsky, G. I. (1980). On flame propagation under conditions of stoichiometry. *SIAM Journal on Applied Mathematics*, *39*(1), 67-82.
177. Grimshaw, R., & Hooper, A. (1991). The non-existence of a certain class of travelling wave solutions of the Kuramoto-Sivashinsky equation. *Physica D: Nonlinear Phenomena*, *50*(2), 231-238.
178. Triki, H., Taha, T. R., & Wazwaz, A. M. (2010). Solitary wave solutions for a generalized KdV–mKdV equation with variable coefficients. *Mathematics and Computers in Simulation*, *80*(9), 1867-1873.
179. Shah, R., Khan, H., Baleanu, D., Kumam, P., & Arif, M. (2020). A semi-analytical method to solve family of Kuramoto–Sivashinsky equations. *Journal of Taibah University for Science*, *14*(1), 402-411.
180. Al Jamal, R., & Morris, K. (2018). Linearized Stability of Partial Differential Equations with Application to Stabilization of the Kuramoto--Sivashinsky Equation. *SIAM Journal on Control and Optimization*, *56*(1), 120-147.
181. Kukavica, I., & Massatt, D. (2021). On the global existence for the Kuramoto-Sivashinsky equation. *Journal of Dynamics and Differential Equations*, 1-17.
182. Brunet, P. (2007). Stabilized Kuramoto-Sivashinsky equation: A useful model for secondary instabilities and related dynamics of experimental one-dimensional cellular flows. *Physical Review E*, *76*(1), 017204.
183. López-Marcos, M. A. (1993). A note on the computation of bifurcation diagrams of the Kuramoto-Sivashinsky equation by pseudospectral methods. *Applied numerical mathematics*, *13*(1-3), 147-154.

184. Uddin, M., & Haq, S. (2009). A mesh-free numerical method for solution of the family of Kuramoto–Sivashinsky equations. *Applied Mathematics and Computation*, *212*(2), 458-469.
185. Mittal, R. C., & Arora, G. (2010). Quintic B-spline collocation method for numerical solution of the Kuramoto–Sivashinsky equation. *Communications in Nonlinear Science and Numerical Simulation*, *15*(10), 2798-2808.
186. Zarebnia, M., & Parvaz, R. (2013). Septic B-spline collocation method for numerical solution of the Kuramoto-Sivashinsky equation. *International Journal of Mathematical and Computational Sciences*, *7*(3), 544-548.
187. Arora, G., & Joshi, V. (2019). Simulation of generalized nonlinear fourth order partial differential equation with quintic trigonometric differential quadrature method. *Mathematical Models and Computer Simulations*, *11*(6), 1059-1083.
188. Fan, E. (2000). Extended tanh-function method and its applications to nonlinear equations. *Physics Letters A*, *277*(4-5), 212-218.
189. Khater, A. H., & Temsah, R. (2008). Numerical solutions of the generalized Kuramoto–Sivashinsky equation by Chebyshev spectral collocation methods. *Computers & Mathematics with Applications*, *56*(6), 1465-1472.
190. Lai, H., & Ma, C. (2009). Lattice Boltzmann method for the generalized Kuramoto–Sivashinsky equation. *Physica A: Statistical Mechanics and its applications*, *388*(8), 1405-1412.
191. Tadmor, E. (1986). The well-posedness of the Kuramoto–Sivashinsky equation. *SIAM Journal on Mathematical analysis*, *17*(4), 884-893.
192. Akgül, A., & Bonyah, E. (2019). Reproducing kernel Hilbert space method for the solutions of generalized Kuramoto–Sivashinsky equation. *Journal of Taibah University for Science*, *13*(1), 661-669.
193. Porshokouhi, M. G., & Ghanbari, B. (2011). Application of He's variational iteration method for solution of the family of Kuramoto–Sivashinsky equations. *Journal of King Saud University-Science*, *23*(4), 407-411.

194. Gomes, S. N., Papageorgiou, D. T., & Pavliotis, G. A. (2017). Stabilizing non-trivial solutions of the generalized Kuramoto–Sivashinsky equation using feedback and optimal controlLighthill–Thwaites prize. *IMA Journal of Applied Mathematics*, *82*(1), 158-194.
195. Rashidinia, J., & Jokar, M. (2017). Polynomial scaling functions for numerical solution of generalized Kuramoto–Sivashinsky equation. *Applicable analysis*, *96*(2), 293-306.
196. Rageha, T. M., Ismaila, H. N., Salemb, G. S., & El-Salamc, F. (2015). Restrictive approximation algorithm for Kuramoto–Sivashinsky equation. *Int. J. Modern Math. Sci*, *13*(1), 29-38.
197. Ersoy, O., & Dag, I. (2016). The Exponential Cubic *B*-Spline Collocation Method for the Kuramoto-Sivashinsky Equation. *Filomat*, *30*(3), 853-861.
198. Iqbal, M. K., Abbas, M., Nazir, T., & Ali, N. (2020). Application of new quintic polynomial B-spline approximation for numerical investigation of Kuramoto–Sivashinsky equation. *Advances in Difference Equations*, *2020*(1), 1-21.
199. Tomasiello, S. (2010). Numerical solutions of the Burgers–Huxley equation by the IDQ method. *International Journal of Computer Mathematics*, *87*(1), 129-140.
200. Tomasiello, S. (2003). Stability and accuracy of the iterative differential quadrature method. *International journal for numerical methods in engineering*, *58*(9), 1277-1296.
201. Tomasiello, S. (2011). Numerical stability of DQ solutions of wave problems. *Numerical Algorithms*, *57*, 289-312.
202. Zheng, S. (2004). *Nonlinear evolution equations*. Monographs and surveys in pure and applied mathematics. Chapman & Hall/CRC, CRC Press, Boca Raton.
203. Dehghan, M., & Abbaszadeh, M. (2017). Numerical investigation based on direct meshless local Petrov Galerkin (direct MLPG) method for solving generalized Zakharov system in one and two dimensions and generalized Gross–Pitaevskii equation. *Engineering with Computers*, *33*, 983-996.

204. Coullet, P., Elphick, C., & Repaux, D. (1987). Nature of spatial chaos. *Physical review letters*, *58*(5), 431-434.
205. Van Saarloos, W. (1987). Dynamical velocity selection: marginal stability. *Physical review letters*, *58*(24), 2571-2574.
206. Dee, G. T., & van Saarloos, W. (1988). Bistable systems with propagating fronts leading to pattern formation. *Physical review letters*, *60*(25), 2641-2644.
207. Van Saarloos, W. (1988). Front propagation into unstable states: marginal stability as a dynamical mechanism for velocity selection. *Physical Review A*, *37*(1), 211-229.
208. Kadri, T., & Omrani, K. (2011). A second-order accurate difference scheme for an extended Fisher–Kolmogorov equation. *Computers & Mathematics with Applications*, *61*(2), 451-459.
209. Khiari, N., & Omrani, K. (2011). Finite difference discretization of the extended Fisher–Kolmogorov equation in two dimensions. *Computers & Mathematics with Applications*, *62*(11), 4151-4160.
210. Guozhen, Z. (1982). Experiments on director waves in nematic liquid crystals. *Physical Review Letters*, *49*(18), 1332-1335.
211. Belmonte-Beitia, J., Calvo, G. F., & Perez-Garcia, V. M. (2014). Effective particle methods for Fisher–Kolmogorov equations: theory and applications to brain tumor dynamics. *Communications in Nonlinear Science and Numerical Simulation*, *19*(9), 3267-3283.
212. Hornreich, R. M., Luban, M., & Shtrikman, S. (1975). Critical behavior at the onset of k→-space instability on the λ line. *Physical Review Letters*, *35*(25), 1678-1681.
213. Ahlers, G., & Cannell, D. S. (1983). Vortex-front propagation in rotating Couette-Taylor flow. *Physical Review Letters*, *50*(20), 1583-1586.
214. Aronson, D. G., & Weinberger, H. F. (1978). Multidimensional nonlinear diffusion arising in population genetics. *Advances in Mathematics*, *30*(1), 33-76.
215. Danumjaya, P., & Pani, A. K. (2005). Orthogonal cubic spline collocation method for the extended Fisher–Kolmogorov equation. *Journal of computational and applied mathematics*, *174*(1), 101-117.

216. Mittal, R. C., & Arora, G. (2010). Quintic B-spline collocation method for numerical solution of the extended Fisher–Kolmogorov equation. *Int. J. of Appl. Math and Mech*, *6*(1), 74-85.
217. Danumjaya, P., & Pani, A. K. (2006). Numerical methods for the extended Fisher-Kolmogorov (EFK) equation. *International Journal of Numerical Analysis and Modeling*, *3*(2), 186-210.
218. Liu, F., Zhao, X., & Liu, B. (2017). Fourier pseudo-spectral method for the extended Fisher-Kolmogorov equation in two dimensions. *Advances in Difference Equations*, *2017*(1), 1-17.
219. Ilati, M., & Dehghan, M. (2018). Direct local boundary integral equation method for numerical solution of extended Fisher–Kolmogorov equation. *Engineering with Computers*, *34*, 203-213.
220. Oruç, Ö. (2020). An efficient wavelet collocation method for nonlinear two-space dimensional Fisher–Kolmogorov–Petrovsky–Piscounov equation and two-space dimensional extended Fisher–Kolmogorov equation. *Engineering with Computers*, *36*(3), 839-856.
221. Rohila, R., & Mittal, R. C. (2019). A numerical study of two-dimensional coupled systems and higher order partial differential equations. *Asian-European Journal of Mathematics*, *12*(05), 1950071.
222. Joshi, V. (2018). *The Numerical solutions of some linear and nonlinear partial differential equations using trigonometric B-spline basis functions* [Doctoral thesis, Lovely Professional University].

List of publications

List of Publications

1. **Navneet Kaur**, Varun Joshi "**Soliton solution of coupled Korteweg-de Vries equation by quintic UAH tension B-spline differential quadrature method**" *Journal of Mathematical Analysis and Applications*, Volume 514, 2022, 126355. (Indexed Sci, Scopus)
2. **Navneet Kaur**, Varun Joshi "**Numerical Solution of Coupled Korteweg-de Vries Equation Employing Quintic Hyperbolic B-Spline Based Differential Quadrature Method**" *International Journal of Applied and Computational Mathematics*, Volume 9, 2023, 8. (Indexed Scopus)
3. **Navneet Kaur**, Varun Joshi "**Numerical Solution of Gray Scott Reaction-Diffusion Equation using LagrangePolynomial**" *Journal of Physics: Conference Series*, Volume 1531, 2020, 012058. (Indexed Scopus)
4. **Navneet Kaur**, Varun Joshi, "**Numerical solution to the Gray-Scott Reaction-Diffusion equation using Hyperbolic B-spline**" *Journal of Physics: Conference Series*, Volume 2267, 2022, 012072. (Indexed Scopus)

List of communicated papers

1. **Navneet Kaur**, Varun Joshi "**Quintic Uniform Algebraic Trigonometric tension B-Spline Differential Quadrature approach for the solution of coupled Korteweg-de Vries Equation**" submitted to *Computational and Applied Mathematics*.
2. **Navneet Kaur**, Varun Joshi "**Kuramoto-Sivashinsky equation: Numerical Solution Using Two Quintic B-Splines and Differential Quadrature Method**" submitted to *Mathematics and Computers in Simulation*.

3. **Navneet Kaur**, Varun Joshi "**Adaptation of quintic Hyperbolic B-spline for the numerical solution of the Kuramoto-Sivashinsky equation**" submitted to *Journal of Applied and Computational Mechanics.*

Conferences Attended

1. Attended international conference on "*Recent Advances in Fundamental and Applied Sciences"* (RAFAS 2019) on 5th and 6th November 2019 at LPU, Phagwara and Presented a poster under the topic "Numerical solution of Gray Scott Reaction-Diffusion Equation using LagrangePolynomial".
2. Attended international conference on "*Recent Advances in Fundamental and Applied Sciences"* (RAFAS 2021) on 25th and 26th June 2021 at LPU, Phagwara and Presented a poster under the topic "Numerical solution of Gray Scott Reaction-Diffusion Equation using HBS".
3. Attended international conference on "*Number Theory and Differential Equations (ICND-2021)"* from 20th to 24th December 2021 at Central University of Karnataka, Kalaburaghi.

Workshop/Webinar Attended

1. Attended webinar on "Symmetries of Nonlinear Models of Real-World Processes" on 24th October 2020 .
2. Workshop on "Differential equations and their Applications" from 27th-28th November 2020.
3. Attended Seminar on "International E-Seminar on Advances in Mathematical Sciences" (ISAMS-2022) from 14th-16th March 2022.
4. Workshop on "International workshop on Research method and methodology" from 17th-18th October 2023.