REPUBLIC OF TURKEY
YILDIZ TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

NUMERICAL AND SYNCHRONIZATIONAL BEHAVIORS OF SOME EVOLUTION EQUATIONS

Shko Ali TAHIR

DOCTOR OF PHILOSOPHY THESIS
Department of Mathematics
Program of Mathematics

Advisor
Prof. Dr. Murat SARI

July, 2020

REPUBLIC OF TURKEY

YILDIZ TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

Numerical and Synchronizational Behaviors of Some Evolution Equations

A thesis submitted by **Shko Ali TAHIR** in partial fulfillment of the requirements for the degree of **DOCTOR OF PHILOSOPHY** is approved by the committee on 21.07.2020 in Department of Mathematics, Program of Mathematics.

Prof. Dr. Murat SARI Yildiz Technical University Advisor	Prof. Dr. Abderrahman BOUHAMIDI University of the Littoral Opal Coast Co-Advisor

Approved By the Examining Committee

Prof. Dr. Murat SARI, Advisor
Yildiz Technical University

Prof. Dr. İsmail KÜÇÜK, Member
Istanbul Sabahattin Zaim University

Prof. Dr. İdris DAĞ, Member
Eskişehir Osmangazi University

Prof. Dr. Canan Çelik KARAASLANLI, Member
Yildiz Technical University

Prof. Dr. Bayram Ali ERSOY, Member
Yildiz Technical University

I hereby declare that I have obtained the required legal permissions during data collection and exploitation procedures, that I have made the in-text citations and cited the references properly, that I have not falsified and/or fabricated research data and results of the study and that I have abided by the principles of the scientific research and ethics during my Thesis Study under the title of "Numerical and Synchronizational Behaviors of Some Evolution Equations" supervised by, Prof. Dr. Murat SARI. In the case of a discovery of false statement, I am to acknowledge any legal consequence.

Shko Ali TAHIR

Signature

S. Ari

Dedicated to my beloved parents & all my family members for everything.

ACKNOWLEDGEMENTS

First and foremost I would like to raise my hands with praise and gratitude to all almighty Allah, most gracious and most merciful. Thank God for guiding me throughout my life journey until now.

I want to thank my terrific and creative advisor Prof, Dr. Murat SARI, his support throughout my study was astonishing and his enthusiasm, dynamism and vision motivated me to go extra mile with my PhD project. I also wish to thank Prof. Dr. Abderrahman BOUHAMIDI for giving me chance to work with him and assist me as co-advisor. I would like to thank my thesis monitoring committee, Prof. Dr. Idris DAĞ, Prof Dr. Ismail KÜÇÜK for their deep revisions of my thesis, with astonishing and very important suggestions. I also would like to thank the other committee members. I am so grateful to the YTP and the department of mathematics at the Yildiz technical university for making it possible for me to study here. I would like to thank university of Sulaimani for their continued support. Also I thank the university ULCO for generously hosting during one year of my research. I would like to express my gratitude to Mr. Pierre GENTRIC and my colleagues at Saint Benoît French college, who have been so helpful and cooperative in giving their support at all times to help me achieve my goal. I would like to thank all my friends that I have got to know here, especially Hande for her genuine support.

Last but not the least important, I would like to thank my mother whose her love and guidance were with me in whatever I have pursued, and the support and advices she provide is beyond the measures, and I enormously owe her. My Father, my deepest heartfelt sympathies go out to you, may God rest you in peace and grant you heaven. I came to this point which still I need and use your wisdom quotes. God Bless you. My beloved family deserve a great deal of thanks, for their endless support and encouragement during not only this PhD but all my endeavors. I wish to thank my wife and my two lovely children, Jan and Selin, for their cuddle breaks and their affection and love, they give me unending vitality, fire, passion and innate strength. I thank God every day for giving you all to me.

Shko Ali TAHIR

TABLE OF CONTENTS

LIST OF SYMBOLS

A	Area
ψ	Controller function
k_2	Coupling parameter of synchronization
$\mathscr{D}_{dis}$	Dispersion coefficient
$\nabla\cdot$	Divergence operator
v	Fluid velocity
$\mathscr{F}$	Flux
∇	Gradient operator
$\mathscr{J}$	Jacobian matrix
L	Length
$\mathscr{L}$	Linear partial differential operator
Λ	Lyapunov function
λ_L	Lyapunov exponent
M	Mass
λ	Molecular diffusivity
$\mathscr{N}$	Nonlinear differential part
s	Spline function
s_h	Interpolating cubic spline (1D)
s_{hd}	Interpolating cubic spline (2D)
s_m	Interpolating of the modified cubic B-spline
Υ	Smooth function between coupled systems
p	Step of BDF method
T	Time

$2D$	Two dimensional
$3D$	Three dimensional
V	Volume

LIST OF ABBREVIATIONS

ADR	Advection Diffusion Reaction
BDF	Backward Differentiation Formulae
BDFS	Backward Differentiation Formulae-Spline
CFL	Courant–Friedrichs–Lewy
SSPRK54	Five Stage and Fourth-Order Strong Stability Preserving Runge-Kutta
SSPRK54S	Five Stage and Fourth-Order Strong Stability Preserving Runge-Kutta-Spline
GS	Generalized Synchronization
GBFE	Generalized Burgers-Fisher Equation
GBHE	Generalized Burgers-Huxley Equation
GBFEF	Generalized Burgers-Fisher Equation with Forcing Terms
GBHEF	Generalized Burgers-Huxley Equation with Forcing terms
ODEs	Ordinary Differential Equations
PDEs	Partial Differential Equations
RK	Runge–Kutta

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Numerical and Synchronizational Behaviors of Some Evolution Equations

Shko Ali TAHIR

Department of Mathematics
Doctor of Philosophy Thesis

Advisor: Prof. Dr. Murat SARI
Co-advisor: Prof. Dr. Abderrahman BOUHAMIDI

This study provides several new combined methods to capture the numerical behaviour of nature, governed by the nonlinear advection-diffusion-reaction equation, in one and two dimensions. To achieve this, the implicit backward differentiation formula-spline (BDFS), the optimal five-stage and fourth-order strong stability preserving Runge-Kutta (SSPRK54)-spline and the modified cubic B-spline-SSPRK54 methods are proposed. Without any linearization, the given problems through the proposed schemes are converted to a system of nonlinear and linear differential equations. The current methods are seen to be very reliable alternatives in solving the problem by conserving the physical properties of nature. In addition, the generalized synchronization behaviours of nonlinear advection-diffusion-reaction processes, without losing their natural properties, are investigated to demonstrate the effectiveness of the proposed technique and to reduce computational difficulties in capturing numerical solutions for advection dominant cases. Within the framework of this thesis, a new version of the synchronization methods, based on the design of response systems, is also proposed to solve the synchronization problem discussed here. This technique utilizes the driver configuration to monitor the synchronized motions. To show the effectiveness and feasibility of those approaches, various numerical simulations are carried out.

Keywords: Nonlinear advection-diffusion-reaction, Dynamical system, Chaos, Synchronization, Approximation theory.

YILDIZ TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

ÖZET

Bazı Evolüsyon Denklemlerinin Nümerik ve Senkronizasyonal Davranışları

Shko Ali TAHIR

Matematik Bölümü

Doktora Tezi

Danışman: Prof. Dr. Murat SARI

Eş-Danışman: Prof. Dr. Abderrahman BOUHAMIDI

Bu çalışma, doğrusal olmayan adveksiyon-difüzyon-reaksiyon denklemi tarafından yönetilen doğanın nümerik davranışını bir ve iki boyutta yakalamak için birkaç yeni kombine yöntem sunmaktadır. Bunu gerçekleştirmek için, kapalı geri fark-spline (BDFS) ile SSPRK54-spline ve modifiye kübik B-spline-SSPRK54 yaklaşımlarını barındıran optimum beş aşamalı ve dördüncü mertebeden kuvvetli stabiliteye sahip yöntemler önerilmektedir. Herhangi bir doğrusallaştırma yapmaksızın, önerilen şemalar aracılığıyla ele alınan problemler, doğrusal olmayan ve doğrusal diferansiyel denklem sistemlerine dönüştürülür. Doğal özellikleri muhafaza ederek, problemin çözümünde önerilen yöntemlerin güvenilir alternatifler olduğu görülmektedir. Ayrıca, viskozite katsayısının düşük degerinde, doğal özelliklerini kaybetmeden doğrusal olmayan adveksiyon-difüzyon-reaksiyon süreçlerinin genelleştirilmiş senkronizasyon davranışları, önerilen tekniğin etkinliğini göstermek ve adveksiyon-baskın durumlar için nümerik çözümleri yakalamadaki hesaplama güçlüklerini azaltmak amacıyla araştırılmıştır. Bu tez çerçevesinde, burada ele alınan senkronizasyon problemini çözmek için cevabi sistemlerin tasarlanmasına dayalı, senkronizasyon yöntemlerinin yeni bir versiyonu da önerilmektedir. Bu teknik senkronize edilmiş hareketleri izlemek için ana yapılandırmayı kullanır. Bu yaklaşımların etkinliğini ve uygulanabilirliğini göstermek için çeşitli nümerik simülasyonlar yapılmıştır.

Anahtar Kelimeler: Doğrusal olmayan adveksiyon-difüzyon reaksiyonu, Dinamik sistem, Kaos, Senkronizasyon, Yaklaşım teorisi.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

1
INTRODUCTION

The nonlinear advection-diffusion-reaction (ADR) problems have been taken much attention in studying many problems encountered in science such as viscous fluid flow, filtration of liquid, gas dynamics, heat conduction, biological species and chemical reactions [58, 145]. Such a prediction was investigated to approximate their solutions numerically, while it is not easy to crack these problems analytically. In the process of historical development, these model equations have been considered by many researchers for both conceptual understanding of physical flows and testing various numerical methods with having challenges of small or large values of the viscosity and independent parameters. The major difficulty of the nonlinear ADR equations with forcing terms is producing their numerical solutions without any linearization. Researchers are still investigating new techniques to find the solution of the nonlinear ADR problems with the aim of improving accuracy, especially when the initial and boundary functions are not smooth or are available only at the grid points. For the nonlinear ADR processes with low values of the viscosity coefficient, several interactions between reaction, convection and diffusion mechanisms can be observed [67]. Thence, many characteristics of chaos such as instability and limited predictability in time can be existed in the nonlinear ADR problems. Some researchers have pointed out that there exists close relationship between chaos and nonlinear ADR processes [161, 184, 255]. Further studies carried out herein, the dynamical behavior and generalized synchronization (GS) of two dependent or independent nonlinear processes are discussed. Study of synchronization behaviors of the nonlinear ADR equations remains new and mostly unexplored field. Since the nonlinear coupled ADR model cannot synchronize, some controller functions should be designed and applied to force the driver system to synchronize with response system. One of the reasons for this is that motivated us to examine the phenomenon and develop suitable synchronization control function via the classical Lyapunov direct method. In the present work, due to aforementioned aims, we propose various newly combined techniques for the approximate solution of the nonlinear ADR processes and solving synchronization problems of nonlinear coupled models. Let us now give some important key definitions and properties that will be useful for the later chapters.

1.1 Principal Terminology

Advection, diffusion and synchronization are main parts consisting of principal terminology. Their physical meaning and mathematical representation bring the whole picture of understanding of them and developing numerical solution techniques.

1.1.1 Advection

Advection plays a fundamental role in the field of physics, engineering, and applied mathematics. The amount of substance traverses the cross-section over which the count is performed depending on nature of the transporting process by bulk motion, this process is called advection. A well-known example is the advection of the pollutants in a river by bulk water flow downstream. The one dimensional concentration gradient of the pollutant is described mathematically as a vector field and given by means of partial differentials equations (PDEs) as:

$$\frac{\partial u}{\partial t} + v\frac{\partial u}{\partial x} = 0, \tag{1.1}$$

where v is fluid velocity. u means concentration, amount of heat or mass transfer. The smooth function $u = u(x, t)$ represents physical processes that can be in one or many dimensional space. Figure 1.1 presents the important property of advection as given: the shape and amplitude are unchanged. However, the position moves to the direction of velocity.

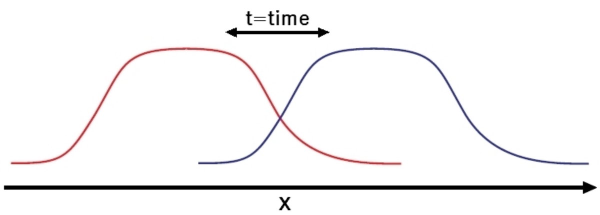


Figure 1.1 Schematic solution of the advection problem in one dimension

1.1.2 Diffusion

The word has a latin root, which means "to spread out". A well-known spread out is a substance transport from an area of high concentration to an area of low concentration in both fluids and solids, without requiring any bulk fluid motion. There are various problems in the physical sciences that we associate with the idea of diffusion, for instance: electrons, ions diffuse, macroscopic, atomistic and molecular approaches. Besides, we introduce the concept of the diffusion in mathematics. It includes all topics of concentration, heat, momentum, information and that can be diffused, by means of the PDEs. One dimensional form of the diffusion equation is given by:

$$\frac{\partial u}{\partial t} = \lambda \frac{\partial^2 u}{\partial x^2}, \tag{1.2}$$

where $\lambda > 0$ is the molecular diffusivity. From Figure 1.2, we can conclude that the diffusion has the following properties: the shape spreads or diffuses, the amplitude decreases and the position spreads but stays at center fixed.

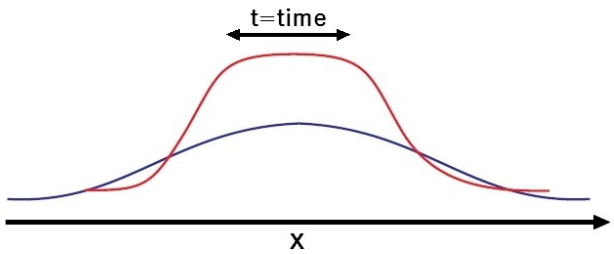


Figure 1.2 Schematic solution of the diffusion problem in one dimension

1.1.3 Model equations

Structure of the nonlinear ADR model plays an important role for describing the relation among the reaction mechanisms, convection effect and diffusion transport. They arise in various fields of science such as fluid dynamics, financial mathematics, turbulence, traffic flow, shock waves, gas dynamics, heat conduction, etc [70, 120, 153, 195, 199]. The nonlinear ADR equation can be expressed as

$$\frac{\partial u}{\partial t}(x,t) = \mathscr{L}(\triangle u, \bigtriangledown u, u, x, t) + \mathscr{N}(\triangle u, \bigtriangledown u, u, x, t), \quad (x,t) \in \Omega = [a,b] \times [t_0, T]. \tag{1.3}$$

Here, $\mathscr{L}(\triangle u, \bigtriangledown u, u, x, t) = a_2 \triangle u(x,t) + a_1 \bigtriangledown u(x,t) + a_0\, u(x,t)$ is a linear partial differential operator of the second order, a_i are constant coefficients, and $\mathscr{N}$ defines a nonlinear differential part. The initial and boundary conditions are given by

$$u(x,t_0) = u_0(x), \quad u(a,t) = g_1(t), \quad u(b,t) = g_2(t), \tag{1.4}$$

where both boundary functions g_1, g_2 and initial function u_0 are known. Even though some researchers assume that the boundary functions g_1 and g_2 are differentiable, it is not necessary for all the times. In the present research, we only assume that the boundary functions g_1 and g_2 are defined on the time interval $[t_0, T]$ without requiring the differentiability of these functions. In the ADR equation, $\frac{\partial u}{\partial t}(x,t)$ is the accumulation term. This term provides the change of concentration over the time. The advection term presents the gradient of concentration corresponding to distances and it is considered by the term $\bigtriangledown u$. The term $\triangle u$ is the diffusion. It provides the divergence of scalar gradient with a constant diffusivity. From Figure 1.3, it can be deduced that the nonlinear ADR equation has the following characteristics: the shape spreads and diffuses, the amplitude decreases and the position changes with the direction of velocity. It is noticeable that, the ADR equations are highly nonlinear equations because they present the interaction between reaction, convection and diffusion mechanisms. The nonlinear ADR equations also contain free parameters. Thus, examination of the physical and numerical properties of the nonlinear ADR equation becomes quite complex. A large number of researchers have mainly carried out to handle such problems by reducing the computational difficulties on capturing their numerical solutions and keeping their real features of the nature at low value of the viscosity at various free parameters. Therefore we concentrate on analysis of the nonlinear physical phenomena without losing their natural properties. To achieve the aforementioned aims, the BDFS, SSPRK54S and modified cubic B-spline SSPRK54 methods are considered. The relative importance of chaotic advection, diffusion and reaction within nonlinear ADR models have extensively been pointed out by [235, 240]. Synchronization can be considered as the adaptation of objects to each other's behavior. Recently, in some researches attention has been paid on generalized synchronization of PDEs [118, 172]. Thus, study of synchronization behaviors of the nonlinear ADR equation remains new and mostly unexplored field. Therefore, next we address the dynamical and generalized synchronization (GS) of coupled chaotic identical and nonidentical models.

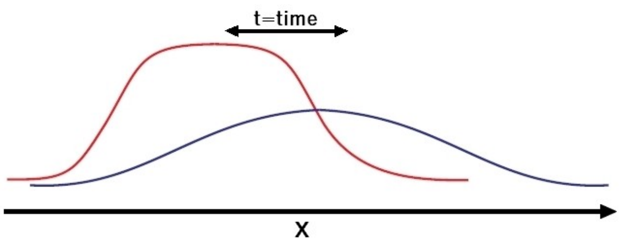


Figure 1.3 Schematic solution of the nonlinear ADR problem in one dimension

1.1.4 Synchronization

The origin of a word has a greek root, syn = common and chronos = time, which means to show the same behaviour over time or to occur at the same time, in which two or more systems interact with each other resulting in a joint evolution on some of their dynamical properties. Thus, synchronization of two or more dynamical systems generally means that one system somehow follows from the behaviour of another. Chaotic phenomena have been seen to be new kinds of oscillating system for the successful applications in different scientific fields including physics, chemistry, ecology, biology, etc. Chaotic oscillators are found in many dynamical systems of various origins. Their behaviors are characterized by instability and limited predictability in time. The original work on synchronization was introduced in coupled pendulum by Huygens [44]. Since this discovery has been carried out, it has attracted very considerable attention over the past three decades in different scientific fields including physical and biological processes. The surprising synchronization phenomena generated between coupled chaotic systems has been discovered by Pecora and Carroll [149]. They proposed that "synchronization can be observed even in chaotic systems" [156]. Then, the synchronization of coupled chaotic systems has been extensively and intensively studied. They split the system into two subsystems, the first one is the driver system and the second one is the response system and may be given in the following form:

$$\begin{cases} \dot{x}(t) &= H(x(t)) & \text{driver,} \\ \dot{y}(t) &= G(y(t)) + \psi(x(t), y(t)) & \text{response,} \end{cases} \tag{1.5}$$

where the functions $H : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ and $G : \mathbb{R}^m \longrightarrow \mathbb{R}^m$ are continuous vector valued functions. The vector $x(t) \in \mathbb{R}^n$ represents the driving signal and $y(t) \in \mathbb{R}^m$ represents the response signal. The function $\psi : \mathbb{R}^{k_1} \longrightarrow \mathbb{R}^m$ is a controller function, here $k_1 = m + n$. After this discovery, several types of synchronization were discovered such as: identical or complete synchronization appears as the coincidence of states of interacting systems, phase synchronization which means the phases of chaotic oscillators in a closely controlled phase relationship, lag synchronization appears as having a parameter mismatch in mutually coupled chaotic oscillator. This type of lag synchronizations has important technological implications in engineering systems. In the case of synchronization of driver-response systems, the designed controller makes the trajectories of the state variables of the driver system to track the trajectories of the response system. This fact may pose a trouble in using the results of theoretical analyses in practical applications of synchronized chaos. Thus, we investigate general methods to detect the existence of the transformation and study this kind of synchronous behavior. Besides, we present the dynamical and GS of two dependent chaotic nonlinear ADR processes with forcing terms, which unidirectionally coupled in the driver-response configuration.

1.2 Literature Review

Numerical solutions of nonlinear ADR models with source functions have been subject to a huge number of studies by many researchers for both conceptual understanding of mathematical and physical reasons. In the process of historical development, various versions of the schemes have been analyzed and implemented successfully to investigate the nonlinear ADR models with challenging values of the viscosity and independent parameters.

In particular, the Burgers equation represents some of the interesting nonlinear ADR phenomena. For the past century, the Burgers equation which has attracted much attention in studying many problems encountered. Bateman [83] considered the ADR equation in his work along with its corresponding initial and boundary conditions. Later, Burgers [120] provided the mathematical modelling of turbulence by using (1.3). Hopf and Cole [57, 114] provided independently that this equation can be transformed to the linear diffusion equation and solved exactly for an arbitrary initial condition. The infinite and finite domains for solving the Burgers equation were suggested by Benton and Platzman [65]. Accurate solution of the Burgers model by using the Galerkin method with fully upwind cubic functions was discussed in the references [37, 96]. Some other studies [66, 90, 97] have considered the space-time finite elements incorporating characteristics for the Burgers equation. Sari and Gurarslan [179] presented the sixth-order compact finite difference method to approximate the solution of the nonlinear Burgers equation. Here, the authors combined the tridiagonal sixth-order compact finite difference scheme in space and the low-storage third-order total variation diminishing RK scheme in time. Many researchers have paid particular attention to solving this problem using various numerical approaches, such as Fourier expansion scheme [205], finite element methods [5, 138, 164, 170, 171, 228, 238, 252], variational method [62, 155, 233], homotopy analysis method [4, 166, 173, 174], spectral collocation method [11, 63], finite volume method [1], differential quadrature methods [16, 18, 210], Haar wavelet [190, 208]. In many studies, researchers have used linearization of nonlinear terms to produce numerical solution of the Burgers equation, which are likely to move away from nature of problems, taking into account various assumptions. For more details see references [10, 68, 87, 139, 207, 218, 227, 232]. The Fisher equation is an important model for describing the process of interaction between diffusion and reaction. There have been vast variety of numerical techniques to obtain solution of the Fisher equation. Canosa [112, 113] considered numerical solutions for the Fisher equation by using a space derivative method. In dealing with the Fisher processes, various numerical techniques to investigate the Fisher models were developed [54, 71, 75, 76, 77, 129, 160, 176, 177, 185, 201, 220, 234, 236, 241, 246, 249, 250].

The generalized Burgers-Fisher equation (GBFE) and the generalized Burgers-Huxley equation (GBHE) with forcing terms can also be presented to be good examples of the nonlinear ADR processes. In fact, the GBFEF and GBHEF have been applied to describe the interaction between diffusion and transports, convection and reaction mechanisms. In the past few years, a great deal of effort has been spent to compute the solution of these models. The GBFEF was first studied by Fisher, with free of forcing terms to describe the propagation of gene in a habitat [199]. Mickens and Gumel [59] gave the non standard finite difference method for the approximation solutions of the GBFE. Kaya and El-Sayed [52] introduced the numerical and explicit solutions of the GBFE. In these works [89, 103, 104, 137], the authors proposed approximate solutions for the GBHE and GBFE by using the adomian decomposition method. The spectral collocation scheme in space and the fourth order Runge Kutta method in time to solve the GBFE were considered by Golbabai and Javidi [8, 168]. Sari et al. [181] investigated the numerical solution of the GBFE and GBHE by using a compact finite difference method with minimal computational cost. Sari et al. [178] also used the higher order finite difference schemes in space and RK4 scheme in time to produce numerical solutions of the GBHE. Dehghan et al. [163] considered the interpolation scaling functions and the mixed collocation difference scheme for solving the GBHE. Recently, many researchers have paid their attention to produce numerical solution of these problems by investigating various methods. For instance, differential quadrature methods [6, 180, 203, 243], series-based methods [21, 22, 34, 86, 95, 110, 175, 215], finite difference schemes [35, 117, 130, 154, 244, 245].

In the last few years, another numerical technique was widely used to solve mathematical models with higher degree piecewise polynomials. Among them, the spline based methods come into existence to solve the nonlinear ADR models in the computational mathematics. First, Schoenberg [105] found mathematical relations of the splines in the context of piecewise polynomial approximations. The continuity, smoothness and local supports of the spline functions were defined by Boor and Prenter [42, 197]. Also these methods have additional advantages over some rival techniques such as they are relatively easy in use and are of computational cost efficiency. Bickley [248] published a study on solutions of the two-point boundary value problem by using the cubic spline interpolation method. Following this research, Fyfe [51] worked on this approach and concluded that the spline method was better than the usual finite difference method. Hence, the applications of spline interpolation of the boundary value problems were developed [12, 20, 55, 125]. Types of piecewise polynomial spline functions are utilized with other numerical techniques for getting the solutions of the nonlinear ADR equations while they are playing important roles together in their computation. This means that any B-spline basis functions are often governed by the spline function that have minimal support with respect to degree, smoothness and domain partition. The first reference to the B-spline function was introduced by Schoenberg [106], who presented it as the piecewise smooth polynomial approximation. Fundamental properties of the spline functions and their limits were proposed by various researchers [40, 41, 82]. Rubin and Graves [223] used the cubic spline interpolation based on the quasi-linearisation scheme to solve the Burgers equation. The spline collocation method was developed to solve the Burgers equation, for instance, see [9, 99, 101, 109, 111, 136, 151, 158, 189, 194, 213, 217, 222, 230]. The finite difference scheme with the cubic splines interpolating space derivatives in solving the Burgers equation was developed by some authors [3, 36, 38, 47, 60, 61, 128, 143, 144, 157, 159, 192, 193, 237]. Zhu and Wang [43] proposed a method for solving the Burgers equation via a spline approach. Various numerical methods based on of the modified cubic B-splines in space were studied in [30, 73, 186, 212]. There has been vast variety of numerical techniques based on splines to obtain solution of the ADR problems such as quadratic B-splines method [15, 93, 182], cubic B-spline methods [17, 98, 231], trigonometric quadratic B-spline algorithms [28, 33, 72, 209, 211], exponential modified cubic B-spline differential quadrature method [29, 183]. Zhu and Kang [49] presented the numerical solution of the Burgers–Fisher equation based on the cubic B-spline scheme and a forward difference to approximate the time derivative of the dependent variable.

For the last few decades, chaotic phenomena were seen to be new types of oscillating system for the successful applications in different scientific fields including physics, chemistry, ecology, biology, etc [91]. In the literature, the large number of researchers had extensively concentrated on the identical synchronization [146, 149, 156], the generalized synchronization [121, 147, 187]. Several different studies of synchronization were also proposed: the active control methods such as adaptive control, feedback control, sliding mode control, adaptive lag synchronization for chaotic system [64, 94, 122, 124, 216, 224, 251, 254]. It is clear that, the behavior of the nonlinear ADR problems can be visualized in chaotic synchronization. Some researchers pointed out that there exists close relationship between chaos and nonlinear ADR problems [26, 56, 78, 260]. However, in the case of synchronization of the transformation between the advection and diffusion terms, the designed controller makes the trajectories of the state variables of the driver to track the trajectories of the response problem. This fact may pose a trouble in using the results of theoretical analyses in practical applications of synchronized chaos of the nonlinear ADR processes. One of the reasons for this is that motivated us to focus on analysis of the nonlinear physical phenomena on capturing numerical behavior of nature governed by the nonlinear coupled ADR equations with source functions.

The iterature tells us that the nonlinear coupled ADR models are characterized by the reaction and diffusion or by the interaction between advection and diffusion

[102, 226, 242]. In recent years, many researchers have paid particular attention to solving these problems using various numerical approaches [27, 123, 131, 133, 135, 162, 165, 169, 229, 239, 253, 257, 259]. Several authors paid their attention to produce approximation solution of the nonlinear coupled ADR problem by taking into account various assumptions [7, 87, 119, 200, 219, 232]. Various numerical techniques to investigate the coupled ADR model by using tensor product were developed [107, 204, 225, 258]. Study of synchronization behaviors of the nonlinear ADR equation remains new and mostly unexplored field. Throughout the last two decades, the ADR models have attracted a lot of attention to get the accurate results by using various methods under synchronization techniques. Basto et al. [161] considered the Chebyshev spectral solutions of the Burgers equation at low values of the viscosity values for synchronization. In the case that the nonlinear coupled ADR model cannot synchronize, some control functions should be designed. Thus, the two theorems for proposing controllers functions for the generalized synchronization were studied in [24, 80, 84, 118, 132, 148, 172, 196]. The Lyapunov method was also studied for the stability of the synchronization of chaotic models for instance, see [23, 32, 256]. Moreover, Yuan et al. [142] studied the synchronization of the PDEs by combining the PDEs theory with the Lyapunov method. Some numerical methods have been developed in trying to get the accurate results of the nonlinear ADR models under various conditions. Our aim is to find efficient schemes for these types of physical problems with conserving the physical properties of nature.

1.3 Objectives

Beyond what has been stated, this research consists of five phases. The first phase mainly focuses on capturing numerical behavior of the nonlinear ADR processes with forcing terms, without doing any linearization. To achieve this, we present the BDFS, SSPRK54S and the modified cubic B-spline-SSPRK54 methods. Comparison between the current methods is carried out in dealing with the nonlinear ADR problems to check the efficiency and utility of the proposed schemes. In the second phase, we propose analysis of a synchronization of coupled chaotic identical and nonidentical dynamical systems producing generalized synchronization in drive-response systems. Thus, we have investigated general methods to detect the existence of the transformation and study this kind of synchronous behavior. In the case of the drive-response methods, efforts to a systematic method that guide the development of solutions to synchronization problems, when trajectories of driving and response systems are strongly connected, then two close states in the state space of the response system correspond to the two close states in the space of the driving system. The third phase mainly focuses on analysis of the physical phenomena without losing their natural properties and reduces the computational difficulties on capturing numerical behavior of nature governed by the coupled Burgers equations with source functions. To achieve this, the BDFS method is proposed, in the sense that, it does not require either linearization, or tensor product. Next phase, by combining the BDFS scheme with the Lyapunov method, the GS is studied for designing control function of the coupled nonlinear ADR equations. The proposed technique effectively guarantees the stability of generalized chaotic synchronization of the nonlinear ADR model by constructing a driver system to implement the generalized synchronization with a response chaotic system by using the Lyapunov stability theory for the low value of the viscosity coefficients. In the last phase, the development of the BDFS scheme for solving the 2D nonlinear ADR model with appropriate initial and boundary conditions.

To accomplish the stated aims, some of the important properties of the current methods are as follows:

1. Neither linearization nor transforming the process is required.
2. Boundary and initial functions are defined on the time interval without requiring the differentiability of these functions.
3. The produced solutions are not presented only at the grid points but also at optional points in the solution domain.
4. The BDFS scheme is unconditionally stable.
5. The proposed methods replace the one and two dimensional ADR problems by ODEs.
6. The designed controller functions enable the state variables of the drive system to globally synchronize with the state variables of the response system in chaotic models.
7. The GS scheme implements directly the synchronization and stabilization of physical, biological and chemical problems which are well-intended chaotic systems with fast synchronization speed.
8. Theoretical results of the current methods are effectively guaranteed for the stability of generalized chaotic synchronization.
9. Development and verification of the BDFS with the Lyapunov method are given to ensure the GS of the coupled nonlinear ADR model.
10. A complementary goal of this work is to investigate and improve the 2D nonlinear ADR problems by the BDFS scheme.

1.4 Thesis Overview

This thesis consists of seven chapters: following the introduction chapter, in Chapter 2, we present the derivation of the nonlinear ADR equation. Chapter 3 presents some properties concerning the cubic splines, B-splines and natural spline in space. Besides, we propose a new scheme for solving the Burgers equation by modifying cubic B-spline approximation in space. Beside, we further propose a generalized method by designing new response systems for solving synchronization problems of coupled chaotic identical and nonidentical dynamical systems. Later, analyses of the BDF and SSPRK54 methods to solve differential equations are discussed. Chapter 4 introduces implementation of the currents methods to handle some nonlinear ADR problems and chaotic systems in time and space. İn Chapter 5, we provide the BDFS method for the 2D nonlinear ADR problems. Chapter 6 is devoted to illustrative examples to discuss the effectiveness of the current methods. Chapter 7 is consisting of final remarks and recommendations in the thesis.

2
NONLINEAR ADR EQUATIONS

This chapter presents derivation of the nonlinear ADR equations. This model is one of the most used models in the computational mathematics and physics. It presents how the concentration of one or more substances distributed in an occasion. For example river moves under the influence of three processes, which are advection, diffusion, and reaction. This model presents the fluid equation introduced by Navier in 1822 [48] and successively studied by several authors: Cauchy in 1823 [19], Poisson in 1829, Saint Venant in 1837, finally, Stokes in 1845 [74]. Thus, they developed the nonlinear ADR problem describing the velocity field and fluid with the initial and boundary conditions based on the conservation of mass.

The nonlinear ADR equation can be derived by using the mass balance equation. First, we derive the nonlinear ADR equation by balancing the difference between the total mass of material entering and leaving the element. To apply the nonlinear ADR equation to the conversation of mass, the difference between the total mass entering and leaving the control volume must be equal to the rate of the total mass inside the control volume. It is considered that the mass balance for the total control volume with the transport occurs in the x-direction can be written as follows (see Figure 2.1)

$$\underbrace{V\frac{\partial u}{\partial t}}_{\substack{change\ of\ the\ mass\\ in\ the\ total\ volume\ in\ \Delta t}} = \underbrace{A\mathscr{F}_{in}}_{\substack{mass\ entering\\ the\ total\ control\ volume\ in\ \Delta t}} - \underbrace{A\mathscr{F}_{out}}_{\substack{mass\ leaving\\ the\ total\ control\ volume\ in\ \Delta t}}, \tag{2.1}$$

where, $V\,[L^3]$ is the volume, $\dfrac{\partial u}{\partial t}\,[ML^{-3}T^{-1}]$ is the concentration over t, $A\,[L^2]$ is the area, $\mathscr{F}_{in}\,[ML^{-2}T^{-1}]$ and $\mathscr{F}_{out}\,[ML^{-2}T^{-1}]$ are the fluxes. By dividing equation (2.1) by volume V, one obtains

$$\begin{aligned}\frac{\partial u}{\partial t} &= \frac{A}{V}\mathscr{F}_{in} - \frac{A}{V}\mathscr{F}_{out}.\\ &= \frac{A}{V}\left(\mathscr{F}_{in} - \mathscr{F}_{out}\right).\end{aligned} \tag{2.2}$$

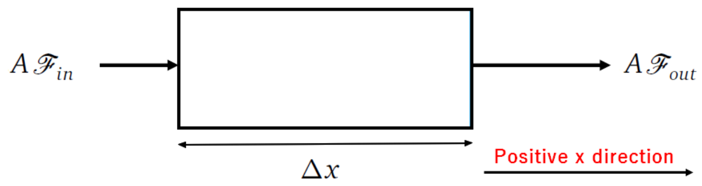


Figure 2.1 Mass balance for a control volume

The flux is changing in the x-direction with the gradient $\frac{\partial \mathscr{F}}{\partial x}$ in the form

$$\mathscr{F}_{out} = \mathscr{F}_{in} + \frac{\partial \mathscr{F}}{\partial x}.\Delta x. \tag{2.3}$$

Substituting equation (2.3) into (2.2) leads to

$$\frac{\partial u}{\partial t} = \frac{A}{V}\left(\mathscr{F}_{in} - \left(\mathscr{F}_{in} + \frac{\partial \mathscr{F}}{\partial x}.\Delta x\right)\right). \tag{2.4}$$

The term $\frac{A}{V}$ indicates the change of x in positive direction, thus $\frac{V}{A} = \Delta x$. Equation (2.4) becomes the general transport equation in x direction as given

$$\frac{\partial u}{\partial t} = -\frac{\partial \mathscr{F}}{\partial x}. \tag{2.5}$$

Equation (2.5) is derived for the conservative tracer of the materials. Here, we consider that all fluids have same the densities of viscosity without loss or addition of matter. Thus, the control volume is not changing as the time progresses. However, $\mathscr{F}$ can be flow, dispersion, advection etc. In this research, we present the advection and dispersion as the two important models of the transport of fluid. We explain these two models of the transport in the x - direction as:

$$\mathscr{F}_{AdvectionFlux} = \frac{\partial x}{\partial t}.u, \tag{2.6}$$

and

$$\mathscr{F}_{DispersiveFlux} = -\mathscr{D}_{dis}\frac{\partial u}{\partial x}. \tag{2.7}$$

$\mathscr{F}_{AdvectionFlux}$ provides the number of particles moving from control volumes in unit time per unit area. $\mathscr{D}_{dis}$ is the dispersion coefficient. By taking into account the notation of the advection and dispersion flux, equation (2.5) becomes:

$$\frac{\partial u}{\partial t} = -\frac{\partial \mathscr{F}}{\partial x} = -\frac{\partial}{\partial x}\left(\mathscr{F}_{AdvectionFlux} + \mathscr{F}_{DispersiveFlux}\right). \tag{2.8}$$

Substitution of equations (2.6) and (2.7) into (2.8), one obtains:

$$\frac{\partial u}{\partial t} = -\frac{\partial}{\partial x}(\underbrace{\frac{\partial x}{\partial t}.}_{\substack{fluid\ velocity \\ v\ in\ x\ direction}}\ u) - \frac{\partial}{\partial x}(-D_{dis}\frac{\partial u}{\partial x}). \tag{2.9}$$

Rearranging the above expressions, we find out

$$\frac{\partial u}{\partial t} = -v\frac{\partial u}{\partial x} + D_{dis}\frac{\partial^2 u}{\partial x^2}. \tag{2.10}$$

In the real life, we live in the three dimensional space, since the same rules exists for the mass balance and transport in all possible dimensions (see Figure 2.2).

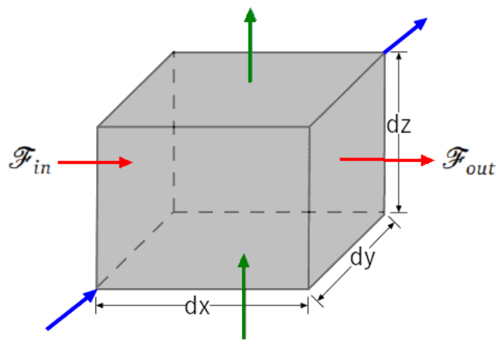


Figure 2.2 Control volume

Now, equation (2.5) is summarized as follows

$$\frac{\partial u}{\partial t} = -\sum_{i=1}^{3}\frac{\partial \mathscr{F}_i}{\partial x_i}, \tag{2.11}$$

where, $x_1 = x$, $x_2 = y$ and $x_3 = z$. Equation (2.11) leads to

$$\frac{\partial u}{\partial t} = -\left(\frac{\partial \mathscr{F}_x}{\partial x} + \frac{\partial \mathscr{F}_y}{\partial y} + \frac{\partial \mathscr{F}_z}{\partial z}\right). \tag{2.12}$$

Rate of mass of fluid within the total control volume is given by using the sum of the net mass flow rates in each direction. Using expressions (2.6)-(2.9) into (2.12), one gets the total mass balance and transport in all dimensions.

$$\frac{\partial u}{\partial t} = \sum_{i=1}^{3}\left(-v_i.\frac{\partial u}{\partial x_i} + (\mathscr{D}_{dis})_i.\frac{\partial^2 u}{\partial x_i^2}\right), \tag{2.13}$$

where, $v_1 = r$, $v_2 = v$, $v_3 = w$, $(\mathscr{D}_{dis})_1 = (\mathscr{D}_{dis})_x$, $(\mathscr{D}_{dis})_2 = (\mathscr{D}_{dis})_y$ and $(\mathscr{D}_{dis})_3 = (\mathscr{D}_{dis})_z$.

Equation (2.13) can be rearranged into the form of the 3D nonlinear ADR model, one obtains

$$\frac{\partial u}{\partial t}+r.\frac{\partial u}{\partial x}+v.\frac{\partial u}{\partial y}+w.\frac{\partial u}{\partial z}=(\mathscr{D}_{dis})_x.\frac{\partial^2 u}{\partial x^2}+(\mathscr{D}_{dis})_y.\frac{\partial^2 u}{\partial y^2}+(\mathscr{D}_{dis})_z.\frac{\partial^2 u}{\partial z^2}. \tag{2.14}$$

It is noticeable that, equation (2.14) demonstrates that the advection processes are governed by the velocity. The model (2.14) is also developed primarily for a non-conservative material, which can be expressed in the following form:

$$\frac{\partial u}{\partial t}+r.\frac{\partial u}{\partial x}+v.\frac{\partial u}{\partial y}+w.\frac{\partial u}{\partial z}=(\mathscr{D}_{dis})_x.\frac{\partial^2 u}{\partial x^2}+(\mathscr{D}_{dis})_y.\frac{\partial^2 u}{\partial y^2}+(\mathscr{D}_{dis})_z.\frac{\partial^2 u}{\partial z^2}+\left(\frac{\partial u}{\partial t}\right)_{\substack{Reaction\\Kinetics}}.$$

We thus consider that the nonlinear ADR problem with the external sources given by (2.14) is

$$\begin{aligned}\frac{\partial u}{\partial t}+r.\frac{\partial u}{\partial x}+v.\frac{\partial u}{\partial y}+w.\frac{\partial u}{\partial z} &= (\mathscr{D}_{dis})_x.\frac{\partial^2 u}{\partial x^2}+(\mathscr{D}_{dis})_y.\frac{\partial^2 u}{\partial y^2}+(\mathscr{D}_{dis})_z.\frac{\partial^2 u}{\partial z^2}\\ &+ \left(\frac{\partial u}{\partial t}\right)_{\substack{Reaction\\Kinetics}}\pm\left(\frac{\partial u}{\partial t}\right)_{External}.\end{aligned} \tag{2.15}$$

Here, we mention that the diffusion coefficient $\mathscr{D}_{dis}$ and the velocity v are assumed to be constant. Then the 3D nonlinear ADR model leads to

$$\frac{\partial u}{\partial t}(x,t)=\bigtriangledown.(\mathscr{D}_{dis}\ \bigtriangledown u)-\vec{v}.\bigtriangledown u+f_1(\chi,t), \tag{2.16}$$

where f_1 is the source function for $\chi=(x,y,z)$, $\bigtriangledown=\left(\frac{\partial}{\partial x},\frac{\partial}{\partial y},\frac{\partial}{\partial z}\right)$ is the gradient, $\bigtriangledown.=\text{div}$ is the divergence operator and $\vec{v}=(r,v,w)$ is the velocity in three dimensions. We implement the current methods to solve these problems by using various numerical approaches, in the following chapters.

3

METHODS

In the field of computational mathematics, the approximate methods are most used ones to solve problem (1.3)-(1.4). The numerical analysis leads us to find the approximate solutions for the nonlinear ADR problems, while the current models are not easy to obtain their analytical solutions. Development of numerical methods for seeking accurate and efficient solutions of these models with small values of the viscosity and free parameters, still remains as a challenging task. To compute the solutions of the proposed problem, we have here developed various combined methods which attempt to combine a natural, spline and B-spline cubic methods in space and backward differentiation formula scheme in time. In the last few decades, spline functions are defined as piecewise polynomial functions being fundamental tools for numerical methods to get solutions of differential equations because of their smoothness and well behavior. The corresponding natural splines are cubic splines whose second derivatives at the boundary points are zero and minimizing strain energy. The B-splines are special spline functions that can be used to define piecewise polynomials by satisfying an appropriate linear combination. The spline functions have minimal support corresponding to the domain partition, degree and smoothness. In the present work, due to aforementioned advantages, we propose three newly combined methods; the BDFS, the SPRK54S and the modified cubic B-splines-SSPRK54 methods. For the last two decades, concepts of synchronization and chaos provide some tools for analyzing nonlinear problems and dynamical systems, with the goal of establishing conditions under which synchronization can occur in such problems. Besides, we further study synchronization behaviors of the nonlinear ADR and well defined chaotic problems. Thence, the concept of the GS of chaotic systems is studied by considering two new techniques for constructing the chaotic synchronization between two identical or non-identical problems.

3.1 Spatial Variation

3.1.1 Splines

In this section, we briefly give some properties concerning interpolating cubic spline, B-spline and natural spline. The first reference to the spline functions in the field of mathematics was given by Schoenberg [105]. Later on, the splines became the most important tool in various fields of mathematics such as approximation theory, numerical analysis and partial differential equations, etc. The main idea of spline functions are defined as piecewise polynomial functions which are the fundamental tool for approximation schemes to obtain the solution of differential equations because of their smoothness and well-posedness. The B-splines are special spline functions that can be used to define piecewise polynomial by satisfying an appropriate linear combination. They have minimal support with respect to a given degree and smoothness. They are also used as basis functions to solve many practical problems, for more details see [42, 152, 165].

Let $\mathscr{C}^l[a,b]$ denotes the classical space of l-times continuously differentiable functions on the interval $[a,b]$. We consider Ω_m as a set of $m+7$ points (see Figures 3.1).

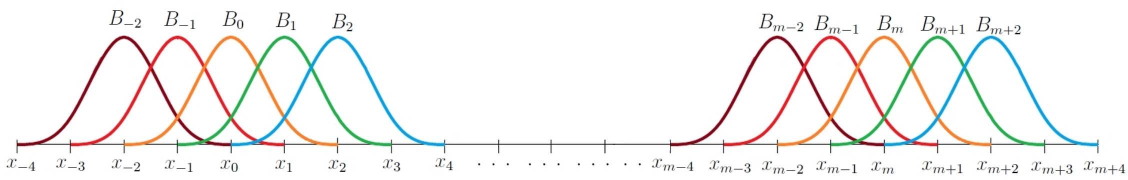


Figure 3.1 B-spline functions with nonuniform knots

$$x_{-3} < x_{-2} < x_{-1} < a = x_0 < x_1 < \ldots < x_m = b < x_{m+1} < x_{m+2} < x_{m+3}, \qquad (3.1)$$

where $x_i = a + ih$ for $i = -3, \ldots, m+3$ with $h = \dfrac{b-a}{m}$. The subset $\{x_0, \ldots, x_m\}$ is a uniform partition of the interval $[a,b] \subset \mathbb{R}$. Let f_s be a function defined on the interval $[a,b]$. The cubic spline s_h interpolating the function f_s at points $x_0, \ldots, x_m$ is the unique function in $\mathscr{C}^2[a,b]$ satisfying the following conditions

$$\begin{cases} s_h(x_i) = f_s(x_i) \quad \text{for} \quad i = 0, \ldots, m, \\ \\ s_h''(a) = s_h''(b), \end{cases} \qquad (3.2)$$

and minimizing the following energy

$$E_s(u) = \int_a^b [u''(x)]^2 dx,$$

where u is an arbitrary function in $\mathscr{C}^2[a,b]$. The cubic spline is the unique function s_h for which

$$\int_a^b [s_h''(x)]^2 dx \leq \int_a^b [u''(x)]^2 dx,$$

holds among all twice continuously differentiable function $u \in \mathscr{C}^2[a,b]$ interpolating the function f_s at the points $x_0, \ldots, x_m$ and satisfying the condition $u''(a) = u''(b)$. Let $\mathscr{S}(\Omega_m)$ denote the space of all cubic splines over the set Ω_m. Recall that the dimension of this space is $dim(\mathscr{S}(\Omega_m)) = m+3$. Then, we recall that the fundamental B-spline function is here the cubic-spline at nodes $-2, -1, 0, 1, 2$, supported by the interval $[-2, 2]$ and is given by the following expression

$$B(x) = \begin{cases} 0 & \text{if} \quad x < -2 \quad \text{or} \quad x \geq 2, \\ \frac{1}{6}(2+x)^3 & \text{if} \quad -2 \leq x < -1, \\ \frac{1}{6}(4-6x^2-3x^3) & \text{if} \quad -1 \leq x < 0, \\ \frac{1}{6}(4-6x^2+3x^3) & \text{if} \quad 0 \leq x < 1, \\ \frac{1}{6}(2-x)^3 & \text{if} \quad 1 \leq x < 2. \end{cases} \tag{3.3}$$

The well-known B-spline functions B_i for $i = -1, \ldots, m+1$ are defined by

$$B_i(x) = B\Big(\frac{x - x_i}{h}\Big). \tag{3.4}$$

The set $\{B_{-1}, \ldots, B_{m+1}\}$ is a basis of the space $\mathscr{S}(\Omega_m)$. The interval support of the function B_i is $[x_{i-2}, x_{i+2}]$ (see Figures 3.2). The values of the B-splines B_i and their derivatives at points x_i are summarized in Table 3.1.

A cubic spline function $s \in \mathscr{S}(\Omega_m)$ over the set Ω_m can be written as a linear combination of the cubic B-splines as

$$s(x) = \sum_{i=-1}^{m+1} \alpha_i B_i(x), \ \forall x \in [a, b]. \tag{3.5}$$

For the interpolating cubic spline s_h satisfying the conditions (3.2) we have

Table 3.1 B-spline values and its derivatives at points x_i

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

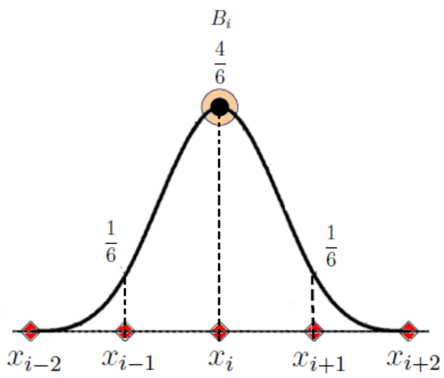


Figure 3.2 Cubic B-spline values at x_i

$$s_h(x_k) = \sum_{i=-1}^{m+1} \alpha_i B_i(x_k) = f_s(x_k), \ 0 \le k \le m, \tag{3.6}$$

with

$$s_h''(a) = \frac{1}{h^2}\alpha_{-1} - \frac{2}{h^2}\alpha_0 + \frac{1}{h^2}\alpha_1, \quad \text{and} \quad s_h''(b) = \frac{1}{h^2}\alpha_{m-1} - \frac{2}{h^2}\alpha_m + \frac{1}{h^2}\alpha_{m+1}.$$

Now, we consider the natural cubic splines which require that the second derivatives vanishing at the boundaries of the interval $[a, b]$. So, the boundary conditions $s_h''(a) = s_h''(b) = 0$ lead to

$$\alpha_{-1} = 2\alpha_0 - \alpha_1 \quad \text{and} \quad \alpha_{m+1} = 2\alpha_m - \alpha_{m-1}. \tag{3.7}$$

By taking into account the interpolating conditions at boundary points $x_0 = a$ and $x_m = b$, we obtain

$$s_h(x_0) = \frac{1}{6}\Big(\alpha_{-1} + 4\alpha_0 + \alpha_1\Big) = f_s(x_0),$$

$$s_h(x_m) = \frac{1}{6}\Big(\alpha_{m-1} + 4\alpha_m + \alpha_{m+1}\Big) = f_s(x_m),$$

together with relations (3.7), we obtain

$$\alpha_0 = f_s(x_0) \quad \text{and} \quad \alpha_m = f_s(x_m). \tag{3.8}$$

Considering the interpolating conditions $s_h(x_i) = f_s(x_i)$ for $i = 1, \ldots, m-1$ and the values given in Table 3.1, we can compute the rest of the coefficients $\alpha_1, \ldots, \alpha_{m-1}$ by solving the linear system $\mathscr{A}_1 \, \phi_s = \Phi_s$ of size $(m-1)\times(m-1)$, where the vectors $\phi_s = (\alpha_1, \ldots, \alpha_{m-1})^T$ and $\Phi_s = (\Phi_{s,1}, \ldots, \Phi_{s,m-1})^T$ with $\Phi_{s,1} = f_s(x_1) - \frac{1}{6} f_s(x_0)$, $\Phi_{s,i} = f_s(x_i)$ for $i = 2, \ldots, m-2$ and $\Phi_{s,m-1} = f_s(x_{m-1}) - \frac{1}{6} f_s(x_m)$. The matrix $\mathscr{A}_1$ is given by

$$\mathscr{A}_1 = \frac{1}{6}\begin{bmatrix} 4 & 1 & 0 & \cdots & 0 \\ 1 & 4 & 1 & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & 1 & 4 & 1 \\ 0 & \cdots & 0 & 1 & 4 \end{bmatrix}. \tag{3.9}$$

As pointed out in reference [165], the following result on a priori bounds for the interpolation error is expressed as follows:

Theorem 3.1. *Let f_s be a function belonging to $\mathscr{C}^4[a,b]$ and s_h be the interpolating cubic spline satisfying the conditions (3.2). Then, for $l = 0, 1, 2, 3$, there exists a non-negative constant $C_k > 0$ such that*

$$||f_s^{(l)} - s_h^{(l)}||_\infty \leq C_k h^{4-l} ||f_s^{(4)}||_\infty \ ,$$

where $||f_s||_\infty$ is the classical L_∞ - norm in $\mathscr{C}[a,b]$ given by

$$||f_s||_\infty = \sup_{x\in[a,b]} |f_s^{(4)}(x)| \,.$$

Even when the interpolated function f_s is only in $\mathscr{C}^2[a,b]$, we have the following a prior bounds for the interpolation error with respect to L_∞ - norm.

Theorem 3.2. *Let f_s be a function belonging to $\mathscr{C}^2[a,b]$ and s_h be the interpolating cubic spline satisfying conditions (3.2). Then, there exists a nonnegative constant $C > 0$ such that*

$$||f_s - s_h||_\infty \leq Ch^2 ||f_s^{(2)}||_\infty \,.$$

Remark 1. The previous theorem is also valid when f_s belongs to the classical Sobolev space $W^{2,\infty}]a,b[$.

This theorem illustrates the perfectiveness of the approximation by the cubic spline. Indeed, by interpolating a continuous twice differentiable function only at discrete points on the interval $[a,b]$, the derivatives of the cubic spline up to the second order are also good approximation of the derivatives of the function f_s. Now, we give a description of the current method for solving the nonlinear ADR equations (1.3)-(1.4). We use the proposed method based on a natural spline defined as a combination of the classical cubic and splines with coefficients depending on the time, by substituting the approximations of the derivatives. The required solution of (1.3) is approximated by a cubic interpolating spline in the following form

$$s_h(x,t)=\sum_{i=-1}^{m+1}\alpha_i(t)B_i(x), \tag{3.10}$$

where $\alpha_i(t)$ are the unknown time dependent coefficients. Let us take the following vector valued functions,

$$\mathbb{B}(x)=\begin{bmatrix} B_1(x)\\ \vdots \\ B_{m-1}(x)\end{bmatrix} \quad \text{and} \quad \phi(t)=\begin{bmatrix} \phi_1(t)\\ \vdots \\ \phi_{m-1}(t)\end{bmatrix}=\begin{bmatrix} \alpha_1(t)\\ \vdots \\ \alpha_{m-1}(t)\end{bmatrix}, \tag{3.11}$$

of size $(m-1)\times 1$. The function $s_h(x,t)$ and their derivatives have the following form

$$\begin{cases} s_h(x,t) &= \alpha_{-1}(t)B_{-1}(x)+\alpha_0(t)B_0(x)+\mathbb{B}(x)^T\phi(t)+\alpha_m(t)B_m(x)\\ &+ \ \alpha_{m+1}(t)B_{m+1}(x),\\ \dfrac{\partial s_h}{\partial t}(x,t) &= \alpha'_{-1}(t)B_{-1}(x)+\alpha'_0(t)B_0(x)+\mathbb{B}(x)^T\phi^{'}(t)+\alpha'_m(t)B_m(x)\\ &+ \ \alpha'_{m+1}(t)B_{m+1}(x),\\ \dfrac{\partial s_h}{\partial x}(x,t) &= \alpha_{-1}(t)B'_{-1}(x)+\alpha_0(t)B'_0(x)+\mathbb{B}'(x)^T\phi(t)+\alpha_m(t)B'_m(x)\\ &+ \ \alpha_{m+1}(t)B'_{m+1}(x),\\ \dfrac{\partial^2 s_h}{\partial x^2}(x,t) &= \alpha_{-1}(t)B''_{-1}(x)+\alpha_0(t)B''_0(x)+\mathbb{B}''(x)^T\phi(t)+\alpha_m(t)B''_m(x)\\ &+ \ \alpha_{m+1}(t)B''_{m+1}(x). \end{cases} \tag{3.12}$$

The current method consists of substituting u and its derivatives in (1.3) by the expression of s_h and its derivatives given by (3.12). So, by evaluating the equation at points x_i for $i=0,\dots,m$, we reach the following relations. For each external points

x_0 and x_m, we have

$$\frac{\partial s_h}{\partial t}(x_0,t) = a_2\frac{\partial^2 s_h}{\partial x^2}(x_0,t) + a_1\frac{\partial s_h}{\partial x}(x_0,t) + a_0 s_h(x_0,t) + F(\phi(t),x_0,t), \tag{3.13}$$

$$\frac{\partial s_h}{\partial t}(x_m,t) = a_2\frac{\partial^2 s_h}{\partial x^2}(x_m,t) + a_1\frac{\partial s_h}{\partial x}(x_m,t) + a_0 s_h(x_m,t) + F(\phi(t),x_m,t), \tag{3.14}$$

where F is the function representing the nonlinear part. The natural spline conditions and the relations (3.7) and (3.8), give rise to

$$\begin{aligned} \alpha_0(t) &= u(x_0,t) = g_1(t), \\ \alpha_m(t) &= u(x_m,t) = g_2(t), \\ \alpha_{-1}(t) &= 2\alpha_0(t) - \alpha_1(t), \\ \alpha_{m+1}(t) &= 2\alpha_m(t) - \alpha_{m-1}(t). \end{aligned} \tag{3.15}$$

By taking the relations (3.13)-(3.15), we obtain

$$\begin{aligned} \alpha_0'(t) &= \big(a_0 + \frac{a_1}{h}\big)g_1(t) - \frac{a_1}{h}\alpha_1(t) + F(\phi(t),x_0,t), \\ \alpha_m'(t) &= \big(a_0 - \frac{a_1}{h}\big)g_2(t) + \frac{a_1}{h}\alpha_{m-1}(t) + F(\phi(t),x_m,t). \end{aligned} \tag{3.16}$$

Now, from (3.12) and (3.16), by evaluating the equation at points x_1 and x_{m-1}, we reach

$$\begin{aligned} \mathbb{B}(x_1)^T\phi'(t) &= \big(\frac{2a_0}{3} + \frac{a_1}{6h} - \frac{2a_2}{h^2}\big)\phi_1(t) + \big(\frac{a_0}{6} - \frac{a_1}{2h} + \frac{a_2}{h^2}\big)\phi_2(t) + \big(\frac{a_2}{h^2} + \frac{a_1}{3h}\big)g_1(t) \\ &+ F(\phi(t),x_1,t) - \frac{1}{6}F(\phi(t),x_0,t), \end{aligned} \tag{3.17}$$

and

$$\begin{aligned} \mathbb{B}(x_{m-1})^T\phi'(t) &= \big(\frac{a_0}{6} + \frac{a_1}{2h} + \frac{a_2}{h^2}\big)\phi_{m-2}(t) + \big(\frac{2a_0}{3} - \frac{a_1}{6h} - \frac{2a_2}{h^2}\big)\phi_{m-1}(t) \\ &+ \big(\frac{a_2}{h^2} - \frac{a_1}{3h}\big)g_2(t) + F(\phi(t),x_{m-1},t) - \frac{1}{6}F(\phi(t),x_m,t). \end{aligned} \tag{3.18}$$

Also, at points x_i for $i = 2,\dots,m-2$, we obtain

$$\mathbb{B}(x_i)^T\phi'(t) = \Big(a_2\mathbb{B}''(x_i) + a_1\mathbb{B}'(x_i) + a_0\mathbb{B}(x_i)\Big)^T\phi(t) + F(\phi(t),x_i,t). \tag{3.19}$$

For $i = 2, \ldots, m-2$, use of Table 3.1 leads to

$$a_2\mathbb{B}''(x_i) + a_1\mathbb{B}'(x_i) + a_0\mathbb{B}(x_i) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \dfrac{a_0}{6} + \dfrac{a_1}{2h} + \dfrac{a_2}{h^2} \quad \longrightarrow i-1 \\ \dfrac{2a_0}{3} - \dfrac{2a_2}{h^2} \quad \longrightarrow i \\ \dfrac{a_0}{6} - \dfrac{a_1}{2h} + \dfrac{a_2}{h^2} \quad \longrightarrow i+1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{3.20}$$

The approximating cubic spline s_h must also satisfy initial condition (1.4) at points $x_0, \ldots, x_m$ and at initial time t_0 :

$$\begin{cases} s_h(x_0, t_0) &= u_0(x_0), \quad \text{for} \quad i = 0, \\ s_h(x_i, t_0) &= u_0(x_i), \quad \text{for} \quad i = 1, ..., m-1, \\ s_h(x_m, t_0) &= u_0(x_m), \quad \text{for} \quad i = m. \end{cases} \tag{3.21}$$

By virtue of (3.12) and the relations (3.15), we end up with the condition

$$\mathscr{A}_1 \phi(t_0) = \phi_0, \tag{3.22}$$

where ϕ_0 is the vector given by

$$\phi_0 = [u_0(x_1) - \frac{1}{6}u_0(x_0), u_0(x_2), \ldots, u_0(x_{m-2}), u_0(x_{m-1}) - \frac{1}{6}u_0(x_m)]^T,$$

and the matrix $\mathscr{A}_1$ of size $(m-1) \times (m-1)$ is given in (3.9).
Now, equations (3.17), (3.18), (3.19) and (3.22) can be written more compactly as follows:

$$\begin{cases} \mathscr{A}_1 \dfrac{d\phi(t)}{dt} &= D\phi(t) + \Phi(\phi(t)), \\ \\ \mathscr{A}_1 \phi(t_0) &= \phi_0, \end{cases} \tag{3.23}$$

where matrix D of size $(m-1)\times(m-1)$ is

$$D=\begin{bmatrix} d_0+\frac{a_1}{6h} & d_1 & 0 & \cdots & 0 \\ d_1' & d_0 & d_1 & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & d_1' & d_0 & d_1 \\ 0 & \cdots & 0 & d_1' & d_0-\frac{a_1}{6h} \end{bmatrix}. \tag{3.24}$$

For $d_0=\frac{2a_0}{3}-\frac{2a_2}{h^2}$, $d_1=\frac{a_0}{6}-\frac{a_1}{2h}+\frac{a_2}{h^2}$, and $d_1'=\frac{a_0}{6}+\frac{a_1}{2h}+\frac{a_2}{h^2}$. The vector valued function Φ is given by

$$\Phi(\phi(t))=[\Phi_1(\phi(t)),\Phi_2(\phi(t)),\ldots,\Phi_{m-2}(\phi(t)),\Phi_{m-1}(\phi(t))]^T$$
$$\Phi_1(\phi(t))=\left(\frac{a_2}{h^2}+\frac{a_1}{3h}\right)g_1(t)+F(\phi(t),x_1,t)-\frac{1}{6}F(\phi(t),x_0,t),$$
$$\Phi_{m-1}(\phi(t))=\left(\frac{a_2}{h^2}-\frac{a_1}{3h}\right)g_2(t)+F(\phi(t),x_{m-1},t)-\frac{1}{6}F(\phi(t),x_m,t),$$
$$\Phi_i(\phi(t))=F(\phi(t),x_i,t)\ \text{ for }\ i=2,\ldots,m-2.$$

Next section presents the modified cubic B-spline scheme to approximate the solution of the nonlinear ADR equation in space.

3.1.2 Modified Cubic B-spline-SSPRK54

Here, we have developed a striking numerical method for solving the nonlinear ADR equation (1.3)-(1.4). To achieve this, we accept a modified cubic B-spline approximation in space. It produces a system of first ODEs and obtains always a diagonal matrix. We do not meet the question of the linearizion and transformation processes. Consider the mesh points $a=x_0<x_1<\ldots<x_m=b$ with uniform length h. Our numerical scheme for solving (1.3) is to find an approximation $s_m(x,t)$ to the exact solution $u(x,t)$ which can be expressed in terms of the cubic B-splines as trial functions as given in equation (3.10). The cubic B-spline $B_j(x)$ with required properties at the knots are given by [197]

$$B_j(x) = \frac{1}{h^3} \begin{cases} 0 & x < x_{j-2} \text{ or } x \geq x_{j-2}, \\ (x - x_{j-2})^3 & x_{j-2} \leq x < x_{j-1}, \\ h^3 + 3h^2(x - x_{j-1}) + 3h(x - x_{j-1})^2 - 3(x - x_{j-1})^3 & x_{j-1} \leq x < x_j, \\ h^3 + 3h^2(x_{j+1} - x) + 3h(x_{j+1} - x)^2 - 3(x_{j+1} - x)^3 & x_j \leq x < x_{j+1}, \\ (x_{j+2} - x)^3 & x_{j+1} \leq x < x_{j+2} \\ 0 & otherwise, \end{cases} \tag{3.25}$$

where the set of splines $\{B_{-1}, B_0, \ldots, B_m, B_{m+1}\}$ construct a basis over the domain $[a, b]$. By using the spline function (3.10) and cubic splines (3.25), the values of B-spline $B_j(x)$ and their derivatives can be calculated at nodes x_j in term of the time parameters α_j by (3.12) where $s_{mj} = s_m(x_j)$. In order to obtain a tridiagonal matrix system of differential equations, we have defined new cubic B-spline basis functions to solve equation (1.3) as follows:

$$\begin{array}{rcll} \mathscr{B}_0(x) & = & B_0(x) & j = 0 \\ \mathscr{B}_1(x) & = & B_1(x) - B_{-1}(x) & j = 1 \\ \mathscr{B}_j(x) & = & B_j(x) & j = 2, 3, \ldots, m-2 \\ \mathscr{B}_{m-1} & = & B_{m-1}(x) - B_{N+1}(x) & j = m-1 \\ \mathscr{B}_m & = & B_m(x) & j = m. \end{array} \tag{3.26}$$

Now assume the approximation solution is given by

$$s_m(x) = \sum_{j=0}^{m} \alpha_j(t) \mathscr{B}_j(x) \quad \forall x \in [a, b]. \tag{3.27}$$

Then, we apply the proposed method to obtain approximate solution (3.27) with the modified set of cubic B-splines given by (3.26) at the knots. The rest of our numerical scheme for solving the nonlinear ADR equation can be seen in the following sections.

3.1.3 Generalized Synchronization (GS)

The problem of chaotic synchronization is related to trajectories starting arbitrarily and close to each other as the time tends to infinity. Identical synchronization of two chaotic systems may occur when the systems are coupled or when one chaotic system drives another chaotic system [90, 156]. However, many real systems are in general nonidentical due to the parameters of two coupled systems do not match, or the coupled systems belong to different classes. So, the possibility of the transformation between drive and response dynamical variables these include the GS can be very complicated. This issue may pose a trouble in practical application of synchronized chaos. Our aim is to analyse synchronization of a coupled chaotic identical and nonidentical dynamical systems producing generalized synchronization in drive-response systems. Thus, we have investigated general methods to detect the existence of the transformation and study this kind of synchronous behavior. In the case of the drive-response methods, efforts to a systematic method that guides the development of solutions to synchronization problems, when trajectories of driving and response systems are strongly connected, then two close states in the space state of the response system correspond to two close states in the space of the driving system. Here, we consider two approaches for constructing chaotic unidirectionally synchronization between the two systems. The systems are either both identical or both nonidentical or each one different from the other. First, we apply the classical Lyapunov stability theory in synchronization of real systems. Secondly, we study a case when the nonlinear part of response system is required to be smooth enough. Then, we use the expansion of such a function to establish the global synchronization of the chaotic dynamical systems. We present that, these techniques can be implemented directly to any experiments and does not require mutual feedback.

Let us consider the first approach (1.5), by assuming the functions H and G as sum of linear and nonlinear parts given by

$$H(x(t)) = Q_1 x(t) + f_g(x(t)) \quad \text{and} \quad G(y(t)) = Q_2 y(t) + g_g(y(t)),$$

where the matrices Q_1 and Q_2 of size $n \times n$ and $m \times m$ are assumed to consist of constants, respectively. The functions $f_g : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ and $g_g : \mathbb{R}^m \longrightarrow \mathbb{R}^m$ represent the nonlinear parts of H and G, respectively. Some of the outputs from the driver system are used to drive the response system. This means that, there exists a relation between the two coupled systems, which could be a smooth function $\Upsilon : \mathbb{R}^n \longrightarrow \mathbb{R}^m$, transforms the trajectories on the attractor of the first system into those on the attractor of the second system. We assume that the driver system in (1.5) is unstable at their equilibrium points. It is suitable to introduce the error system e given by

$$e(t) = y(t) - \Upsilon(x(t)).$$

Definition 3.1. *System (1.5) is global generalized synchronization with respect to vector function Υ, if the controller function ψ exists and satisfies the following property:*

$$\lim_{t \longrightarrow \infty} \|e(t)\| = \lim_{t \longrightarrow \infty} \|y(t) - \Upsilon(x(t))\| = 0\,, \tag{3.28}$$

for all initial conditions.

One can consider the Lyapunov function given by

$$\Lambda(t) = \frac{1}{2} e(t)^T Q_3\, e(t). \tag{3.29}$$

The notation $()^T$ stands for the transpose operator and Λ is positive definite function and is independent of time. In a practical example, we select the matrix Q_3 starts to be equal to the identity matrix. We assume that the error system $e(t)$ is small enough and satisfies a differential equation of the form

$$\dot{e}(t) = -Q_4(t)e(t), \tag{3.30}$$

where Q_4 is an appropriate matrix. We have

$$\dot{e}(t) = \dot{y}(t) - \mathscr{J}_\Upsilon(x(t))\dot{x}(t) = Q_2\, y(t) + g_g(y(t)) + \psi(x(t), y(t)) - \mathscr{J}_\Upsilon(x(t))H(x(t)), \tag{3.31}$$

where $\mathscr{J}_\Upsilon$ is the Jacobian matrix of the function Υ. According to condition (3.30) it follows that, the corresponding controller function ψ exists and is given by

$$\psi(x(t), y(t)) = -Q_4(t)e(t) + \mathscr{J}_\Upsilon(x(t))H(x(t)) - Q_2 y(t) - g_g(y(t)). \tag{3.32}$$

Then, system (1.5) becomes

$$\begin{cases} \dot{x}(t) &= H(x(t)) & \text{driver,} \\ \dot{y}(t) &= -Q_4(t)e(t) + \mathscr{J}_\Upsilon(x(t))H(x(t)) & \text{response.} \end{cases} \tag{3.33}$$

Thus, we have the following results:

Theorem 3.3. *Assume that*

(i) Υ *is a continuously differentiable function,*

(ii) The matrix $Q_4^T(t)Q_3 + Q_3Q_4(t)$ *is a positive definite matrix.*

Then, system (3.33) is a global generalized synchronization with respect to the vector function Υ.

Proof: The derivative of the Lyapunov function Λ is given by

$$\begin{aligned}\dot{\Lambda}(t) &= \frac{1}{2}\Big(\dot{e}(t)^T Q_3 e(t) + e(t)^T Q_3 e(t)\Big) = \frac{-1}{2}\Big((Q_4(t)e(t))^T Q_3 + e^T Q_3 Q_4(t)e(t)\Big),\\ &= \frac{-1}{2} e^T(t)\Big(Q_4^T(t)Q_3 + Q_3 Q_4(t)\Big)e(t).\end{aligned} \tag{3.34}$$

Since $Q_4^T(t)Q_3 + Q_3Q_4(t)$ is a positive definite matrix and from the Lyapunov stability theory, it follows that $||e(t)|| \longrightarrow 0$ as $t \longrightarrow \infty$ and system (1.5) is globally generalized synchronous with respect to the vector function Υ. It is also possible to consider another hypothesis which guarantees the global generalized synchronization of chaotic systems.

In the second approach, we assume that function g_g in (1.5) is sufficiently smooth and $||e(t)||$ is small enough. So, we have the following expansion,

$$g_g(y(t)) = g_g(\Upsilon(x(t)) + e(t)) = g_g(\Upsilon(x(t))) + J_g(\Upsilon(x(t)))e(t) + o(||e(t)||),$$

where $J_g(\Upsilon(x(t)))$ is the Jacobian matrix of g_g at point $\Upsilon(x(t))$. It follows that $g_g(y(t))$ may be approximated by the sum $g_g(\Upsilon(x(t))) + J_g(\Upsilon(x(t)))e(t)$. The response system in (1.5) can be approximated by

$$\dot{y}(t) = Q_2 y(t) + g_g(\Upsilon(x(t))) + J_g(\Upsilon(x(t)))e(t) + \psi(x(t), y(t)).$$

Then,

$$\dot{e}(t) = Q_2 y(t) + g_g(\Upsilon(x(t))) + J_g(\Upsilon(x(t)))e(t) + \psi(x(t), y(t)) - \mathscr{J}_\Upsilon(x(t))H(x(t)) = -Q_4(t)e(t).$$

One can thus obtain

$$\psi(x(t), y(t)) = -\Big(J_g(\Upsilon(x(t))) + Q_4(t)\Big)e(t) + \mathscr{J}_\Upsilon(x(t))H(x(t)) - Q_2\, y(t) - g_g(\Upsilon(x(t))). \tag{3.35}$$

Here, system (1.5) becomes

$$\begin{cases} \dot{x}(t) = H(x(t)) & \text{driver,}\\ \dot{y}(t) = g_g(y(t)) - g_g(\Upsilon(x(t))) - \Big(J_g(\Upsilon(x(t))) + Q_4(t)\Big)e(t) & \\ \qquad + \mathscr{J}_\Upsilon(x(t))H(x(t)) & \text{response.}\end{cases} \tag{3.36}$$

Now, if we set $Q_5(t) = Q_4(t) - Q_2$ and if matrix $Q_4(t)$ commutes with Q_2 then $Q_5(t)$

commutes with Q_2 and the solution of the differential equation (3.30) is given by

$$e(t) = e^{-Q_5(t)}\omega(t), \tag{3.37}$$

where $\omega(t)$ is the solution of the differential system

$$\dot{\omega}(t) = -Q_2\, \omega(t).$$

The solution $\omega(t)$ satisfies the condition

$$||\omega(t)|| \leq C_1\, e^{\lambda_L\, t}, \tag{3.38}$$

where λ_L denotes the maximum Lyapunov exponent of the response system in (1.5) and C_1 is a positive constant. Furthermore, we assume that matrix Q_5 satisfies the condition

$$||e^{-Q_5(t)}|| \leq C_2\, e^{-\vartheta(t)}, \tag{3.39}$$

where C_2 is a positive constant and function ϑ is assumed to be a non-negative function satisfying the following property

$$\lim_{t\to+\infty} \frac{\vartheta(t)}{t} = \ell > \lambda_L. \tag{3.40}$$

Thus, we reach the following results:

Theorem 3.4. *Assume that*

(i) Υ *and* g_g *are continuously differentiable functions,*

(ii) Matrices Q_2 *and* $Q_4(t)$ *commute and matrix* $Q_5(t) = Q_4(t) - Q_2$ *satisfies conditions (3.39)-(3.40).*

Then, system (3.36) is global generalized synchronization with respect to vector function Υ.

Proof: From (3.37), we have

$$||e(t)|| \leq ||e^{-Q_5(t)}||\, ||w(t)||.$$

According to (3.38) and (3.39) it follows that

$$||e(t)|| \leq C\, e^{\lambda_L\, t - \vartheta(t)}, \tag{3.41}$$

where C is a positive constant. The property (3.40) gives that $||e(t)|| \longrightarrow 0$ as $t \longrightarrow \infty$, for any set of initial conditions. Hence we have completed the proof of system (3.36) that it is global generalized synchronization with respect to the vector function Υ.

Remark 2. In a practical example we select matrix Q_5 to be independent of time in the form $Q_5 = k_2 I_m$, where I_m is the identity matrix of size $m \times m$ and k_2 is a coupling parameter of synchronization. So, matrix Q_5 commutes with any matrix and we have $Q_4 = Q_2 + k_2 I_m$. It follows that

$$e^{-Q_5 t} = e^{-k_2 t} I_m,$$

and

$$||e^{-Q_5 t}|| = e^{-k_2 t}.$$

In this case, function ϑ is given by $\vartheta(t) = k_2 t$. Condition (3.39) is satisfied and we have thus condition (3.41) in the form

$$||e(t)|| \leq C\, e^{(\lambda_L - k_2)t}.$$

Condition (3.40) is satisfied for

$$k_2 > \lambda_L,$$

where the maximum Lyapunov exponent is approximately equal to the largest eigenvalue of matrix Q_2. To ensure that $||e(t)||$ is small enough for all t, and the value of the parameter k_2 must be large enough.

Thence, the proposed algorithms replace equations (1.3)-(1.4) and system (1.5) by an ODE system. Then, we solve the resulting system in time by the following schemes.

3.2 Temporal Variation

In the literature, it is possible to find several methods to solve the resulting ODEs (3.23), (3.33) and (3.36) in time. It is noticeable that, these problems are highly nonlinear equations because they present the interaction between reaction, convection and diffusion mechanisms [67] and contain free parameters. Since stiffness is a property of differential equations widely varying time scales which means some components of the solution decay much more rapidly than others. So, the explicit methods do not work with stiff problems or even if work they are extremely slow. Due to stiffness of the obtained ODEs, in this study we focus on the BDF and SSPRK54 methods for solving the resulting ODEs in time. The BDF method is one of the most important tool to solve differential equations. For comparison purposes, we also provide the SSPRK54 method for solving ODEs in time. Note that in this method, the SSP property also guarantees the stability properties which are necessary in the numerical solutions of ODEs.

3.2.1 BDF

Backward differentiation formulae (BDF) are implicit multi-step methods for numerically solving the initial-value problems (3.23). They are the most widely used methods for solving ODEs due to their stability properties. In addition, the BDF formulae are based on numerical differentiation. The time interval $[t_0, T]$ is divided into N subintervals with the time step $\Delta t = \dfrac{T - t_0}{N}$ with the knots $t_n = t_0 + n\,\Delta t$ for $n = 0, \ldots, N$. The BDF method applied to (3.23) gives rise to the following approximations

$$\mathscr{A}_1 \phi_n - \tau h\big[D\phi_n + \Phi(\phi_n)\big] - \sum_{j=0}^{p} \eta_j A \phi_{n-j} = 0, \tag{3.42}$$

where $\phi_n = [\phi_{1,n}, \ldots, \phi_{m-1,n}]^T$ is an approximation obtained by the BDF method of vector $\phi(t)$ given by (3.11) at $t = t_n$. The coefficients η_j and τ are given in Table 3.2 for the p-step BDF formula.

Table 3.2 Coefficients of the BDF p-step method for $p = 6$

p	τ	η_0	η_1	η_2	η_3	η_4	η_5	η_6
1	1	-1						
2	$\frac{2}{3}$	$\frac{4}{3}$	$\frac{-1}{3}$					
3	$\frac{6}{11}$	$\frac{18}{11}$	$\frac{-9}{11}$	$\frac{2}{11}$				
4	$\frac{22}{25}$	$\frac{48}{25}$	$\frac{-36}{25}$	$\frac{16}{25}$	$\frac{-3}{25}$	$\frac{3}{25}$		
5	$\frac{60}{137}$	$\frac{300}{137}$	$\frac{-300}{137}$	$\frac{200}{137}$	$\frac{-75}{137}$	$\frac{-12}{137}$	$\frac{-12}{137}$	
6	$\frac{60}{137}$	$\frac{300}{137}$	$\frac{-300}{137}$	$\frac{200}{137}$	$\frac{-75}{137}$	$\frac{-72}{147}$	$\frac{-75}{147}$	$\frac{10}{147}$

At each time step n, we have to solve equation (3.42) for ϕ_n by rearranging in the following form

$$\mathscr{G}(\phi_n) = (\mathscr{A}_1 - \eta_0\, I)\phi_n - \tau h\big[D\phi_n + \Phi(\phi_n)\big] - \sum_{j=1}^{p} \eta_j\, \mathscr{A}_1\, \phi_{n-j} = 0, \tag{3.43}$$

where I is the $(m-1) \times (m-1)$ identity matrix. Equation (3.43) can efficiently be solved by using the Newton method with starting guess taken from the last time step. Here, the Newton method for the approximation of ϕ_n generates iterations (ξ_k) given by

$$\begin{cases} \xi_0 & \\ \xi_{k+1} = \xi_k - [J_{\mathscr{G}}(\xi_k)]^{-1}\mathscr{G}(\xi_k), & k \geqslant 0 \end{cases} \tag{3.44}$$

where $J_{\mathscr{G}}(\xi_k)$ is the Jacobian matrix of $\mathscr{G}$ at point ξ_k. We have

$$J_{\mathscr{G}}(\xi_k) = (\mathscr{A}_1 - \eta_0 I) - \tau h(D + J_{\Phi}(\xi_k)), \tag{3.45}$$

with J_{Φ} being the Jacobian matrix of Φ. The value of the interpolating spline s_h given by (3.10) at time t_n is

$$s_h(x,t_n) = \alpha_{-1}(t_n)B_{-1}(x) + \alpha_0(t_n)B_0(x) + \mathbb{B}(x)^T y(t) + \alpha_m(t_n)B_m(x) + \alpha_{m+1}(t_n)B_{m+1}(x).$$

Here the coefficients $\alpha_i(t_n)$ are not known exactly. But they are approximated by the coefficients denoted by $\widehat{\alpha}_{i,n}$ and which are computed by the BDF method and given by

$$\begin{array}{lcccl} \widehat{\alpha}_{i,n} & = & \phi_{i,n}, \quad i = 1,\ldots,m-1, & & \\ \widehat{\alpha}_{0,n} & = & u(x_0,t_n) & = & g_1(t_n), \\ \widehat{\alpha}_{m,n} & = & u(x_m,t_n) & = & g_2(t_n), \\ \widehat{\alpha}_{-1,n} & = & 2\widehat{\alpha}_{0,n} - \phi_{1,n} & = & 2g_1(t_n) - \phi_{1,n}, \\ \widehat{\alpha}_{m+1,n} & = & 2\widehat{\alpha}_{m,n} - \phi_{m-1,n} & = & 2g_2(t_n) - \phi_{m-1,n}. \end{array} \tag{3.46}$$

The value $s_h(x,t_n)$ approximated by spline s_h given by (3.10) at time t_n for $n = 0,\ldots,N$ are expressed in terms of the values $\widehat{s}_{n,h}(x)$ where $\widehat{s}_{n,h}$ be the cubic spline as

$$\widehat{s}_{n,h}(x) = \sum_{i=-1}^{m+1} \widehat{\alpha}_{i,n} B_i(x).$$

We have thus $s_h(x,t_n) \simeq \widehat{s}_{n,h}(x)$ for all $x \in [a,b]$.

3.2.1.1 Convergence of the BDFS method

Here, we give a result on the convergence of the proposed method. For the sake of simplicity, we consider that the errors stemmed from the Newton method in the BDF method are neglected. Our study is based on the following theorem which gives some results on the convergence of the BDF method applied to ODEs of type (3.23). The theorem can be found in reference [79].

Theorem 3.5. *For* $\dfrac{d\phi(t)}{dt} = \mathscr{A}_1^{-1} D\phi(t) + \mathscr{A}_1^{-1}\Phi(\phi(t))$ *if* $\mathscr{A}_1^{-1}D$ *is diagonalizable and the derivatives of* Φ *are bounded (up to order* $p+1$*). Then, the p-step BDF method is* *convergent for* $t_{n+p} \in [t_0, T]$ *and*

$$\|\phi(t_{n+p}) - \phi_{n+p}\|_\infty \leq C_p\Big(e^{(n+2)C_p\Delta t} \max_{i=0,\ldots,p-1} \|\phi(t_i) - \phi_i\|_\infty + \frac{(\Delta t)^p \max\limits_{t\in[t_0,T]} \|\phi^{(p+1)}(t)\|_\infty (e^{C_p \Delta t(n+1)} - 1)}{\min\limits_{i=2,\ldots,m-1} |\eta_p - \Delta t \upsilon_i|}\Big),$$

where $C_p > 0$ *is a non-negative constant and* υ_i *are the eigenvalues of the matrix* $\mathscr{A}_1^{-1}D$.

The following theorems give error estimates when the function $u(\,.\,, t_n)$ is approximated by the function $\widehat{s}_{n,h}$. In the rest of this section we assume that $\mathscr{A}_1^{-1}D$ is diagonalizable and the derivatives of Φ are bounded (up to order $p+1$).

Theorem 3.6. *We assume that the solution u of (1.3) is such that the functions* $u(\,.\,, t_n) : x \in [a, b] \longmapsto u(x, t_n)$ *are in* $\mathscr{C}^2[a, b]$ *for* $n = 0, \ldots, N$. *Then, we have the error bounds*

$$\|u(\,.\,, t_n) - \widehat{s}_{n,h}\|_\infty \leq Ch^2 \|\frac{\partial^2 u(\,.\,, t_n)}{\partial x^2}\|_\infty + (m+1)\,\|\phi(t_n) - \phi_n\|_\infty\,, \tag{3.47}$$

where $C > 0$ *is a non-negative constant.*

Proof: For all $x \in [a, b]$, we have

$$|u(x, t_n) - \widehat{s}_{n,h}(x)| \leq |u(x, t_n) - s_h(x, t_n)| + |s_h(x, t_n) - \widehat{s}_{n,h}(x)|.$$

By virtue of Theorem 3.2, we get the error estimates

$$|u(x, t_n) - s_h(x, t_n)| \leq Ch^2 \|\frac{\partial^2 u(\,.\,, t_n)}{\partial x^2}\|_\infty.$$

We recall that

$$s_h(x, t_n) - \widehat{s}_{n,h}(x) = \sum_{i=-1}^{m+1} \Big(\alpha_i(t_n) - \widehat{\alpha}_{i,n}\Big) B_i(x).$$

Since $0 \leq B_i(x) \leq 1$, it follows that

$$|s_h(x, t_n) - \widehat{s}_{n,h}(x)| \leq \sum_{i=-1}^{m+1} \Big|\alpha_i(t_n) - \widehat{\alpha}_{i,n}\Big|.$$

So, by considering (3.7) and (3.8), we reach

$$\begin{aligned} \sum_{i=-1}^{m+1} \Big|\alpha_i(t_n) - \widehat{\alpha}_{i,n}\Big| &= 2|\alpha_1(t_n) - \widehat{\alpha}_{1,n}| + \sum_{i=2}^{m-2} |\alpha_i(t_n) - \widehat{\alpha}_{i,n}| + 2|\alpha_{m-1}(t_n) - \widehat{\alpha}_{m-1,n}| \\ &\leq (m+1)\,\|\phi(t_n) - \phi_n\|_\infty. \end{aligned}$$

Then, for all $x \in [a,b]$, we obtain

$$|u(x,t) - \widehat{s}_{n,h}(x)| \leq Ch^2 \| \frac{\partial^2 u(\,.\,,t_n)}{\partial x^2} \|_\infty + (m+1)\, \|\phi(t_n) - \phi_n\|_\infty \,.$$

It follows that

$$\|u(\,.\,,t_n) - \widehat{s}_{n,h}\|_\infty \leq Ch^2 \| \frac{\partial^2 u(\,.\,,t_n)}{\partial x^2} \|_\infty + (m+1)\, \|\phi(t_n) - \phi_n\|_\infty \,, \tag{3.48}$$

for $n = 0, \ldots, N$.

Theorem 3.7. *If we assume that the solution u of (1.3) is such that the functions $u(\,.\,,t_n)$: $x \in [a,b] \longmapsto u(x,t_n)$ are in $\mathscr{C}^4[a,b]$ for $n = 0,\ldots,N$. Then, we have the error bounds*

$$\|u(\,.\,,t_n) - \widehat{s}_{n,h}\|_\infty \leq Ch^4 \| \frac{\partial^4 u(\,.\,,t_n)}{\partial x^4} \|_\infty + (m+1)\, \|\phi(t_n) - \phi_n\|_\infty \,, \tag{3.49}$$

where $C > 0$ is a non-negative constant.

Proof: The proof of this theorem is similar to Theorem 3.6 by applying Theorem 3.1 instead of Theorem 3.2.

Theorem 3.8. *We assume that the solution u of (1.3) is such that the functions $u(\,.\,,t_n)$: $x \in [a,b] \longmapsto u(x,t_n)$ are in $\mathscr{C}^2[a,b]$ for $n = 0,\ldots,N$. Then, we have the error bounds*

$$\begin{aligned} \|u(.,t_n) - \widehat{s}_{n,h}\|_\infty \;&\leq\; C\Big(h^2 \| \frac{\partial^2 u(\,.\,,t_n)}{\partial x^2} \|_\infty \\ &+\; (m+1)\, e^{C\Delta t}\Big(\max_{i=0,\ldots,p-1} \|\phi(t_i) - \phi_i\|_\infty + \frac{(\Delta t)^p \max\limits_{t\in[t_0,T]} \|\phi^{(p+1)}(t)\|_\infty}{\min\limits_{i=2,\ldots,m-1} |\eta_p - (\Delta t)\upsilon_i|} \Big)\Big), \end{aligned} \tag{3.50}$$

where $C > 0$ is a non-negative constant. Furthermore, if $\Delta t \leq \epsilon_0$ and $h \leq h_0$ where ϵ_0 and h_0 are sufficiently small non-negative constants and if the starting values for the BDF method are supposed to be in a sufficiently small neighborhood of the exact solution then the convergence of the proposed method holds.

Proof: According to Theorem 3.6, we have

$$\|u(\,.\,,t_n) - \widehat{s}_{n,h}\|_\infty \leq Ch^2 \| \frac{\partial^2 u(\,.\,,t_n)}{\partial x^2} \|_\infty + (m+1)\, \|\phi(t_n) - \phi_n\|_\infty \,. \tag{3.51}$$

Using Theorem 3.5, the p-step BDF method is convergent for $t_{n+p} \in [t_0, T]$ and

$$\begin{aligned}\|\phi(t_{n+p}) - \phi_{n+p}\|_\infty &\leq C_p\Big(e^{(n+2)C_p\Delta t} \max_{i=0,\dots,p-1} \|\phi(t_i) - \phi_i\|_\infty \\ &+ \frac{(\Delta t)^p \max_{t\in[t_0,T]} \|\phi^{(p+1)}(t)\|_\infty (e^{C_p\Delta t(n+1)} - 1)}{\min_{i=2,\dots,m-1} |\eta_p - \Delta t \upsilon_i|}\Big),\end{aligned}$$

where $C_p > 0$ is a non-negative constant and υ_i are the eigenvalues of the matrix $A^{-1}D$. It follows that

$$\|\phi(t_{n+p}) - \phi_{n+p}\|_\infty \leq C_p e^{(N+2)C_p\Delta t}\Big(\max_{i=0,\dots,p-1} \|\phi(t_i) - \phi_i\|_\infty + \frac{(\Delta t)^p \max_{t\in[t_0,T]} \|\phi^{(p+1)}(t)\|_\infty}{\min_{i=2,\dots,m-1} |\eta_p - \Delta t \upsilon_i|}\Big). \tag{3.52}$$

By considering (3.51) and (3.52), we obtain the error estimate (3.50). The error estimates hold for $\Delta t \leq \epsilon_0$ and $h \leq h_0$ where ϵ_0 and h_0 are sufficiently small non-negative constants. Inequality (3.50) shows that if the starting values are supposed to be in a sufficiently small neighbourhood of the exact solution then the convergence of the method holds.

3.2.2 SSPRK54

Now, we present the SSPRK54 methods to numerically approximate the solution of the ODE (3.23), (3.33) and (3.36). The SSPRK54 method has order at most four. However, we pay attention to the optimal five-stage, fourth order method [206]. The SSP is a more suitable approach in high order time discretization schemes preserve the strong stability properties in any norm of the spatial discretization with first-order Euler time stepping. In order to have stability when using explicit numerical schemes, we require to apply the CFL (Courant–Friedrichs–Lewy) condition [141]. Thus, the optimal SSPRK54 scheme is made more efficient by the CFL. The optimally of this scheme is guaranteed by using an approach based on global optimization. Therefore, the proposed method needs less storage space and low cost. In addition this is why we interested in the SSPRK54 scheme. For starting the current scheme, let the time interval $[t_0, T]$ is divided into N subintervals as previously mentioned. At each time step n, we have to solve ϕ_n of equation (3.23) and rearrange it in the following form

$$\mathscr{R}(\phi_n) = (\mathscr{A}_1 - I)\phi_n - \big[D\phi_n + \Phi(\phi_n)\big] = 0, \tag{3.53}$$

where I is an $(m-1)\times(m-1)$ identity matrix thus,

$$
\begin{aligned}
\phi_1 &= \phi_n + 0.391752226571890\Delta t\mathscr{R}(\phi_n) \\
\phi_2 &= 0.444370493651235\phi_n + 0.555629506348765\phi_1 + 0.368410593050371\Delta t\mathscr{R}(\phi_1) \\
\phi_3 &= 0.620101851488403\phi_n + 0.379898148511597\phi_2 + 0.251891774271694\Delta t\mathscr{R}(\phi_2) \\
\phi_4 &= 0.178079954393132\phi_n + 0.821920045606868\phi_3 + 0.544974750228521\Delta t\mathscr{R}(\phi_3) \\
\phi_{n+1} &= 0.517231671970585\phi_2 + 0.096059710526147\phi_3 + 0.063692468666290\Delta t\mathscr{R}(\phi_3) \\
&+ 0.386708617503269\phi_4 + 0.226007483236906\Delta t\mathscr{R}(\phi_4).
\end{aligned}
$$

The efficiency and accuracy of the BDFS, SSPRK54S, modified B-spline-SSPRK54 and GS methods have been tested for different cases of the nonlinear ADR and chaotic dynamical problems, in later chapters.

4

IMPLEMENTATION TO NONLINEAR ADR EQUATIONS AND CHAOTIC SYSTEMS

In this chapter, we demonstrate the applicability of the previous methods to some model problems of nonlinear ADR equations with initial and boundary conditions and well-defined chaotic systems. The proposed schemes for solving the models (1.3)-(1.4) can be categorized in three essential groups: the BDFS, SSPRK54S and modified cubic B-spline-SSPRK54 methods. The proposed methods are realized to be efficient for these types of nonlinear ADR physical problems. Moreover, we demonstrate the effectiveness of the proposed control function via GS method. We address the problem of synchronization of identical and nonidentical chaotic systems (1.5) by considering physical and biological problems. Then, the concepts between the properties of chaotic and coupled nonlinear ADR problems are going to be discussed as well.

4.1 Numerical Solutions of Nonlinear ADR Equations by using BDFS and SSPRK54S Schemes

For the approximate solution of the nonlinear ADR problems (1.3)-(1.4), we accept the BDFS and SSPRK54S techniques in different cases. The generalized Burgers-Fisher equation with forcing terms (GBFEF) and the generalized Burgers-Huxley equation with forcing terms (GBHEF) can be considered to be good examples of the nonlinear ADR models. They present the high importance for describing the interaction between diffusion and transports, convection and reaction mechanisms.

4.1.1 GBFEF

The GBFEF was first studied by Fisher, with free of forcing term, to describe the propagation of gene in a habitat [39, 199]. The GBFEF as the dynamic spread of a combustion front was presented by Kolmogorov et al. [14]. Consider the GBFEF of the form

$$\frac{\partial u}{\partial t} - \lambda \frac{\partial^2 u}{\partial x^2} + \gamma_1 u^\delta \frac{\partial u}{\partial x} - \gamma_2 u(1 - u^\delta) - f(x,t) = 0, \quad (x,t) \in \Omega_m = [a,b] \times [t_0, T], \tag{4.1}$$

with the initial and boundary conditions given by

$$u(x, t_0) = u_0(x), \tag{4.2}$$

$$u(a,t) = g_1(t), \quad u(b,t) = g_2(t). \tag{4.3}$$

The functions g_1, g_2 and the initial function u_0 are known. The λ, γ_1, γ_2 are real parameters, δ is a positive integer, $0 < \lambda \leq 1$ and $0 < C \leq 1$. The structure of the GBFEF can be seen as a useful model for describing the relation between the reaction mechanisms, convection effect and diffusion transport. It also arises in various fields such as financial mathematics, turbulence, fluid mechanics, traffic flow, shock waves and gas dynamics. In this example, the presented numerical schemes in solving (4.1) are to find an approximation $s_h(x,t)$ to the exact solution $u(x,t)$ given in equation (3.10). By rearranging equation (4.1) as the form of (1.3), we obtain linear part and nonlinear part, involving the forcing term, respectively as

$$\mathscr{L}(\frac{\partial^2 u}{\partial x^2}, \frac{\partial u}{\partial x}, u, x, t) = \lambda u_{xx} + \gamma_2 u,$$

and

$$\mathscr{N}(\frac{\partial^2 u}{\partial x^2}, \frac{\partial u}{\partial x}, u, x, t) = -\gamma_1 u^\delta u_x - \gamma_2 u u^\delta + f(x,t).$$

Now, $a_0 = \gamma_2$, $a_1 = 0$ and $a_2 = \lambda$. Considering the relations (3.16), we obtain

$$\begin{aligned} \alpha_0'(t) &= \gamma_2 g_1(t) + F(\phi(t), x_0, t), \\ \alpha_m'(t) &= \gamma_2 g_2(t) + F(\phi(t), x_m, t), \end{aligned} \tag{4.4}$$

where

$$F(\phi(t), x_0, t) = \left(1 - (g_1(t))^\delta\right) - \frac{\gamma_1}{h}(g_1(t))^\delta \left(g_1(t) - \alpha_1(t)\right) + f(x_0, t),$$

and

$$F(\phi(t), x_m, t) = \left(1 - (g_2(t))^\delta\right) - \frac{\gamma_1}{h}(g_2(t))^\delta \left(\alpha_{m-1}(t) - g_2(t)\right) + f(x_m, t).$$

Thus, by evaluating equations (3.12) and (4.4) at points x_1 and x_{m-1}, one obtains

$$\frac{4}{6}\alpha_1'(t)+\frac{1}{6}\alpha_2'(t) = \big(\frac{2\gamma_2}{3}-\frac{2\lambda}{h^2}\big)\alpha_1(t)+\big(\frac{\gamma_2}{6}+\frac{\lambda}{h^2}\big)\alpha_2(t)+\big(\frac{\lambda}{h^2}\big)g_1(t) + F(\phi(t),x_1,t)-\frac{1}{6}F(\phi(t),x_0,t), \tag{4.5}$$

$$\frac{1}{6}\alpha_{m-2}'(t)+\frac{4}{6}\alpha_{m-1}'(t) = \big(\frac{\gamma_2}{6}+\frac{\lambda}{h^2}\big)\alpha_{m-2}(t)+\big(\frac{2\gamma_2}{3}-\frac{2\lambda}{h^2}\big)\alpha_{m-1}(t)+\big(\frac{\lambda}{h^2}\big)g_2(t) + F(\phi(t),x_{m-1},t)-\frac{1}{6}F(\phi(t),x_m,t). \tag{4.6}$$

At points x_i, $i=2,\ldots,m-2$, we give

$$\frac{1}{6}\alpha_{i-1}'+\frac{4}{6}\alpha_i'+\frac{1}{6}\alpha_{i+1}' = \big(\frac{\gamma_2}{6}+\frac{\lambda}{h^2}\big)\alpha_{i-1}(t)+\big(\frac{2\gamma_2}{3}-\frac{2\lambda}{h^2}\big)\alpha_i(t)+\big(\frac{\gamma_2}{6}-\frac{\lambda}{h^2}\big)\alpha_{i+1}(t), + F(\phi(t),x_i,t), \tag{4.7}$$

where

$$\begin{aligned}
F(\phi(t),x_1,t) &= -\frac{\gamma_1}{2h}\big(\frac{1}{6}g_1(t)+\frac{4}{6}\alpha_1(t)+\frac{1}{6}\alpha_2(t)\big)^{\delta}\big(g_1(t)-\alpha_2(t)\big)\\
&- \frac{\gamma_2}{6}\big(g_1(t)+4\alpha_1(t)+\alpha_2(t)\big)\big(\frac{1}{6}g_1(t)+\frac{4}{6}\alpha_1(t)+\frac{1}{6}\alpha_2(t)\big)^{\delta}\big)+f(x_1,t),\\
F(\phi(t),x_{m-1},t) &= -\frac{\gamma_1}{2h}\big(\frac{1}{6}\alpha_{m-2}(t)+\frac{4}{6}\alpha_{m-1}(t)+\frac{1}{6}g_2(t)\big)^{\delta}\big(\alpha_{m-2}(t)-g_2(t)\big)\\
&- \big(\frac{1}{6}\alpha_{m-2}(t)+\frac{4}{6}\alpha_{m-1}(t)+\frac{1}{6}g_2(t)\big)^{\delta}\big)+f(x_{m-1},t),\\
F(\phi(t),x_i,t) &= -\frac{\gamma_1}{2h}\big(\frac{1}{6}\alpha_{i-1}(t)+\frac{4}{6}\alpha_i(t)+\frac{1}{6}\alpha_{i+1}(t)\big)^{\delta}\big(\alpha_{i-1}(t)-\alpha_{i+1}(t)\big)\\
&- \frac{\gamma_2}{6}\big(\alpha_{i-1}(t)+4\alpha_i(t)+\alpha_{i+1}(t)\big)-\big(\frac{1}{6}\alpha_{i-1}(t)+\frac{4}{6}\alpha_i(t)+\frac{1}{6}\alpha_{i+1}(t)\big)^{\delta}+f(x_i,t).
\end{aligned}$$

The approximating cubic spline s_h must also satisfy the initial condition (4.2) at points $x_0,\ldots,x_m$ and at initial time t_0 :

$$\begin{cases} s_h(x_0,t_0) = u_0(x_0), & \text{for } i=0,\\ s_h(x_i,t_0) = u_0(x_i), & \text{for } i=1,...,m-1,\\ s_h(x_m,t_0) = u_0(x_m), & \text{for } i=m. \end{cases} \tag{4.8}$$

By virtue of (3.7), (3.8) and (3.12), we obtain

$$\mathscr{A}_1\, \phi(t_0) = \phi_0, \tag{4.9}$$

where $\phi_0 = [6u_0(x_1) - g_1(t_0), 6u_0(x_2), \ldots, 6u_0(x_{m-2}), 6u_0(x_{m-1}) - g_2(t_0)]^T$ and the matrix $\mathscr{A}_1$ of size $(m-1) \times (m-1)$ is given by (3.9).
Now, equations (4.5), (4.6), (4.7) and (4.9) are expressed as in the following ODEs

$$\begin{cases} \mathscr{A}_1 \dfrac{d\phi(t)}{dt} &= D\phi(t) + \Phi(\phi(t)), \\ \\ \mathscr{A}_1\, \phi(t_0) &= \phi_0. \end{cases} \tag{4.10}$$

The matrix D of size $(m-1)\times(m-1)$ is given (3.24) for $d_0 = \dfrac{2\gamma_2}{3} - \dfrac{2\lambda}{h^2}$, $d_1 = \dfrac{\gamma_2}{6} + \dfrac{\lambda}{h^2}$ and $d_1' = \dfrac{\gamma_2}{6} + \dfrac{\lambda}{h^2}$. The vector valued function Φ is given by

$$\Phi(\phi(t)) = [\Phi_1(\phi(t)), \Phi_2(\phi(t)), \ldots, \Phi_{m-2}(\phi(t)), \Phi_{m-1}(\phi(t))]^T,$$

$$\Phi_1(\phi(t)) = (\frac{\lambda}{h^2})g_1(t) + F(\phi(t), x_1, t) - \frac{1}{6}F(\phi(t), x_0, t),$$

$$\Phi_{m-1}(y(t)) = (\frac{\lambda}{h^2})g_2(t) + F(\phi(t), x_{m-1}, t) - \frac{1}{6}F(\phi(t), x_m, t),$$

$$\Phi_i(\phi(t)) = F(\phi(t), x_i, t) \ \text{ for } \ i = 2, \ldots, m-2.$$

Now, we solve the system (4.10) by using the BDF and SSPRK54 methods, as mentioned in previous chapter.

4.1.2 GBHEF

The GBHEF being a nonlinear ADR model is of high importance for presenting the interaction between advection, diffusion, reaction and transports mechanisms. First, the GBHEF equation was investigated in references [126, 167], with free of forcing term. The GBHEF can be presented by the following form:

$$\frac{\partial u}{\partial t} - \lambda \frac{\partial^2 u}{\partial x^2} + \gamma_1 u^\delta \frac{\partial u}{\partial x} - \gamma_2 u(1-u^\delta)(u^\delta - C) - f(x,t) = 0, \quad (x,t) \in \Omega_m = [a,b]\times[t_0,T], \tag{4.11}$$

with the initial and boundary conditions given by

$$u(x, t_0) = u_0(x), \tag{4.12}$$

$$u(a,t) = g_1(t), \quad u(b,t) = g_2(t). \tag{4.13}$$

Here, λ, γ_1, γ_2 and δ are physical constants. In this example, we use the proposed numerical schemes to find an approximation $s_h(x,t)$ to the exact solution $u(x,t)$ given by (3.10). By rearranging equation (4.11) in the form of (1.3), we can define the linear part and nonlinear part, involving the forcing term, respectively as

$$\mathscr{L}(\frac{\partial^2 u}{\partial x^2}, \frac{\partial u}{\partial x}, u, x, t) = \lambda u_{xx} - \gamma_2\, Cu,$$

$$\mathscr{N}(\frac{\partial^2 u}{\partial x^2}, \frac{\partial u}{\partial x}, u, x, t) = -\gamma_1 u^{\delta} u_x + \gamma_2 u u^{\delta} - \gamma_2 C u - \gamma_2 u u^{2\delta} + \gamma_2 C u u^{\delta} + f(x,t).$$

Now, from the above parts, we get $a_0 = -\gamma_2\, C$, $a_1 = 0$ and $a_2 = \lambda$. Considering the relations (3.16), one finds

$$\begin{aligned} \alpha_0'(t) &= -\gamma_2 C g_1(t) + F(\phi(t), x_0, t), \\ \alpha_m'(t) &= -\gamma_2 C g_2(t) + F(\phi(t), x_m, t), \end{aligned} \tag{4.14}$$

where

$F(\phi(t), x_0, t) = \big(\gamma_2 g_1(t) g_1(t))^{\delta}\big)\big(1 - (g_1(t))^{\delta} + C\big) - \frac{\gamma_1}{h}(g_1(t))^{\delta}\big(g_1(t) - \alpha_1(t)\big) + f(x_0, t)$,

$F(\phi(t), x_m, t) = \big(\gamma_2 g_2(t) g_2(t))^{\delta}\big)\big(1 - (g_2(t))^{\delta} + C\big) - \frac{\gamma_1}{h}(g_2(t))^{\delta}\big(\alpha_{m-1}(t) - g_2(t)\big) + f(x_m, t)$.

Now, from (3.12) and (4.14), by evaluating these equations at points x_1 and x_{m-1}, one obtains

$$\begin{aligned} \frac{4}{6}\alpha_1'(t) + \frac{1}{6}\alpha_2'(t) &= \big(\frac{-2\gamma_2 C}{3} - \frac{2\lambda}{h^2}\big)\alpha_1(t) + \big(\frac{-\gamma_2 C}{6} + \frac{\lambda}{h^2}\big)\alpha_2(t) + \big(\frac{\lambda}{h^2}\big)g_1(t) \\ &+ F(\phi(t), x_1, t) - \frac{1}{6}F(\phi(t), x_0, t), \\ \frac{1}{6}\alpha_{m-2}'(t) + \frac{4}{6}\alpha_{m-1}'(t) &= \big(\frac{\gamma_2}{6} + \frac{\lambda}{h^2}\big)\alpha_{m-2}(t) + \big(\frac{2\gamma_2}{3} - \frac{2\lambda}{h^2}\big)\alpha_{m-1}(t) + \big(\frac{\lambda}{h^2}\big)g_2(t) \\ &+ F(\phi(t), x_{m-1}, t) - \frac{1}{6}F(\phi(t), x_m, t). \end{aligned} \tag{4.15}$$

(4.16)

At points x_i, $i = 2, \ldots, m-2$, we obtain

$$\begin{aligned} \frac{1}{6}\alpha_{i-1}' + \frac{4}{6}\alpha_i' + \frac{1}{6}\alpha_{i+1}' &= \big(\frac{-\gamma_2 C}{6} + \frac{\lambda}{h^2}\big)\alpha_{i-1}(t) + \big(\frac{-2\gamma_2 C}{3} - \frac{2\lambda}{h^2}\big)\alpha_i(t) \\ &+ \big(\frac{-\gamma_2 C}{6} - \frac{\lambda}{h^2}\big)\alpha_{i+1}(t) + F(\phi(t), x_i, t), \end{aligned} \tag{4.17}$$

where

$$\begin{aligned}
F(\phi(t),x_1,t) &= \Big(-\gamma_2\big(g_1(t)+\alpha_1(t)+\alpha_2(t)\big)\big(\frac{1}{6}g_1(t)+\frac{4}{6}\alpha_1(t)+\frac{1}{6}\alpha_2(t)\big)^{\delta}\Big)\\
&\times \Big(\big(\frac{1}{6}g_1(t)+\frac{4}{6}\alpha_1(t)+\frac{1}{6}\alpha_2(t)\big)^{\delta}-C\Big)\\
&- \frac{\gamma_1}{2h}\big(\frac{1}{6}g_1(t)+\frac{4}{6}\alpha_1(t)+\frac{1}{6}\alpha_2(t)\big)^{\delta}\big(g_1(t)-\alpha_2(t)\big),\\
F(\phi(t),x_m,t) &= \Big(-\gamma_2\big(\alpha_{m-2}(t)+4\alpha_{m-1}(t)+g_2(t)\big)\big(1-\big(\frac{1}{6}\alpha_{m-2}(t)+\frac{4}{6}\alpha_{m-1}(t)+\frac{1}{6}g_2(t)\big)^{\delta}\Big)\\
&\times \Big(\big(\frac{1}{6}\alpha_{m-2}(t)+\frac{4}{6}\alpha_{m-1}(t)+\frac{1}{6}g_2(t)\big)^{\delta}-C\Big)\\
&- \frac{\gamma_1}{2h}\big(\frac{1}{6}\alpha_{m-2}(t)+\frac{4}{6}\alpha_{m-1}(t)+\frac{1}{6}g_2(t)\big)^{\delta}\big(\alpha_{m-2}(t)-g_2(t)\big),\\
F(\phi(t),x_i,t) &= \Big(-\gamma_2\big(\alpha_{i-1}(t)+4\alpha_i(t)+\alpha_{i+1}(t)\big)\big(1-\big(\frac{1}{6}\alpha_{i-1}(t)+\frac{4}{6}\alpha_i(t)+\frac{1}{6}\alpha_{i+1}(t)\big)^{\delta}\Big)\\
&- \frac{\gamma_1}{2h}\big(\frac{1}{6}\alpha_{i-1}(t)+\frac{4}{6}\alpha_i(t)+\frac{1}{6}\alpha_{i+1}(t)\big)^{\delta}\big(\alpha_{i-1}(t)-\alpha_{i+1}(t)\big).
\end{aligned}$$

The approximating cubic spline s_h must also satisfy the initial condition (4.2) at points $x_0,\ldots,x_m$ and at initial time t_0 as given in (4.8). Now, equations (4.9), (4.15), (4.16) and (4.17) are expressed compactly as in the following ODEs

$$\begin{cases} \mathscr{A}_1 \dfrac{d\phi(t)}{dt} = D\phi(t)+\Phi(\phi(t)), \\ \mathscr{A}_1\,\phi(t_0) = \phi_0, \end{cases} \tag{4.18}$$

where the matrix D of size $(m-1)\times(m-1)$ is given (3.24) for $d_0 = \dfrac{-2\gamma_2 C}{3} - \dfrac{2\lambda}{h^2}$, $d_1 = \dfrac{-\gamma_2 C}{6} + \dfrac{\lambda}{h^2}$, and $d_1' = \dfrac{-\gamma_2 C}{6} + \dfrac{\lambda}{h^2}$. The vector valued function Φ is given by

$$\begin{gathered}
\Phi(\phi(t)) = [\Phi_1(\phi(t)),\Phi_2(\phi(t)),\ldots,\Phi_{m-2}(\phi(t)),\Phi_{m-1}(\phi(t))]^T,\\
\Phi_1(\phi(t)) = \big(\frac{\lambda}{h^2}\big)g_1(t)+F(\phi(t),x_1,t)-\frac{1}{6}F(\phi(t),x_0,t),\\
\Phi_{m-1}(\phi(t)) = \big(\frac{\lambda}{h^2}\big)g_2(t)+F(\phi(t),x_{m-1},t)-\frac{1}{6}F(\phi(t),x_m,t),\\
\Phi_i(\phi(t)) = F(\phi(t),x_i,t) \ \text{ for } \ i=2,\ldots,m-2.
\end{gathered}$$

Now, we solve the system (4.18) by using the BDF and SSPRK54 methods, as proposed in the previous section.

4.1.3 A New Approach for the Nonlinear Coupled ADR Equations with Source Functions via the BDFS Scheme

This section focuses on analysis of the nonlinear physical phenomena of coupled ADR models via the BDFS scheme. We capture numerical behavior of the physical environment governed by the nonlinear coupled Burgers equations with source functions. It is recognized that these models are characterized by the interaction of reaction and diffusion [69, 85, 191]. The Coupled Burgers equation was first presented by Esipov [221]. It can be seen that examination of the physical and numerical properties of the nonlinear coupled Burgers equation is quite complex. Thus, by preserving the actual physical properties of nature, and not using matrix or tensor products, behaviour of the physical environment governed by the coupled Burgers equation with source functions has thus been investigated effectively. To achieve this, the BDFS method combines the cubic spline defined in space with the BDF scheme in time. Hence this work produces a block matrix system of first ODEs in time. The currently combined approaches are directly applicable to solve our problems without any further transformation. The block matrix system is solved by the BDF scheme which is usually implemented together with the Newton method to solve nonlinear differential equations at each step. The Thomas algorithm is also used in the solution of the linear part of the system obtained as a result of the application of the BDFS method.

Consider the coupled nonlinear Burgers equation with source functions in the form

$$\begin{aligned} \frac{\partial u_1}{\partial t} - \lambda_1 \frac{\partial^2 u_1}{\partial x^2} + \lambda_2 u_1 \frac{\partial u_1}{\partial x} + \gamma_1 (u_1 u_2)_x &= f_2(x,t), \\ \frac{\partial u_2}{\partial t} - \lambda_3 \frac{\partial^2 u_2}{\partial x^2} + \lambda_4 u_2 \frac{\partial u_2}{\partial x} + \gamma_2 (u_1 u_2)_x &= f_3(x,t), \end{aligned} \tag{4.19}$$

with the initial conditions

$$u_1(x,t_0) = u_{1,0}(x), \;\; u_2(x,t_0) = u_{2,0}(x), \tag{4.20}$$

and the boundary conditions

$$\begin{aligned} u_1(a,t) &= g_1(t), \;\; u_1(b,t) = g_2(t), \\ u_2(a,t) &= g_3(t), \;\; u_2(b,t) = g_4(t), \end{aligned} \tag{4.21}$$

where $(x,t) \in \Omega = [a,b] \times [t_0, T]$; λ_1, λ_2, λ_3, λ_4, γ_1 and γ_2 are real parameters. The functions g_1, g_2, g_3 and g_4 are known. The functions f_2 and f_3 are the source terms. The required solutions of (4.19)-(4.21) $u_1(x,t)$ and $u_2(x,t)$ are approximated by the cubic interpolating splines $s_{1,h}$ and $s_{2,h}$ respectively, as

$$
\begin{aligned}
s_{1,h}(x,t) &= \sum_{i=-1}^{m+1} \alpha_i(t) B_i(x), \\
s_{2,h}(x,t) &= \sum_{j=-1}^{m+1} \beta_j(t) B_j(x),
\end{aligned}
\tag{4.22}
$$

where $\alpha_i(t)$ and $\beta_j(t)$ are unknowns based on the time. The cubic splines $s_{1,h}$ and $s_{2,h}$ are the unique functions in $\mathscr{C}^2([a,b])$ and satisfying the following conditions

$$
\begin{cases}
s_{1,h}(x_i,t) = u_1(x_i,t) \quad \text{for} \quad i = 0,\ldots,m, \\
s''_{1,h}(a,t) = s''_{1,h}(b,t),
\end{cases}
\tag{4.23}
$$

$$
\begin{cases}
s_{2,h}(x_j,t) = u_2(x_j,t) \quad \text{for} \quad j = 0,\ldots,m, \\
s''_{2,h}(a,t) = s''_{2,h}(b,t).
\end{cases}
\tag{4.24}
$$

By using the cubic splines $s_{1,h}$ and $s_{2,h}$ in conditions (4.23) and (4.24), one can obtain

$$
s_{1,h}(x_k,t) = \sum_{i=-1}^{m+1} \alpha_i(t) B_i(x_k) = u_1(x_k,t),\ 0 \le k \le m,
\tag{4.25}
$$

$$
s_{2,h}(x_k,t) = \sum_{j=-1}^{m+1} \beta_j(t) B_j(x_k) = u_2(x_k,t),\ 0 \le k \le m,
\tag{4.26}
$$

with

$$
s''_{1,h}(a,t) = \frac{1}{h^2}\alpha_{-1} - \frac{2}{h^2}\alpha_0 + \frac{1}{h^2}\alpha_1, \quad \text{and} \quad s''_{1,h}(b,t) = \frac{1}{h^2}\alpha_{m-1} - \frac{2}{h^2}\alpha_m + \frac{1}{h^2}\alpha_{m+1},
$$

$$
s''_{2,h}(a,t) = \frac{1}{h^2}\beta_{-1} - \frac{2}{h^2}\beta_0 + \frac{1}{h^2}\beta_1, \quad \text{and} \quad s''_{2,h}(b,t) = \frac{1}{h^2}\beta_{m-1} - \frac{2}{h^2}\beta_m + \frac{1}{h^2}\beta_{m+1}.
$$

According to the natural cubic splines at the boundary conditions, the expression

$$
\begin{cases}
\alpha_{-1}(t) = 2\alpha_0(t) - \alpha_1(t), \\
\alpha_{m+1}(t) = 2\alpha_m(t) - \alpha_{m-1}(t),
\end{cases}
\tag{4.27}
$$

$$
\begin{cases}
\beta_{-1}(t) = 2\beta_0(t) - \beta_1(t), \\
\beta_{m+1}(t) = 2\beta_m(t) - \beta_{m-1}(t),
\end{cases}
\tag{4.28}
$$

can be obtained. Taking into account of the interpolating conditions at the boundary points $x_0 = a$ and $x_m = b$ yields

$$s_{1,h}(x_0,t) = \frac{1}{6}\Big(\alpha_{-1}(t) + 4\alpha_0(t) + \alpha_1(t)\Big) = u_1(x_0,t),$$

$$s_{2,h}(x_0,t) = \frac{1}{6}\Big(\beta_{-1}(t) + 4\beta_0(t) + \beta_1(t)\Big) = u_2(x_0,t),$$

and

$$s_{1,h}(x_m,t) = \frac{1}{6}\Big(\alpha_{m-1}(t) + 4\alpha_m(t) + \alpha_{m+1}(t)\Big) = u_1(x_m,t),$$

$$s_{2,h}(x_m,t) = \frac{1}{6}\Big(\beta_{m-1}(t) + 4\beta_m(t) + \beta_{m+1}(t)\Big) = u_2(x_m,t).$$

Then substitution of the above expressions into (4.27) and (4.28) as:

$$\alpha_0(t) = u_1(x_0,t) \quad \text{and} \quad \alpha_m(t) = u_1(x_m,t), \tag{4.29}$$

$$\beta_0(t) = u_2(x_0,t) \quad \text{and} \quad \beta_m(t) = u_2(x_m,t). \tag{4.30}$$

Now, according to the above procedure, numerical solutions of the coupled Burgers equation with source functions are produced. One can then consider the following vector valued functions

$$w(t) = \begin{bmatrix} \phi_1(t) \\ \phi_2(t) \end{bmatrix}, \text{ where } \quad \phi_1(t) = \begin{bmatrix} \alpha_1(t) \\ \vdots \\ \alpha_{m-1}(t) \end{bmatrix} \text{ and } \phi_2(t) = \begin{bmatrix} \beta_1(t) \\ \vdots \\ \beta_{m-1}(t) \end{bmatrix}. \tag{4.31}$$

By substituting $s_{1,h}$ and $s_{2,h}$ with their derivatives u_1 and u_2 in (4.19) at points x_i and x_j for $i = 0$ and m, $j = 0$ and m, one reaches

$$\frac{\partial s_{1,h}}{\partial t}(x_0,t) = \lambda_1 \frac{\partial^2 s_{1,h}}{\partial x^2}(x_0,t) + F_1(\phi_1(t), x_0, t), \tag{4.32}$$

$$\frac{\partial s_{1,h}}{\partial t}(x_m,t) = \lambda_1 \frac{\partial^2 s_{1,h}}{\partial x^2}(x_m,t) + F_1(\phi_1(t), x_m, t), \tag{4.33}$$

and

$$\frac{\partial s_{2,h}}{\partial t}(x_0,t) = \lambda_3 \frac{\partial^2 s_{2,h}}{\partial x^2}(x_0,t) + F_2(\phi_2(t), x_0, t), \tag{4.34}$$

$$\frac{\partial s_{2,h}}{\partial t}(x_m,t) = \lambda_3 \frac{\partial^2 s_{2,h}}{\partial x^2}(x_m,t) + F_2(\phi_2(t), x_m, t), \tag{4.35}$$

where F_1 and F_2 are the functions representing the nonlinear parts. By taking into account the relations (4.27)-(4.35), one obtains

$$\begin{aligned} \alpha_0'(t) &= F_1(w(t), x_0, t), \\ \alpha_m'(t) &= F_1(w(t), x_m, t), \end{aligned} \tag{4.36}$$

where

$$F_1(w(t), x_0, t) = \frac{\lambda_2}{2h} g_1(g_1 - \alpha_1) + \frac{\gamma_1}{h}\big(g_1(g_3 - \beta_1) + g_3(g_1 - \alpha_1)\big) - f_2(x_0, t),$$

$$F_1(w(t), x_m, t) = \frac{\lambda_2}{h} g_2(\alpha_{m-1} - g_2) + \frac{\gamma_1}{h}\big(g_2(\beta_{m-1} - g_4) + g_4(\alpha_{m-1} - g_4)\big) - f_2(x_m, t),$$

$$\begin{aligned} \beta_0'(t) &= F_2(w(t), x_0, t), \\ \beta_m'(t) &= F_2(\alpha(t), \beta(t), x_m, t), \end{aligned} \tag{4.37}$$

$$F_2(w(t), x_0, t) = \frac{\lambda_4}{2h} g_3(g_3 - \beta_1) + \frac{\gamma_2}{h}\big(g_3(g_1 - \alpha_1) + g_1(g_3 - \beta_1)\big) - f_3(x_0, t),$$

$$F_2(w(t), x_m, t) = \frac{\lambda_4}{h} g_4(\beta_{m-1} - g_4) + \frac{\gamma_2}{h}\big(g_4(\alpha_{m-1} - g_4) + g_2(\beta_{m-1} - g_2)\big) - f_3(x_m, t).$$

Now, by evaluating (4.36) and (4.37), in (4.19) at points x_i and x_j for $i = 1, \dots, m-1$, $j = 1, \dots, m-1$, one finds

$$\frac{4}{6}\alpha_1'(t) + \frac{1}{6}\alpha_2'(t) = \frac{-2}{h^2}\alpha_1(t) + \frac{1}{h^2}\alpha_2(t) + \frac{1}{h^2}g_1(t) + F_1(w(t), x_1, t) - \frac{1}{6}F_1(w(t), x_0, t), \tag{4.38}$$

and

$$\begin{aligned} \frac{1}{6}\alpha_{m-2}'(t) + \frac{4}{6}\alpha_{m-1}'(t) &= \frac{1}{h^2}\alpha_{m-2}(t) + \frac{1}{h^2}\alpha_{m-1}(t) + \frac{1}{h^2}g_2(t) + F_1(w(t), x_{m-1}, t) \\ &- \frac{1}{6}F_1(w(t), x_m, t). \end{aligned} \tag{4.39}$$

Thus, at points x_i for $i = 2, \dots, m-2$, we obtain

$$\frac{1}{6}\alpha_{i-1}' + \frac{4}{6}\alpha_i' + \frac{1}{6}\alpha_{i+1}' = \frac{1}{h^2}\alpha_{i-1}(t) + \frac{-2}{h^2}\alpha_i(t) + \frac{1}{h^2}\alpha_{i+1}(t) + F_1(w(t), x_i, t), \tag{4.40}$$

where

$$\begin{aligned} F_1(w(t), x_1, t) &= -\frac{\lambda_2}{2h}\Big(\frac{1}{6}g_1(t) + \frac{4}{6}\alpha_1(t) + \frac{1}{6}\alpha_2(t)\Big)\big(g_1(t) - \alpha_2(t)\big) \\ &- \frac{\gamma_1}{12h}\big((g_1(t) + 4\alpha_1(t) + \alpha_2(t))(g_1 - \alpha_2(t)) \\ &+ (g_1(t) + 4\alpha_1(t) + \alpha_2(t))(g_3 - \beta_2(t))\big) + f_2(x_1, t), \end{aligned}$$

$$
\begin{aligned}
F_1(w(t),x_{m-1},t) &= -\frac{\lambda_2}{2h}\Big(\frac{1}{6}\alpha_{m-2}(t)+\frac{4}{6}\alpha_{m-1}(t)+\frac{1}{6}g_2(t)\Big)(\alpha_{m-2}(t)-g_2(t)) \\
&- \frac{\gamma_1}{12h}\Big((\beta_{m-2}(t)+4\beta_{m-1}(t)+g_4(t))(\alpha_{m-2}(t)-g_2)+(\alpha_{m-2}(t)+4\alpha_{m-1}(t) \\
&+ g_2(t))(\beta_{m-2}(t)-g_4)\Big)+f_2(x_{m-1},t), \\
F_1(w(t),x_i,t) &= -\frac{\lambda_2}{2h}\Big(\frac{1}{6}\alpha_{i-1}(t)+\frac{4}{6}\alpha_i(t)+\frac{1}{6}\alpha_{i+1}(t)\Big)\Big(\alpha_{i-1}(t)-\alpha_{i+1}(t)\Big) \\
&- \frac{\gamma_1}{12h}\Big((\alpha_{i-1}(t)+4\alpha_i(t)+\alpha_{i+1}(t))(\beta_{i-1}(t)-\beta_{i+1}(t))+(\beta_{i-1}(t)+4\beta_i(t) \\
&+ \beta_{i+1}(t))(\alpha_{i-1}(t)-\alpha_{i+1}(t))\Big)+f_2(x_i,t),
\end{aligned}
$$

$$
\frac{4}{6}\beta_1'(t)+\frac{1}{6}\beta_2'(t)=\frac{-2}{h^2}\beta_1(t)+\frac{1}{h^2}\beta_2(t)+\frac{1}{h^2}g_3(t)+F_2(\alpha(t),\beta(t),x_1,t)-\frac{1}{6}F_2(w(t),x_0,t), \tag{4.41}
$$

and

$$
\begin{aligned}
\frac{1}{6}\beta_{m-2}'(t)+\frac{4}{6}\beta_{m-1}'(t) &= \frac{1}{h^2}\beta_{m-2}(t)+\frac{1}{h^2}\beta_{m-1}(t)+\frac{1}{h^2}g_4(t)+F_2(w(t),x_{m-1},t) \\
&- \frac{1}{6}F_2(w(t),x_m,t).
\end{aligned} \tag{4.42}
$$

Then, at points x_j for $j=2,\dots,m-2$, one can have

$$
\frac{1}{6}\beta_{i-1}'+\frac{4}{6}\beta_i'+\frac{1}{6}\beta_{i+1}' = \frac{1}{h^2}\beta_{i-1}(t)+\frac{-2}{h^2}\beta_i(t)+\frac{1}{h^2}\alpha_{i+1}(t)+F_2(w(t),x_i,t), \tag{4.43}
$$

where

$$
\begin{aligned}
F_2(w(t),x_1,t) &= -\frac{\lambda_4}{2h}\Big(\frac{1}{6}g_3(t)+\frac{4}{6}\beta_1(t)+\frac{1}{6}\beta_2(t)\Big)\Big(g_3(t)-\beta_2(t)\Big) \\
&- \frac{\gamma_2}{12h}\Big((g_2(t)+4\beta_1(t)+\beta_2(t))(g_3-\beta_2(t))+(g_2(t)+4\beta_1(t) \\
&+ \beta_2(t))(g_1-\alpha_2(t))\Big)+f_3(x_1,t), \\
F_2(w(t),x_{m-1},t) &= -\frac{\lambda_4}{2h}\Big(\frac{1}{6}\beta_{m-2}(t)+\frac{4}{6}\beta_{m-1}(t)+\frac{1}{6}g_4(t)\Big)(\beta_{m-2}(t)-g_4(t)) \\
&- \frac{\gamma_2}{12h}\Big((\alpha_{m-2}(t)+4\alpha_{m-1}(t)+g_2(t))(\beta_{m-2}(t)-g_4) \\
&+ (\beta_{m-2}(t)+4\beta_{m-1}(t)+g_4(t))(\alpha_{m-2}(t)-g_2)\Big)+f_3(x_{m-1},t),
\end{aligned}
$$

$$
\begin{aligned}
F_2(w(t),x_i,t) &= -\frac{\lambda_4}{2h}\Big(\frac{1}{6}\beta_{i-1}(t)+\frac{4}{6}\beta_i(t)+\frac{1}{6}\beta_{i+1}(t)\Big)\big(\beta_{i-1}(t)-\beta_{i+1}(t)\big) \\
&- \frac{\gamma_2}{12h}\big((\beta_{i-1}(t)+4\beta_i(t)+\beta_{i+1}(t))(\alpha_{i-1}(t)-\alpha_{i+1}(t)) \\
&+ (\alpha_{i-1}(t)+4\alpha_i(t)+\alpha_{i+1}(t))(\beta_{i-1}(t)-\beta_{i+1}(t))\big)+f_3(x_i,t).
\end{aligned}
$$

The approximated cubic splines $s_{1,h}$ and $s_{2,h}$ must also satisfy the initial conditions (4.20) at points $x_0,\dots,x_m$ and at the initial time t_0, one can have

$$
\begin{cases}
s_{1,h}(x_0,t_0) = u_{1,0}(x_0), & \text{for} \quad i=0, \\
s_{1,h}(x_i,t_0) = u_{1,0}(x_i), & \text{for} \quad i=1,...,m-1, \\
s_{1,h}(x_m,t_0) = u_{1,0}(x_m), & \text{for} \quad i=m,
\end{cases}
\tag{4.44}
$$

and

$$
\begin{cases}
s_{2,h}(x_0,t_0) = u_{2,0}(x_0), & \text{for} \quad j=0, \\
s_{2,h}(x_j,t_0) = u_{2,0}(x_j), & \text{for} \quad j=1,...,m-1, \\
s_{2,h}(x_m,t_0) = u_{2,0}(x_m), & \text{for} \quad j=m.
\end{cases}
\tag{4.45}
$$

Thus, by virtue of (4.44) and (4.45), we have

$$
\mathscr{A}_2 w(t_0) = w_0, \tag{4.46}
$$

where $\mathscr{A}_2 = \left[\begin{array}{c|c} \mathscr{A}_1 & 0 \\ \hline 0 & \mathscr{A}_1 \end{array}\right]$ with size $2(m-1)\times 2(m-1)$ and the $(m-1)\times(m-1)$ matrix $\mathscr{A}_1$ is tridiagonal matrix.

$$
w_0 = \begin{bmatrix} \alpha_1(t_0) \\ \vdots \\ \alpha_{m-1}(t_0) \\ \beta_1(t_0) \\ \vdots \\ \beta_{m-1}(t_0). \end{bmatrix} = \begin{bmatrix} 6u_{1,0}(x_1)-g_1(t_0) \\ 6u_{1,0}(x_2) \\ \vdots \\ 6u_{1,0}(x_{m-2}) \\ 6u_{1,0}(x_{m-1})-g_2(t_0) \\ 6u_{2,0}(x_1)-g_3(t_0) \\ 6u_{2,0}(x_2) \\ \vdots \\ 6u_{2,0}(x_{m-2}) \\ 6u_{2,0}(x_{m-1})-g_4(t_0) \end{bmatrix}.
$$

Now, equations (4.38)-(4.43) and (4.46) are summarized as ODEs given by

$$\begin{cases} \mathscr{A}_2 \dfrac{dw(t)}{dt} &= \mathbb{D}w(t) + \Phi(w(t)), \\ \\ \mathscr{A}_2 w(t_0) &= w_0. \end{cases} \tag{4.47}$$

Here the matrix $\mathbb{D} = \left[\begin{array}{c|c} D & 0 \\ \hline 0 & D \end{array}\right]$ is of the size $2(m-1)\times 2(m-1)$ and the $(m-1)\times(m-1)$ matrix D is a tridiagonal matrix (3.24). The vector valued function is then given by

$$\Phi(w(t)) = \begin{bmatrix} \zeta_1(w(t)) \\ \zeta_2(w(t)) \end{bmatrix} = \begin{bmatrix} \zeta_{1,1}w(t) \\ \zeta_{1,2}w(t) \\ \vdots \\ \zeta_{1,m-2}w(t) \\ \zeta_{1,m-1}w(t) \\ \zeta_{2,1}w(t) \\ \zeta_{2,2}w(t) \\ \vdots \\ \zeta_{2,m-2}w(t) \\ \zeta_{2,m-1}w(t) \end{bmatrix},$$

where

$$\zeta_{1,1}(w(t)) = \frac{\lambda_1}{h^2} g_1(t) + F_1(w(t), x_1, t) - \frac{1}{6} F_1(w(t), x_0, t),$$

$$\zeta_1, m-1(w(t)) = \frac{\lambda_1}{h^2} g_2(t) + F_1(w(t), x_{m-1}, t) - \frac{1}{6} F_1(w(t), x_m, t),$$

$$\zeta_{1,i}(w(t)) = F_1(w(t), x_i, t) \quad \text{for} \quad i = 2, \ldots, m-2,$$

and

$$\zeta_{2,1}(w(t)) = \frac{\lambda_3}{h^2} g_3(t) + F_2(w(t), x_1, t) - \frac{1}{6} F_2(w(t), x_0, t),$$

$$\zeta_{2,m-1}(w(t)) = \frac{\lambda_3}{h^2} g_4(t) + F_2(w(t), x_{m-1}, t) - \frac{1}{6} F_2(w(t), x_m, t),$$

$$\zeta_{2,i}(w(t)) = F_2(w(t), x_i, t) \quad \text{for} \quad i = 2, \ldots, m-2.$$

Now, by using the BDF method, the first order ordinary differential equation system (4.47) is solved. The time interval $[t_0, T]$ is divided into N subintervals with time step $\Delta t = \dfrac{T - t_0}{N}$ and knots $t_n = t_0 + n\Delta t$ for $n = 0, \ldots, N$. The BDF method applied to (4.47) gives arise to the following approximations

$$\begin{cases} \mathscr{A}_1 y_n - \tau h\big[D y_n + \zeta_1(w_n)\big] - \sum_{j=0}^{p} \eta_j \mathscr{A}_1 y_{n-j} &= 0, \\ \mathscr{A}_1 z_n - \tau h\big[D z_n + \zeta_2(w_n)\big] - \sum_{j=0}^{p} \eta_j \mathscr{A}_1 z_{n-j} &= 0, \end{cases} \tag{4.48}$$

where $w_n = [\phi_{1,n}, \phi_{2,n}]^T$ is an approximation obtained by the BDF method of the vector $w(t)$ given by (4.31). The coefficients η_j and τ are known. At each time step n, we have to solve equation (4.48) for w_n by rearranging it

$$\begin{cases} \mathscr{G}_1(\phi_{1,n}) = (\mathscr{A}_1 - \eta_0\, I)\phi_{1,n} - \tau h\big[D\phi_{1,n} + \zeta_1(w_n)\big] - \sum_{j=1}^{p} \eta_j \mathscr{A}_1 \phi_{1,n-j} = 0, \\ \mathscr{G}_2(\phi_{2,n}) = (\mathscr{A}_1 - \eta_0\, I)\phi_{2,n} - \tau h\big[D\phi_{2,n} + \zeta_2(w_n)\big] - \sum_{j=1}^{p} \eta_j \mathscr{A}_1 \phi_{2,n-j} = 0, \end{cases} \tag{4.49}$$

where I is the $(m-1)\times(m-1)$ identity matrix. Equation (4.49) can efficiently be solved by using the Newton method. Here, the Newton method for the approximation of w_n generates iterations, ξ_k, given by

$$\begin{cases} \xi_0 & \\ \xi_{k+1} &= \xi_k - [J_{\mathscr{G}}(\xi_k)]^{-1}\mathscr{G}(\xi_k), \qquad k \geqslant 0 \end{cases} \tag{4.50}$$

where, $J_{\mathscr{G}}(\xi_k)$ is the Jacobian matrix of $\mathscr{G}$ at the point ξ_k. We have

$$J_{\mathscr{G}}(\xi_k) = (\mathscr{A}_2 - \eta_0 \mathbb{I}) - \tau h(\mathbb{D} + J_\Phi(\xi_k)), \tag{4.51}$$

where $\mathbb{I} = \left[\begin{array}{c|c} I & 0 \\ \hline 0 & I \end{array}\right]$ is the $2(m-1)\times 2(m-1)$ identity matrix and J_Φ is the Jacobian matrix of Φ given by

$$J_\Phi(w(t)) = \left[\begin{array}{ccc|ccc} \frac{\partial \zeta_{1,1}}{\partial \alpha_1} & \cdots & \frac{\partial \zeta_{1,1}}{\partial \alpha_{m-1}} & \frac{\partial \zeta_{1,1}}{\partial \beta_1} & \cdots & \frac{\partial \zeta_{1,1}}{\partial \beta_{m-1}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \zeta_{1,m-1}}{\partial \alpha_1} & \cdots & \frac{\partial \zeta_{1,m-1}}{\partial \alpha_{m-1}} & \frac{\partial \zeta_{1,m-1}}{\partial \beta_1} & \cdots & \frac{\partial \zeta_{1,m-1}}{\partial \beta_{m-1}} \\ \hline \frac{\partial \zeta_{2,1}}{\partial \alpha_1} & \cdots & \frac{\partial \zeta_{2,1}}{\partial \alpha_{m-1}} & \frac{\partial \zeta_{2,1}}{\partial \beta_1} & \cdots & \frac{\partial \zeta_{2,1}}{\partial \beta_{m-1}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \zeta_{2,m-1}}{\partial \alpha_1} & \cdots & \frac{\partial \zeta_{2,m-1}}{\partial \alpha_{m-1}} & \frac{\partial \zeta_{2,m-1}}{\partial \beta_1} & \cdots & \frac{\partial \zeta_{2,m-1}}{\partial \beta_{m-1}} \end{array}\right].$$

The linear part of the Jacobian matrix whose solution can be solved by using the Thomas algorithm. The approximate solutions $s_{1,h}$ and $s_{2,h}$ given by (4.22) at time t_n

are

$$\begin{cases} s_{1h}(x,t_n) &= \alpha_{-1}(t_n)B_{-1}(x)+\alpha_0(t_n)B_0(x)+\mathbb{B}(x)^T y(t)+\alpha_m(t_n)B_m(x)+\alpha_{m+1}(t_n)B_{m+1}(x), \\ s_{2h}(x,t_n) &= \beta_{-1}(t_n)B_{-1}(x)+\beta_0(t_n)B_0(x)+\mathbb{B}(x)^T y(t)+\beta_m(t_n)B_m(x)+\beta_{m+1}(t_n)B_{m+1}(x). \end{cases}$$

Here, the coefficients $\alpha_i(t_n)$ and $\beta_i(t_n)$ are approximated by $\widehat{\alpha}_{i,n}$ and $\widehat{\beta}_{i,n}$ and given by

$$\begin{array}{lcccl} \widehat{\alpha}_{i,n} &=& y_{i,n}, \quad \text{for}, \quad i=1,\ldots,m-1, && \\ \widehat{\alpha}_{0,n} &=& u_1(x_0,t_n) &=& g_1(t_n), \\ \widehat{\alpha}_{m,n} &=& u_1(x_m,t_n) &=& g_2(t_n), \\ \widehat{\alpha}_{-1,n} &=& 2\widehat{\alpha}_{0,n}-y_{1,n} &=& 2g_1(t_n)-y_{1,n}, \\ \widehat{\alpha}_{m+1,n} &=& 2\widehat{\alpha}_{m,n}-y_{m-1,n} &=& 2g_2(t_n)-y_{m-1,n}, \end{array} \tag{4.52}$$

$$\begin{array}{lcccl} \widehat{\beta}_{i,n} &=& z_{i,n}, \quad \text{for}, \quad i=1,\ldots,m-1, && \\ \widehat{\beta}_{0,n} &=& u_2(x_0,t_n) &=& g_3(t_n), \\ \widehat{\beta}_{m,n} &=& u_2(x_m,t_n) &=& g_4(t_n), \\ \widehat{\beta}_{-1,n} &=& 2\widehat{\beta}_{0,n}-z_{1,n} &=& 2g_3(t_n)-z_{1,n}, \\ \widehat{\beta}_{m+1,n} &=& 2\widehat{\beta}_{m,n}-z_{m-1,n} &=& 2g_4(t_n)-z_{m-1,n}. \end{array} \tag{4.53}$$

The values $s_{1h}(x,t_n)$ and $s_{2h}(x,t_n)$ of the spline s_{1h} and s_{2h} at time t_n for $n=0,\ldots,N$ are presented in terms of the values $\widehat{s}_{n,1h}(x)$ and $\widehat{s}_{n,2h}(x)$. Here $\widehat{s}_{n,1h}$ and $\widehat{s}_{n,2h}$ indicate the cubic splines given in the form, respectively,

$$\begin{aligned} \widehat{s}_{n,1h}(x) &= \sum_{i=-1}^{m+1} \widehat{\alpha}_{i,n}B_i(x), \\ \widehat{s}_{n,2h}(x) &= \sum_{i=-1}^{m+1} \widehat{\beta}_{i,n}B_i(x). \end{aligned} \tag{4.54}$$

We thus have $s_{1h}(x,t_n) \backsimeq \widehat{s}_{n,1h}(x)$ and $s_{2h}(x,t_n) \backsimeq \widehat{s}_{n,2h}(x)$ for all $x \in [a,b]$.

4.2 Modified Cubic B-spline Basis Functions

In this section, a striking approximation method for solving the Burgers equation with source term is considered. The structure of the Burgers equation takes into account both nonlinear advection and diffusion terms for simulating the physical behavior of the motion and its shock wave behavior when the viscosity value is small. This numerical scheme is based on the modified cubic B-splines in space variable. The obtained results have been computed without using any linearization and transformation processes. The produced diagonal system has been solved by the SSPRK54 scheme.

Consider the Burgers equation with source term of equation (1.3) as follows:

$$\frac{\partial u}{\partial t}(x,t)-\lambda\frac{\partial^2 u}{\partial x^2}+u\frac{\partial u}{\partial x}-f(x,t)=0,\quad (x,t)\in\Omega_m=[a,b]\times[t_0,T], \tag{4.55}$$

with the initial and boundary conditions are given by

$$u(x,t_0)=u_0(x), \tag{4.56}$$

$$u(a,t)=g_1(t),\quad u(b,t)=g_2(t), \tag{4.57}$$

where g_1, g_2 and u_0 are known functions. Here, $f(x,t)$ represents the source term. The rest of our numerical scheme can be expressed as follows:
At boundaries (4.57) for $x = x_0$ and $x = x_m$ the approximation solution (3.27) becomes

$$\begin{array}{rcccl} s_m(x_0,t) & = & \alpha_0(t)\mathscr{B}_0(x_0)+\alpha_1(t)\mathscr{B}_1(x_0) & = & g_1(t) \\ s_m(x_N,t) & = & \alpha_{m-1}(t)\mathscr{B}_{m-1}(x_m)+\alpha_m(t)\mathscr{B}_N(x_m) & = & g_2(t). \end{array} \tag{4.58}$$

Substitution of the approximate solution (3.27) in (4.55) leads to

$$\sum_{j=0}^{m}\alpha_j'(t)\mathscr{B}_j(x)=-\big(\sum_{j=0}^{m}\alpha_j(t)\mathscr{B}_j(x)\big)\big(\sum_{j=0}^{m}\alpha_j(t)\mathscr{B}_j'(x)\big)+\lambda\big(\sum_{j=0}^{m}\alpha_j(t)\mathscr{B}''_j(x)\big)+f(x,t), \tag{4.59}$$

where $\alpha'(t)$ is the first derivative with respect to t. The cubic B-splines basis $\mathscr{B}'_j(x)$ and $\mathscr{B}''_j(x)$ denote the first and second differentiation with respect to x. Let us discretize the domain $[a,b]$ into grid points and let us take $x = x_j$ for $j = 0,\ldots,m$ in equation (4.59). We thus obtain

$$\sum_{j=0}^{m}\alpha_j'(t)\mathscr{B}_j(x_j)=-\big(\sum_{j=0}^{m}\alpha_j(t)\mathscr{B}_j(x_j)\big)\big(\sum_{j=0}^{m}\alpha_j(t)\mathscr{B}_j'(x_j)\big)+\lambda\big(\sum_{j=0}^{m}\alpha_j(t)\mathscr{B}''_j(x_i)\big)+f(x_j,t). \tag{4.60}$$

By using the approximation values of $s_m(x_j)$, $s_m^{'}(x_j)$ and $s_m^{''}(x_j)$ given by equations (3.12) at the knots in equation (4.60), we find the following difference equations with the variables $\alpha(t)$,

$$\begin{cases} 4\alpha'_0 &= g'_0(t) & j=0, \\ \alpha'_{j-1}+4\alpha'_j+\alpha'_{j+1} &= \dfrac{-3}{h}(\alpha_{j-1}+4\alpha_j+\alpha_{j+1})(\alpha_{j+1}-\alpha_{j-1}) & \\ &+ \dfrac{6\lambda}{h^2}(\alpha_{j-1}-2\alpha_j+\alpha j+1) & j=1,2,\ldots,m-1, \\ 4\alpha'_m &= g'_1(t) & j=m. \end{cases} \tag{4.61}$$

Now, using equations (4.61), we obtain the following system with $m+1$ equations and $m+1$ unknowns, as follows

$$\mathscr{A}_3\,\phi' = \Phi_1, \tag{4.62}$$

where

$$\mathscr{A}_3 = \begin{bmatrix} 4 & 0 & 0 & \cdots & 0 \\ 1 & 4 & 1 & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & 1 & 4 & 1 \\ 0 & \cdots & 0 & 0 & 4 \end{bmatrix}.$$

For $\phi' = \left[\alpha'_0, \alpha'_1, \ldots, \alpha'_{m-1}, \alpha'_m\right]^T$, $\Phi_1 = \left[\Phi_{1,0}, \Phi_{1,1}, \ldots, \Phi_{1,N-1}, \Phi_{1,m}\right]^T$, corresponding to the knots are evaluated as:
$\Phi_{1,0} = g'_0(t)$ for $j=0$;
$\Phi_{1,j} = -\dfrac{3}{h}(\alpha_{j-1}+4\alpha_j+\alpha_{j+1})(\alpha_{j+1}-\alpha_{j-1}) + \dfrac{6\lambda}{h^2}(\alpha_{j-1}-2\alpha_j+\alpha_{j+1})$ for $j = 1, \ldots, m-1$,
$\Phi_{1,m} = g'_1(t)$ for $j=m$.
Initial vector ϕ^0 can be obtained by using the initial and boundary conditions at $t=0$. We then have the following relations
$s_m(x_0, 0) = g_0(0)$ for $j=0$;
$s_m(x_j, 0) = u_0(x_j)$ for $j=1,\ldots,m-1$,
$s_m(x_m, 0) = g_1(0)$ for $j=m$.
The above equations yield a tridiagonal matrix system by using the approximate solution (3.27) as given

$$\mathscr{A}_3\,\phi = \Phi_2. \tag{4.63}$$

$\phi^0 = \left[\alpha^0_0, \alpha^0_1, \ldots, \alpha^0_{m-1}, \alpha^0_m\right]^T$, $\Phi_2 = \left[g_0(0), u_0(x_1), \ldots, u_0(x_{m-1}), g_1(0)\right]^T$.

Then, we apply the SSPRK54 method to solve the first order ordinary differential equation system (4.62). Once the parameter $\alpha^0 = \alpha(t_0)$ has been determined at a specified time level, we can compute the solution at the required time level by using iterations. We use the Thomas algorithm to solve the tridiagonal system encountered in (4.63), and then the SSPRK54 method to solve the ODE system.

4.3 Designing a Response Approach in Chaotic Systems

This section is to discuss the GS method by designing new response systems for solving synchronization problems of coupled chaotic identical and nonidentical models. To demonstrate the effectiveness of the proposed control functions, we address the problem of synchronization of identical and nonidentical chaotic systems by considering various numerical examples from physical and biological problems. We consider here the chaotic systems to find an access to Section 4.4.

4.3.1 Synchronization of Two Identical Systems

4.3.1.1 Memristor system

Consider the problem of synchronization of identical systems with dimension spaces $n = m = 4$ concerning the Memristor chaotic systems. The Memristor was postulated as the fourth nonlinear circuit element by Chua [150]. This Memristor system may be described via the following nonlinear differential equations with respect to the fundamental basic circuit elements, resistance, capacitance, inductance and Memristor [45]. We then have

$$\begin{cases} C_{11}\dfrac{dv_{11}}{dt} &= i_3 - W_1(\varrho_1)v_{11} \\ L_{22}\dfrac{di_3}{dt} &= v_{22} - v_{11} \\ C_{22}\dfrac{dv_{22}}{dt} &= -i_3 + G_1 v_{22} \\ \dfrac{d\varrho_1}{dt} &= v_{11} \end{cases} . \tag{4.64}$$

The parameters v_{11}, v_{22} are the voltages. i_3 is the current. Nonlinear function W_1 is called the Memristance. We set $x_1 = v_{11}$, $x_2 = i_3$, $x_3 = v_{22}$, $x_4 = \varrho_1$, $\iota_1 = \dfrac{1}{C_{11}}$,

$\iota_2 = \dfrac{1}{C_{22}}$, $\iota_3 = \dfrac{G_1}{C_{22}}$ and $L_{22} = 1$. Then, system (4.64) can be transformed to a first order differential equation system as

$$\begin{cases} \dfrac{dx_1}{dt} &= \iota_1(x_2 - W_1(x_4)x_1) \\ \dfrac{dx_2}{dt} &= x_3 - x_1 \\ \dfrac{dx_3}{dt} &= -\iota_2 x_2 + \iota_3 x_3 \\ \dfrac{dx_4}{dt} &= x_1 \end{cases}, \tag{4.65}$$

where function $W_1(x_4)$ is defined as

$$W_1(x_4) = \begin{cases} \rho_1 \quad if \quad |x_4| < 1 \\ \rho_2 \quad if \quad |x_4| > 1 \end{cases}. \tag{4.66}$$

The response is similarly chosen to the Memristor system (4.65) given by

$$\begin{cases} \dfrac{dy_1}{dt} &= \iota_1(y_2 - W_1(y_4)y_1) + \psi_1(x(t), y(t)) \\ \dfrac{dy_2}{dt} &= y_3 - y_1 + \psi_2(x(t), y(t)) \\ \dfrac{dy_3}{dt} &= -\iota_2 y_2 + \iota_3 y_3 + \psi_3(x(t), y(t)) \\ \dfrac{dy_4}{dt} &= y_1 + \psi_4(x(t), y(t)). \end{cases} \tag{4.67}$$

To achieve the reduced order synchronization behavior between two identical Memristor systems, we consider that the Memristor systems as the driver system (4.65) and as the response system (4.67). Then, we apply the first approach method for this problem, by rewriting the driver-response system as (3.33). One can rewrite the response system (4.67) in the form

$$\dot{y}(t) = -Q_4(t)e(t) + \mathscr{J}_{\Upsilon}(x(t))H(x(t)),$$

where

$$Q_2 = \begin{bmatrix} 0 & \iota_1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -\iota_2 & \iota_3 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \text{ and } g_g(y) = \begin{bmatrix} -\alpha W_1(y_4)y_1 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Now, to solve the synchronization of this problem with the control functions $\psi(x, y)$ as calculated by (3.32), one can define the identity vector function as $\Upsilon(x_1, x_2, x_3, x_4) = (y_1, y_2, y_3, y_4)^T$. For the sake of the simplicity, we consider $Q_4 = k_2 I_4$. Thence, to demonstrate the effectiveness of the proposed control function, we solve the driver-response systems by using the RK4 scheme with the initial conditions $x(0)$ and $y(0)$ by presenting simulation results.

4.3.1.2 Hindmarsh-Rose (HR) neuronal system

Here, we mainly study the role of neural synchronization in physical diseases, and particularly in the case of heart attack, where the neural activity takes place in many part of human body, such as the heart muscles. The problem of chaos in the heart muscles will decrease when neurons begin to convince to fire in synchronous with them. The dynamic variables during this process are the neurons membrane potential, which are changed and control a vast number of ionic channels. In general, it describes three different states of the membrane potential which can be Resting, Spiking and Bursting. Some papers investigated synchronization of two HR neurons [50]. Hereafter, some information about neural activity and synchronization, we present the dynamics of the membrane potential in the axon of neuron with a three dimensional system which is known as the HR model

$$\begin{cases} \dfrac{dx_1}{dt} &= x_2 + \iota_4 x_1^2 - x_1^3 - x_3 + \rho_3(t) \\ \dfrac{dx_2}{dt} &= 1 - \iota_5 x_1^2 - x_2 \\ \dfrac{dx_3}{dt} &= \iota_6(\iota_7(x_1 - C_{33}) - x_3) \end{cases}. \tag{4.68}$$

where x_1, x_2 and x_3 represent the membrane potential, the recovery variable and the exchange of ions through slow ionic channels respectively. $\rho_3(t)$ is the externally applied current at time t, ι_6 is a recovery variable; which is very small. The parameter C_{33} is the x-coordinate of the leftmost equilibrium point of the model without adaptation. Parameters ι_4,ι_5, ι_6 and ι_7 are given in biological phenomena. To study synchronization motions of the two identity coupled HR neuronal systems, it is assumed that system (4.68) is considered to be the drive system, and the response system is given by

$$\begin{cases} \dfrac{dy_1}{dt} &= y_2 + \iota_4 y_1^2 - y_1^3 - y_3 + \rho_3(t) + \psi_1(x(t), y(t)) \\ \dfrac{dy_2}{dt} &= 1 - \iota_5 y_1^2 - y_2 + \psi_2(x(t), y(t)) \\ \dfrac{dy_3}{dt} &= \iota_6(\iota_7(y_1 - y_\mu) - C) + \psi_3(x(t), y(t)) \end{cases} . \tag{4.69}$$

To solve the synchronization problems, one can rewrite systems (4.68) and (4.69) as (3.33). Thus, the response system (4.69) can be rewritten as:

$$\dot{y}(t) = -Q_4(t)e(t) + \mathscr{J}_\Upsilon(x(t))H(x(t)),$$

where

$$Q_2 = \begin{bmatrix} 0 & 1 & -1 \\ 0 & -1 & 0 \\ \iota_6\iota_7 & 0 & -\iota_6 \end{bmatrix} \text{ and } g_g(y) = \begin{bmatrix} \iota_4 y_2^2 - y_2^3 + I \\ 1 - \iota_5 y_2^2 \\ 0 \end{bmatrix}.$$

The control function $\psi(x, y)$ can be determined by (3.32), one can propose the identity vector function as $\Upsilon(x_1, x_2, x_3) = (y_1, y_2, y_3)^T$. We produce numerical results using the RK4 method for the driver-response systems, by considering $Q_4 = k_2 I_3$ at initial points, $x(0)$ and $y(0)$.

4.3.1.3 Belousov-Zhabotinsky (BZ) reaction

We suggest modeling of the BZ reaction in chemistry. The reaction is important mathematically because exhibits many characteristics of chaos. Considering the reaction rates and flow rate, the simple mathematical model consisting of two "rate" equations can be written as [108].

$$\begin{cases} \dfrac{dx_1}{dt} &= (-{x_1}^3 - \iota_8 x_1 + \iota_9) - \iota_{10} x_2 \\ \dfrac{dx_2}{dt} &= \dfrac{(x_1 - x_2)}{\iota_{11}}, \end{cases} , \tag{4.70}$$

where $x_1 = [HBrO_2]$ and $x_2 = [Br^-]$. The characterization of chaos in the BZ reaction relied on the rate of parameters which are fed into the system. Here, to study synchronization motions of the two identity modeling of the BZ reaction system, it is assumed that system (4.70) is considered to be the drive system, and thus the response system is given by

$$\begin{cases} \frac{dy_1}{dt} &= (-{y_1}^3 - \iota_8 y_1 + \iota_9) - \iota_{10} y_2 + \psi_1(x(t), y(t)) \\ \frac{dy_2}{dt} &= \frac{(y_1 - y_2)}{\iota_{11}} + \psi_2(x(t), y(t)). \end{cases} \quad . \tag{4.71}$$

Then, system (4.71) can be rewritten as

$$\dot{y}(t) = -Q_4(t)e(t) + \mathscr{J}_\Upsilon(x(t))H(x(t)),$$

where

$$Q_2 = \begin{bmatrix} -\iota_8 & -\iota_9 \\ 1/\iota_{11} & -1/\iota_{11} \end{bmatrix} \text{ and } g_g(y) = \begin{bmatrix} -y_1^3 + \gamma \\ 0 \end{bmatrix}.$$

The control function $\psi(x,y)$ is given by (3.32), one obtains the identity vector function as $\Upsilon(x_1, x_2) = (y_1, y_2)^T$. Thus, we provide numerical results using the RK4 method for solving the driver-response systems with initial conditions $x(0)$ and $y(0)$. For the sake of the simplicity, we consider $Q_4 = k_2 I_2$.

4.3.2 Synchronization of Two Nonidentical Systems

4.3.2.1 Lorenz and Rössler systems

Here, we consider two nonidentical chaotic systems in both cases of $n < m$ and $n > m$. We take the well systems of Lorenz and Rössler [91]. The Lorenz system is defined by the three dimensional ordinary differential equations as follows

$$\begin{cases} \frac{dx_1}{dt} &= \iota_{12}(x_2 - x_1) \\ \frac{dx_2}{dt} &= \iota_{13} x_1 - x_2 - x_1 x_3 \\ \frac{dx_3}{dt} &= -\iota_{14} x_3 + x_1 x_2 \end{cases} \quad . \tag{4.72}$$

The Rössler system is designed by four nonlinear ordinary differential equation system

$$\begin{cases} \dfrac{dy_1}{dt} &= -y_2 - y_3 \\ \dfrac{dy_2}{dt} &= y_1 + \iota_{15} y_2 + y_4 \\ \dfrac{dy_3}{dt} &= y_1 y_3 + \iota_{16} \\ \dfrac{dy_4}{dt} &= -\iota_{17} y_3 + \iota_{18} y_4 \end{cases}. \tag{4.73}$$

To illustrate Theorem 3.4 for the first case $n < m$, the Lorenz system is chosen as the drive system and the Rössler as the response system which can be redefined as

$$\dot{y}(t) = g_g(y(t)) - g_g(\Upsilon(x(t))) - \Big(J_g(\Upsilon(x(t))) + Q_4(t)\Big)e(t) + \mathscr{J}_\Upsilon(x(t))H(x(t)),$$

where

$$Q_2 = \begin{bmatrix} 0 & -1 & -1 & 0 \\ 1 & \iota_{15} & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\iota_{17} & \iota_{18} \end{bmatrix} \text{ and } g_g(y) = \begin{bmatrix} 0 \\ 0 \\ y_1 y_3 + \iota_{16} \\ 0 \end{bmatrix}.$$

In the second case, for $n > m$, we adopt the Rössler system as the drive, and making the Lorenz system as the response. Then, we rewrite the system in the form

$$\dot{y} = -Q_4(t)e(t) + \mathscr{J}_\Upsilon(x(t))H(x(t)),$$

where

$$Q_2 = \begin{bmatrix} -\iota_{12} & \iota_{12} & 0 \\ \iota_{13} & -1 & 0 \\ 0 & 0 & -\iota_{14} \end{bmatrix} \text{ and } g_g(y) = \begin{bmatrix} 0 \\ -y_1 y_3 \\ y_1 y_3 \end{bmatrix}.$$

The control function $\psi(x, y)$ can be determined by (3.35). The vector function is given by $\Upsilon(x_1, x_2, x_3) = (y_1, y_2, y_3, y_1 + y_2 + y_3)^T$ which is the nonidentity function. We choose the matrix to be $Q_4 = Q_2 + Q_5$, where $Q_5 = k_2 I_4$. We obtain the behaviour of synchronization by using the very large value of the coupling strength. We produce numerical results using the RK4 method for the nonidentical driver-response systems at initial points $x(0)$ and $y(0)$.

4.4 Synchronization of the Nonlinear ADR Processes via the BDFS and Lyapunov Methods

This section dynamical and GS of two dependent chaotic nonlinear ADR processes with forcing term, which unidirectionally coupled in the driver-response configuration. By combining the BDFS scheme with the Lyapunov direct method, the GS is studied for designing controller function of the coupled nonlinear ADR equations without any linearization. This technique utilizes the driver configuration to monitor the synchronized motions. The nonlinear coupled model is described by the incompressible fluid flow coupled to thermal dynamics, and motivated by the Boussinesq equations. Let us consider the nonlinear coupled Burgers equations with source functions as:

$$\begin{aligned} \frac{\partial u_1}{\partial t} - \lambda_1 \frac{\partial^2 u_1}{\partial x^2} + \lambda_2 u_1 \frac{\partial u_1}{\partial x} &= f_1(x,t) + k_2 u_2(x,t) \qquad \text{driver}, \\ \frac{\partial u_2}{\partial t} - \lambda_3 \frac{\partial^2 u_2}{\partial x^2} + \lambda_4 u_1 \frac{\partial u_2}{\partial x} &= f_2(x,t) \qquad \text{response}, \end{aligned} \tag{4.74}$$

with initial conditions

$$u_1(x,t_0) = u_{1,0}(x), \quad u_2(x,t_0) = u_{2,0}(x), \tag{4.75}$$

and boundary conditions

$$\begin{aligned} u_1(a,t) &= g_1(t), \; u_1(b,t) = g_2(t), \\ u_2(a,t) &= g_3(t), \; u_2(b,t) = g_4(t). \end{aligned} \tag{4.76}$$

Here, $(x,t) = [a,b] \times [t_0, T]$. The function u_2 can be viewed as a temperature field. λ_3 is the thermal conductivity, λ_1 is a viscosity coefficient, λ_2 and λ_4 are constants. k_2 is the coefficient of the thermal expansion and also the coupling strength. The function u_1 represents the velocity. The functions g_1, g_2, g_3 and g_4 are known. The f_1 and f_2 are the source terms. The outputs from the driver are used to drive the response. Thus, there exists a relation between the coupled Burgers equation, which could be a temperature u_2, transforms the trajectories on the attractor of the first equation into those on the attractor of the second equation. Numerical results have been produced by using the BDFS method for the proposed driver-response. The required solutions of (4.74)-(4.76) $u_1(x,t)$ and $u_2(x,t)$ are approximated by the cubic interpolating splines $s_{1,h}$ and $s_{2,h}$, respectively, as given in (4.22). By determining unknown time dependent coefficients $\alpha_i(t)$ and $\beta_j(t)$, now, we consider the natural cubic splines at the boundary conditions, we can obtain the expressions $\alpha_{-1}(t)$, $\alpha_{m+1}(t)$, $\beta_{-1}(t)$ and $\alpha_{m+1}(t)$ as given in (4.27) - (4.28). Later, by interpolating the conditions at the boundary points $x_0 = a$

and $x_m = b$ yields

$$s_{1,h}(x_0,t) = \frac{1}{6}\Big(\alpha_{-1}(t) + 4\alpha_0(t) + \alpha_1(t)\Big) = u_1(x_0,t),$$

$$s_{2,h}(x_0,t) = \frac{1}{6}\Big(\beta_{-1}(t) + 4\beta_0(t) + \beta_1(t)\Big) = u_2(x_0,t),$$

$$s_{1,h}(x_m,t) = \frac{1}{6}\Big(\alpha_{m-1}(t) + 4\alpha_m(t) + \alpha_{m+1}(t)\Big) = u_1(x_m,t),$$

$$s_{2,h}(x_m,t) = \frac{1}{6}\Big(\beta_{m-1}(t) + 4\beta_m(t) + \beta_{m+1}(t)\Big) = u_2(x_m,t).$$

Now, $\alpha_0(t)$, $\alpha_m(t)$, $\beta_0(t)$ and $\beta_m(t)$ substitution of the above expressions into (4.27) and (4.28) leads to

$$\alpha_0(t) = u_1(x_0,t) \quad \text{and} \quad \alpha_m(t) = u_1(x_m,t), \tag{4.77}$$

$$\beta_0(t) = u_2(x_0,t) \quad \text{and} \quad \beta_m(t) = u_2(x_m,t). \tag{4.78}$$

Then, we use $s_{1,h}$ and $s_{2,h}$ with their derivatives in (4.74) at points x_i and x_j for $i = 0$ and m, $j = 0$ and m, one reaches

$$\frac{\partial s_{1,h}}{\partial t}(x_0,t) = \lambda_1 \frac{\partial^2 s_{1,h}}{\partial x^2}(x_0,t) + F_1(\alpha(t),\beta(t),x_0,t), \tag{4.79}$$

$$\frac{\partial s_{1,h}}{\partial t}(x_m,t) = \lambda_1 \frac{\partial^2 s_{1,h}}{\partial x^2}(x_m,t) + F_1(\alpha(t),\beta(t),x_m,t), \tag{4.80}$$

and

$$\frac{\partial s_{2,h}}{\partial t}(x_0,t) = \lambda_3 \frac{\partial^2 s_{2,h}}{\partial x^2}(x_0,t) + F_2(\alpha(t),\beta(t),x_0,t), \tag{4.81}$$

$$\frac{\partial s_{2,h}}{\partial t}(x_m,t) = \lambda_3 \frac{\partial^2 s_{2,h}}{\partial x^2}(x_m,t) + F_2(\alpha(t),\beta(t),x_m,t), \tag{4.82}$$

where F_1 and F_2 are the functions indicating the nonlinear parts. By taking into account the relations (4.27), (4.28), (4.77), (4.78) (4.79), (4.80), (4.81) and (4.82) , one can have

$$\begin{aligned} \alpha_0'(t) &= F_1(\alpha(t),\beta(t),x_0,t), \\ \alpha_m'(t) &= F_1(\alpha(t),\beta(t),x_m,t). \end{aligned} \tag{4.83}$$

$$F_1(\alpha(t),\beta(t),x_0,t) = \frac{\lambda_2}{2h} g_1(g_1 - \alpha_1) + \frac{k_2}{2h} g_3(g_3 - \beta_1) + g_3(g_3 - \alpha_1)\Big) - f_1(x_0,t),$$

$$F_1(\alpha(t),\beta(t),x_m,t) = \frac{\lambda_2}{h} g_2(\alpha_{m-1} - g_2) + \frac{k_2}{h} g_4(\beta_{m-1} - g_4) + g_4(\alpha_{m-1} - g_4)\Big) - f_1(x_m,t).$$

$$\begin{aligned} \beta_0'(t) &= F_2(\alpha(t),\beta(t),x_0,t), \\ \beta_m'(t) &= F_2(\alpha(t),\beta(t),x_m,t), \end{aligned} \tag{4.84}$$

$$F_2(\alpha(t),\beta(t),x_0,t) = \frac{\lambda_4}{2h} g_3(g_3-\beta_1)+g_1(g_3-\beta_1)\big)-f_2(x_0,t),$$

$$F_2(\alpha(t),\beta(t),x_m,t) = \frac{\lambda_4}{h} g_4(\beta_{m-1}-g_4)+g_2(\beta_{m-1}-g_2)\big)-f_2(x_m,t).$$

Thus, by using (4.83) and (4.84), in (4.74) at points x_i and x_j for $i=1,\dots,m-1$, $j=1,\dots,m-1$, one finds

$$\begin{aligned} \frac{4}{6}\alpha_1'(t)+\frac{1}{6}\alpha_2'(t) &= \frac{-2}{h^2}\alpha_1(t)+\frac{1}{h^2}\alpha_2(t)+\frac{1}{h^2}g_1(t) \\ &+ F_1(\alpha(t),\beta(t),x_1,t)-\frac{1}{6}F_1(\alpha(t),\beta(t),x_0,t), \end{aligned} \tag{4.85}$$

$$\begin{aligned} \frac{1}{6}\alpha_{m-2}'(t)+\frac{4}{6}\alpha_{m-1}'(t) &= \frac{1}{h^2}\alpha_{m-2}(t)+\frac{1}{h^2}\alpha_{m-1}(t)+\frac{1}{h^2}g_2(t) \\ &+ F_1(\alpha(t),\beta(t),x_{m-1},t)-\frac{1}{6}F_1(\alpha(t),\beta(t),x_m,t). \end{aligned} \tag{4.86}$$

Then, at points x_i for $i=2,\dots,m-2$, we obtain

$$\frac{1}{6}\alpha_{i-1}'+\frac{4}{6}\alpha_i'+\frac{1}{6}\alpha_{i+1}' = \frac{1}{h^2}\alpha_{i-1}(t)+\frac{-2}{h^2}\alpha_i(t)+\frac{1}{h^2}\alpha_{i+1}(t)+F_1(\alpha(t),\beta(t),x_i,t), \tag{4.87}$$

where

$$\begin{aligned} F_1(\alpha(t),\beta(t),x_1,t) &= -\frac{\lambda_2}{2h}\Big(\frac{1}{6}g_1(t)+\frac{4}{6}\alpha_1(t)+\frac{1}{6}\alpha_2(t)\Big)\big(g_1(t)-\alpha_2(t)\big) \\ &+ \frac{k}{12h}\big((g_2(t)+4\beta_1(t)+\beta_2(t))\big)+f_1(x_1,t), \\ F_1(\alpha(t),\beta(t),x_{m-1},t) &= -\frac{\lambda_2}{2h}\Big(\frac{1}{6}\alpha_{m-2}(t)+\frac{4}{6}\alpha_{m-1}(t)+\frac{1}{6}g_2(t)\Big)(\alpha_{m-2}(t)-g_2(t)) \\ &+ \frac{k_2}{12h}\big((\beta_{m-2}(t)+4\beta_{m-1}(t)+g_4(t))\big)+f_1(x_{m-1},t), \\ F_1(\alpha(t),\beta(t),x_i,t) &= -\frac{\lambda_2}{2h}\Big(\frac{1}{6}\alpha_{i-1}(t)+\frac{4}{6}\alpha_i(t)+\frac{1}{6}\alpha_{i+1}(t)\Big)\big(\alpha_{i-1}(t)-\alpha_{i+1}(t)\big) \\ &+ \frac{k_2}{12h}\big((\beta_{i-1}(t)+4\beta_i(t)+\beta_{i+1}(t))\big)+f_1(x_i,t), \end{aligned}$$

$$\begin{aligned} \frac{4}{6}\beta_1'(t)+\frac{1}{6}\beta_2'(t) &= \frac{-2}{h^2}\beta_1(t)+\frac{1}{h^2}\beta_2(t)+\frac{1}{h^2}g_3(t) \\ &+ F_2(\alpha(t),\beta(t),x_1,t)-\frac{1}{6}F_2(\alpha(t),\beta(t),x_0,t), \end{aligned} \tag{4.88}$$

$$\frac{1}{6}\beta'_{m-2}(t)+\frac{4}{6}\beta'_{m-1}(t) = \frac{1}{h^2}\beta_{m-2}(t)+\frac{1}{h^2}\beta_{m-1}(t)+\frac{1}{h^2}g_4(t) + F_2(\alpha(t),\beta(t),x_{m-1},t)-\frac{1}{6}F_2(\alpha(t),\beta(t),x_m,t). \tag{4.89}$$

Then, at points x_j for $j=2,\dots,m-2$, one finds

$$\frac{1}{6}\beta'_{i-1}+\frac{4}{6}\beta'_i+\frac{1}{6}\beta'_{i+1} = \frac{1}{h^2}\beta_{i-1}(t)+\frac{-2}{h^2}\beta_i(t)+\frac{1}{h^2}\alpha_{i+1}(t)+F_2(\alpha(t),\beta(t),x_i,t), \tag{4.90}$$

$$F_2(\alpha(t),\beta(t),x_1,t) = -\frac{\lambda_4}{2h}\Big(\frac{1}{6}g_3(t)+\frac{4}{6}\beta_1(t)+\frac{1}{6}\beta_2(t)\Big)\big(g_3(t)-\beta_2(t)\big)+f_2(x_1,t),$$

$$F_2(\alpha(t),\beta(t),x_{m-1},t) = -\frac{\lambda_4}{2h}\Big(\frac{1}{6}\beta_{m-2}(t)+\frac{4}{6}\beta_{m-1}(t)+\frac{1}{6}g_4(t)\Big)(\beta_{m-2}(t)-g_4(t)) + f_1(x_{m-1},t),$$

$$F_2(\alpha(t),\beta(t),x_i,t) = -\frac{\lambda_4}{2h}\Big(\frac{1}{6}\beta_{i-1}(t)+\frac{4}{6}\beta_i(t)+\frac{1}{6}\beta_{i+1}(t)\Big)\big(\beta_{i-1}(t)-\beta_{i+1}(t)\big) + f_2(x_i,t).$$

One has (4.46), and equations (4.87)-(4.89) are summarized as the system of ordinary differential equations as given in 4.47. From the hypothesis of the Lyapunov method, synchronization of the proposed model is studied at optimal value of the coupling strength. Again the system (4.47) is given

$$\frac{dw(t)}{dt}=\mathscr{A}_2^{-1}\left(\mathbb{D}w(t)+\Phi(w(t))\right), \tag{4.91}$$

where $\mathscr{A}_2$ is a tridiagonal matrix defining a regular system when the eigenvalues are all different and the real parts of them are negative. The matrix $\mathscr{A}_2$ is considered to be negative definite. Thus, (4.74)-(4.76) is globally generalized synchronous with respect to the coupling strength at the optimal value.

Furthermore, the stability requires that $t>t_0+T$ and $k>0$. Then, the BDF method is applied, the first-order ordinary differential equation system (4.91) is solved. The time interval $[t_0,T]$ is divided into N subintervals with the time step $\Delta t=\dfrac{T-t_0}{N}$ with the knots $t_n=t_0+n\,\Delta t$ for $n=0,\dots,N$. The method applied to (4.91) gives arise to the following approximations. The values $s_{1,h}(x,t_n)$ and $s_{2,h}(x,t_n)$ of the spline $s_{1,h}$ and $s_{2,h}$ at time t_n for $n=0,\dots,N$ are presented in terms of the values $\widehat{s}_{n,1h}(x)$ and $\widehat{s}_{n,2h}(x)$. Here $\widehat{s}_{n,1h}$ and $\widehat{s}_{n,2h}$ indicate the cubic splines given in (4.54). Thus, we obtain $s_{1,h}(x,t_n)\simeq\widehat{s}_{n,1h}(x)$ and $s_{2,h}(x,t_n)\simeq\widehat{s}_{n,2h}(x)$ for all $x\in[a,b]$. In the following chapters, numerical examples illustrating the accuracy of the present approach are given.

5

TWO DIMENSIONAL NONLINEAR ADR PROBLEMS

Two-dimensional ADR processes with forcing terms have various kinds of practical applications in applied mathematics, such as turbulence and viscous fluid [81, 116, 247]. In this chapter, we provide the BDFS method for solving the 2D ADR processes with forcing terms, without any linearization, and keeping the originality of nature. Here, we are going to develop our earlier work by using the 2D spline, B-spline and natural spline methods in space. A system of first order ODEs is produced. The BDF scheme is particularly suitable for the large scale and stiff ODE problems. Thus, after successful discretization in space, the BDF method allows an efficient implementation for solving the resulting system in time. In recent years, many researchers have paid their particular attention to solving these problems using various numerical approaches, particularly interested in the 2D Burgers equation. Fletcher [115] found the exact solution of the 2D Burgers equation by using the Hopf-Cole transformation. Many authors have used many numerical and analytical techniques for solving the 2D Burgers equation such as: finite element and finite difference methods [46], the similarity reductions [2], finite difference scheme [25], Eulerian Lagrange method [53], Lattice Boltzmann method [127], Haar wavelet method [258], Galerkin method [250], the modified bi-cubic B-spline functions [31, 92].

The 2D ADR equation arising in various fields of science is considered as

$$\frac{\partial u}{\partial t}(x,y,t)=\mathscr{L}_d(\frac{\partial^2 u}{\partial x^2},\frac{\partial u}{\partial x},\frac{\partial u}{\partial y},\frac{\partial^2 u}{\partial x\partial y},\frac{\partial^2 u}{\partial y^2},u,x,y,t)+\mathscr{N}_d(\frac{\partial^2 u}{\partial x^2},\frac{\partial u}{\partial x},\frac{\partial u}{\partial y},\frac{\partial^2 u}{\partial x\partial y},\frac{\partial^2 u}{\partial y^2},u,x,y,t), \tag{5.1}$$

for $(x,y,t)\in\Omega_d\times[t_0,T]$, where $\Omega_d=[a,b]\times[c,d]$. $\mathscr{L}_d$ is a linear partial differential operator of second order and $\mathscr{N}_d$ defines a non-linear differential part. The initial condition at t_0

$$u(x,y,t_0)=u_0(x,y),\quad (x,y)\in\Omega_d\cup\partial\Omega_d, \tag{5.2}$$

and boundary conditions are given by

$$\begin{aligned}
u_a(y,t,u(a,y,t),u_x(a,y,t),u_y(a,y,t)) &= g_1(y,t),\\
u_b(y,t,u(b,y,t),u_x(b,y,t),u_y(b,y,t)) &= g_2(y,t),\\
u_c(x,t,u(x,c,t),u_x(x,c,t),u_y(x,c,t)) &= g_3(x,t),\\
u_d(x,t,u(x,d,t),u_x(x,d,t),u_y(x,d,t)) &= g_4(x,t).
\end{aligned} \tag{5.3}$$

The boundary functions g_1, g_2, g_3, g_4 and initial function u_0 are known. In many practical and physical situations the boundary functions are not differentiable or their derivatives are not available. So, in this study, we only assume that the boundary conditions are defined on the time interval without any further requirement. We give an introduction of the BDFS technique to analyse the 2D ADR equation in detail, in the following section.

5.1 Description of the Method

For the approximate solution of the 2D ADR processes (5.1)-(5.3) in $\mathscr{C}^l(\Omega_d)$ where the classical space of l-times continuously differentiable functions on the interval Ω_d is considered. Then, we define the solution u on a rectangle Ω_d (see Figure 5.1)

$$\Omega_d = \{(x,y)|\ a \le x \le b \ \text{ and } \ c \le y \le d\}.$$

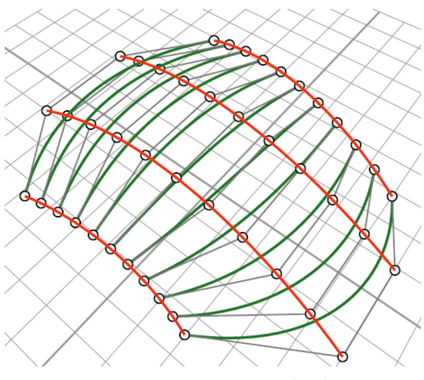


Figure 5.1 Knot insertions into product surface of two dimensions

Let us consider uniform subdivisions of the space interval Ω_d as set of $m+7$ knots

$$\begin{cases} x_{-3} < x_{-2} < x_{-1} < a = x_0 < x_1 < \quad \dots \quad < x_{m_1} = b < x_{m_1+1} < x_{m_1+2} < x_{m_1+3} \\ \text{and} \\ y_{-3} < y_{-2} < y_{-1} < c = y_0 < y_1 < \quad \dots \quad < y_{m_2} = d < y_{m_2+1} < y_{m_2+2} < y_{m_2+3} \end{cases} \tag{5.4}$$

with $x_i = a + ih_1$ and $y_j = c + jh_2$ for $i, j = -3, \dots, m_{1,2} + 3$ where $h_1 = \dfrac{b-a}{m_1}$ and $h_2 = \dfrac{d-c}{m_2}$. The subset $\{x_0, \dots, x_{m_1}\}$ and $\{y_0, \dots, y_{m_2}\}$ are uniform partitions of the domain $[a, b] \times [c, d] \subset \mathbb{R} \times \mathbb{R}$. The exact solution of (5.1)-(5.3) is approximated by a cubic interpolating spline on a rectangle Ω_d. The cubic spline s_{hd} interpolating the function u at the knots $x_0, \dots, x_{m_1}$ and $y_0, \dots, y_{m_2}$ is the unique function in $\mathscr{C}^2(\Omega_d)$ satisfying the following conditions

$$\begin{cases} s_{hd}(x_i, y_j) = u(x_i, y_j) \quad \text{for} \quad i = 0, \dots, m_1, j = 0, \dots, m_2 \\ s''_{hd}(a, y_j) = s''_{hd}(b, y_j), \\ s''_{hd}(x_i, c) = s''_{hd}(x_i, d), \end{cases} \tag{5.5}$$

Let $\mathscr{S}(\Omega_d)$ denote the space of all the cubic splines over the set Ω_d. It is well known that the dimension of this space is $dim(\mathscr{S}(\Omega_d)) = (m_1 + 3)(m_2 + 3)$. We denote by $\{B_i(x)\}_{i=0}^{m_1}$ and $\{B_j(y)\}_{j=0}^{m_2}$ basis of the space $\mathscr{S}(\Omega_d)$. The values of the B-splines $B_i(x)$ and $B_j(y)$ with their derivatives at knots x_i and y_j, respectively are given in Table 3.1. It can be shown that the basis $B_{ij}(x, y)$ as the linear space of the cubic splines defined on the rectangle Ω_d (see Figure 5.2).

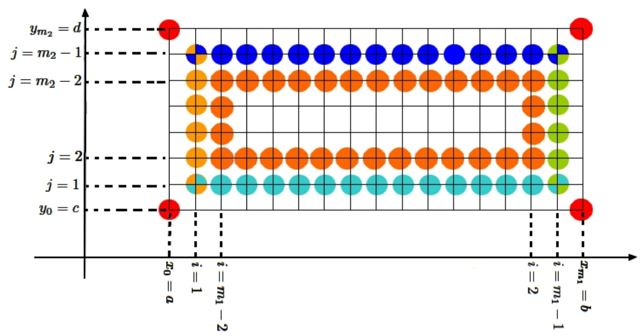


Figure 5.2 Division of the rectangle Ω_d by knots

A cubic spline function $s_{hd} \in \mathscr{S}(\Omega_d)$ over the set Ω_d may be written as a linear combination of the cubic $B_{ij}(x, y)$ of the form

$$s_{hd}(x, y) = \sum_{i=-1}^{m_1+1} \sum_{j=-1}^{m_2+1} \alpha_{i,j} B_i(x) B_j(y), \ \forall (x, y) \in [a, b] \times [c, d], \tag{5.6}$$

where, the coefficients $\{\alpha\}_{i,j=-1}^{m_1,m_2}$ are unknown coefficients. Let us take the following vector valued functions,

$$\mathbb{B}_1(x) = \begin{bmatrix} B_{-1}(x) \\ \vdots \\ B_{m_1-1}(x) \end{bmatrix} \quad \text{and} \quad \mathbb{B}_2(x) = \begin{bmatrix} B_{-1}(y) \\ \vdots \\ B_{m_2-1}(y) \end{bmatrix} \tag{5.7}$$

For the interpolating cubic spline s_{hd} by satisfying the conditions (5.5), one has

$$s_{hd}(x_{k_3}, y_{k_4}) = \sum_{i=-1}^{m_1+1} \sum_{j=-1}^{m_2+1} \alpha_{i,j} B_i(x_{k_3}) B_j(y_{k_4}), \;\; 0 \leq k_3 \leq m_1, \;\; 0 \leq k_4 \leq m_2, \tag{5.8}$$

By considering the natural cubic splines which require that the second derivatives are neglected at the boundaries of the rectangle Ω_d. So, the boundary conditions are given by

$$s''_{hd}(a, y_{k_4}) = s''_{hd}(b, y_{k_4}) = 0, \tag{5.9}$$

and

$$s''_{hd}(x_{k_3}, c) = s''_{hd}(x_{k_3}, d) = 0. \tag{5.10}$$

It can be required that the interpolating conditions $s_{hd}(x_i, y_j) = u(x_i, y_j)$ with the $B_{ij}(x, y)$ and their derivative values taken at certain lines. This approach is presented in more detail in the following steps:

1. ● at $i = 1$ for $1 \leq j \leq m_2 - 1$,
2. ● at $i = m_1 - 1$ for $1 \leq j \leq m_2 - 1$,
3. ● at $j = 1$ for $1 \leq i \leq m_1 - 1$,
4. ● at $j = m_2 - 1$ for $1 \leq i \leq m_1 - 1$,
5. ● at $2 \leq i \leq m_1 - 2$ and $2 \leq i \leq m_1 - 2$.

Considering equations (5.8)-(5.10) and the interpolating conditions at boundary points, we have

$$\begin{cases} s''_{hd}\left(a, y_{k_4}, t\right) & = & 0 \\ s_{hd}(a, y_{k_4}, t) & = & g_1(a, y_{k_4}, t) \qquad k_4 = 0, \ldots, m_2 \end{cases} . \tag{5.11}$$

By taking into account equation (5.8), the equation (5.11) becomes

$$\alpha_{-1,k_4-1}+4\alpha_{-1,k_4}+\alpha_{-1,k_4+1}=2\left(\alpha_{0,k_4-1}+4\alpha_{0,k_4}+\alpha_{0,k_4+1}\right)-\left(\alpha_{1,k_4-1}+4\alpha_{1,k_4}+\alpha_{1,k_4+1}\right), \tag{5.12}$$

and

$$\alpha_{0,k_4-1}+4\alpha_{0,k_4}+\alpha_{0,k_4+1}=6g_1(a,y_{k_4},t),\ \text{for}\ k_4=0,\ldots,m_2 \tag{5.13}$$

For $k_4=0$, by using the relation (5.12)-(5.13), one finds the vector values of α_{0,k_4} as follows

$$\begin{cases} 4\alpha_{0,0}+\alpha_{0,1} &= 6g_1(a,y_0,t)-\alpha_{0,-1}, \\ \alpha_{0,0}+4\alpha_{0,1}+\alpha_{0,2} &= 6g_1(a,y_1,t), \\ \vdots &= \vdots \\ \alpha_{0,m_2-1}+4\alpha_{0,m_2} &= 6g_1(a,y_{m_2},t)-\alpha_{0,m_2+1}. \end{cases} \tag{5.14}$$

We continue to use the 2D natural spline conditions and the above relations at boundary points, we get

$$\begin{cases} s''_{hd}\left(b,y_{k_4},t\right) &= 0 \\ s_{hd}(b,y_{k_4},t) &= g_2(b,y_{k_4},t) \qquad k_4=0,\ldots,m_2 \end{cases}. \tag{5.15}$$

By using expression (5.8) into the equation (5.15), one obtains

$$\begin{aligned} \alpha_{m_1+1,k_4-1}+4\alpha_{m_1+1,k_4}+\alpha_{m_1+1,k_4+1} &= 2\left(\alpha_{m_1,k_4-1}+4\alpha_{m_1,k_4-1}+\alpha_{m_1,k_4+1}\right) \\ &- \left(\alpha_{m_1-1,k_4-1}+4\alpha_{m_1-1,k_4}+\alpha_{m_1-1,k_4+1}\right), \end{aligned} \tag{5.16}$$

and

$$\alpha_{m_1,k_4-1}+4\alpha_{m_1,k_4}+\alpha_{m_1,k_4+1}=6g_1(b,y_{k_4},t),\ \text{for}\ k_4=0,\ldots,m_2. \tag{5.17}$$

By taking into account the relations (5.16)-(5.17) at $k_4=m_2$, we obtain the vector values of α_{m_1,k_4} as

$$\begin{cases} 4\alpha_{m_1,m_2} + \alpha_{m_1,m_2+1} &= 6g_2(b, y_0, t) - \alpha_{m_1,m_2-1}, \\ \alpha_{m_1,m_2} + 4\alpha_{m_1,m_2+1} + \alpha_{m_1,m_2+2} &= 6g_2(b, y_1, t), \\ \vdots &= \vdots \\ \alpha_{m_1,m_2-1} + 4\alpha_{m_1,m_2} &= 6g_2(b, y_{m_2}, t) - \alpha_{m_1,m_2+1}. \end{cases} \tag{5.18}$$

The 2D natural spline conditions with the above relations at the other boundary points are considered as

$$\begin{cases} s''_{hd}\left(x_{k_3}, c, t\right) &= 0 \\ s_{hd}(x_{k_3}, c, t) &= g_3(x_{k_3}, c, t) \qquad k_3 = 0, \ldots, m_1 \end{cases}. \tag{5.19}$$

Rewriting the above equation (5.19) into (5.8), we have

$$\alpha_{k_3-1,-1} + 4\alpha_{k_3,-1} + \alpha_{k_3+1,-1} = 2\left(\alpha_{k_3-1,0} + 4\alpha_{k_3,0} + \alpha_{k_3+1,0}\right) - \left(\alpha_{k_3-1,1} + 4\alpha_{k_3,1} + \alpha_{k_3+1,1}\right), \tag{5.20}$$

and

$$\alpha_{k_3-1,0} + 4\alpha_{k_3,0} + \alpha_{k_3+1,0} = 6g_3(y_{k_3}, c, t), \text{ for } k_3 = 0, \ldots, m_1. \tag{5.21}$$

By using the expressions (5.20)-(5.21) at $k_3 = 0$, we find out the vector values of $\alpha_{k_3,0}$

$$\begin{cases} 4\alpha_{0,0} + \alpha_{1,0} &= 6g_3(y_0, c, t) - \alpha_{-1,0}, \\ \alpha_{0,0} + 4\alpha_{1,0} + \alpha_{2,0} &= 6g_3(y_1, c, t), \\ \vdots &= \vdots \\ \alpha_{m_1-1,0} + 4\alpha_{m_1,0} &= 6g_3(y_{m_1,0}, t) - \alpha_{m_1+1,0}. \end{cases} \tag{5.22}$$

Considering the 2D natural spline conditions and the above relations at boundary points together, one obtains

$$\begin{cases} s''_{hd}\left(x_{k_3}, d, t\right) &= 0 \\ \\ s_{hd}(x_{k_4}, d, t) &= g_4(x_{k_3}, d, t) \qquad k_3 = 0, \ldots, m_1 \end{cases} . \tag{5.23}$$

Substituting s_{hd} with their derivatives from (5.8) into (5.23) gives

$$\begin{aligned} \alpha_{k_3-1,m_2+1} + 4\alpha_{k_3,m_2+1} + \alpha_{k_3+1,m_2+1} &= 2\left(\alpha_{k_3-1,m_2} + 4\alpha_{k_3-1,m_2} + \alpha_{k_3+1,m_2}\right) \\ &- \left(\alpha_{k_3-1,m_2-1} + 4\alpha_{k_3,m_2-1} + \alpha_{k_3+1,m_2-1}\right), \end{aligned} \tag{5.24}$$

and

$$\alpha_{k_3-1,m_2} + 4\alpha_{k_3,m_2} + \alpha_{k_3+1,m_2} = 6g_4(x_{k_3}, d, t), \text{ for } k_3 = 0, \ldots, m_1. \tag{5.25}$$

For $k_3 = m_1$, by taking the relation (5.24)-(5.25) we then have the vector values of α_{k_3,m_2}

$$\begin{cases} 4\alpha_{m_1,m_2} + \alpha_{m_1+1,m_2} &= 6g_4(x_0, d, t) - \alpha_{m_1-1,m_2}, \\ \\ \alpha_{m_1,m_2} + 4\alpha_{m_1+1,m_2} + \alpha_{m_1+2,m_2} &= 6g_4(x_1, d, t), \\ \\ \quad \vdots &= \quad \vdots \\ \\ \alpha_{m_1-1,m_2} + 4\alpha_{m_1,m_2} &= 6g_4(x_{m_1}, d, t) - \alpha_{m_1+1,m_2}. \end{cases} \tag{5.26}$$

Now, the relations (5.12) and (5.21) at $k_3 = 0$ and $k_4 = 0$ are expressed as in the following cases, where $k_3 = 0$, equation (5.12) is

$$\alpha_{0,-1} + 4\alpha_{0,0} + \alpha_{0,1} = 6g_1(a, c, t). \tag{5.27}$$

For $k_4 = 0$ equation (5.21) becomes

$$\alpha_{-1,0} + 4\alpha_{0,0} + \alpha_{1,0} = 6g_3(a, c, t). \tag{5.28}$$

One can consider the two boundary points to be equal at the points a and c and then we obtain

$$\alpha_{0,-1} + \alpha_{0,1} = \alpha_{-1,0} + \alpha_{1,0}. \tag{5.29}$$

Also, at points $k_3 = 1, \ldots, m_1 - 1$ and $k_4 = 1, \ldots, m_2 - 1$, we then reach

$$\begin{aligned} \mathbb{B}_1(x_{k_3})^T \alpha'_{i,j}(t) \mathbb{B}_2(y_{k_4}) &= -\left[\mathbb{B}_1(x_{k_3})^T \alpha_{i,j}(t) \mathbb{B}_2(y_{k_4})\right] \\ &\quad \left[\mathbb{B}'_1(x_{k_3})^T \alpha_{i,j}(t) \mathbb{B}_2(y_{k_4}) + \mathbb{B}_2(x_{k_3})^T \alpha_{i,j}(t) \mathbb{B}'_2(x_{k_4})\right] \\ &= F(\alpha_{i,j}(t), x_{k_3}, y_{k_4}, t). \end{aligned} \tag{5.30}$$

The approximating 2D cubic spline s_{hd} must also satisfy initial condition (5.2) at points $x_0, \ldots, x_{m_1}$ and $y_0, \ldots, y_{m_2}$ at initial time t_0:

$$\begin{cases} s_{hd}(x_0, y_{k_4}, t_0) = u_0(x_0, y_{k_4}), & \text{for} \quad k_4 = 0, \ldots, m_2, \\ s_{hd}(x_{k_3}, y_{k_4}, t_0) = u_0(x_{k_3}, y_{k_4}), & \text{for} \quad k_3, k_4 = 1, ..., m_{1,2} - 1, \\ s_{hd}(x_{m_1}, y_{m_2}, t_0) = u_0(x_{m_1}, y_{m_2}), & \text{for} \quad k_3, k_4 = m_1, m_2. \end{cases} \tag{5.31}$$

By using the above equation, we find out the condition

$$\mathbb{A}_d \; \phi_d(t_0) = (\phi_d)_0, \tag{5.32}$$

where

$$\mathbb{A}_d = \begin{bmatrix} \mathscr{A}_1 & 0 & 0 & \cdots & 0 \\ 0 & \mathscr{A}_1 & 0 & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & 0 & \mathscr{A}_1 & 0 \\ 0 & \cdots & 0 & 0 & \mathscr{A}_1 \end{bmatrix}, \tag{5.33}$$

and

$$\phi_d = \left[\alpha_{i,j}(t)\right]_{\substack{1 \le i \le m_1 - 1 \\ 1 \le j \le m_2 - 1}} = \begin{bmatrix} \phi_{1,1}(t) & \phi_{1,2}(t) & \cdots & \phi_{1,m_2-1}(t) \\ \phi_{2,1}(t) & \phi_{2,2}(t) & \cdots & \phi_{2,m_2-1}(t) \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \phi_{m_1-1,1}(t) & \phi_{m_1-1,2}(t) & \cdots & \phi_{m_1-1,m_2-1}(t) \end{bmatrix}. \tag{5.34}$$

$(\phi_d)_0$ is the matrix of size $(m_1 - 1) \times (m_2 - 1)$ and $\mathscr{A}_1$ is given in (3.9).

Now, the above equations can be written more compactly as the ODEs form:

$$\begin{cases} \mathbb{A}_d\, \phi_d^{'}\, \mathbb{B}_d &= -\big[\mathbb{A}_d^T\, \phi_d\, \mathbb{B}_d\big]\big[(\mathbb{A}_d^{'})^T\, \phi_d\, \mathbb{B}_d^{'}\big] + F_d(\alpha_d(t), x, y, t), \\ \mathbb{A}_d\, \phi_d(t_0) &= \phi_{d,0}, \end{cases} \tag{5.35}$$

where

$$\mathbb{B}_d(x) = \begin{bmatrix} \mathbb{B}_1(x) \\ \mathbb{B}_2(y) \end{bmatrix}. \tag{5.36}$$

We will consider the BDF scheme or any other suitable method for solving the large scale (5.35) ODE system in time. They from (5.35)

$$\begin{cases} \phi_d^{'} &= -\phi_d\big[(\mathbb{A}_d^{'})^T\, \phi_d\, \mathbb{B}_d^{'}\big] + \left(\mathbb{A}_d^T\right)^{-1} F_d(\phi_d(t), x, y, t)\, \left(\mathbb{B}_d^T\right)^{-1} \equiv \mathbb{F}_d(t, \alpha_d(t)) \\ \phi_d(t_0) &= \left(\mathbb{A}_d^T\right)^{-1}\, \phi_{d,0}, \quad t_0 \leq t \leq T. \end{cases} \tag{5.37}$$

The BDF scheme applied to the (5.37) yields

$$(\phi_d)_{\varrho+1} = \sum_{j=0}^{p} \eta_j\, (\phi_d)_{\varrho-j} + h_d\, \tau\, \mathbb{F}_d(t_{\varrho+1}, (\phi_d)_{\varrho+1}). \tag{5.38}$$

Here h_d is the mesh grid, $t_{\varrho+1} = h_d + t_\varrho$, $(\phi_d)_{\varrho+1} \equiv \phi_d(t_{\varrho+1})$, τ and η_j are the coefficients for the p-step BDF formula given in Table 3.2. Hence, we obtain the 2D ADR equation in a large matrix valued form. We will then rewrite the BDF scheme for ODEs in term of matrix operations. An implementation of the proposed method is in preparation and will be done for the 2D Burgers equation.

6
NUMERICAL ILLUSTRATIONS

In this chapter we demonstrate the accuracy and efficiency of the proposed methods given in previous chapters for different models. For this purpose, we first give various numerical results of the nonlinear ADR problems with forcing terms to investigate the accuracy of the BDFS, SSPRK54S and modified B-spline-SSPRK54 methods. Besides, we address the problem of GS of identical and nonidentical chaotic systems. By considering various numerical examples from physical and biological problems to demonstrate the effectiveness of the proposed control function, the results are presented. In order to show that there is close relation between the generalized chaotic synchronization and nonlinear ADR models, various numerical examples illustrating the accuracy of the present approach are given. Accuracy of the proposed methods are assessed in terms of the relative and absolute errors. Comparison between the proposed methods is carried out in dealing with various problems to check the efficiency and utility of the proposed schemes. The numerical solutions obtained by these methods are tabulated and compared with some works in the literature for different meaningful physical parameters. All computations have been carried out using the currently produced computer codes in MATLAB 2018 on a workstation with 16 significant decimal digits.

6.1 Numerical Examples

The computational domain $[a,b]$ is discredited on the equally spaced points $x_i' = a + i\dfrac{b-a}{k}$, for $i = 0,\dots,k$. It is important to note that the produced solutions are not presented only at the grid points but also at optional points in the solution domain. In order to measure the accuracy of the proposed schemes, the relative errors e_1, e_2 and e_∞ are defined by

$$k \longrightarrow e_1(k) = \frac{\|U - S_k\|_1}{\|U\|_1} = \frac{\sum_{n=0}^{N}\left(\sum_{i=0}^{k} |u(x_i', t_n) - s_{h,n}(x_i')|\right)}{\sum_{n=0}^{N}\left(\sum_{n=0}^{N} |u(x_i', t_n)|\right)}, \tag{6.1}$$

$$k \longrightarrow e_2(k) = \frac{\|U - S_k\|_2}{\|U\|_2} = \frac{\sqrt{\sum_{n=0}^{N}\left(\sum_{i=0}^{k} |u(x_i', t_n) - s_{h,n}(x_i')|^2\right)}}{\sqrt{\sum_{n=0}^{N}\left(\sum_{n=0}^{N} |u(x_i', t_n)|^2\right)}}, \tag{6.2}$$

$$k \longrightarrow e_\infty(k) = \frac{\|U - S_k\|_\infty}{\|U\|_\infty} = \frac{\max_{0\le n\le N}\left(\max_{0\le i\le k} |u(x_i', t_n) - s_{h,n}(x_i')|\right)}{\max_{0\le n\le N}\left(\max_{0\le i\le k} |u(x_i', t_n)|\right)}, \tag{6.3}$$

where $U = (u(x_i', t_n))$ and $S_k = (s_{h,n}(x_i'))$ are the matrices of size $(N+1)\times(N+1)$ whose entries are the values of the exact and numerical solutions, respectively, at points (x_i', t_n) with step size $h = \frac{b-a}{k}$ and time step size $\Delta t = \frac{T - t_0}{N}$. The numerical solutions produced here are at a set of points x_i' which are different from the set of points on the B-spline discretizations.

Example 1

In this example we consider the nonlinear problem, with free external force, known as the Burgers equation

$$u_t = \lambda u_{xx} - u u_x, \tag{6.4}$$

for $(x,t) \in [0,1]\times[1,2]$. The initial condition is

$$u(x,1) = \frac{x}{1 + \frac{1}{\sqrt{C}}\exp(\frac{x^2}{4\lambda})} = u_0(x), \tag{6.5}$$

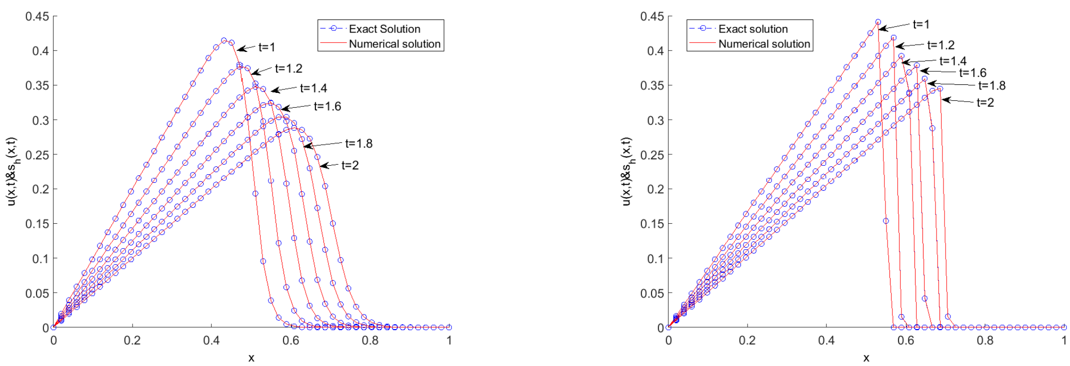
where $C = e^{1/(8\lambda)}$, and the boundary conditions are

$$u(0,t) = g_1(t) = 0 \text{ and } u(1,t) = \frac{1}{t\big(1 + C\sqrt{\frac{t}{C}}\exp(\frac{1}{4\lambda t})\big)} = g_2(t)\,. \tag{6.6}$$

The exact solution [100] is given by

$$u(x,t) = \frac{x}{t\left(1 + C\sqrt{\frac{t}{C}}\exp\left(\frac{x^2}{4\lambda t}\right)\right)}, \quad (x,t) \in [0,1] \times [1,2]. \tag{6.7}$$

For various time values, comparison between the numerical and exact solutions is carried out as seen in Figures 6.1a and 6.1b. In these figures, we observe that the numerical and the exact solutions are in good agreement. The behaviour of the solution is exhibited under the consideration of the physical constants. At any time t with small value of λ, the solution curves are very steep. For the time passed the steepness remains unchanged. Thus, this is a challenging situation that we have obtained the steep solutions. The relative and absolute errors for the computation are presented for $\Delta t = 1E-03$ and $\Delta t = 1E-04$ with different values of λ and k in Tables 6.1 and 6.2. It is concluded from the comparison of the results in these tables that the proposed scheme is very accurate for for different values of λ. The relative error e_∞ is plotted as a function in Figures 6.2a and 6.2b for $\Delta t = 1E-03$ and $\Delta t = 1E-04$, respectively. Notice that the error decreases as k increases for different values of λ. It can be seen that the theoretical convergence and the computational error results are found to be in good agreement when the relatively smaller spatial steps are used. The numerical results are performed for various values of the parameters Δt, λ and k. Note that the numerical convergence is in agreement with the theoretical convergence given in Theorem 3.8. The physical behavior of the solutions is presented in Figures 6.3a and 6.3b. By comparing various λ values, effects of the advection dominant cases are clearly exhibited.



(a) $\lambda = 0.005$ and $\Delta t = 1e-03$ **(b)** $\lambda = 0.0005$ and $\Delta t = 1e-04$

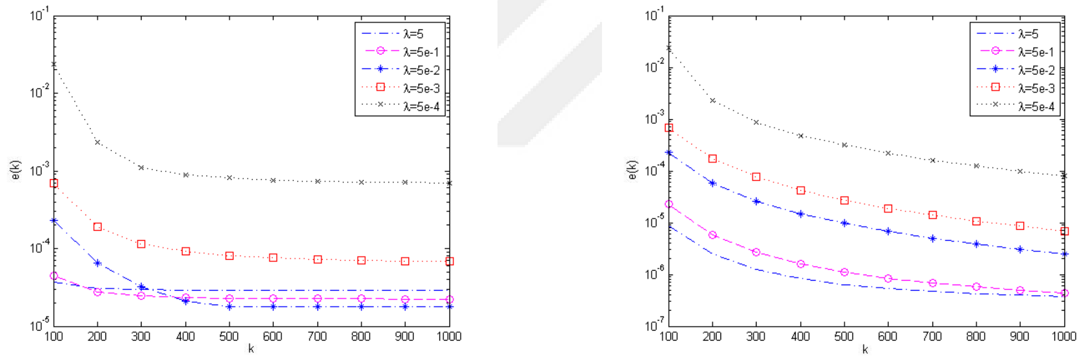
Figure 6.1 Numerical and exact solutions of the Burgers equation

Table 6.1 Comparison of the errors for various values of λ, k with $\Delta t = 1E-03$

BDFS ($e_\infty(k)$)					
k	$\lambda = 5$	$\lambda = 5E-01$	$\lambda = 5E-02$	$\lambda = 5E-03$	$\lambda = 5E-04$
100	3E-07	4E-07	2E-06	6E-06	2E-05
300	2E-07	2E-06	3E-07	1E-05	3E-04
500	2E-06	1E-06	3E-05	7E-04	5E-05
700	2E-05	5E-05	3E-04	4E-03	8E-03
900	2E-06	2E-05	1E-04	6E-04	7E-04

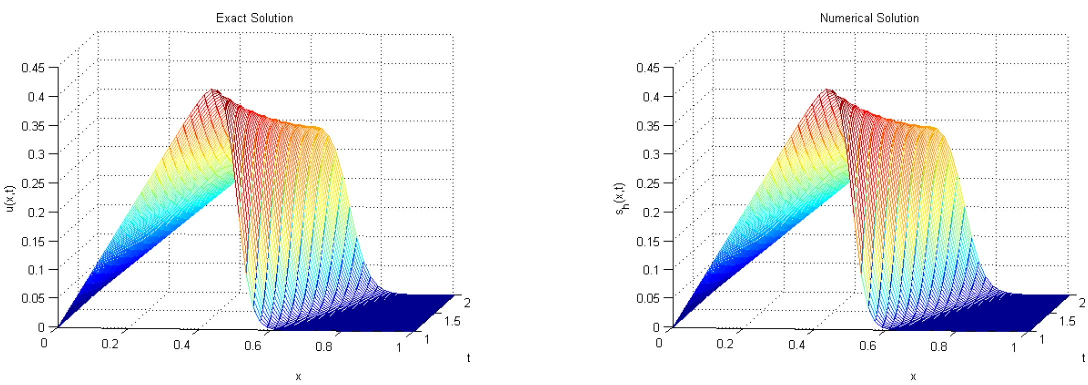
Table 6.2 Comparison of the errors for different values of λ, k with $\Delta t = 1E-04$

BDFS ($e_\infty(k)$)					
k	$\lambda = 5$	$\lambda = 5E-01$	$\lambda = 5E-02$	$\lambda = 5E-03$	$\lambda = 5E-04$
200	2E-08	5E-08	5E-05	1E-04	2E-05
400	8E-08	1E-07	14E-07	4E-05	48E-04
600	5E-08	8E-07	6E-06	1E-05	2E-04
800	4E-07	57E-07	3E-06	1E-05	12E-04
1000	3E-07	44E-07	2E-06	6E-04	7E-03



(a) $\Delta t = 1E-03$ and $t = 2$ **(b)** $\Delta t = 1E-04$ and $t = 2$

Figure 6.2 The presentation of relative errors e_∞ for various time steps



(a) Exact solution **(b)** Numerical solution

Figure 6.3 Exact and numerical solutions for $\Delta t = 1E-04$ and $\lambda = 0.005$

Example 2

In the second example, we consider the Burgers equation with the forcing function

$$u_t = \lambda u_{xx} - uu_x + f(x,t), \tag{6.8}$$

in the domain $[0,1] \times [0,1]$ with the boundary conditions and the initial condition, given by

$$\begin{cases} u(x,0) &= 0, \\ u(0,t) &= 0, \\ u(1,t) &= 0. \end{cases} \tag{6.9}$$

The external force is taken to be

$$\begin{aligned} f(x,t) &= \pi sin(\pi x)cos(\pi t) + \pi sin(\pi t)sin(\pi t)sin(\pi x)cos(\pi x) \\ &+ \lambda \pi^2 sin(\pi t)sin(\pi x), \end{aligned}$$

such that the exact solution reads

$$u(x,t) = sin(x\pi)sin(t\pi), \quad \text{for} \quad (x,t) \in [0,1] \times [0,1]\,. \tag{6.10}$$

Comparison of the computed and exact solutions is presented in Figure 6.4. As in the previous example, we observe that the proposed method exhibits highly accurate results. Tables 6.3 and 6.4 give various values of the relative error $e_\infty(k)$ for $\Delta t = 1E-03$ and $\Delta t = 1E-04$, respectively, for different values of λ and k. The relative errors e_∞ is plotted as a function for $\Delta t = 1E-03$ and $\Delta t = 1E-04$ in Figures 6.5a and 6.5b, respectively. From these tables and figures, we can see that the relative error $e_\infty(k)$ decreases as the value of k increases. We also observe in this example that the theoretical results on the convergence are confirmed by the numerical ones. Physical behaviour of the problem has been presented in a comparative way in Figures 6.6a and 6.6b.

Table 6.3 Relative error $e_\infty(k)$ for various values of λ, k with $\Delta t = 1E-03$ in Example 2

k	$\lambda=5$	$\lambda=5E-01$	$\lambda=5E-02$	$\lambda=5E-03$	$\lambda=5E-04$
200	3E-06	41E-05	1E-06	3E-04	1E-04
600	2E-06	1E-06	3E-05	7E-04	5E-05
1000	1E-06	2E-06	21E-05	5E-04	6E-05

Table 6.4 Relative error $e_\infty(k)$ for various values of λ, k with $\Delta t = 1E-04$ for Example 2

k	$\lambda = 5$	$\lambda = 5E-01$	$\lambda = 5E-02$	$\lambda = 5E-03$	$\lambda = 5E-04$
100	7E-07	4E-06	1E-07	3E-06	1E-05
500	6E-07	3E-07	5E-06	3E-04	1E-05
900	8E-07	2E-07	1E-05	9E-04	2E-04

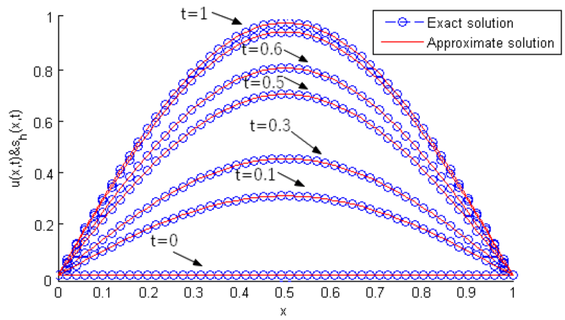
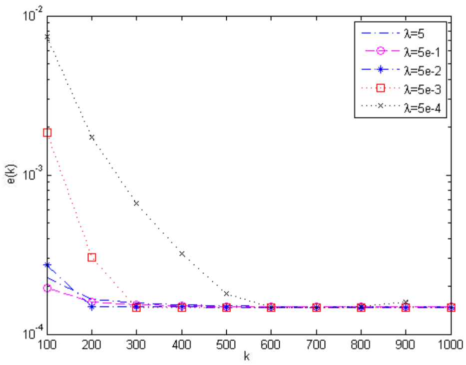
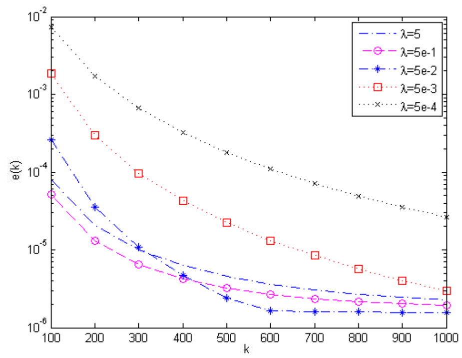


Figure 6.4 Exact and numerical solutions for $\lambda = 0.05$ and $\Delta t = 1E-03$

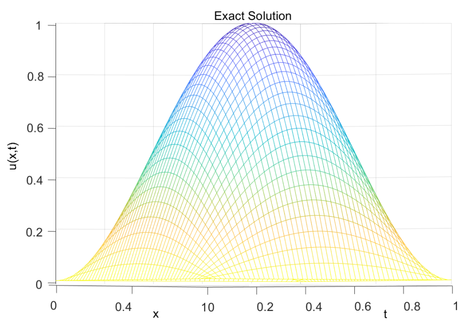


(a) $\Delta t = 1E-03$ and $t = 1$

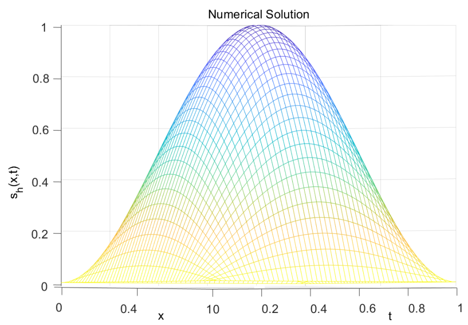


(b) $\Delta t = 1E-04$ and $t = 1$

Figure 6.5 The relative errors e_∞



(a) Exact solution



(b) Numerical solution

Figure 6.6 Exact and numerical solutions for $\Delta t = 1E-04$ and $\lambda = 0.005$

Example 3

In the present example, we consider the nonlinear problem as the Fisher equation, with free external force, given by

$$u_t = \lambda u_{xx} + \mu u(1-u), \quad (x,t) \in [a,b] \times [0,1], \tag{6.11}$$

where $a = -0.2$ and $b = 0.8$, with the initial and boundary conditions

$$u(x,0) = \left(1 + e^{\sqrt{\frac{\mu}{6}}x}\right)^{-2}, \; u(a,t) = u(b,t) = 0 \; .$$

The exact solution of the Fisher equation (6.11) is given by

$$u(x,t) = \left(1 + e^{\sqrt{\frac{\mu}{6}}x - \frac{5\mu}{6}t}\right)^{-2} . \tag{6.12}$$

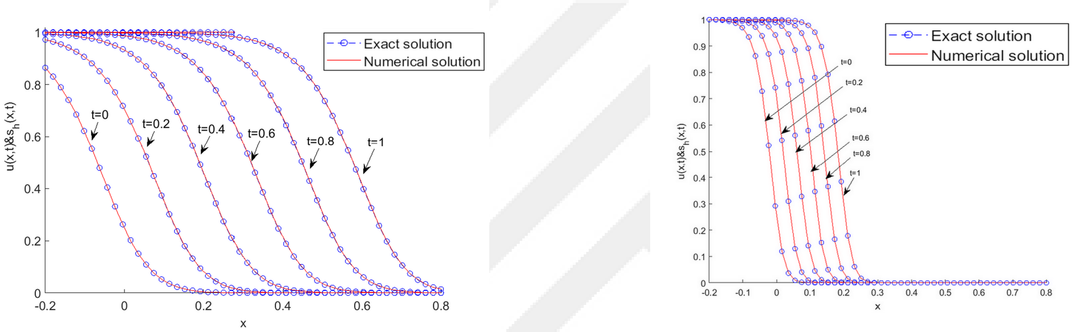
The numerical solutions are computed with the parameters $\lambda = 1$, $\Delta t = 1E-03$, $\Delta t = 1E-04$ for different values of k and μ. In Figures 6.7a and 6.7b the exact and numerical solutions are presented for comparison purposes. The numerical solutions are seen to be good agreement with the exact ones and the sharp behaviours come out. As λ decreases, the curves become very steep. Thus, the results obtained by the proposed method captures the nature of the problem. The numerical solutions to the reaction-diffusion equation (6.11) shows that the reaction is more effective than diffusion, this is why effects of the reaction clearly are seen in the solutions. The relative and absolute errors are presented in Tables 6.5 and 6.6 for different values of the parameters k, Δt and μ. We have seen from the corresponding table that the errors obtained by the BDFS scheme is quite small. The quantitative and qualitative results of the proposed method are highly challenging. The relative errors for small and large values of μ are plotted in Figures 6.8a and 6.8b. The behavior of the numerical solutions is in agreement with the exact solution as seen in Figures 6.9a and 6.9b.

Table 6.5 Relative errors of the proposed methods for the Fisher equation with $\Delta t = 1E-04$

	BDFS ($e_\infty(k)$)					
k	$\mu = 1000$	$\mu = 2000$	$\mu = 3000$	$\mu = 4000$	$\mu = 5000$	$\mu = 5000$
300	1E-04	7E-05	3E-05	7E-04	2E-02	1E-06
420	6E-05	3E-05	1E-05	4E-04	1E-02	7E-02
540	4E-05	2E-05	1E-05	2E-04	6E-03	4E-02

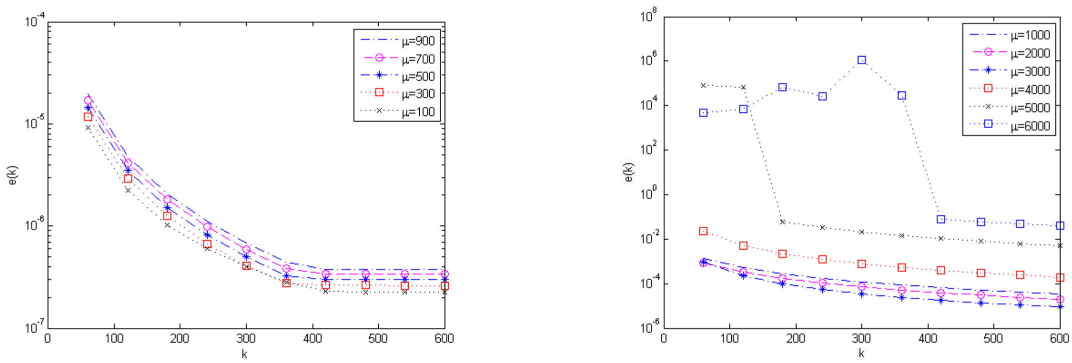
Table 6.6 Relative errors of the proposed methods for the Fisher equation with $\Delta t = 1E-03$

	BDFS ($e_\infty(k)$)				
k	$\mu = 100$	$\mu = 300$	$\mu = 500$	$\mu = 700$	$\mu = 900$
120	2E-08	22E-08	2E-06	28E-06	4E-05
240	64E-07	6E-07	61E-07	6E-05	98E-05
360	2E-07	29E-07	28E-07	2E-05	3E-05
480	1E-06	16E-06	19E-06	2E-05	3E-05
600	1E-06	4E-05	18E-05	2E-04	3E-04



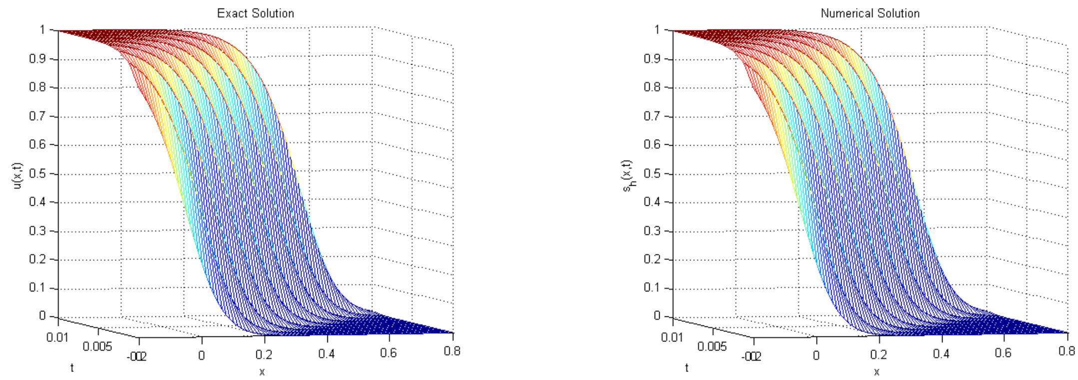
(a) $\lambda = 1$ and $\Delta t = 1E-03$ **(b)** $\lambda = 0.05$ and $\Delta t = 1E-04$

Figure 6.7 Exact and numerical solutions of the Fisher equation for $\mu = 1000$



(a) $\Delta t = 1E-03$ and $t = 1$ **(b)** $\Delta t = 1E-04$ and $t = 1$

Figure 6.8 Relative errors e_∞ of the Fisher equation for various values of Δt and μ



(a) Exact solution　　**(b)** Numerical solution

Figure 6.9 Exact and numerical solutions corresponding to $\Delta t = 1E-04$, $\mu = 120$ and $\lambda = 1$

Example 4

Consider equation (4.1) in the following form

$$u_t - \lambda u_{xx} + \gamma_1 u^{\delta} u_x - \gamma_2 u(1 - u^{\delta}) = 0, \tag{6.13}$$

with the initial condition

$$u(x,0) = (\frac{1}{2} + \frac{1}{2} tanh((\frac{-\gamma_1 \delta}{2(\delta+1)})x))^{1/\delta} = u_0(x), \tag{6.14}$$

and the boundary conditions

$$u(0,t) = (\frac{1}{2} + \frac{1}{2} tanh[\frac{-\gamma_1 \delta}{2(\delta+1)}(-(\frac{\gamma_1}{\delta+1} + \frac{\gamma_2(\delta+1)}{\gamma_1}))t])^{1/\delta} = g_1(t), \tag{6.15}$$

$$u(1,t) = (\frac{1}{2} + \frac{1}{2} tanh[\frac{-\gamma_1 \delta}{2(\delta+1)}(1-(\frac{\gamma_1}{\delta+1} + \frac{\gamma_2(\delta+1)}{\gamma_1}))t])^{1/\delta} = g_2(t). \tag{6.16}$$

Exact solution of equation (6.13) is given by

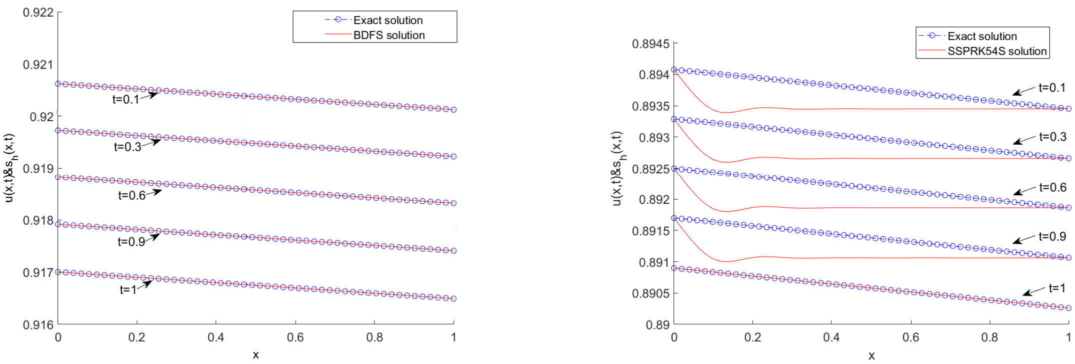
$$u(x,t) = (\frac{1}{2} + \frac{1}{2} tanh[\frac{-\gamma_1 \delta}{2(\delta+1)}(x-(\frac{\gamma_1}{\delta+1} + \frac{\gamma_2(\delta+1)}{\gamma_1}))t])^{1/\delta}. \tag{6.17}$$

This example is provided in the domain $(x,t) \in \Omega = [-1,1] \times [0,1]$ with various values of parameters λ, δ, γ_1, γ_2 and Δt by the current methods. In Table 6.7, the relative errors are computed with various parameters.The BDFS results are still very accurate while the SSPRK54S did not work (N.W.) for larger values of δ. Also, the BDFS results demonstrated that the relative errors decrease as the parameter δ increases. The relative and absolute errors for the computation are considered for various time levels and $\delta = 1,\ 4,\ 40,\ 50$ in Table 6.8 and compared with those available in the literature. It is concluded from the comparison of the obtained results in these tables that the proposed methods are very accurate for all values of $\delta > 0$. We have depicted the BDFS, SSPRK54S solutions and exact solution for different time values in Figures 6.10a and 6.10b. It can be seen from the produced results that the BDFS presents more accurate results than the SSPRK54S. The physical behavior of the solutions is illustrated in Figure 6.11. In conclusion, the computed results in the present problem, show that, the BDFS method has no restriction on the choice of parameter values.

Table 6.7 Relative errors of the proposed methods for Problem (6.13)

$\lambda = 0.01,\ \delta = 500,\ \gamma_1 = \gamma_2 = 0.01,\ \Delta t = 1E-4$		
Errors	SSPRK54S	BDFS
e_1	N.W.	$4.03E-7$
e_2	N.W.	$4.13E-7$
e_∞	N.W.	$3.82E-6$

$\lambda = 0.0005,\ \delta = 10000,\ \gamma_1 = \gamma_2 = 1,\ \Delta t = 1E-4$		
Errors	SSPRK54S	BDFS
e_1	N.W.	$2.29E-8$
e_2	N.W.	$2.01E-8$
e_∞	N.W.	$6.97E-7$



(a) BDFS and exact solutions

(b) SSPRK54S and exact solutions

Figure 6.10 Computed solutions of Problem (6.13) for $\lambda = 1,\ \gamma_1 = \gamma_2 = 0.01,\ \delta = 8,\ h = 0.002$ and $\Delta t = 1E-03$

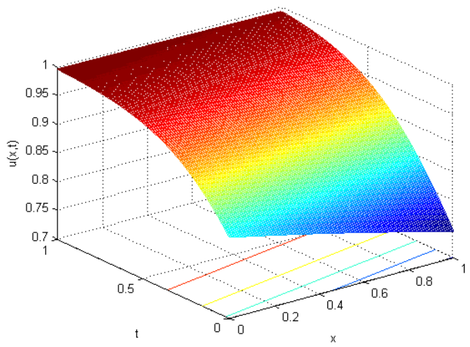
Table 6.8 Comparisons of the errors for Problem (6.13)

$\lambda=\delta=1,\ \gamma_1=\gamma_2=0.001$					
	SSPRK54S	BDFS	Ref. [88]	Ref. [202]	Ref. [214]
Errors	$\Delta t=1E-2$		$\Delta t=1E-4$		
e_1	$3.72E-4$	$2.38E-3$	- - -	- - -	- - -
e_2	$3.78E-4$	$4.15E-4$	- - -	- - -	- - -
e_∞	$4.64E-4$	$3.41E-4$	- - -	- - -	- - -
L_∞	$5.93E-9$	$5.82E-8$	$1.93E-5$	$6.44E-7$	$1.22E-9$

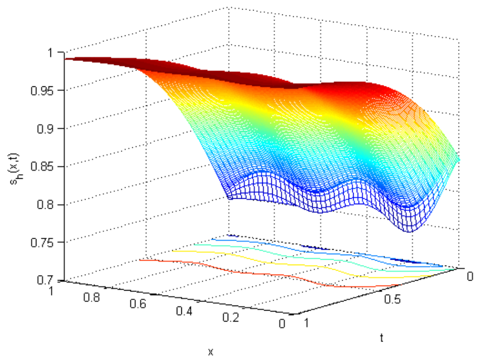
$\delta=4,\ \lambda=1,\ \Delta t=1E-4$					
	SSPRK54S	BDFS	Ref. [202]	Ref. [214]	Ref. [49]
Errors	$\gamma_1=-0.01,\gamma_2=1$		$\beta=1,\gamma_2=0.5$		
e_1	$1.07E-2$	$1.55E-4$	- - -	- - -	- - -
e_2	$1.16E-2$	$5.85E-5$	- - -	- - -	- - -
e_∞	$1.43E-2$	$5.64E-5$	- - -	- - -	- - -
L_∞	$2.03E-5$	$4.82E-8$	$1.22E-5$	$1.08E-8$	$1.44E-6$

$\lambda=1,\ \Delta t=1E-4$					
	SSPRK54S	BDFS	Ref. [88]	Ref. [202]	Ref. [214]
Errors	$\delta=50,\gamma_1=\gamma_2=0.001$		$\delta=2,\beta=\gamma_2=1$		
e_1	$1.67E-3$	$1.02E-5$	- - -	- - -	- - -
e_2	$3.52E-4$	$4.15E-7$	- - -	- - -	- - -
e_∞	$1.33E-4$	$6.34E-7$	- - -	- - -	- - -
L_∞	$3.07E-7$	$3.37E-9$	$2.5E-4$	$2.1E-6$	$1.7E-7$

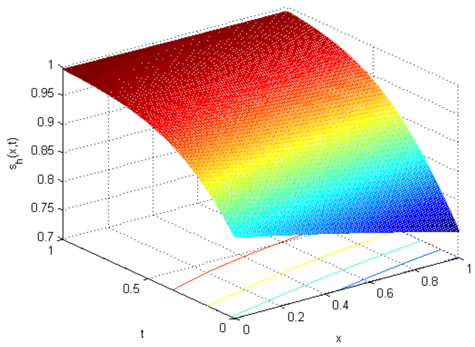
$\lambda=1,\gamma_1=0\ \Delta t=1E-4$					
	SSPRK54S	BDFS	Ref. [202]	Ref. [214]	Ref. [181]
Errors	$\delta=40,\gamma_2=0.001$		$\delta=8,\gamma_2=1$		
e_1	$2.47E-5$	$3.88E-7$	- - -	- - -	- - -
e_2	$2.82E-5$	$4.01E-7$	- - -	- - -	- - -
e_∞	$1.83E-5$	$3.94E-10$	- - -	- - -	- - -
L_∞	$4.85E-9$	$2.03E-17$	$1.19E-11$	$5.5E-16$	$3.6E-11$



(a) Exact solution



(b) SSPRK54S solution



(c) BDFS solution

Figure 6.11 Computed solutions of Problem (6.13) for $\lambda = 1$, $\gamma_1 = \gamma_2 = 0.01$, $\delta = 8$, $h = 0.002$ and $\Delta t = 1E-03$

Example 5

We consider the GBFE with an external force $f(x,t)$

$$u_t - \lambda u_{xx} + \gamma_1 u^{\delta} u_x - \gamma_2 u(1-u^{\delta}) = f(x,t), \tag{6.18}$$

in the domain $[-1,1] \times [-1,1]$ with the homogeneous Dirichlet boundary conditions and the homogeneous initial condition, namely

$$\begin{cases} u(0,t) &= 0 \\ u(1,t) &= 0 \\ u(x,0) &= 0. \end{cases} \tag{6.19}$$

We choose the external source as

$$\begin{aligned} f(x,t) &= (\pi)^2 sin(x\pi)sin(t\pi) + \pi\lambda cos(x\pi)sin(t\pi)(sin(x\pi)sin(t\pi))^{\delta} \\ &- \gamma_1(sin(\pi x)sin(\pi t)(1-(sin(x\pi)sin(t\pi))^{\delta}) + \pi sin(\pi x)cos(\pi t). \end{aligned}$$

The exact solution is

$$u(x,t) = sin(x\pi)sin(t\pi) \qquad (x,t) \in [-1,1] \times [-1,1]. \tag{6.20}$$

Various relative errors for problem (6.18) by using the BDFS and SSPRK54S methods have been considered in Table 6.9. Thus, the results produced by the BDFS method are accurate while the SSPRK54S does not work for large values of δ. For various time values, comparison between the BDFS, SSPRK54S approximation and exact solution is carried out as seen in Figures 6.12a and 6.12b. In the figures, we observe that the BDFS and exact solutions are in good agreement. Physical behaviour of the problem (6.18) has been captured in Figure 6.13. It can be seen that the proposed scheme is in very good agreement with the exact one and exhibits physical characteristics of the problem correctly.

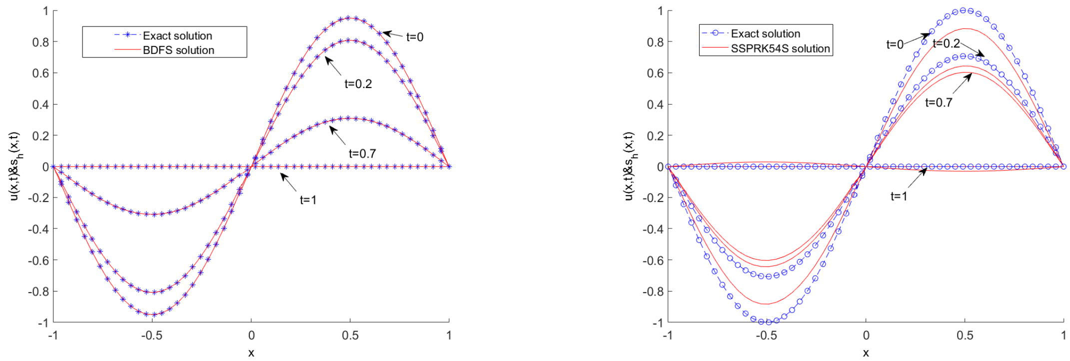
Table 6.9 Relative errors of the GBFEF for the proposed methods

$\lambda = 0.01$, $\delta = 1$, $\gamma_1 = \gamma_2 = 1$, $\Delta t = 1E-3$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	$2.19E-1$	$8.49E-3$
e_2	$1.44E-1$	$8.55E-3$
e_∞	$1.86E-1$	$7.88E-3$

$\lambda = 0.001$, $\delta = 4$, $\gamma_1 = -0.01$, $\gamma_2 = 1$, $\Delta t = 1E-4$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	$1.04E-1$	$3.47E-3$
e_2	$1.34E-1$	$4.95E-4$
e_∞	$1.63E-1$	$5.74E-4$

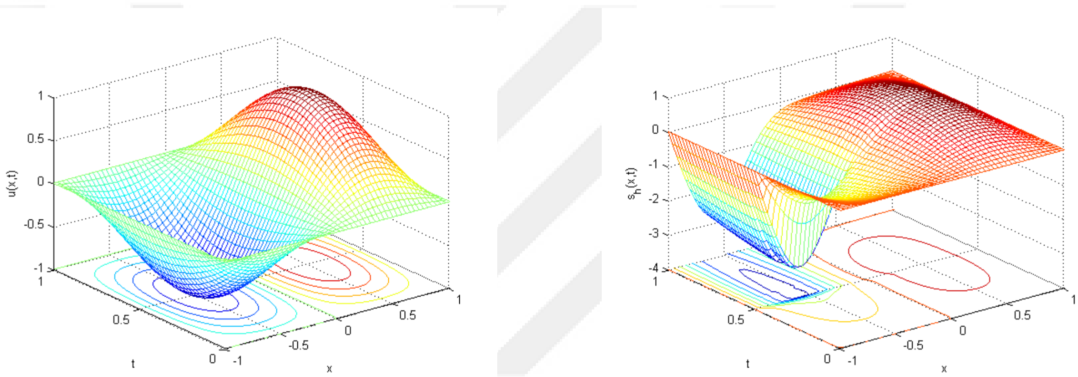
$\lambda = 0.0001$, $\delta = 500$, $\gamma_1 = \gamma_2 = 0.01$, $\Delta t = 1E-4$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	N.W.	$4.17E-3$
e_2	N.W.	$4.92E-5$
e_∞	N.W.	$4.33E-5$

$\lambda = 0.0001$, $\delta = 10000$, $\gamma_1 = \gamma_2 = 1$, $\Delta t = 1E-5$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	N.W.	$2.48E-3$
e_2	N.W.	$3.66E-3$
e_∞	N.W.	$3.28E-3$

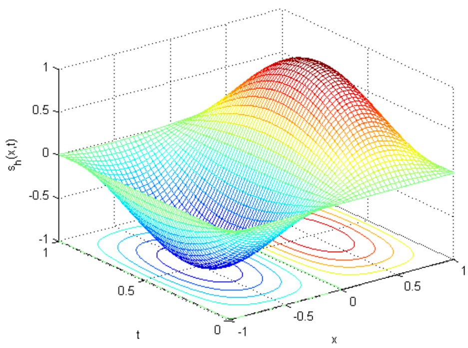


(a) BDFS and exact solutions **(b)** SSPRK54S and exact solutions

Figure 6.12 Computed solutions of Problem (6.18) for $\lambda = 0.01, \gamma_1 = \gamma_2 = 0.001$, $\delta = 4$ and $\Delta t = 1E-02$



(a) Exact solution **(b)** SSPRK54S solution



(c) BDFS solution

Figure 6.13 Computed solutions of Problem (6.18) for $\lambda = 0.001$, $\gamma_1 = 0.001$, $\gamma_2 = 1$, $\delta = 8$ and $\Delta t = 1E-03$

Example 6

In this problem, we consider the GBHE of the form of equation (4.11) with the exact solution given by

$$u(x,t) = \Big(\frac{C}{2} + \frac{C}{2} tanh\big[a_1\big(x - a_2 t\big)\big]\Big)^{1/\delta}, \tag{6.21}$$

where

$$a_1 = \frac{-\gamma_1\delta + \delta\sqrt{\gamma_1^2 + 4\gamma_2(1+\delta)}}{4(1+\delta)} C,$$

and

$$a_2 = \frac{C\gamma_1}{(1+\delta)} - \frac{(1+\delta - C)(-\gamma_1 + \sqrt{\gamma_1^2 + 4\gamma_1(1+\delta)})}{2(1+\delta)}.$$

Here, γ_1, γ_2, δ, λ are parameters and $C = 0.1$. Initial and boundary conditions are taken from the exact solution. Approximate solutions of this problem are obtained by using δ as 1, 8, 500, 10000 for various values of γ_1, γ_2 and λ in the domain $(x,t) \in \Omega = [-1,1] \times [0,1]$. In Table 6.10, accuracy of the current schemes is examined by computing the relative errors for large values of δ and smaller values of λ. Here, it can be concluded that the BDFS results are in good agreement with the exact solution for large values of δ while the SSPRK54S does not work (N.W.). The relative and absolute errors are documented in Table 6.11 and are compared with some previous works. We have noticed from the corresponding table that the errors obtained by the BDFS and SSPRK54S methods are quite small and furthermore, better than most of the schemes available in the literature. Physical behavior of the problem is captured in Figure 6.14. Note that behaviour of the problem with the BDFS is in good agreement with exact solution at free of choice of the physical parameters. The BDFS and SSPRK54S solutions with exact solution of this example are plotted in Figure 6.15. It can be concluded that the BDFS scheme solutions are very compatible with the exact solution and, more accurate than the SSPRK54S method.

Table 6.10 Relative errors of the GBHE for the proposed methods

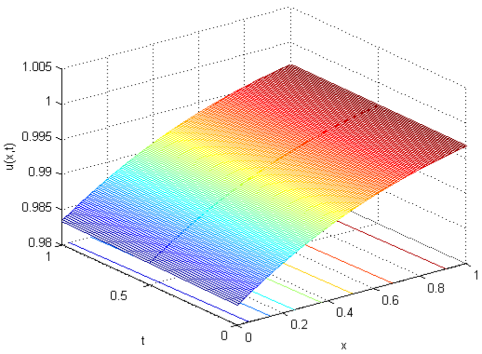
$\lambda = 0.001$, $\gamma_1 = \gamma_2 = 1000$, $\delta = 500$ $\Delta t = 1E-4$, $C = 0.0001$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	N.W.	$7.94E-3$
e_2	N.W.	$8.07E-3$
e_∞	N.W.	$1.08E-2$

$\lambda = 0.0001$ $\gamma_1 = 1$, $\gamma_2 = 5$, $\delta = 10000$ $\Delta t = 1E-4$, $C = 1$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	N.W.	$4.00E-4$
e_2	N.W.	$2.33E-4$
e_∞	N.W.	$3.15E-5$

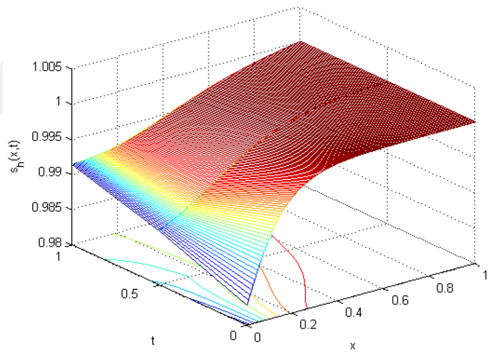
Table 6.11 Comparison of the errors for Problem (6.21)

$\lambda=\delta=1,\ \gamma_1=\gamma_2=0.001,\ C=0.001$					
	SSPRK54S	BDFS	Ref. [88]	Ref. [202]	Ref. [214]
Errors	$\Delta t=1E-3$		$\Delta t=1E-4$		
e_1	$5.10E-5$	$6.67E-7$	- - -	- - -	- - -
e_2	$5.22E-5$	$6.90E-7$	- - -	- - -	- - -
e_∞	$7.63E-6$	$8.91E-8$	- - -	- - -	- - -
L_∞	$6.73E-11$	$1.02E-17$	$1.93E-7$	$3.74E-8$	$4.26E-17$

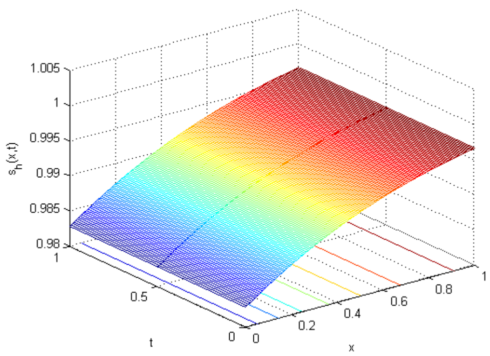
$\delta=8,\ \lambda=\gamma_2=1,\ \Delta t=1E-4$					
	SSPRK54S	BDFS	Ref. [89]	Ref. [202]	Ref. [214]
Errors	$C=0.01,\gamma_1=80$		$C=0.0001,\gamma_1=100$		
e_1	$1.07E-2$	$1.55E-4$	- - -	- - -	- - -
e_2	$1.16E-2$	$5.85E-5$	- - -	- - -	- - -
e_∞	$1.43E-2$	$5.64E-5$	- - -	- - -	- - -
L_∞	$2.03E-5$	$4.82E-8$	$4.58E-8$	$1.27E-8$	$5.55E-17$



(a) Exact solution

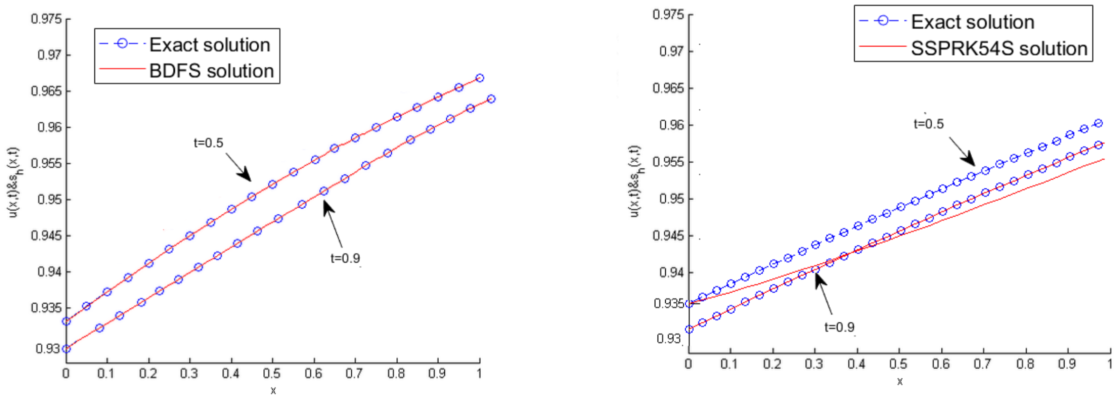


(b) SSPRK54S solution



(c) BDFS solution

Figure 6.14 Computed solutions of Problem (6.21) for $\lambda=0.01$, $\gamma_1=\gamma_2=0.1,\delta=40$, $\Delta t=1E-3$ and $C=0.1$ with $h=0.002$



(a) BDFS and exact solutions **(b)** SSPRK54S and exact solutions

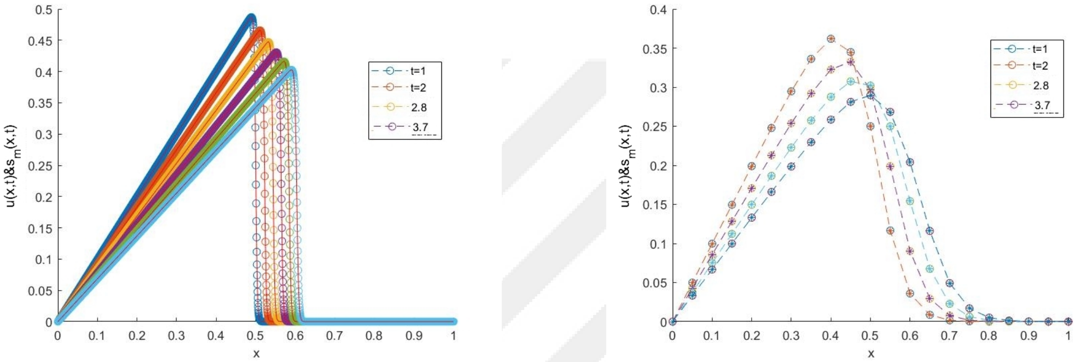
Figure 6.15 Computed solutions of Problem (6.21) for $\lambda = 1$, $\gamma_1 = 0.01$, $\gamma_2 = 0.5$, $\delta = 80$ and $\Delta t = 1e-04$ with $h = 0.02$

Example 7

In this problem, we demonstrate accuracy of the modified B-spline-SSPR54 scheme. In order to measure the accuracy of the current scheme, we discretize the solution domain $[a, b]$ into uniformly grid sizes h by new equally spaced points x'_i, for $i = 0, \ldots, k$. For our discretization with the forward Euler, the linear stability yields the restriction and given by: CFL= $\lambda\Delta t/h^2$ and we present the relative error as defined in (6.3). We also consider the exact solution of equation (4.55) as given in reference [99]

$$u(x,t) = \frac{x/t}{1 + \sqrt{t/t_0} e^{x^2/4\lambda t}}, \quad t \geq 1; \quad 0 \leq x \leq 1,$$

where $t_0 = e^{1/8\lambda}$. The initial condition is taken from the exact solution when $t = 1$. Boundary conditions are $u(0,t) = u(1,t) = 0$. The numerical solution provides shock like solution of the Burgers equation with those given in the paper [99]. Approximate solutions of this problem illustrate the propagation of shock for $\lambda = 5E-03, 5E-04, 5E-05$ and $\lambda = 5E-06$ at various values of the time and space steps. Figures 6.16a and 6.16b show the shock for the viscosity λ as $5E-04$ and $5E-03$, respectively. From these figures, we have observed the initial shocks are of sharp behaviour and, this steepness continues during time progression. In the same figures, the agreement between the numerical and the exact solutions appears satisfactorily. So that, the behaviours of the problem are in very good agreement for different kinematic viscosities. They also keep the correct physical characteristics of the current problem. Figures 6.17a and 6.17b provide that the produced solutions become sharper, than the previous ones, with various kinematic viscosity values $\lambda = 5E-05$ and $\lambda = 5E-06$, respectively. Here, we have concluded that the steepness remains almost unchanged as the time progresses. The same figures also show the effectiveness of the current scheme with small kinematic viscosity. In Table 6.12, we present e_∞ errors at various values of x, t for various time and spatial increments and, compare with some works presented in the literature. We have seen that the numerical solutions are observed to be very close to the exact solution. And also, the comparisons presented that the current method offers better results than the numerical schemes considered by the literature [99]. The physical behavior of the current problem at various λ values are depicted in Figure 6.18.

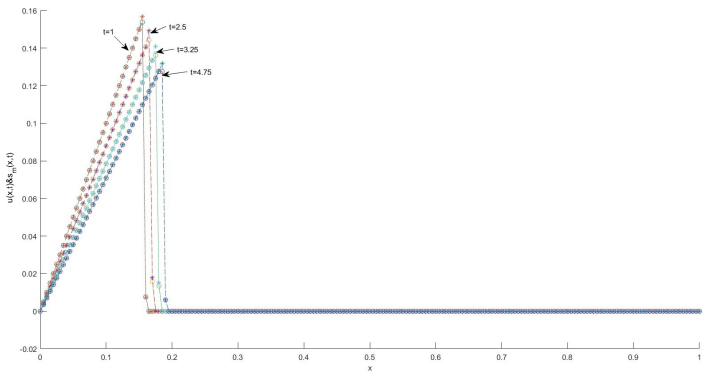


(a) $\lambda = 5E-04$ **(b)** $\lambda = 5E-03$

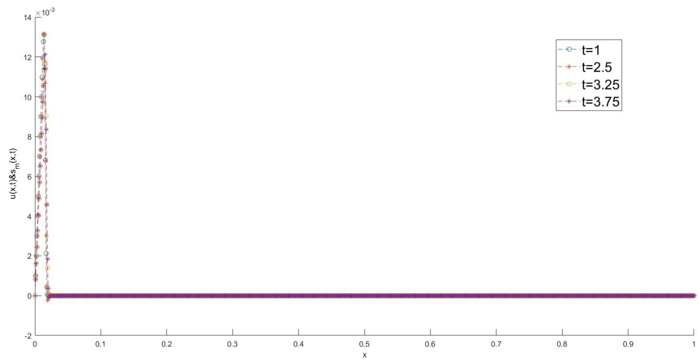
Figure 6.16 Numerical and exact solutions of Example 7 at different times produced for the various parameters

Table 6.12 Comparison of the present results with the literature for $\lambda = 5E-04$, $\Delta t = 1E-02$ and $h = 5E-03$

x	t	Ref. [99]	Present Method	Exact
0.1	1.7	0.058830	0.058821	0.058820
0.3	1.7	0.176480	0.176472	0.176470
0.5	2.5	0.200010	0.200004	0.200000
0.7	3.25	0.215390	0.215380	0.215380
0.9	3.25	0.123580	0.124354	0.124350

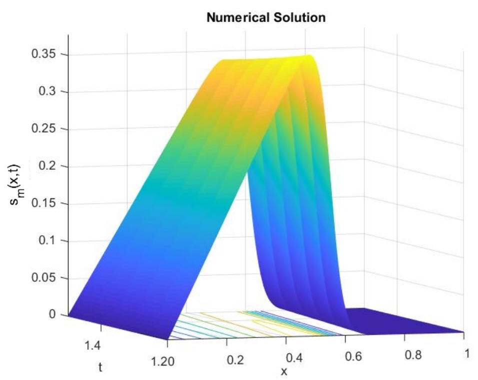


(a) CFL=1.4$E-03$, $\lambda = 5E-05$, $\Delta t = 1.1E-04$

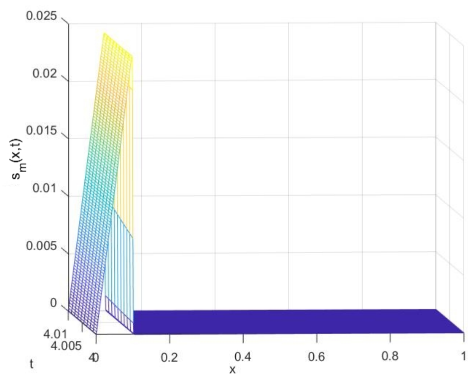


(b) CFL=1.4$E-02$, $\lambda = 5E-06$, $\Delta t = 1.1E-02$

Figure 6.17 Numerical and exact solutions of Example 7 at $t = 1.5$



(a) CFL=1.1$E-02$, $\lambda = 5E-03$



(b) CFL=1.4$E-02$, $\lambda = 5E-06$

Figure 6.18 Physical behavior of the computed solution for Example 7 at various λ values

Example 8

The exact solution for the problem (4.55) is [114]

$$u(x,t) = 2\pi\nu \frac{\sum_{j=1}^{\infty} j a_j sin(j\pi x) e^{-j^2\pi^2\nu t}}{a_0 + 2\sum_{j=1}^{\infty} a_j cos(j\pi x) e^{-j^2\pi^2\nu t}},$$

where

$$a_j = \int_0^1 e^{-(2\pi\nu)^{-1}(1-cos(\pi\nu))} cos(j\pi x)dx \quad \text{for all} \quad j \geq 1.$$

The initial and boundary conditions for this example are

$$u(0,t) = u(1,t) = 0 \text{ and } u(x,0) = sin(\pi x) \quad (x,t) \in \Omega = [0,1] \times [0,T].$$

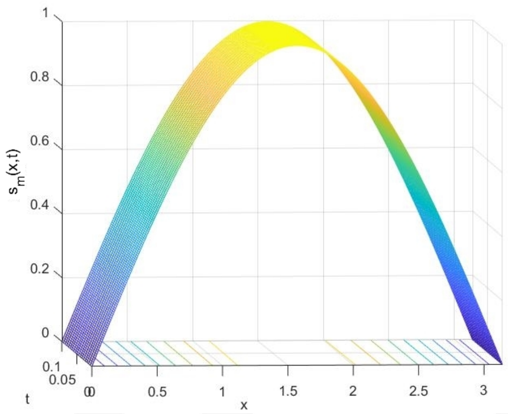
The produced results together with exact solutions are documented in Tables 6.13 and 6.14. It is seen that the agreement between the exact and numerical solutions appear satisfactorily. In the same tables, we compare between the proposed scheme and some previous works [188, 99]. Figure 6.19 provides the physical behavior of the various λ values for various CFL conditions. Here, we have observed that the physical behavior of this problem cannot be kept for $\lambda < 1E-03$. The current scheme exhibits more accurate results than the rival methods. The approximation solutions are visualized at various values of the parameters λ, Δt and CFL in Figures 6.20a, 6.20b and 6.20c. The initial shock is very steep $\lambda = 1E-02$. Here, it can be observed that the discrete results are found to be in very good agreement with the exact solution. The exact values are not practical to make comparison for the small values of λ because of slow convergence of the Fourier series result.

Table 6.13 Comparison of the present results with the literature for $\lambda = 1$, $\Delta t = 1E-04$ and $h = 1E-01$

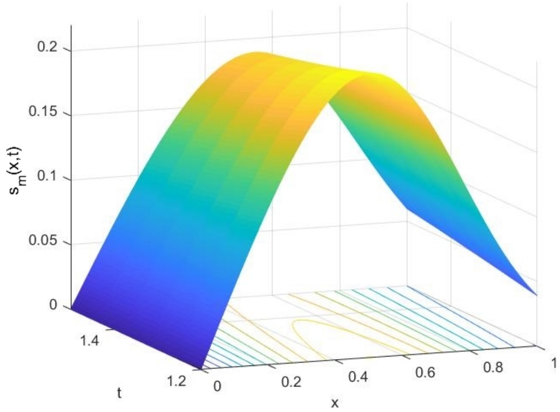
x	Ref. [188]	Ref. [99]	Present Method	Exact
0.1	0.10831	0.10898	0.10818	0.10954
0.2	0.20724	0.20862	0.20709	0.20979
0.3	0.28799	0.29013	0.28788	0.29190
0.4	0.34273	0.34564	0.34273	0.37158
0.5	0.36531	0.36895	0.36551	0.35905
0.6	0.35223	0.35633	0.35266	0.30991
0.7	0.30400	0.30748	0.30396	0.30991
0.8	0.22358	0.22606	0.22318	0.12069
0.9	0.11860	0.11988	0.11813	0.12069

Table 6.14 Comparison of the present results with the literature for $\lambda = 1$, $\Delta t = 1E-04$ and $h = 5E-02$

x	Ref. [188]	Ref. [99]	Present Method	Exact
0.1	0.10920	0.10937	0.10914	0.10954
0.2	0.20912	0.20946	0.20900	0.20979
0.3	0.29088	0.29140	0.29071	0.29190
0.4	0.34658	0.34728	0.34639	0.34792
0.5	0.36997	0.37083	0.36979	0.37158
0.6	0.35740	0.35826	0.35716	0.35905
0.7	0.30847	0.30917	0.30814	0.30991
0.8	0.22676	0.22725	0.22644	0.22782
0.9	0.12012	0.12038	0.11992	0.12069

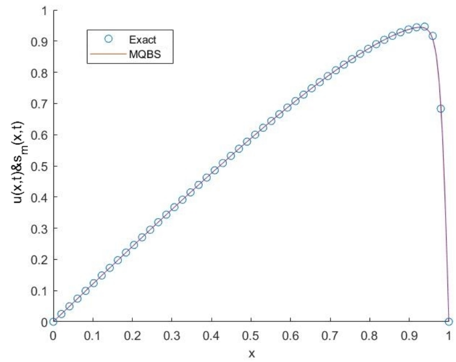


(a) CFL=$1.1E-02$, $\lambda = 1E-03$

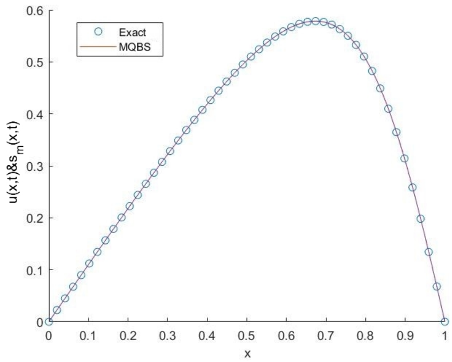


(b) CFL=$1.4E-02$, $\lambda = 1E-04$

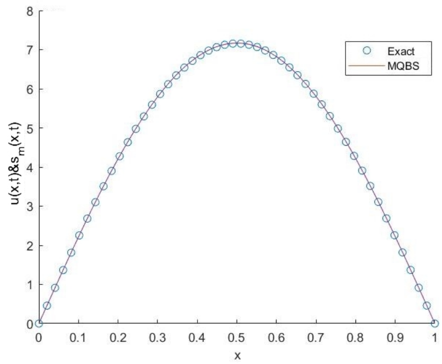
Figure 6.19 Physical behavior of the computed solution for Example 8 at various values of λ and CFL



(a) $\lambda = 1E-02$, $\Delta t = 1E-03$, and CFL=$4E-03$



(b) $\lambda = 1E-01$, $\Delta t = 1E-03$ and CFL=$4E-02$



(c) $\lambda = 1$, $\Delta t = 1E-03$ and CFL=$4E-01$

Figure 6.20 Numerical and exact solutions of Example 8 at $t = 5E-01$ for various values of the parameters

6.2 Numerical Results of Synchronization of Identical and Nonidentical Chaotic Systems

Various numerical simulations are performed to verify the effectiveness of the proposed control function. We provide the problem of GS of identical and nonidentical systems with various dimension spaces. The interval time $[t_0, T]$ is dividing into N subintervals $[t_n, t_{n+1}]$ with $t_n = t_0 + n\Delta t$ with $n = 0, \cdots, N \quad \Delta t = \dfrac{T - t_0}{N}$. Let x_n and y_n present the approximation of the vectors $x(t_n)$ and $y(t_n)$, respectively. To measure the accuracy of the proposed methods, the relative error R_e defined by

$$t_n \longmapsto R_e(t_n) = \sqrt{\frac{\sum_{n=0}^{N} \|y_n - \Upsilon(x_n)\|^2}{\sum_{n=0}^{N} \|\Upsilon(x_n)\|^2}}, \tag{6.22}$$

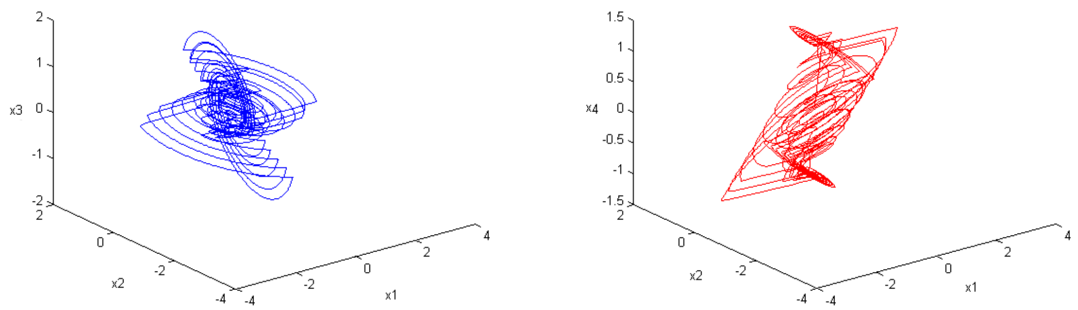
and the partial relative error function r_e defined by

$$t_n \longmapsto r_e(t_n) = \frac{\|y_n - \Upsilon(x_n)\|}{\|\Upsilon(x_n)\|}. \tag{6.23}$$

The globally generalized synchronization with respect to control function Υ is also confirmed by the following simulation results.

Example 9

In this example, we propose the GS of identical systems with spatial dimension $n = m = 4$ concerning the Memristor chaotic systems (4.65)-(4.67) with initial conditions $x(0) = (1,1,1,1)^T$ and $y(0) = (10,20,50,20)^T$, respectively. The interval time is $[t_0, T] = [0,20]$ with $N = 50000$. The orbit states of the Memristor system (4.65) has a chaotic attractor portrayed for the parameter values fixed as $\iota_1 = 4$, $\iota_2 = 1$, $\iota_3 = 0.65$, $\rho_1 = 0.2$ and $\rho_2 = 10$, as shown in Figure 6.21. Figure 6.22 provides the relative error given by (6.22). The relative error r_e represented by (6.23) is in Figure 6.23, and it illustrates the behavior of the error, properly. Figure 6.24 depicts the behaviour between various components of the systems for different values of parameter k_2. The synchronization is seen when k_2 becomes the larger and larger. We observe that the error decreases as k_2 increases for the synchronizational motion of the chaotic system (4.67).



(a) (b)

Figure 6.21 Chaotic attractors of system (4.65)

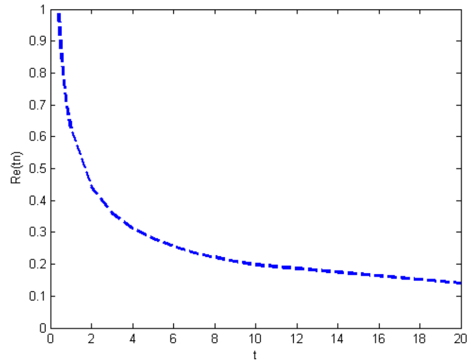


Figure 6.22 Behaviour of the relative error $R_e(t_n)$

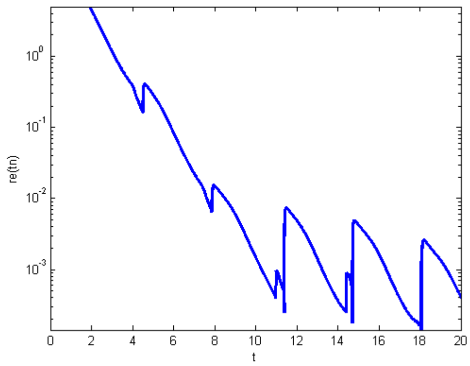


Figure 6.23 Behaviour of the relative error $r_e(t_n)$

Example 10

In this problem, the proposed method is applied to synchronizational behaviour of two identical HR neuron systems (4.68). We focus on the GS of two well defined chaotic systems in which the control method can be applied to the drive-response synchronization of the HR neurons. The time interval is taken to be $[t_0, T] = [0, 200]$. The chaotic Bursting system (4.68) exhibits a well defined chaotic attractor with constant values: $\iota_4 = 3$; $\iota_5 = 5$; $\iota_6 = 4$; $C_{33} = -8/5$; $\rho_3 = 3.25$ and $\iota_7 = 0.005$, and the initial conditions $x(0) = (-0.54, -1, 3)$ and $y(0) = (0.54, 1, -3)$ of the drive and response systems respectively as shown in Figure 6.25. The hypothesis of Theorem 3.3 are confirmed and we have the synchronization analysis between the drive and response systems. The results are also confirmed by various simulations for a coupling strength k_2 which is small enough. The relative error r_e (6.23) is presented in Figure 6.26. We demonstrate that the convergence of the relative error converges to zero. Figure 6.27 presents the time series of component x_1 from the drive system and component y_1 from the response system.

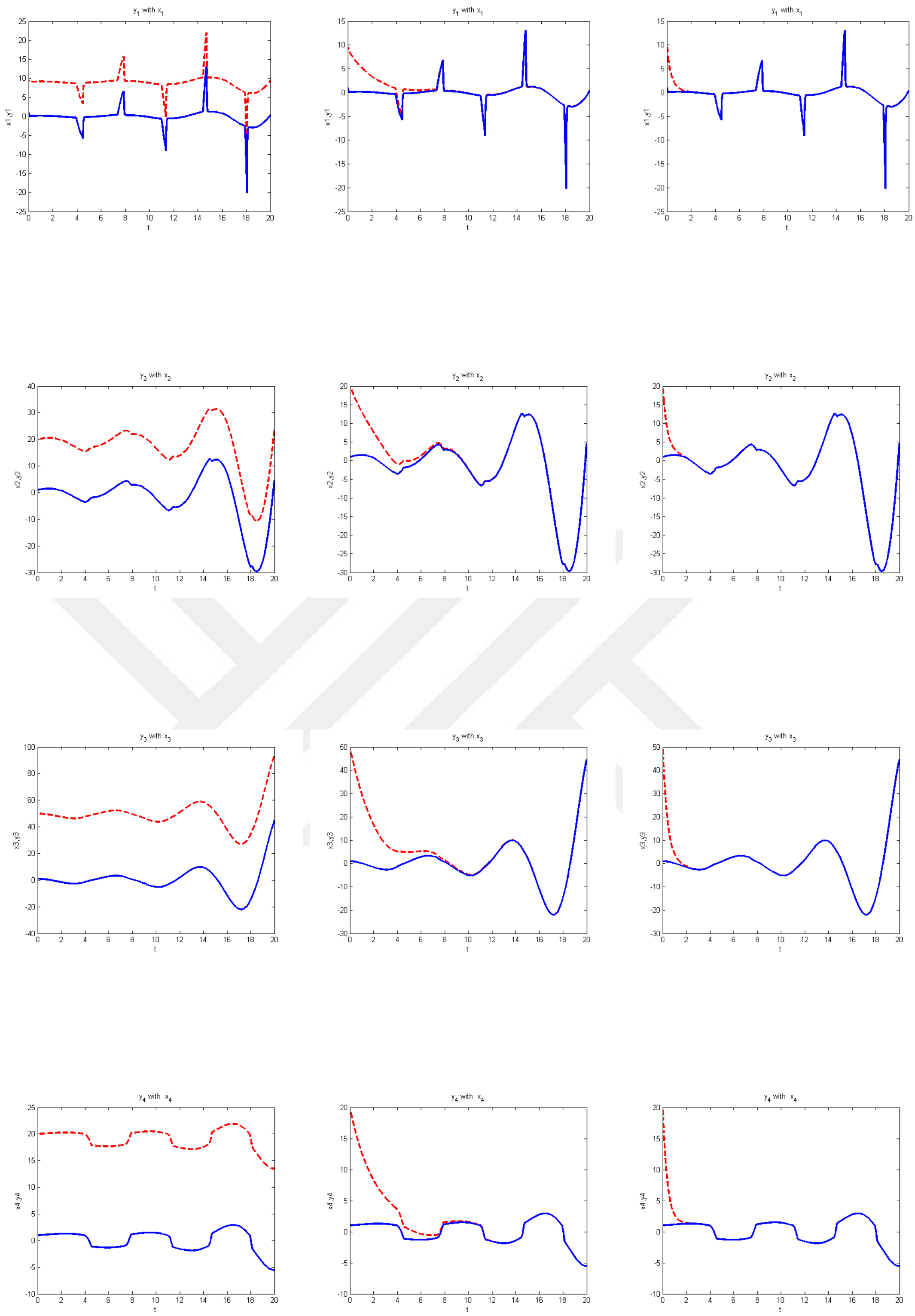
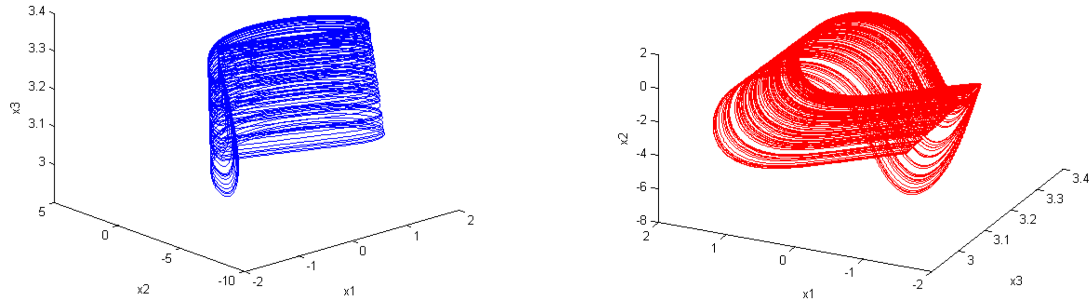


Figure 6.24 Time series for $x_i(t),\ y_i(t)(i = 1, 2, 3, 4)$ at various values of the coupling constant $k_2 = 0,\ 0.5,\ 1.5$



(a) **(b)**

Figure 6.25 Chaotic bursting of neuronal system

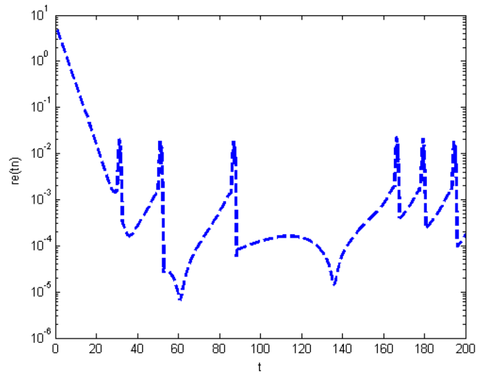


Figure 6.26 Behaviour of the relative error $r_e(t_n)$

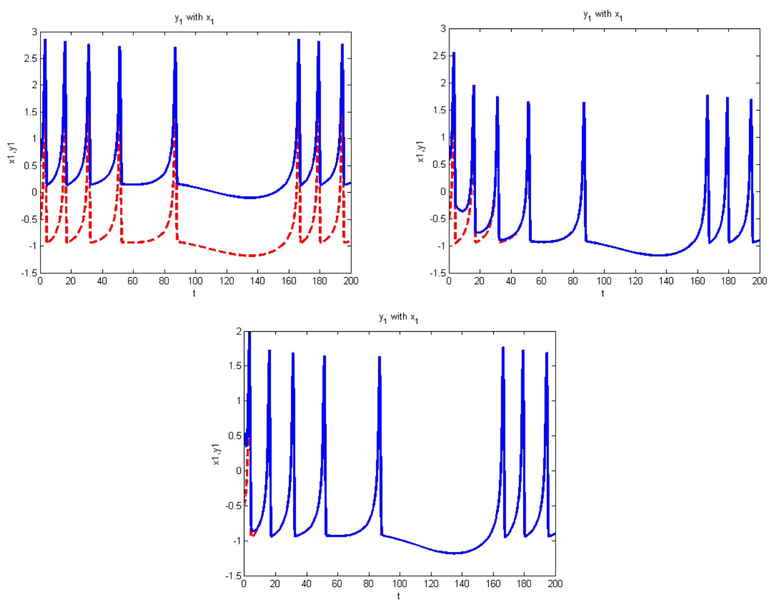


Figure 6.27 Synchronization between two identical HR neurons systems: amplitudes y_1 according x_1 at various coupling strengths $k_2 = 0,\ 0.1,\ 0.2$

Example 11

In this example, the GS is applied to study synchronization motions of two identical (4.70). The chaotic attractor (4.70) displays at parameter values: $\iota_8 = 0.000005$; $\iota_9 = 0.00009$; $\iota_{10} = 10000$ and $\iota_{11} = 0.5$, and the initial condition $x(0) = (0, 0)$ as shown in Figure 6.28. In this figure, one obtains the chaotic attractors by choosing the sufficiently best parameter values in problem (4.70).

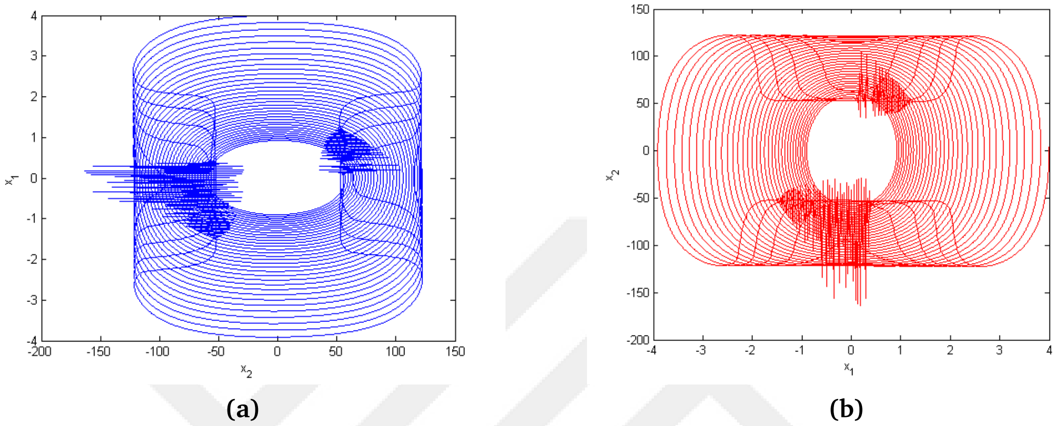


Figure 6.28 Chaotic modeling of the BZ reaction

The time interval is taken to be $[t_0, T] = [0, 1.5]$. Notice that the theoretical results of Theorem 3.3 is also satisfied by the numerical results, and we illustrate the behaviour of the GS by using a small value of the coupling strength k_2. Also, successes in designing of coupled control functions have the fast GS in the mechanistic understanding of these often complex reactions. In Figure 6.30, we conclude that the drive system is synchronized with the response system for the coupling strengths $k_2 \geq 0.013$. The graph of the relative error $r_e(t_n)$ (6.23) is illustrated in Figure 6.29 and it figures out that the convergence of the results has currently been satisfied.

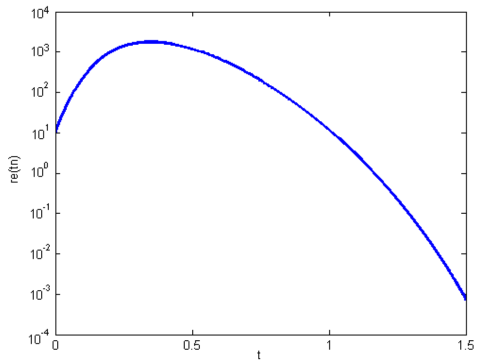


Figure 6.29 Behaviour of the relative error $r_e(t_n)$

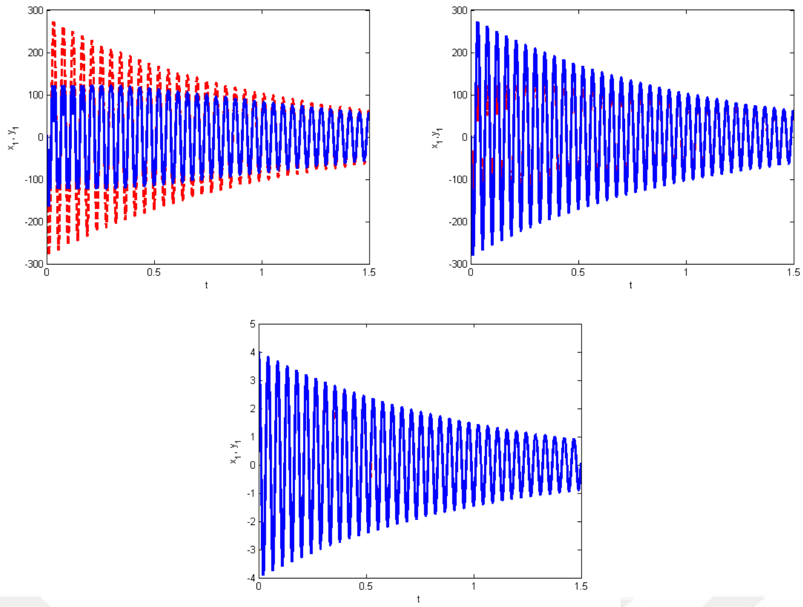
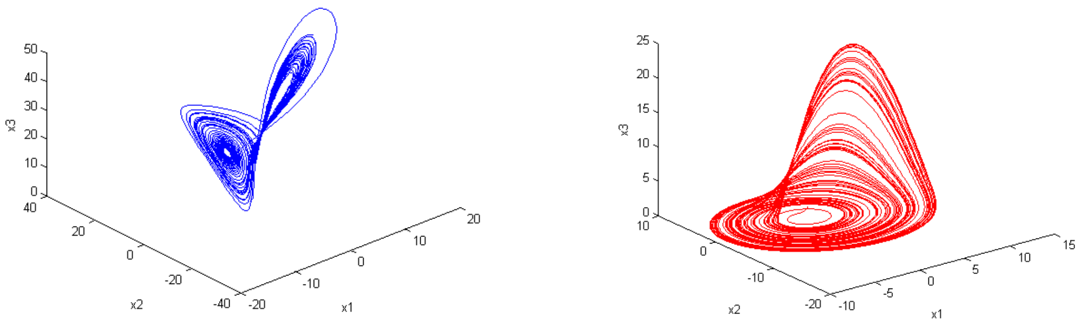


Figure 6.30 Synchronization between two identical BZ reaction systems: amplitudes y_1 according x_1 at various values of the coupling strengths $k_2 = 0.001,\ 0.01,\ 0.013$

Example 12

Here, the current approaches are applied to investigate the GS behaviour of two non identical chaotic systems (4.72) and (4.73) in both cases of $n < m$ and $n > m$. One can obtain that the Lyapunov exponents are positive, and showing that the famous Lorenz and Rössler systems exhibit the chaotic attractor with the parameter values $\iota_{12} = 10$, $\iota_{13} = 8/3$, $\iota_{14} = 28$, $\iota_{15} = 0.25$, $\iota_{16} = 3$, $\iota_{17} = 0.5$ and $\iota_{18} = 0.05$. The initial conditions are $x(0) = (10, 10, 10)^T$ and $y(0) = (1, 1, 1, 1)^T$, respectively; in the time interval $[t_0, T] = [0, 20]$ with $N = 50000$ (see Figuer 6.31). Notice that the theoretical results of Theorem 3.4 are also satisfied by the approximate results, and we catch the synchronization by using the large value of the coupling strength k_2. In Figure 6.32, we obtain that the drive system is synchronized with the response system for the coupling $k_2 \geq 7.58$. The graph of the relative error r_e (6.23) is represented in Figure 6.33 and it finds out the convergence of the currently results computed. The same systems were also studied in references [13, 134, 198].

In the second case for $n > m$, we consider the Rössler system as the drive, and making the Lorenz system as the response. In our simulation, we set the coupling $k_2 = 0.5$, while the initial conditions are given: $x(0) = (-5, -5, 10, 10)^T$ and $y(0) = (10, 10, 10)^T$ and the time interval is taken to be $[t_0, T] = [0, 2500]$. Figure 6.34 represents that the error state converges to zero, in this case; we confirmed the Theorem 3.3 to estimate the small coupling strength. The designed controller functions, the drive and response system are well synchronized.



(a) **(b)**

Figure 6.31 Chaotic attractor of the Lorenz and Rössler systems

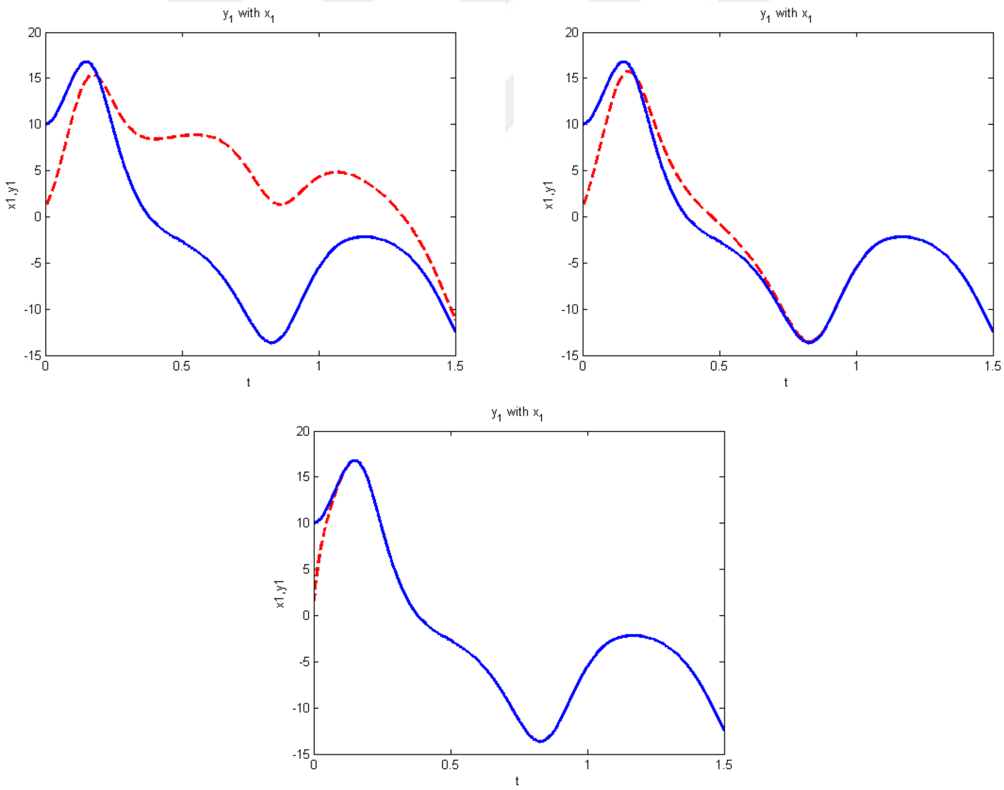


Figure 6.32 Time series for $x_i(t), y_i(t) (i = 1, 2, 3, 4)$ at various values of the coupling constant $k_2 = 7.58,\ 10,\ 20$

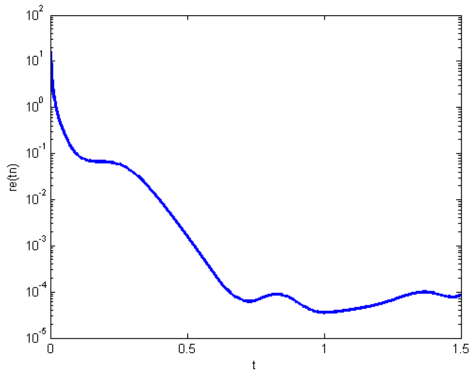


Figure 6.33 Behaviour of the relative error $r_e(t_n)$

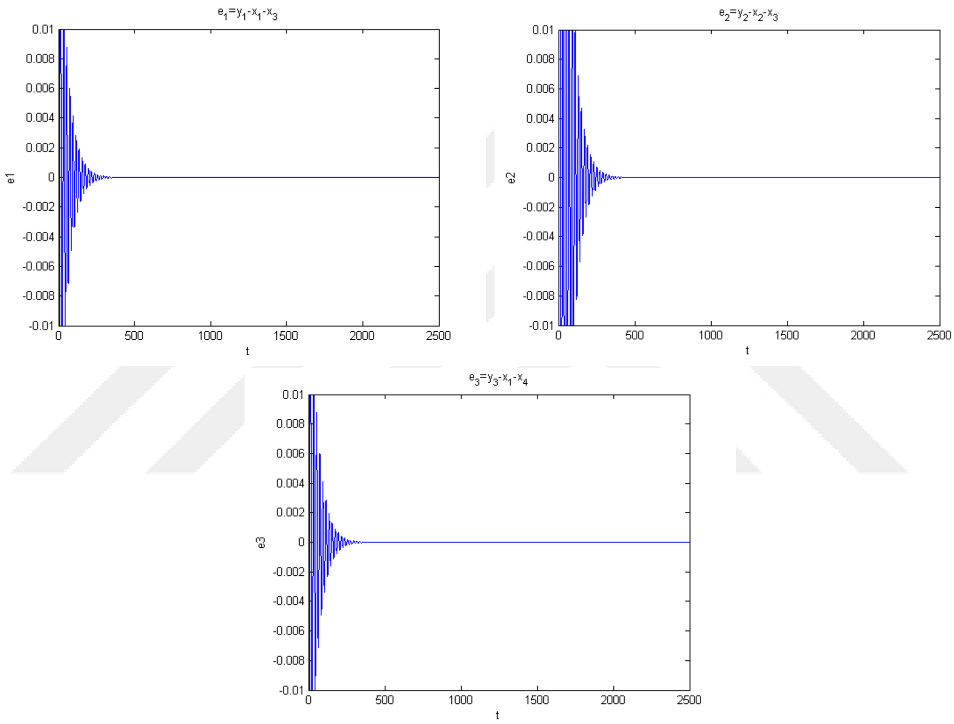


Figure 6.34 Error state (e_1, e_2, e_3)

6.3 Numerical Solutions of Coupled ADR Equations

In this section, we investigate the accuracy of the BDFS method governed by the nonlinear coupled Burgers equations with source functions. We consider the discrete approximation of $u_1(x,t)$ and $u_2(x,t)$ by $s_{1,h}$ and $s_{2,h}$. Here, the BDFS solutions are not presented only at the grid points but also at optional points in the solution domain. To measure the accuracy of the proposed scheme, the relative error $L_\infty(k)$ defined by

$$k \longrightarrow L_\infty(k) = \|\mathbb{W} - \mathbb{S}_k\|_\infty = \max_{0\le n\le 2N}\left(\max_{0\le i\le 2k} |\mathbb{W}(x_i', t_{2n}) - \mathbb{S}(x_i')|\right), \tag{6.24}$$

where the vector function $\mathbb{W}(\,.\,,t_{2n}) = \begin{bmatrix} u_1(\,.\,,t_n) \\ u_2(\,.\,,t_n) \end{bmatrix}$ is approximated by the vector function $\mathbb{S}(x_{2n},\,.) = \begin{bmatrix} \widehat{s}_{n,1h} \\ \widehat{s}_{n,2h} \end{bmatrix}$.

Example 13

We set the parameters $\lambda_2 = \lambda_4 = -2$ and $\gamma_1 = \gamma_2 = 1$ in equation (4.19) leads to

$$\begin{aligned} \frac{\partial u_1}{\partial t} - \lambda_1 \frac{\partial^2 u_1}{\partial x^2} - 2u_1 \frac{\partial u_1}{\partial x} + (u_1 u_2)_x &= f_2(x,t), \\ \frac{\partial u_2}{\partial t} - \lambda_3 \frac{\partial^2 u_2}{\partial x^2} - 2u_2 \frac{\partial u_2}{\partial x} + (u_1 u_2)_x &= f_3(x,t). \end{aligned} \tag{6.25}$$

The initial and boundary conditions are taken from the exact solution, where exact solution of equation (6.25) is given by [123]

$$\begin{cases} u_1(x,t) = e^{-t} sin(x) & x \in [-3,3], \ \ t > 0, \\ u_2(x,t) = e^{-t} sin(x). \end{cases} \tag{6.26}$$

Numerical solutions have been produced by taking time steps $\Delta t = 0.001$ for the values of the parameters $\lambda_1 = \lambda_3 = 1$, 0.05 and 0.005, respectively. In this problem, the source functions are free. In Table 6.15, absolute errors for the computation are calculated and compared with the literature [140, 204]. From the tabulated results, it can be noted that, the BDFS methods have been seen to be accurate in comparison with the exact solution and the available literature. The absolute errors are documented in Table 6.16 for small values of the viscosity. Here, it is concluded that the presented scheme appears very satisfactory for low viscosity, while it is not the case in the corresponding literature. Figure 6.35 shows the numerical and exact solutions of $u_1(x,t)$ and $u_2(x,t)$ with $\Delta t = 0.01$ at $t = 1$. Here, it can be deduced that there is an excellent agreement between the numerical and exact solutions. Behavior of the solutions are presented in Figure 6.36. It can be seen that, the proposed scheme is in very good agreement with the exact one and exhibits physical characteristics of the problem correctly.

Table 6.15 Absolute errors at various time values for $u_1(x,t)$ with $\Delta t = 0.001$ in Example 13

Errors	t	BDFS	Ref. [140]	Ref. [204]
	0.1	$3.45E-08$	$5.30E-05$	$2.05E-06$
L_2	0.5	$4.09E-07$	$2.67E-04$	$1.02E-05$
	1	$1.43E-07$	$5.38E-04$	$2.04E-05$
	0.1	$2.15E-08$	$4.08E-05$	$1.86E-06$
L_∞	0.5	$4.00E-07$	$1.62E-04$	$6.22E-06$
	1	$1.20E-07$	$1.98E-04$	$7.56E-06$

Table 6.16 Absolute errors for $u_1(x,t)$ at $\Delta t = 0.001$ in Example 13

Errors	t	$\lambda_1 = \lambda_3 = 0.05$	$\lambda_1 = \lambda_3 = 0.005$
	0.1	$5.22E-06$	$2.02E-03$
L_2	0.5	$1.01E-04$	$4.61E-02$
	0.9	$1.13E-03$	$7.19E-02$
	0.1	$5.21E-05$	$4.91E-03$
L_∞	0.5	$1.25E-03$	$9.85E-02$
	0.9	$2.01E-03$	$9.71E-02$

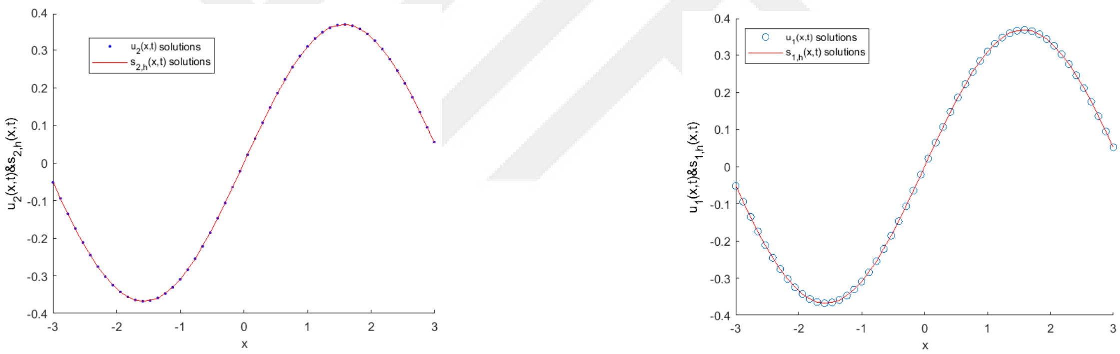


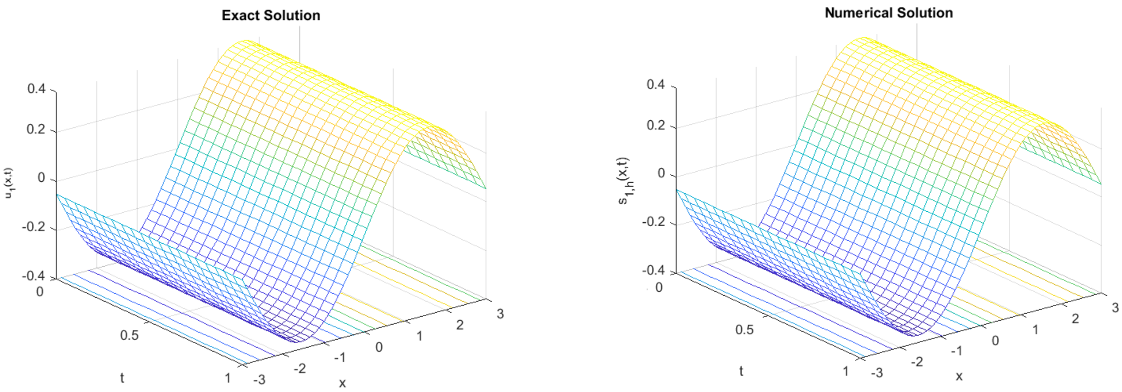
Figure 6.35 Computed solutions for (a) $u_1(x,t)$ and (b) $u_2(x,t)$ of Example 13 at $t = 1$ with $\lambda_1 = \lambda_3 = 0.05$

Example 14

Consideration of the parameters $\lambda_1 = \lambda_3 = 1$ and $\lambda_2 = \lambda_4 = 2$, in equation (4.19) gives

$$\begin{aligned} \frac{\partial u_1}{\partial t} - \frac{\partial^2 u_1}{\partial x^2} + 2u_1\frac{\partial u_1}{\partial x} + \gamma_1(u_1u_2)_x &= f_2(x,t), \\ \frac{\partial u_2}{\partial t} - \frac{\partial^2 u_2}{\partial x^2} + 2u_2\frac{\partial u_2}{\partial x} + \gamma_2(u_1u_2)_x &= f_3(x,t). \end{aligned} \tag{6.27}$$

The exact solution is given by



(a) (b)

Figure 6.36 Computed solutions of Example 13 for $u_1(x,t)$ with $\Delta t = 0.001$ for $\lambda_1 = \lambda_3 = 0.05$

$$\begin{cases} u_1(x,t) &= a_1\left(1-2a_2\left(\dfrac{2\gamma_1-1}{4\gamma_1\gamma_2-1}\right)\right)tanh\left(a_2(x-2a_2t)\right), \\ u_2(x,t) &= a_1\left(\left(\dfrac{2\gamma_2-1}{2\gamma_1-1}\right)-2a_2\left(\dfrac{2\gamma_1-1}{4\gamma_1\gamma_2-1}\right)\right)tanh\left(a_2(x-2a_2t)\right), \end{cases} \tag{6.28}$$

where $a_2 = a_1\dfrac{4\gamma_1\gamma_2-1}{4\gamma_1-2}$ and a_1, γ_1, γ_2 are arbitrary constants. The initial and boundary conditions are taken from the exact solution. The source functions are neglected in this example. Numerical solutions of this problem are obtained for the domain $(x,t) \in [-10,10]$ with various values of γ_1 and γ_2. The BDFS solutions have been computed and compared with the exact solution in Table 6.17 at different time levels where $t > 0$. It can be seen that, the BDFS is more accurate than those available in the literature [140, 204]. In Figure 6.37, we present the numerical and exact solutions of $u_1(x,t)$. Here, it can be noted that the BDFS results show excellent agreement with the exact solution. We have depicted the behavior of the solution $u_1(x,t)$ in Figure 6.38. In conclusion, we can see that the theoretical results on the convergence are confirmed by the numerical counterparts.

Table 6.17 The errors at various time values for $u_1(x,t)$ at $\Delta t = 1E-03$

Errors	t	γ_1	γ_2	BDFS	Ref. [140]	Ref. [204]
L_2	0.5	1E-01	3E-01	$4.879E-07$	$6.631E-04$	$6.736E-04$
		3E-01	3E-02	$6.060E-07$	$6.903E-04$	$7.326E-04$
L_∞	1	1E-01	3E-01	$5.455E-08$	$8.151E-05$	$8.258E-05$
		3E-01	3E-02	$9.142E-08$	$8.541E-05$	$9.182E-05$

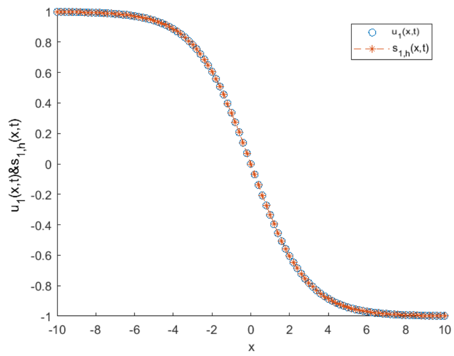


Figure 6.37 Computed solutions of Example 14 for $u_1(x,t)$ with $t = 0.5$ and $\gamma_1 = \gamma_2 = 0.01$

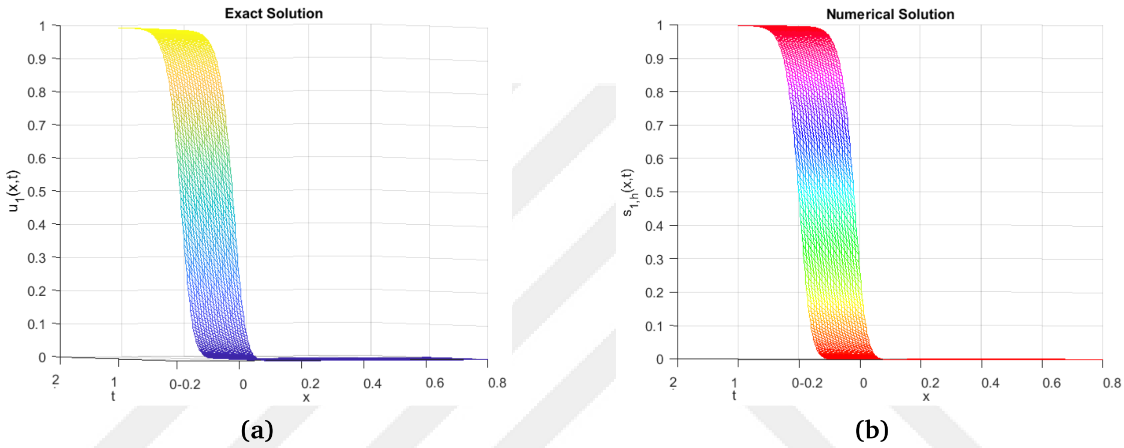


Figure 6.38 Computed solutions of Example 14 for $u_1(x,t)$ with $\Delta t = 0.01$ for $\gamma_1 = \gamma_2 = 0.01$

Example 15

Here, we consider the nonlinear coupled Burgers equation (4.19) with free source functions given by

$$\begin{aligned} \frac{\partial u_1}{\partial t} - \lambda_1 \frac{\partial^2 u_1}{\partial x^2} + 2u_1 \frac{\partial u_1}{\partial x} + \gamma_1 (u_1 u_2)_x &= f_2(x,t), \\ \frac{\partial u_2}{\partial t} - \lambda_3 \frac{\partial^2 u_2}{\partial x^2} + 2u_2 \frac{\partial u_2}{\partial x} + \gamma_2 (u_1 u_2)_x &= f_3(x,t). \end{aligned} \tag{6.29}$$

The exact solution of equation (6.29) is given by [204]

$$\begin{cases} u_1(x,t) = e^{-t} sin(x) & x \in [1,4], t > 0, \\ u_2(x,t) = cos(t)cos(x). \end{cases} \tag{6.30}$$

This problem is solved for various selections of λ_1 , λ_3, γ_1 and γ_2 at different time levels. Absolute errors for $u_2(x,t)$ have been calculated and compared with some previous works in Table 6.18. Here, it can be observed that, the errors obtained by the BDFS scheme are quite small and furthermore, better than most of available methods in the literature. In Table 6.19, the accuracy of the proposed schemes is examined by computing the errors for small values of the viscosity. It can be deduced that, the BDFS results are in good agreement with the exact solutions. Behavior of the BDFS solutions for $u_1(x,t)$ and $u_2(x,t)$ and exact solutions are exhibited in Figure 6.39. The results are illustrated in a qualitative way in Figure 6.40. It reveals that the BDFS solutions are highly accurate and very close to the exact solutions.

Table 6.18 The errors at various times for $u_2(x,t)$ with $\Delta t = 1E-03$, $\lambda_1 = \lambda_3 = 2$

Errors	t	γ_1	γ_2	BDFS	Ref. [140]	Ref. [204]
L_2	0.5	1E-01	3E-01	$1.300E-06$	$4.890E-04$	$9.057E-04$
		3E-01	3E-02	$4.356E-06$	$7.056E-04$	$1.591E-04$
L_∞	1	1E-01	3E-01	$9.005E-07$	$4.113E-05$	$4.770E-05$
		3E-01	3E-02	$9.789E-07$	$9.779E-05$	$3.617E-05$

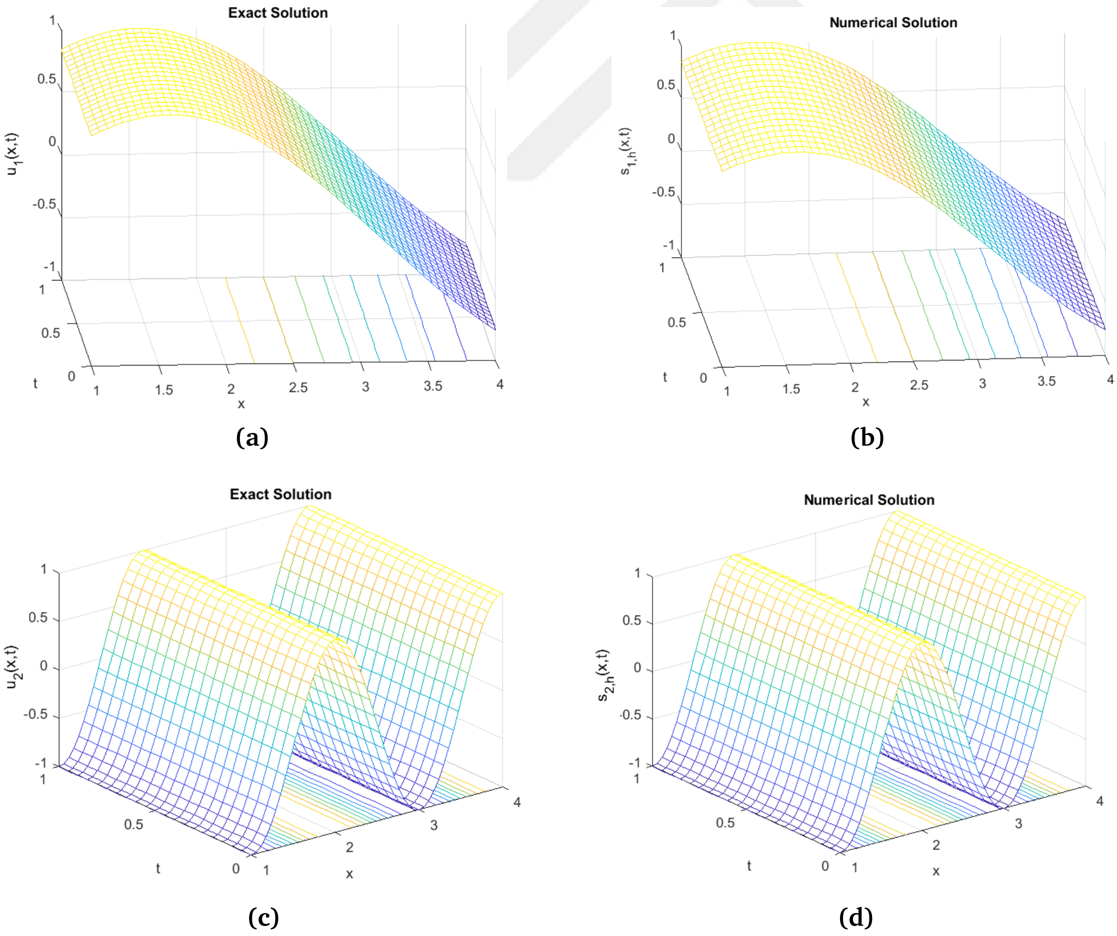
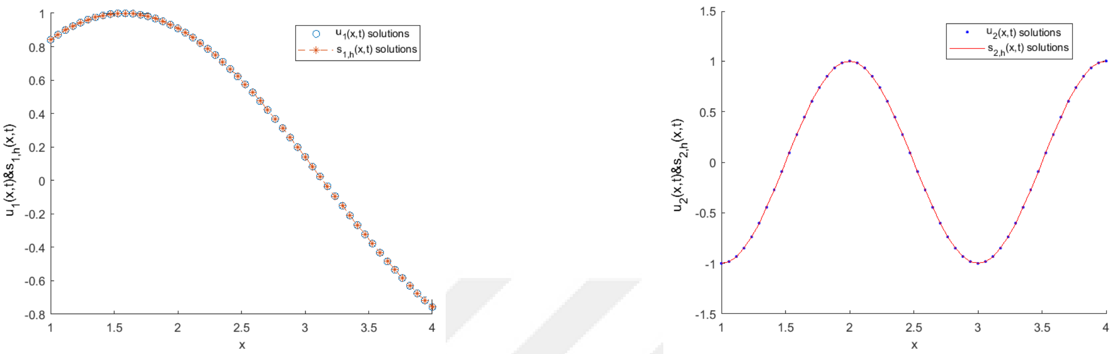


Figure 6.39 Computed solutions of Example 15 with $u_1(x,t)$ and $u_2(x,t)$ for $\lambda_1 = \lambda_3 = 0.001$, $\Delta t = 0.001$

Table 6.19 The errors at various time values with $\Delta t = 0.005$, $\lambda_1 = \lambda_3 = 0.001$

Errors	t	BDFS
L_2	0.5	$2.23E-04$
	0.9	$1.03E-04$
L_∞	0.5	$3.46E-03$
	0.9	$1.99E-02$



(a) **(b)**

Figure 6.40 Comparison between the BDFS and exact solutions (a) $u_1(x,t)$ and (b) $u_2(x,t)$ for Example 15 with $t = 0.7$, $\lambda_1 = \lambda_3 = 0.001$

Example 16

Consider the nonlinear coupled Burgers equation with source functions, namely

$$\begin{aligned} \frac{\partial u_1}{\partial t} - \lambda_1 \frac{\partial^2 u_1}{\partial x^2} + 2u_1 \frac{\partial u_1}{\partial x} + \gamma_1 (u_1 u_2)_x &= f_2(x,t), \\ \frac{\partial u_2}{\partial t} - \lambda_3 \frac{\partial^2 u_2}{\partial x^2} + 2u_2 \frac{\partial u_2}{\partial x} + \gamma_2 (u_1 u_2)_x &= f_3(x,t). \end{aligned} \tag{6.31}$$

in the domain $[0,1] \times [t_0, T]$ with the boundary and initial conditions, given by

$$\begin{cases} u_1(x,0) = e^x, & u_2(x,0) = x^2 + 1, \\ u_1(0,t) = 1 + t^2, & u_2(0,t) = e^t, \\ u_1(1,t) = e + t^2 & u_2(1,t) = 1 + e^t. \end{cases} \tag{6.32}$$

The source functions are taken to be

$$f_2(x,t) = 2t - \lambda_1 e^x + 2e^x(e^x + t^2) + 2\gamma_1 x e^x,$$

$$f_3(x,t) = e^t - 2\lambda_3 + 4x(x^2 + e^t) + 2\gamma_2 x e^x,$$

such that the exact solutions are

$$u_1(x,t) = e^x + t^2, \qquad \text{and} \qquad u_2(x,t) = x^2 + t^2. \tag{6.33}$$

The comparison between the numerical and exact solutions for various time values are shown in Figure 6.41. In these figures, we can see that, the numerical and exact solutions are in good agreement. Behaviour of the problem has been explained in a comparative way in Figure 6.42 for $u_1(x,t)$. The numerical solutions are seen to be good agreement with the exact ones. In Table 6.20, accuracy of the proposed schemes is examined by computing the errors for $\Delta t = 0.001$ with $\lambda_1 = \lambda_3 = 1$. The errors are presented in Tables 6.21-6.22 for $u_2(x,t)$ for different time values with $\Delta t = 1E-4$, $\lambda_1 = \lambda_3 = 0.1$. It can be seen that, the theoretical convergence and the computational errors are found to be in good agreement. In conclusion, the BDFS is seen to be a very good choice for solving the nonlinear coupled Burgers equations with source functions.

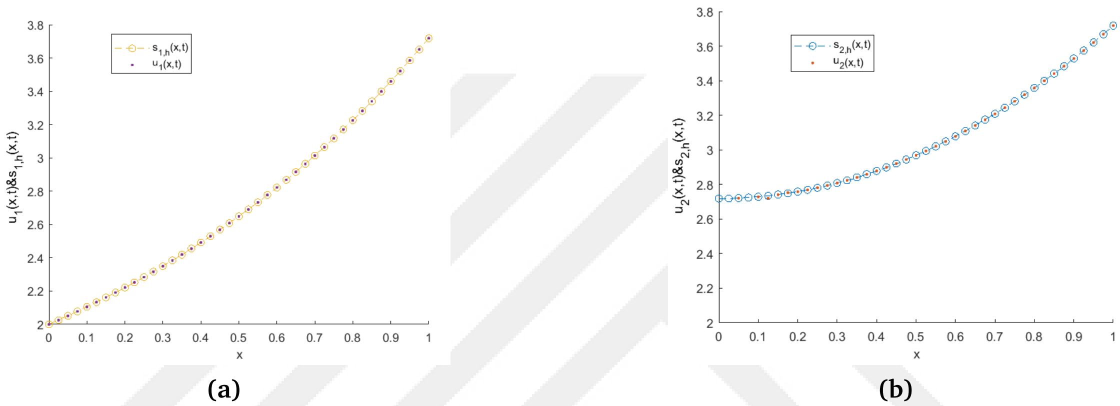


Figure 6.41 Computed solutions of Example 16 for (a) $u_1(x,t)$ and (b) $u_2(x,t)$ with $t = 1.5$, $\lambda_1 = \lambda_3 = 1$

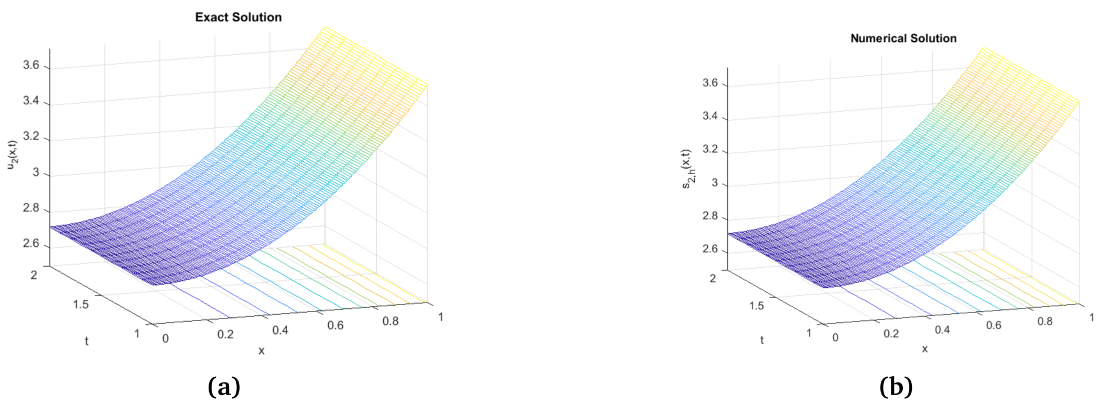


Figure 6.42 Computed solutions of Example 16 for $u_2(x,t)$ with $t \geq 1$, $\Delta t = 0.001$ over $[1,2]$

Table 6.20 Absolute errors of Example 16 for $u_1(x,t)$ at various time values with $\Delta t = 0.001$, $\lambda_1 = \lambda_3 = 1$

Errors	t	BDFS
L_2	1.1	$4.01E-05$
	1.9	$3.03E-05$
L_∞	1.1	$3.06E-04$
	1.9	$3.99E-03$

Table 6.21 Absolute errors for $u_2(x,t)$ at $\Delta t = 1E-4$ and $\lambda_1 = \lambda_3 = 0.1$ over $[1,2]$ in Example 16

Errors	t	BDFS
L_2	1.1	$6.01E-05$
	1.9	$7.07E-03$
L_∞	1.1	$8.01E-04$
	1.9	$5.99E-03$

Table 6.22 Absolute errors for $u_2(x,t)$ at $\Delta t = 1E-4$, $\lambda_1 = \lambda_3 = 0.001$ over $[1,10]$ in Example 16

Errors	t	BDFS
L_2	1.5	$2.41E-03$
	9	$4.48E-02$
L_∞	1.5	$6.61E-02$
	9	$2.69E-02$

6.4 Simulation Results of Synchronization of the Nonlinear ADR Processes

In this section, a numerical example illustrating the accuracy of the present approach is given. The solution domain $[a,b]$ is disretized using the equally spaced points. In order to explain the synchronization of the driver and the response (4.74)-(4.76) equations, the error norm is given by

$$\lim_{t\to\infty} \|L_\infty\| = \lim_{t\to\infty} \|\mathbb{V} - \mathbb{S}_k\| = \lim_{t\to\infty} \left(\max_{0\le n\le 2N} \left(\max_{0\le i\le 2q} |\mathbb{V}(x_i', t_{2n}) - \mathbb{S}(x_i')| \right) \right). \quad (6.34)$$

The vector function $\mathbb{V}(.,t_{2n}) = \begin{bmatrix} u_1(.,t_n) \\ u_2(.,t_n) \end{bmatrix}$ is approximated by the vector function $\mathbb{S}(x_{2n},.) = \begin{bmatrix} \widehat{s}_{n,1h} \\ \widehat{s}_{n,2h} \end{bmatrix}$.

Example 17

In this example, the parameters are set as: $\lambda_1 = \lambda_3 = 0.00001$ and $\lambda_2 = \lambda_4 = 1$. The initial and boundary conditions are obtained from the exact solution. The exact solution of equation (4.74) are represented by the velocity u_1 and the temperature u_2 as given by

$$u_1(x,t) = e^{-t}(x - \frac{x^2}{2}),$$
$$u_2(x,t) = x(1-x).$$

The source functions are taken to be

$$f_1(x,t) = e^{-t}\left(-x + \frac{x^2}{2} + \lambda_1 + \lambda_2\left(x - \frac{x^2}{2}(1-x)\right) - k_2(x(1-x))\right),$$

$$f_2(x,t) = 2\lambda_4 e^{-t}\left(x - \frac{x^2}{2}\right).$$

The computational domain for this problem is $[0,1]\times[1,T]$. Various cases for $\lim_{t\to\infty} \|.\|$ are given at various values of k_2 and t in Table 6.23. Full synchronization of the proposed coupled model has been observed for $k_2 \geq 0.24$. As shown in the simulation, the synchronizational behaviour in the fluid that occurs with decreasing viscosity constant. We can see that the chaotic behaviour and instability situations of the nonlinear coupled equations with forcing function in Figure 6.43. As can be seen, the synchronization is observed in Figure 6.44 when k_2 becomes larger with small value of the viscosity coefficient. The results are also confirmed by the simulations for a nonlinear coupling ADR model.

Table 6.23 Absolute errors for $u_2(x,t)$ at various time and k_2 values for $\Delta t = 0.001$ and $\lambda_1 = \lambda_3 = 0.00001$

$\lim_{t\to\infty} \|\|.\|\|$	t	k_2	BDFS
L_∞	1	0.01	$4.01E-05$
	2		$1.04E-01$
	3.5		$3.02E-03$
	5		$4.44E-02$
	7		$4.01E-03$
	10		$6.98E-02$
$\lim_{t\to\infty} \|\|.\|\|$	t	k_2	BDFS
L_∞	1	0.24	$4.01E-05$
	2		$3.03E-09$
	3.5		$2.22E-08$
	5		$1.66E-08$
	7		$6.09E-07$
	10		$3.77E-09$

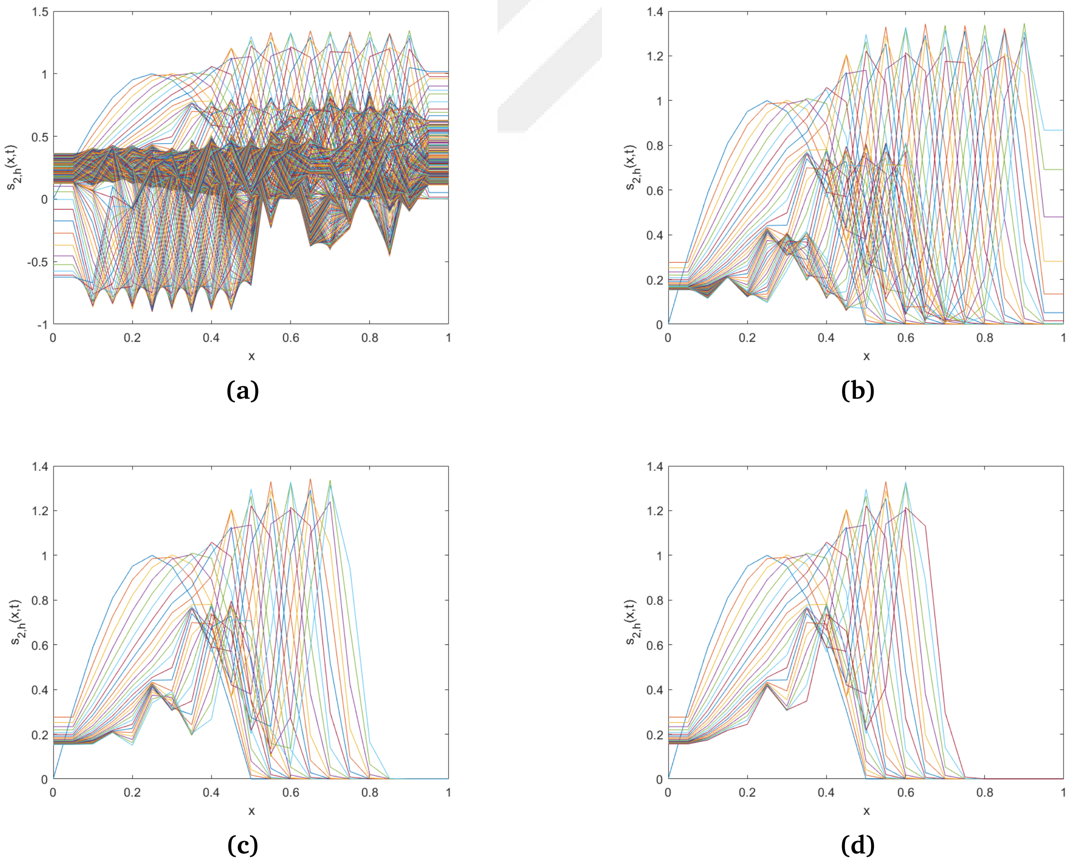


Figure 6.43 Chaotic attractors for the nonlinear coupling in the temperature field at $k_2 = 0,\ 0.0001,\ 0.005,\ 0.01$

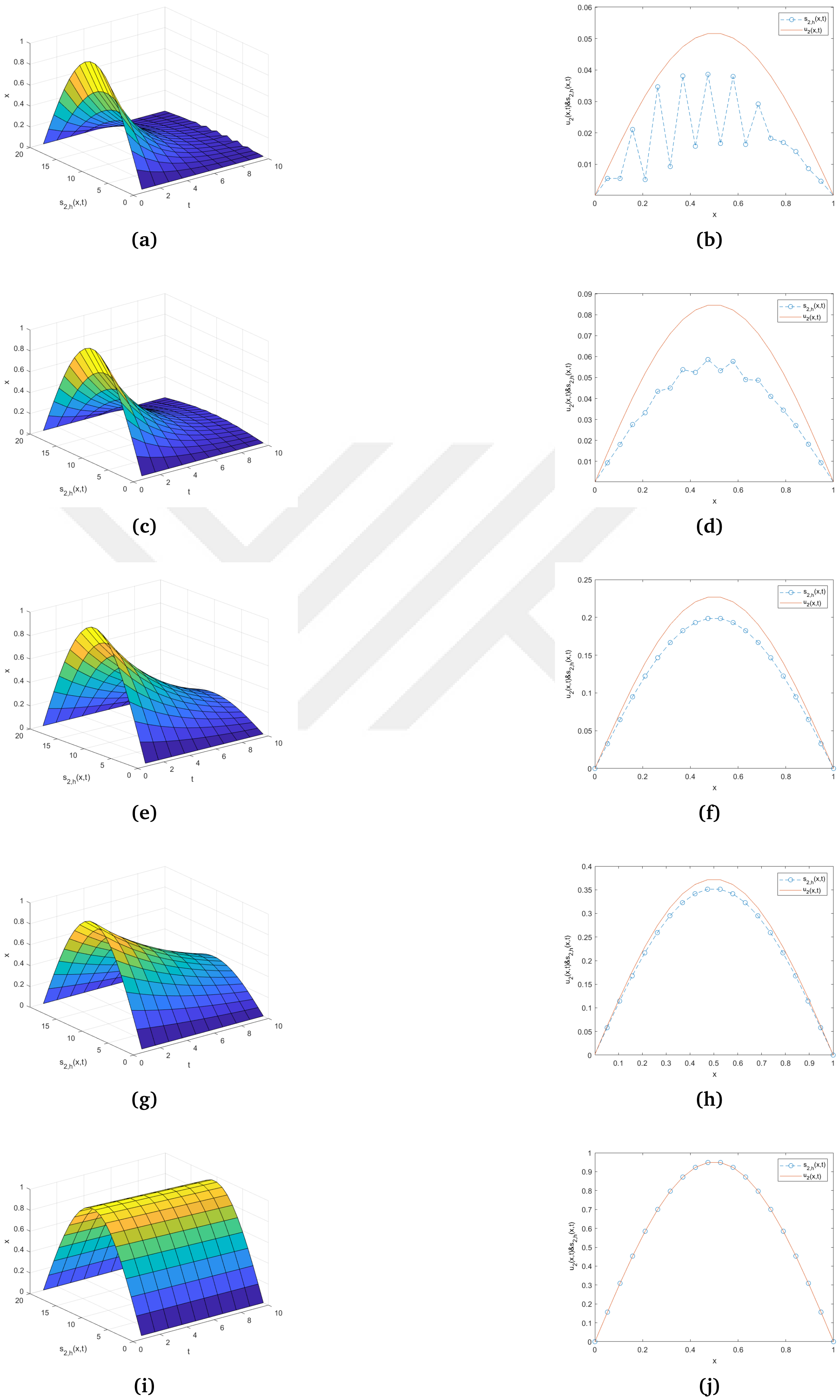


Figure 6.44 Nonlinear coupling with the driver for $k_2 = 0.001, 0.01, 0.1, 0.16, 0.24$

7

RESULTS AND DISCUSSION

The motivation for this research is to investigate the synchronization, stabilization, convergence in capturing numerical solutions of the nonlinear ADR processes and some chaotic problems. With keeping real features of nature, reducing the computational cost and also without requirement of extra storage space, the BDFS and SSPRK54S methods have been proposed to numerically capture the behaviour of physical environment represented by the nonlinear ADR equations with forcing terms. It should be pointed out that the schemes lead to an ordinary differential equation without using any transformation for the given model. Since the linearization of the systems loses their real features, the proposed schemes have been shown to be effectively applicable to such problems in terms of numerical and theoretical results. The produced results revealed that the proposed approach is a rapidly convergent and a reliable alternative in solving the nonlinear ADR equation with also source functions. Notice that the current methods have been figured out to be more effective than the literature for the problem of interest. The computed results have revealed that the BDFS method is more accurate and computationally more economical in comparison with the SPRK54S method. The BDFS method has also been realized to be more reliable than the SSPRK54S, even a very important alternative for the research society, in analysing the problem by conserving the physical properties of nature. The results showed that the BDFS scheme is relatively free of choice of the physical parameters. Yet, we have explored the utility of a combined scheme based on modified cubic B-spline basis functions in space with the SSPRK54 scheme in time for solving the ADR equation. The results have been computed without using any linearization or transformation. The produced results show that the proposed scheme is efficient and reliable for solving these models for quite small values of the viscosity constant. The instabilities observed for the interaction between reaction, convection and diffusion mechanisms, since the ADR equations are highly nonlinear models. Many chaotic behaviors characterized by instability and limited predictability in time. Thus, the relative importance of chaotic advection and diffusion in nonlinear ADR processes can be connected with the GS dynamics of nonlinear models.

The synchronization problems of coupled chaotic identical and nonidentical models has also been studied. Next aim of this thesis is to study coupled systems that do not have mutual feedback but they were organized as a drive system and a response system through some communication channels between them. We have also discussed the asymptotic stability by considering various coupling strengths and a new response system proposed in each case of the present methods are given as: First, we have proposed that this phenomenon of chaotic synchronism may construct a response system via the Lyapunov stability to carry out the generalized synchronization with the derive system for a given smooth invertible function. We have then considered a new hypothesis from the nonlinear part of the response system under some sufficient conditions which showed that the global generalized synchronization between chaotic systems. The methods may be implemented directly in any numerical simulations for synchronization of chaotic systems with different dimensions and have the fast synchronization speed. Numerical results have also illustrated the effectiveness of the proposed approaches. Thus, synchronization of two models generally means that one model somehow follows the motion of another. As a result, let us recall that synchronization is observed that even chaotic problems could synchronize when they are coupled. In this work, some numerical analysis of the nonlinear physical phenomena without losing their natural properties and by reducing the computational difficulties on capturing numerical behavior of nature governed by the nonlinear coupled ADR equations have been done. With keeping real features of nature and also reducing the computational cost and without requirement extra storage space, the BDFS method has been proposed to numerically capture the behaviour of physical environment represented by the nonlinear coupled ADR equation with forcing terms. Under these natural circumstances, the proposed scheme has been shown to be effectively applicable to such problems. The current results revealed that the proposed approach is a rapidly convergent and a reliable alternative in solving the nonlinear coupled ADR equation with source functions. As a further contribution of this thesis, the dynamical and GS of two dependent chaotic nonlinear ADR processes with forcing terms, which unidirectionally coupled in the driver-response configuration, by combining the BDFS scheme with the Lyapunov method, the GS has been studied for designing a control function of the coupled nonlinear ADR equations. Since the nonlinear coupled ADR model cannot synchronize itself, some control functions should be designed and applied to synchronize such problems. In the investigation of the real-world processes without losing their natural properties, this article has addressed the GS behaviour defined by the nonlinear coupled ADR equations, by combing the BDFS and Lyapunov methods. In the current method, it has been importantly concluded that the nonlinear coupled ADR model can be synchronized under the consideration of a proposed control function. As a result, from the produced numerical simulation, the temperature drives the velocity field and the velocity field provides the advection term at very small viscosity value. Further work addresses in detail the improvement of a combination of the classical 2D cubic B-splines and natural splines in space. This scheme appears to be an interesting approach for the approximations of 2D nonlinear ADR problems. Since, without any linearization, the given problem through the BDFS scheme is converted to a system of nonlinear and linear equations. To conclude, we have observed the following results:

- Without requiring differentiability of the initial and boundary functions, we have obtained the numerical solution of the nonlinear ADR problems,
- By using equally spaced points, the produced solutions have not only been obtained at the grid points but also at optional points for various choices of grid sizes and time steps.
- The computed results have been computed without using either any linearization or transforming the model,
- The BDF scheme has been implemented using the Newton and Thomas algorithms to solve the nonlinear and linear parts of the resulting system at each iteration respectively,
- With keeping real features of nature, all of the current schemes illustrated the behaviour of shock behaviours,
- The results showed that the proposed schemes have relatively been free of choice of the physical parameters,
- The current methods have been figured out to be more effective than the literature for the problem of interest,
- The designed controllers enabled the state variables of the response system to globally synchronize the state variables of the driver system in current problems,
- Some control functions have been designed and applied to synchronize such coupled ADR chaotic problems,
- It has been concluded that the nonlinear coupled ADR chaotic model can be synchronized under the consideration of a proposed control function at a very small viscosity value.

Open problems and some recommendations are presented as follows:

» By considering the proposed schemes, the numerical solution of the nonlinear ADR problems involving Neumann or Robin boundary conditions can be investigated.

» The chaotic network and the phenomena of synchronization in the network of the nonlinear ADR equations can be studied.

» Synchronization of the 2D nonlinear ADR problems can be investigated.

Bibliography

[1] A. A. Nascimento, F. P. Mariano, A. Silveria-Neto, E. L. M. Padilla. A comparison of Fourier pseudospectral method and finite volume method used to solve the Burgers equation. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 36:737–742, 2014.

[2] A. A. Soliman. New numericl technique for Burgers equation based on similarity reductions. *International Conference on Computational Fluid Dyanamics. Beinjing, China*, 17:559–566, 2000.

[3] A. A. Soliman. A Galerkin solution for Burgers equation using cubic B-spline finite elements. *Abstract and Applied Analysis*, 2012:1–15, 2012.

[4] A. Asaithambi. Numerical solution of the Burgers equation by automatic differentiation. *Applied Mathematics and Computation*, 216:2700–2708, 2010.

[5] A. Dogan, A. Galerkin. Finite element approach to Burgers equation. *Applied Mathematics and Computation*, 157:331–346, 2004.

[6] A. G. Bratsos. A fourth order improved numerical scheme for the generalized Burgers-Huxley equation. *Journal of Computational and Applied Mathematics*, 3:152–158, 2011.

[7] A. G. Bratsos, A. Q. M. Khaliq. An exponential time differencing method of lines for Burgers-Fisher and coupled Burgers equations. *Journal of Computational and Applied Mathematics*, 356:182–197, 2019.

[8] A. Golbabai, M. Javidi. A spectral domain decomposition approach for the generalized Burgers-Fisher equation. *Chaos, Solitons and Fractals*, 39:385–392, 2009.

[9] A. H. A. Ali, L. R. T. Gardner, G. A. Gardner. A Galerkin approach to the solution of Burgers equation. *UCNW Mathematices*, 4:90–94, 1990.

[10] A. J. M. Jawad, M. D. Petkovic, A. Biswas. Solutions of Burgers equations and perturbed Burgers equation. *Applied Mathematics and Computation*, 216:3370–3377, 2010.

[11] A. K. Khalifa, K. I. Noor, M. A. Noor. Some numerical methods for solving Burgers equation. *International Journal of Physical Sciences*, 6:1702–1710, 2011.

[12] A. Khan. Parametric cubic spline solution of two point boundary value problems. *Applied Mathematics and Computation*, 175:175–182, 2004.

[13] A. Khan, P. Singh. Chaos synchronization in Lorenz system. *Applied Mathematics*, 6:1864–1872, 2015.

[14] A. Kolmogrov, I. Petrovski, N. A. Piskunov. *A study of the equation of diffusion with increase in the equation of matter and its application to a biological problem*. Bjul, Moskov Skogogos, 1 edition, 1937.

[15] A. Korkmaz, A. M. Aksoy, I. Dag. Quartic B-spline differential quadrature method. *International Journal of Nonlinear*, 11:403–411, 2011.

[16] A. Korkmaz, I. Dag. Polynomial based differential quadrature method for numerical solution of nonlinear Burgers equation. *Journal of Franklin Institute*, 348:2863–2875, 2011.

[17] A. Korkmaz, I. Dag. Cubic B-spline differential quadrature methods and stability for Burgers equation. *International Journal of Computer Aided Engineering*, 30:320–344, 2013.

[18] A. Korkmaz, İ. Dag. Shock wave simulations using sinc differential quadrature method. *Engineering Computations*, 28:654–674, 2011.

[19] A. L. Cauchy. Exercises de mathematique. *Mémoires de l'Académie Royale des Sciences (Paris)*, pages 183–, 1828.

[20] A. Lamnii, H. Mraoui. Spline collocation method for solving boundary value problems. *International Journal of Mathematical Modelling and Computations*, 3:11–23, 2013.

[21] A. M. Al-Rozbayani, M. O. Al-Amr. Discrete a domian decomposition method for solving Burgers-Huxley equation. *International Journal of Contemporary Mathematical Sciences*, 8:623–631, 2013.

[22] A. Molabahramia, F. Khani. The homotopy analysis method to solve the Burgers-Huxley equation. *Nonlinear Analysis Real World Applications*, 10:589–600, 2009.

[23] A. Ouannas, X. Wang, V. T. Pham, G. Grassi, V. V. Huynh. Synchronization results for a class of fractional-order spatio temporal partial differential systems based on fractional Lyapunov approach. *Boundary Value Problems*, 74:1–12, 2019.

[24] A. Pogromsky, G. Santoboni, H. Nijmeijer. Partial synchronization from symetry toward stability. *Physica D.*, 172:65–87, 2002.

[25] A. R. Bahadir. A fully implicit finite-difference scheme for two-dimensional Burgers equations. *Applied Mathematics and Computation*, 137:131–137, 2003.

[26] A. R. Mitchell, J. C. Bruch Jr. A numerical study of chaos in a reaction-diffusion equation. *Numerical Methods for Partial Differential Euations*, 1:13–23, 2018.

[27] A. Rashid, A. I. Ismail. A fourier pseudo spectral method for solving coupled viscous Burgers equations. *Computational Methods in Applied Mathematics*, 9:4–20, 2009.

[28] A. Sahin, I. Dag, B. Saka. A B-spline algorithm for the numerical solution of Fisher's equation. *Kybernetes*, 37:326–342, 2008.

[29] A. Sahin, O. Ozmen. Usage of higher order B-splines in numerical solution of Fisher's equation. *International Journal of Nonlinear Science*, 17:241–253, 2014.

[30] A. Tripathi, R. C. Mittal. Numerical solutions of generalized Burgers-Fisher and generalized Burgers-Huxley equations using collocation of cubic B-Splines. *International Journal of Computer Mathematics*, 92:1053–1077, 2014.

[31] A. Tripathi, R. C. Mittal. Numerical solutions of two-dimensional Burgers equations using modified Bi-cubic B-spline finite elements. *Engineering Computations*, 32:1–15, 2015.

[32] A. V. Prasolov. A direct Lyapunov method for delay differential equations. *International Journal of pure and applied mathematics*, 964:823–834, 2019.

[33] B. Ay, I. Dag, M. Z. Gorgulu. Trigonometric quadratic B-spline subdomain Galerkin algorithm for the Burgers equation. *Open Physics*, 13:400–406, 2015.

[34] B. Batiha, M. S. M. Noorani, I. Hashim. Application of variational iteration method to the generalized Burger-Huxley equation. *Chaos Solitons* & *Fractals*, 36:660–663, 2008.

[35] B. Inan, A. R. Bahadir. Numerical solutions of the generalized Burger-Huxley equation by implicit exponential finite difference method. *Journal of Applied Mathematics*, 11:1–17, 2015.

[36] B. L. Lohar, P. C. Jain. Variable mesh cubic spline technique for N-wave solution of Burgers equation. *Journal Computational Physices*, 39:433–442, 1981.

[37] B. M. Herbst, S. W. Schoombie, A. R. Mitchell. A moving Petrov-Galerkin method for transport equations. *International Journal for Numerical Methods in Engineering*, 18:1321–1336, 1982.

[38] B. Saka, I. Dag. Quartic B-spline collocation method to the numerical solutions of the Burgers equation. *Chaos Solitons Fractals*, 32:1125–1137, 2007.

[39] C. D. Boor. *Partial differential equations of parabolic type*. Prentice-Hall INC NEW Jersey, 1 edition, 1964.

[40] C. D. Boor. On uniform approximation by splines. *Journal of Approximation Theory*, 1:219–235, 1968.

[41] C. D. Boor. On calculating with B-splines. *Journal of Approximation Theory*, 6:50–62, 1972.

[42] C. D. Boor. *A practical guide to spline*. Mathematics of Computation, 2 edition, 1978.

[43] C. G. Zhu, R. H. Wang. Numerical solution of Burgers equation by cubic B-spline quasi interpolation. *Applied Mathematics and Computation*, 208:260–272, 2009.

[44] C. H. Huygens. *The pendulum clock - (English translation)*. Lowa State University Press - Ames, 2 edition, 1986.

[45] C. H. K. Volos, I. M. Kyprianidis, I. N. Stouboulos, J. M. Munoz-Pacheco, V. T. Pham. Synchronization of chaotic nonlinear circuits via a Memristor. *Journal of Engineering Science and Technology Review*, 8:44–51, 2015.

[46] C. J. Fletcher. A comparison of finite element and finite difference solutions of the one and two dimensional Burgers equation. *Journal of Computational Physics*, 51:51–159, 1983.

[47] C. Koroglu. Exact and nonstandard finite difference schemes for the generalized KdV Burgers equation. *Advances in Difference Equations*, 134:235–245, 2020.

[48] C. L. M. Henri Navier. Efficient numerical treatment of nonlinearities in the advection-diffusion reaction equations. *Mémoires de l'Académie Royale des Sciences (Paris)*, 6:389–416, 1823.

[49] C. Zhu, W.S. Kang. Numerical solution of Burgers-Fisher equation by cubic B-spline quasi-interpolation. *Applied Mathematics and Computation*, 216:2679–2686, 2010.

[50] D. Hrg. Synchronization of two Hindmarsh-Rose neurons with unidirectional coupling. *Neural Network*, 40:73–79, 2018.

[51] D. J. Fyfe. The use of cubic splines in the solutions to two-point boundary value problems. *Computer Journal*, 12:188–192, 1969.

[52] D. Kaya, S. M. El-Sayed. A numerical simulation and explicit solutions of the generalized Burger-Fisher equation. *Applied Mathematics and Computation*, 152:403–413, 2004.

[53] D. L. Young, C. M. Fan, S. P. Hu, S. N. Atluri. The Eulerian lagranging method of funcdamental solutions for two dimensional unsteady Burgers equation. *Engineering Analysis with Boundary Elements*, 32:395–412, 2008.

[54] D. Olmos, B. D. Shizgal. A pseudospectral method of solution of Fisher's equation. *Journal of Computational and Applied Mathematics*, 193:219–242, 2006.

[55] E. A. Al-Said. Cubic spline method for solving two-point boundary value problems. *Korean Journal Computational and Applied*, 5:669–680, 1998.

[56] E. H. Twizell, Y. Wang, W. G. Price. Chaos free numerical solutions of reaction-diffusion equations. *Proceedings of the Royal Society of London A*, 430:541–576, 1990.

[57] E. Hopf. The partial differential equation $U_t + UU_x = \lambda U_{xx}$. *Communications on Pure and Applied Mathematics*, 3:201–230, 1950.

[58] E. L. Miller. Predictor-corrector studies of Burgers model of turbulent flow. *Quarterly of Applied Mathematics*, 12:255–276, 1966.

[59] E. Mickens, A. B. Gumel. Construction and analysis of a non-standard finite difference scheme for the Burgers-Fisher equation. *Journal of Sound and Vibration*, 257:791–797, 2002.

[60] E. N. Aksan. Quadratic B-spline finite element method for numerical solution of the Burgers equation. *Applied Mathematics and Computation*, 174:884–896, 2006.

[61] E. N. Aksan. An application of cubic B-spline finite element method for the Burgers equation. *Thermal Science*, 22:195–202, 2018.

[62] E. N. Aksan, A. Ozdes. A numerical solution of Burgers equation. *Applied Mathematics and Computation*, 156:395–402, 2004.

[63] E. Ngondiep. An efficient three-level explicit time-split scheme for solving two-dimensional unsteady nonlinear coupled Burgers equations. *International Journal for Numerical Methods in Fluid*, 47:20–39, 2019.

[64] E. Ott, C. Grebogi, J. A. Yorke. Controlling chaos. *Physical Review Letters*, 64:1196–1199, 1990.

[65] E. R. Benton, G. W. Platzman. A table of solutions of the one dimensional Burger equations. *Quarterly of Applied Mathematics*, 30:195–212, 1972.

[66] E. Varoglu, W. D. L. Finn. Space-time finite elements incorporating characteristics for the Burgers equation. *International Journal for Numerical Methods in Engineering*, 16:171–184, 1980.

[67] E. Yanagida. Standing pulse solutions in reaction-diffusion systems with skew gradient structure. *Journal of Dynamics and Differential Equations*, 10:189–205, 2002.

[68] F. B. Majid, A. I. Ranasinghe. Solution of the Burgers equation using an implicit linearizing scheme. *Communications in Nonlinear Science and Numerical Simulation*, 14:1861–1867, 2009.

[69] F. Liu, Y. Wang, S. Li. Barycentric interpolation collocation method for solving the coupled viscous Burgers equations. *International Journal of Computer Mathematics*, 95:1–7, 2018.

[70] F. T. Nieuwstadt, J. A. Steketee. *Selected papers of J M Burgers*. Springer Science+Business Media, BV, Dordrecht, 1995.

[71] G. Adomian. An efficient numerical scheme for Burgers equation. *Computers Mathematics Applications*, 29:1–3, 1995.

[72] G. Arora, B. K. Singh. Numerical solution of Burgers equation with modified cubic B-spline differential quadrature method. *Communications on Applied Mathematics and Computation*, 224:166–177, 2019.

[73] G. Arora, V. Joshi. A computational approach using modified trigonometric cubic B-spline for numerical solution of Burgers equation in one and two dimensions. *Alexandria Engineering Journal*, 57:1087–1098, 2018.

[74] G. G. Stokes. Exercises de mathematique. *Transactions of the Cambridge Philosophical Society*, 8:305–1845, 1845.

[75] G. Gurarslan. Numerical modelling of linear and nonlinear diffusion equations by compact finite difference method. *Applied Mathematics and Computation*, 216:2472–2578, 2010.

[76] G. Gurarslan, H. Karahan, D. Alkaya, M. Sari, M. Yasar. Numerical solution of advection-diffusion equation using a sixth-order compact finite difference method. *Mathematical Problems in Engineering*, 2014:1–7, 2013.

[77] G. Hariharan, K. Kannan, K. R. Sharma. Haar wavelet method for solving Fisher's equation. *Applied Mathematics and Computation*, 211:284–292, 2009.

[78] G. J. Brett, L. Pratt, I. Rypina, P. Wang. Competition between chaotic advection and diffusion: stirring and mixing in a 3-D eddy model. *Nonlinear Processes in Geophysics*, 26:37–60, 2019.

[79] G. Kirlinger. On the convergence of backward differentiation formulas for stiff initial value problems. *BIT Numerical Mathematics*, 41:1039–1049, 2001.

[80] G. Santoboni, A. Pogromsky, H. Nijmeijer. Partial observer and partial synchronization. *Internat Journal Bifurcation and Chaos*, 13:453–458, 2015.

[81] G. Zhao, X. Yu, R. Zhang. The new numerical method for solving the system of two-dimensional Burgers equations. *Computers* & *Mathematics with Applications*, 62:3279–3291, 2011.

[82] H. B. Curry, I. J. Schoenberg. *On polya frequency functions IV: the fundamental spline functions and their limits*. Oxford University Press, London, 2nd edition edition, 1977.

[83] H. Bateman. Some recent researches on the motion of fluids. *Monthly Weather Review*, 43:163–170, 1915.

[84] H. Chen, P. Shi, C. C. Lim. Pinning impulsive synchronization for stochastic reaction-diffusion dynamical networks with delay. *Neural Network*, 106:281–293, 2018.

[85] H. Eltayeb, S. Meslouba, A. Kilicman. A note on a singular coupled Burgers equation and double Laplace transform method. *Journal of Nonlinear Sciences and Applications*, 11:635–643, 2019.

[86] H. Gao , R. Zhao. New exact solutions to the generalized Burgers-Huxley equation. *Applied Mathematics and Computation*, 217:1598–1603, 2010.

[87] H. M. Salih, N. M. Tawfiq, Z. R. Yahya. Solution of system of viscous Burgers equation via collocation method. *International Journal of Modern Mathematical Sciences*, 2:106–123, 2018.

[88] H. N. A. Ismail, A. A. A. Rabboh. A restrictive Pade approximation for the solution of the generalized Fisher and Burger-Fisher equation. *Applied Mathematics and Computation*, 154:203–210, 2004.

[89] H. N. A. Ismail, K. Raslam, A. A. Abd Rabboh. A domian decomposition method for Burgers–Huxley and Burgers-Fisher equations. *Applied Mathematics and Computation*, 159:291–301, 2004.

[90] H. Nguyen, J. Reynen. A space-time finite element approach to Burgers equation. *Numerical Methods for Nonlinear Problems*, 3:718–728, 1987.

[91] H. O. Peitgen, H. Jurgens, D. Saupe. *Chaos and fractals*. Springer-Verlag, 1 edition, 1992.

[92] H. S. Shukla, M. Tamsir, V. K. Srivastava, J. Kumar. Numerical solution of two dimensional coupled viscous Burger equation using modified cubic B-spline differential quadrature method . *AIP Advances*, 4:1–10, 2018.

[93] H. V. Chapani, V. H. Pradhan, M. N. Mehta. Numerical simulation of Burgers equation using Quadratic B-splines. *International Journal of Applied Mathematics and Mechanics*, 11:18–32, 2018.

[94] H. Y. Du, P. Shi, N. Lu. Function projective synchronization in complex dynamical networks with time delay via hybrid feedback control. *Nonlinear AnalReal*, 13:82–90, 2013.

[95] I. Celik. Haar wavelet method for solving generalized Burgers-Huxley equation. *Arab Journal of Mathematical Sciences*, 18:25–37, 2012.

[96] I. Christie, A.R. Mitchell. Up winding of high order Galerkin methods in conduction-convection problems. *International Journal for Numerical Methods in Engineering*, 12:1764–1771, 1978.

[97] I. Christie, D. F. Griffiths, A. R. Mitchell, J. M. Sanz-Serna. Product approximation for nonlinear problems in the finite element method. *Journal of Numerical Analysis*, 1:253–266, 1981.

[98] I. Dag, B. Saka, A. Boz. Quintic B-spline Galerkin method for numerical solutions of the Burgers equation. *Dynamical Systems and Applications*, 11:295–309, 2004.

[99] I. Dag, D. Irk, A. Sahin. B-Spline collocation methods for numerical solutions of the Burgers equation. *Mathematical Computational Engineering*, 5:521–538, 2005.

[100] I. Dag, D. Irk, B. Saka. A numerical solution of the Burgers equation using cubic B-splines. *Applied Mathematics and Computation*, 163:199–211, 2005.

[101] I. Dag, O. Ersoy. The exponential cubic B-spline algorithm for Fisher equation. *Chaos, Solitons and Fractals*, 86:101–106, 2016.

[102] I. E. Horia. Waves in fluids by James Light ill. *Bulletin Mathématiques de la Société des Sciences Mathématiques de Roumanie*, 24:200–216, 1980.

[103] I. Hashim, M. S. M. Noorani, B. Batiha. A note on the a domian decomposition method for the generalized Huxley equation. *Applied Mathematics and Computation*, 181:1439–1445, 2006.

[104] I. Hashim, M. S. M. Noorani, M. R. Said Al-Hadidi. Solving the generalized Burgers-Huxley equation using the a domian decomposition method. *Mathematical and Computer Modelling*, 43:1404–1411, 2016.

[105] I. J. Schoenberg. Contribution to the problem of approximation of equidistant data by analytic functions. *Quarterly of Applied Mathematics*, 4:45–99, 1946.

[106] I. J. Schoenberg. Spline functions and the problem of graduation. *Proceedings of the National Academy of Sciences*, 52:947–950, 1946.

[107] I. Kucuk, I. Sadek, Y. Yilmaz. Active control of a smart beam with time delay by Legendre wavelets. *Applied Mathematics and Computation*, 218:8968–8977, 2012.

[108] I. R. Epstein. Nonlinear chemical dynamics: oscillations, patterns and chaos. *Journal of Physical Chemistry*, 100:13132–13147, 1996.

[109] I. Wasim, M. Abbas, M. Amin. Hybrid B-spline collocation method for solving the generalized Burgers-Fisher and Burgers-Huxley equations. *Mathematical Problems in Engineering*, 2018:1–18, 2018.

[110] J. Biazar, F. Muhammadi. Application of differential transform method to the generalized Burgers-Huxley equation. *Applications and Applied Mathematics, An International Journal*, 5:1726–1740, 2010.

[111] J. Caldwell. Application of cubic splines to the nonlinear Burgers equation. *Numerical Methods for Nonlinear Problems*, 3:253–261, 1987.

[112] J. Canosa. Diffusion in nonlinear multiplicative media. *Journal of Mathematical Physics*, 10:1862–1868, 1968.

[113] J. Canosa. On a nonlinear diffusion equation describing population growth. *Journal of Mathematical Physics*, 17:307–313, 1973.

[114] J. D. Cole. On a quasi-linear parabolic equations occurring in aerodynamics. *Quarterly of Applied Mathematics*, 9:225–236, 1951.

[115] J. D. Flether. Generating exact solutions of the two dimensional Burgers equation. *International Journal for Numerical Methods in Fluids*, 3:213–216, 1983.

[116] J. D. Logan. *An introduction to nonlinear partial differential equations*. Wily-Interscience, New York, 1 edition, 1994.

[117] J. E. M. Diaz, J. R. Ramirez, J. Villa. The numerical solution of a generalized Burgers-Huxley equation through a conditionally bounded and symmetry-preserving method. *Applied Mathematics and Computation*, 61:3330–3342, 2011.

[118] J. Jing, L. Min, G. Zhao. Partial generalized synchronization theorems of differential and discrete systems. *Kybernetika*, 44:511–521, 2008.

[119] J. Liu, G. Hou. Numerical solutions of the space and time fractional coupled Burgers equations by generalized differential transform method. *Applied Mathematics and Computation*, 217:7001–7008, 2011.

[120] J. M. Burgers. Mathematical example illustrating relations occurring in the theory of turbulent fluid motion. *Trans Roy Academic Science, Amsterdam*, 17:1–53, 1939.

[121] J. M. Gonzalez-Miranda. Generalized synchronization in directionaly coupled systems with identical individual dynamics. *Physical Review E*, 65:202–224, 2002.

[122] J. Mei, M. H. Jiang, W. M. Xu, B. Wang. Finite-time synchronization control of complex dynamical networks with time delay. *Communications in Nonlinear Science and Numerical*, 13:2262–478, 2018.

[123] J. Nee, J. Duan. Limit set of trajectories of the coupled viscous Burgers equations. *Applied Mathematics Letters*, 11:57–61, 1998.

[124] J. P. Zhou, Z. Wang, Y. Wang, Q. K. Kong. Synchronization in complex dynamical networks with interval time-varying coupling delays. *Nonlinear Dynamical*, 72:2487–2498, 2014.

[125] J. Rashidinia, R. Mohammadi, R. Jalilian. Cubic spline method for two-point boundary value problems. *International Journal of Engineering Science*, 19:39–43, 2008.

[126] J. Satsuma. Exact solutions of Burgers equation with reaction terms topics in soliton theory and exactly solvable nonlinear equations. *Conference on Nonlinear Evolution Equations, Solitons and the Inverse Scattering Transform, Oberwolfach, Germany*, pages 255–262, 1986.

[127] J. Zhang, G. Yan. Lattice Boltzmann method for one and two dimensional Burgers equation. *Journal of Physics A*, 387:4771–4786, 2009.

[128] J. Zhao, H. Li, Z. Fang, X. Bai. Numerical solution of Burgers equation based on mixed finite volume element methods. *Discrete Dynamics in Nature and Society*, 2020:1–14, 2020.

[129] K. Al-Khaled. Numerical study of Fisher's reaction-diffusion equation by the sinc collocation method. *Journal of Computational and Applied Mathematics*, 137:245–255, 2001.

[130] K. M. Owolabi. Numerical solution of the generalized Burgers-Huxley equation by exponential time differencing scheme. *International Journal of Biomedical Engineering and Science*, 1:43–52, 2015.

[131] K. Maleknejad, M. Hadizade. A new computational method for Volterra - Fredholm integral equations. *Computers and Mathematics with Applications*, 37:1–8, 1999.

[132] K. N. Wu, T. Tian, L. Wang. Synchronization for a class of coupled linear partial differential systems via boundary control. *Journal of the Franklin Institute*, 353:4062–4073, 2016.

[133] K. Pan, X. Wu, X. Yue, R. Ni. A spatial sixth-order CCD-TVD method for solving multidimensional coupled Burgers equation. *Computational and Applied Mathematics*, 39:50–61, 2020.

[134] K. Pyragas. Weak and strong synchronization of chaos. *Physical Review E*, 54:45–08, 1996.

[135] K. R. Desai, V. H. Pradhan. Solution of Burgers equation and coupled Burgers equations by homotopy perturbation method. *International Journal of Engineering Research and Applications*, 2:2033–2040, 2012.

[136] K. R. Raslan. A collocation solution for Burgers equation using quadratic B-spline finite elements. *International Journal of Computer Mathematics*, 7:931–938, 2003.

[137] K. R. Raslan. A domian decomposition method for Burgers-Huxley and Burgers-Fisher equations. *Applied Mathematics and Computation*, 150:281–292, 2005.

[138] K. T. Elgindy, B. Karasozen. High-order integral nodal discontinuous Gegenbauer-Galerkin method for solving viscous Burgers equation. *International Journal of Computer Mathematics*, 96:2039–2078, 2019.

[139] K. T. Elgindy, B. Karasozen. Distributed optimal control of viscous Burgers equation via a high-order, linearization, integral, nodal discontinuous Gegenbauer-Galerkin method. *Optimal Control Applications and Methods*, 41:253–277, 2020.

[140] K. V. Srivastava, K. Mukesh Awasthi, M. Tamsir. A fully implicit finite difference solution to one dimensional coupled nonlinear Burgers equations. *International Journal of Mathematical and Computational Sciences*, 7:4–2, 2013.

[141] K. W. Morton, D. F. Mayers. *Numerical solution of partial differential equations*. Cambridge University Press, 1994.

[142] K. Yuan, A. Alofi, J. Cao, A. Al-Mazrooei, A. Elaiw. Synchronization of the coupled distributed parameter system with time delay via proportional spatial derivative control. *Discrete Dynamics in Nature and Society*, 2014:1–7, 2014.

[143] L. Andallah, M. Khatun. Numerical solution of advection-diffusion equation using finite difference schemes. *Bangladesh Journal of Scientific and Industrial Research*, 1:15–22, 2020.

[144] L. C. Nair , A. Awasthi. Quintic trigonometric spline based numerical scheme for nonlinear modified Burgers equation. *Applied Mathematics and Computation*, 35:1430–1442, 2019.

[145] L. Debtnath. *Nonlinear partial differential equations for scientist and engineers*. Birkhauser, Boston, 2nd edition edition, 1997.

[146] L. H. Nguyena, K. S. Hongb. Synchronization of coupled chaotic Fitz Hugh Nagumo neurons via Lyapunov functions. *Mathematics and Computers in Simulation*, 82:590–603, 2020.

[147] L. Kocarev, U. Partlitz. Generalized synchronization predictability and equivalence of undirectionally coupled dynamical systems. *Physical Review Letters*, 76:1816–9, 1996.

[148] L. Lin, G. Moutsinas, C. Wu, W. Guo. Synchronization with molecular signals on spatial-temporal complex networks. *Conference: ACM International Conference on Nanoscale Computing and Communication (NanoCom) - Dublin*, 12:1–6, 2019.

[149] L. M. Pecora, T. L. Carroll. Synchronization in chaotic systems. *Physical Review Letters*, 64:821–824, 1990.

[150] L. O. Chua. Memristor the missing circuit element. *IEEE Transactions on Circuit Theory*, 18:507–519, 1971.

[151] L. R. T. Gardner, G. A. Gardner, A. H. A. Ali. A method of lines solutions for Burgers equation. *Computer Mechaine*, 1:1551–1561, 1991.

[152] L. Schumaker. *Spline functions and basic theory*. Bjul, Moskov Skogogos, 1 edition, 1981.

[153] L. V. Wijngaarden. On the equation of motion for mixtures of liquid and gas bubbles. *Journal Fluid Mech*, 33:465–474, 1968.

[154] L. Zhang, L. Wang, X. Ding. Exact finite difference scheme and nonstandard finite difference scheme for Burgers and Burgers-Fisher equations. *Journal of Applied Mathematics*, 2014:1–12, 2014.

[155] M. A. Abdou, A. A. Soliman. Variational iteration method for solving Burgers and coupled Burgers equations. *Journal of Computational and Applied Mathematics*, 2:245–251, 2005.

[156] M. A. Aziz-Alaoui. Synchronization of chaos. *Encyclopedia of Mathematical Physics*, 5:213–226, 2006.

[157] M. A. Ramadan , T. S. El-Danaf, F. E. I. Abd Alaal. A numerical solution of the Burgers equation using septic B-splines. *Chaos, Solitons* & *Fractals*, 26:795–804, 2005.

[158] M. Addam, A. Bouhamidi, K. Jbilou. A numerical method for one-dimensional diffusion problem using Fourier transform and the B-spline Galerkin method. *Applied Mathematics and Computation*, 215:4067–4079, 2010.

[159] M. Aghamohamadi, J. Rashidinia, R. Ezzati. Tension spline method for solution of non-linear Fisher equation. *Applied Mathematics and Computation*, 249:399–407, 2014.

[160] M. Bastani, D. K. Salkuyeh. A highly accurate method to solve Fisher's equation. *Pramana*, 78:335–346, 2012.

[161] M. Basto, V. Semiao, F. Calheiros. Dynamics and synchronization of numerical solutions of the Burgers equation. *Journal of Computational and Applied Mathematics*, 231:793–806, 2010.

[162] M. Dehghan, A. Hamidi, M. Shakourifar. The solution of coupled Burgers equations using adomian Pade technique. *Applied Mathematics and Computation*, 189:1043–1047, 2007.

[163] M. Dehghan, B. N. Saray, M. Lakestani. Three methods based on the interpolation scaling functions and the mixed collocation finite difference schemes for the numerical solution of the nonlinear generalized Burgers-Huxley equation. *Mathematical and Computer Modelling*, 55:1129–1142, 2012.

[164] M. Gulsu. A finite difference approach for solution of Burgers equation. *Applied Mathematics and Computation*, 175:1245–1255, 2006.

[165] M. H. Schultz. *Spline analysis*. Prentice-hall, ING, 1 edition, 1973.

[166] M. Inc. On numerical solution of Burgers, equation by homotopy analysis method. *Physics Letters A*, 372:356–360, 2009.

[167] M. J. Ablowitz, B. Fuchssteiner, M. Kruskal. Topics in solution theory and exactly solvable nonlinear equations. *Topics in Soliton Theory and Exactly Solvable Nonlinear Equations*, 1:1–354, 1987.

[168] M. Javidi. A modified Chebyshev pseudospectral algorithm for the generalized Burgers-Huxley equation. *Computers and Mathematics with Applications*, 62:3366–3377, 2011.

[169] M. Jima, A. Shiferaw, A. Tsegaye. Numerical solution of the coupled Burgers equation using differential quadrature method based on fourier expansion basis. *Applied Mathematics*, 9:821–835, 2018.

[170] M. K. Kadalbajoo, K. K. Sharma, A. Awasthi. A parameter-uniform implicit difference scheme for solving time-dependent Burgers equations . *Applied Mathematics and Computation*, 170:1365–1393, 2005.

[171] M. M. Cecchi, R. Nociforo, P. P. Grego. Space-time finite elements numerical solution of Burgers problems. *Le Matematiche LI*, 1:43–57, 1996.

[172] M. M. El-borai, W. G. Elsayed, T. A. Aljamal. Synchronization and impulsive control of some parabolic partial differential equations. *American Journal of Theoretical and Applied Statistics*, 6:30–39, 2017.

[173] M. M. Rashidi, E. Erfani. New analytic method for solving Burgers and nonlinear heat transfer equations and comparison with homotopy. *Communications in Nonlinear Science and Numerical Simulation*, 180:1539–1544, 2009.

[174] M. M. Rashidi, G. Domairry, S. Dinarvand. Approximate solutions for the Burgers and regularized long wave equations by means of the homotopy analysis method. *Communications in Nonlinear Science and Numerical Simulation*, 14:708–717, 2016.

[175] M. Moghimi, S. A. H. Fatemeh. Variational iteration method for solving generalized Burger-Fisher and Burger equations. *Chaos Solitons & Fractals*, 33:1756–1761, 2007.

[176] M. S. El-Azab. An approximation scheme for a nonlinear diffusion Fisher's equation. *Applied Mathematics and Computation*, 186:579–588, 2007.

[177] M. Sari. Fisher's Equation. *Encyclopedia of Applied and Computational Mathematics*, Chapter-F:550–553, 2015.

[178] M. Sari , G. Gurarslan , A. Zeytinoglu. High-order finite difference schemes for numerical solutions of the generalized Burgers-Huxley equation. *Numerical Methods for Partial Differential Equations*, 27:1313–1326, 2010.

[179] M. Sari, G. Gurarslan. A sixth-order compact finite difference scheme to the numerical solutions of Burgers equation. *Applied Mathematics and Computation*, 208:475–483, 2009.

[180] M. Sari, G. Gurarslan. Numerical solutions of the generalized Burgers-Huxley equation by a differential quadrature method. *Mathematical Problems in Engineering*, 11:1–11, 2009.

[181] M. Sari, G. Gurarslan, I. Dag. A compact finite difference method for the solution of the generalized Burgers-Fisher equation. *Numerical Methods for Partial Differential Equations*, 26:125–134, 2010.

[182] M. Tamsir, N. Dhiman, V. K. Srivastava. Cubic trigonometric B-spline differential quadrature method for numerical treatment of Fisher's reaction-diffusion equations. *Alexandria Engineering Journal*, 57:2019–2026, 2019.

[183] M. Tamsir, V. K. Srivastava, R. Jiwari. An algorithm based on exponential modified cubic B-spline differential quadrature method for nonlinear Burgers equation. *Applied Mathematics and Computation*, 209:111–124, 2016.

[184] M. U. Ahmet, D. Arugaslan, E. Yilmaz. Method of Lyapunov functions for differential equations with piecewise constant delay. *Journal of computational and applied mathematics*, 235:4554–4569, 2011.

[185] N. A. Kudryashov. On exact solutions of families of Fisher equation. *Theoretical and Mathematical Physics*, 94:211–218, 1993.

[186] N. Dhiman, M. Tamsir, N. Dhiman. A collocation technique based on modified form of trigonometric cubic B-spline basis functions for Fisher's reaction-diffusion equation. *Multi discipline Modeling in Materials and Structures*, 14:923–939, 2018.

[187] N. Rulkov, M. Sushchik, L. Tsimring, H. Abarbanel. Generalized synchronization of chaos in directionally coupled chaotic systems. *Physical Review E*, 51:980–904, 1995.

[188] O. Chakrone, O. Diyer, D. Sbibih. Numerical solution of Burgers equation based on cubic B-splines quasi-interpolants and matrix arguments. *Boletim da Sociedade Paranaense de Matemática*, 33:109–119, 2015.

[189] O. Ersoy, I. Dag. The extended B-spline collocation method for numerical solutions of Fisher equation. *AIP Conference Proceedings*, 12:345–359, 2015.

[190] O. Oruc, F. Bulut, A. Esen. A Haar wavelet-finite difference hybrid method for the numerical solution of the modified Burgers equation. *Journal of Mathematical Chemistry*, 53:1592–1607, 2015.

[191] O. Oruc, F. Bulut, A. Esen. Chebyshev wavelet method for numerical solutions of coupled Burgers equation. *Hacettepe Journal of Mathematics* & *Statistics*, 48:1–16, 2019.

[192] P. C. Jain, B. L. Lohar. Cubic spline technique for coupled non-linear parabolic equations. *Applied Mathematics and Computation*, 5:179–185, 1979.

[193] P. C. Jain, D. N. Holla. Numerical solutions of coupled Burgers equations. *International Journal of Non-linear Mechaine*, 13:213–222, 1978.

[194] P. C. Jain, R. Shankar, T. V. Singh. Numerical technique for solving convective-reaction-diffusion equation. *Mathematices Computational Modelling*, 9:113–125, 1995.

[195] P. Fletcher, S. Haswell, V. Paunov. Theoritical considerations of chemical reactions in micro reactors operating under electro osmotic and electrophoretic control. *Analyst*, 124:1273–1282, 1969.

[196] P. Garcia, A. Acosta, H. Leiva. Synchronization conditions for master-slave reaction diffusion systems. *Europhys Lettere*, 88:06–36, 2009.

[197] P. M. Prenter. *Splines and vartational methods*. John wiley, New York, 2 edition, 1975.

[198] P. Milan, K. Vladimír, H. Zbynek, S. Katalin. Synchronization as adjustment of information rates detection from bivariate time series. *Physical Review E*, 63:046–211, 2001.

[199] R. A. Fisher. The wave of advance of advantageous genes. *Annals of Eugenics*, 7:355–369, 1937.

[200] R. Abazari, A. Borhanifar. Numerical study of the solution of the Burgers and coupled Burgers equations by a differential transformation method. *Computers* & *Mathematics with Applications*, 59:2711–2722, 2010.

[201] R. Anguelov, P. Kama, J. M. S. Lubuma. On non-standard finite difference models of reaction–diffusion equations. *Journal of Computational and Applied Mathematics*, 175:11–29, 2005.

[202] R. C. Mittal. Tripartite numerical solutions of generalized Burgers-Fisher and generalized Burgers-Huxley a compact finite difference method for the generalized Burgers-Fisher equation. *International Journal of Computer Mathematics*, 92:1053–1077, 2015.

[203] R. C. Mittal , R. Jiwari. A numerical study of Burger-Huxley equation by differential quadrature method. *Journal of Applied Mathematics and Mechanics*, 5:1–9, 2009.

[204] R. C. Mittal, A. Tripathi. A collocation method for numerical solutions of coupled Burgers equations. *International Journal for Computational Methods in Engineering Science and Mechanics*, 15:457–471, 2012.

[205] R. C. Mittal, P. Singhal. Numerical solution of Burgers equation. *Communications in Numerical Methods in Engineering*, 9:397–406, 1993.

[206] R. J. Spiteri, S. J. Ruuth. A new class of optimal high-order strong-stability-preserving time discretization methods. *SIAM Journal on Numerical Analysis*, 40:469–491, 2002.

[207] R. Jiwari. A Haar wavelet quasi linearization approach for numerical simulation of Burgers equation. *Computer Physics Communications*, 183:2413–2423, 2012.

[208] R. Jiwari. A hybrid numerical scheme for the numerical solution of the Burgers equation. *Computer Physics Communications*, 188:59–67, 2015.

[209] R. Jiwari, A. Alshomrani. A new algorithm based on modified trigonometric cubic B-splines functions for nonlinear Burgers type equations. *International Journal of Numerical Methods for Heat* & *Fluid Flow*, 27:1638–1661, 2017.

[210] R. Jiwari, R. C. Mittal, K. K. Sharma. A numerical scheme based on weighted average differential quadrature method for the numerical solution of Burgers equation. *Applied Mathematics and Computation*, 219:6680–6691, 2013.

[211] R. Jiwari, S. Pandit, M. E. Koksal. A class of numerical algorithms based on cubic trigonometric B-spline functions for numerical simulation of nonlinear parabolic problems. *Computational and Applied Mathematics*, 38:140–152, 2019.

[212] R. K. Jain, R. C. Mittal. Numerical solutions of nonlinear Burgers equation with modified cubic B-splines collocation method. *Applied Mathematics and Computation*, 218:7839–7855, 2012.

[213] R. Mohammadi. B-spline collocation algorithmfor numerical solution of the generalized Buegers-Huxley equation. *Numerical Methods for PartialDifferential Equations*, 29:1173–1191, 2013.

[214] R. Mohanty, S. Sharma. High accuracy quasi variable mesh method for the system of 1D quasi linear parabolic partial differential equations based on off-step spline in compression approximations. *Advances in Difference Equation*, 3:1274–3, 2017.

[215] R. Nawaz, H. Ullah, S. Islam, M. Idrees. Application of optimal homotopy asymptotic method to Burger equations. *Journal of Applied Mathematics*, 2013:1–8, 2013.

[216] R. Q. Quiroga, J. Arnhold, P. Grassberger. Learning driver-response relationships from synchronization patterns. *Physical Review E*, 61:5142–5148, 2000.

[217] R. Rohila, R. C. Mittal. Numerical study of reaction diffusion Fisher's equation by fourth order cubic B-spline collocation method. *Mathematical Sciences*, 12:79–89, 2018.

[218] S. Abbasbandy, M. T. Darvishi. A numerical solution of Burgers equation by modified adomian method. *Applied Mathematics and Computation*, 3:1265–1272, 2005.

[219] S. Bu, S. Bak. Simulation of advection-diffusion–dispersion equations based on a composite time discretization scheme. *Advances in Difference Equations*, 132:240–253, 2020.

[220] S. E. A. Alhazmi. Numerical solution of Fisher's equation using finite difference. *Bulletin of Mathematical Sciences and Applications*, 12:27–34, 2015.

[221] S. E. Esipov. Coupled Burgers equations: a model of polydispersive sedimentation. *Physical Review Letters*, 52:3711–3718, 1995.

[222] S. G. Rubin, P. K. Khosla. Higher-order numerical solutions using cubic splines. *AIAA Journal*, 14:851–858, 1976.

[223] S. G. Rubin, R. A. Graves. Cubic spline approximation for problems in fluid mechanics. *Computers* & *Fluids*, 3:1–36, 1975.

[224] S. H. Strogatz. *Nonlinear dynamics and chaos: With applications to physics, biology, chemistry, and engineering*. Perseus Books, Massachusetts, USA, 1 edition, 1994.

[225] S. Kazema, M. Shaban, J. Amani Rad. Solution of the coupled Burgers equation based on operational matrices of d-Dimensional orthogonal functions. *Zeitschrift fur Naturforschung A*, 67:267–274, 2012.

[226] S. Kumar, D. Baleanu, A. Kumar. Two analytical methods for time fractional nonlinear coupled Boussinesq Burgers equations arise in propagation of shallow water waves. *Nonlinear Dynamics*, 2:699–715, 2016.

[227] S. Kutluay, A. Esen. A linearized numerical scheme for Burgers like equations. *Applied Mathematics and Computation*, 156:295–305, 2004.

[228] S. Kutluay, A. Esen. A lumped Galerkin method for solving the Burgers equation. *International Journal of Computer Mathematics*, 81:1433–1444, 2004.

[229] S. Kutluay, Y. Ucar. Numerical solutions of the coupled Burgers equation by the Galerkin quadratic B-spline finite element method. *Mathematical Methods in the Applied Sciences*, 36:2403–2415, 2013.

[230] S. Kutluay, Y. Ucar, N. M. Yagmurlu. Numerical solutions of the modified Burgers equation by a cubic B-spline collocation method. *Bulletin of the Malaysian Mathematical Sciences Society*, 39:1603–1614, 2016.

[231] S. Kutulay, A. Esen, I. Dag. Numerical solutions of the Burgers equation by the least squares quadratic B-spline finite element method. *Journal of Computational and Applied Mathematics*, 167:21–33, 2004.

[232] S. Momani, Z. Odibat, V. S. Erturk. Generalized differential transform method for solving a space and time fractional diffusion-wave equation. *Physics Letters A*, 370:379–387, 2007.

[233] S. S. Xie, S. Heo, S. Kim, G. Woo, S. Yi. Numerical solution of one-dimensional Burgers equation using reproducing kernel function. *Journal of Computational and Applied Mathematics*, 214:417–434, 2008.

[234] S. Tang, R. O. Weber. Numerical study of Fisher's equation by a Petrov-Galerkin finite element method. *Journal of Mathematics and Mathematical Sciences*, 33:27–38, 1991.

[235] S. W. Jones. Interaction of chaotic advection and diffusion. *Chaos, Solitons* & *Fractals*, 4:929–940, 1994.

[236] T. Mavoungou, Y. Cherruault. Numerical study of Fisher's equation by a domian's method. *Numerical Methods for Partial Differential Equations*, 19:89–95, 1994.

[237] T. Ozis, A. Esen, S. Kutluay. Numerical solution of Burgers equation by quadratic B-spline finite elements. *Applied Mathematics and Computation*, 165:237–249, 2005.

[238] T. Ozis, E. N. Aksan, A. Ozdes. A finite element approach for solution of Burgers equation. *Applied Mathematics and Computation*, 139:417–428, 2003.

[239] T. S. El-Danaf, K. R. Raslan, K. K. Ali. New numerical treatment for the generalized regularized long wave equation based on finite difference scheme. *International Journal of Soft Computing and Engineering*, 4:16–24, 2019.

[240] T. Tel, A. Pentek, I. Scheuring, Z. Toroczkai, C. Grebogi, J. Kadtke. Chaotic advection, diffusion, and reactions in open flows. *Chaos*, 10:89–98, 2000.

[241] T. Zhao, C. Li, Z. Zang, Y. Wu. Chebyshev-Legendre pseudo spectral method for the generalised Burgers-Fisher equation. *Applied Mathematical Modelling*, 36:1046–1056, 2012.

[242] U. Erdogan, M. Sari, H. Kocak. Efficient numerical treatment of nonlinearities in the advection-diffusion-reaction equations. *International Journal of Numerical Methods for Heat* & *Fluid Flow*, 29:1–20, 2019.

[243] V. Chandraker, A. Awasthi, S. Jayaraj. Numerical treatment of Burgers-Fisher equation. *Procedia Technology*, 25:1217–1225, 2016.

[244] V. Gupta, M. K. Kadalbajoo. A singular perturbation approach to solve Burgers-Huxley equation via monotone finite difference scheme on layer adaptive mesh. *Communications in Nonlinear Science and Numerical Simulation*, 16:1825–1844, 2011.

[245] V. J. Ervin, J. E. Macias-Diaz, J. Ruiz-Ramirez. A positive and bounded finite element approximation of the generalized Burgers-Huxley equation. *Journal of Mathematical Analysis and Applications*, 424:1143–1160, 2015.

[246] V. S. Borkar. On extremal solutions to stochastic control problems. *Applied Mathematics* & *Optimization*, 28:49–56, 1993.

[247] W. C. Q. Xu, Z. Zheng. Solution of two-dimensional time-fractional Burgers equation with high and low Reynolds numbers. *Advances in Difference Equations*, 3:338–343, 2018.

[248] W. G. Bickley. Piece wise cubic interpolation and two-point boundary value problem. *Computer Journal*, 11:206–208, 1968.

[249] W. Liao. An implicit fourth-order compact finite difference scheme for one-dimensional Burgers equation. *Applied Mathematics and Computation*, 206:755–764, 2008.

[250] X. H. Zhang, J. Ouyang, L. Zhang. Element-free characteristic Galerkin method for Burgers equation. *Engineering Analysis with Boundary Elements*, 10:356–362, 2009.

[251] X. Z. Jin, G. H. Yang. Adaptive sliding mode fault-tolerant control for nonlinearly chaotic systems against network faults and time-delays. *Journal Franklin Institute*, 13:1260–1220, 2019.

[252] Y. C. Hon, X. Z. Mao. An efficient numerical scheme for Burgers equation. *Applied Mathematics and Computation*, 95:37–50, 1998.

[253] Y. Chen, T. Zhang. A weak Galerkin finite element method for coupled Burgers equation. *Journal of Computational and Applied Mathematics*, 348:103–119, 2020.

[254] Y. Chen, Z. Jia, G. Deng. Adaptive Lag synchronization of Lorenz chaotic system with uncertain parameters. *Applied Mathematics*, 3:549–553, 2012.

[255] Y. Kawamura, S. Shirasaka, T. Yanagita, H. Nakao. Optimizing mutual synchronization of rhythmic spatio temporal patterns in reaction-diffusion systems. *Physical Review E*, 96:12–24, 2018.

[256] Y. Wen, X. F. Zhou, Z. Zhang, S. Liu. Lyapunov method for nonlinear fractional differential systems with delay. *Nonlinear Dynamics*, 82:1015–1025, 2018.

[257] Z. Rong-Pei, Y. Xi-Jun, Z. Guo-Zhong. Local discontinuous Galerkin method for solving Burgers and coupled Burgers equations. *Chinese Physics B*, 20:1–11, 2011.

[258] Z. Shi, Y. Cao, Q. J. Chen. Solving 2D and 3D poisson equations and biharmonic equations by the Haar wavelet method. *Applied Mathematical Modelling*, 36:5143–5161, 2012.

[259] Z. Y. Ali Allawee. A spatial sixth order CCD - TVD method for solving multi dimensional coupled Burgers equation. *International Journal of Computer Science*, 15:31–39, 2018.

[260] Z. Zhang, G. E. Karniadakis. *Numerical methods for stochastic partial differential equations with white noise*. Springer Nature, 11 edition, 2010.

PUBLICATIONS FROM THE THESIS

Contact Information: shko.ali.tahir@std.yildiz.edu.tr

Papers

1. S. Ali Tahir, M. Sari, A. Bouhamidi. Generalized synchronization of identical and nonidentical chaotic dynamical systems via master approaches. An International Journal of Optimization and Control: Theories & Applications, 7 (3):248-254, 2017.

2. S. Ali Tahir, M. Sari, A. Bouhamidi. Designing a response approach in chaotic systems. Sigma Journal of Engineering and Natural Sciences, 36 (2):445-323, 2018.

3. M. Sari, S. Ali Tahir, A. Bouhamidi. Behaviour of advection-diffusion-reaction processes with forcing terms. Carpathian Journal of Mathematics, 35 (2):1843-4401, 2019.

4. S. Ali Tahir, M. Sari. A B-spline-SSPRK54 method for advection-diffusion processes. Bulletin of Mathematical Analysis and Applications, 11 (3):1-10, 2019.

5. S. Ali Tahir, M. Sari. Simulations of nonlinear parabolic PDEs with forcing function without linearization, 2019. (Submitted).

6. M. Sari and S. Ali Tahir. Synchronization of the advection-diffusion-reaction processes, 2019. (Submitted).

7. S. Ali Tahir, M. Sari. A new approach for the nonlinear coupled advection-diffusion-reaction processes with source functions, 2020. (Submitted).

8. S. Ali Tahir, M. Sari. Historical literature review: Nonlinear ADR problems, 2020. (Submitted).

Conference Papers

1. S. Ali Tahir, M. Sari. Fundamental of generalized synchronization in chaotic systems. Computational Sciences Research, May-16, Istanbul, Turkey, 2016.

2. S. Ali Tahir, M. Sari, A. Bouhamidi. Synchronization of identical and nonidentical chaotic dynamical systems. 2^{nd} international conference on computational mathematics and engineering sciences, May-20-22, Istanbul, Turkey, 2017.

3. M. Sari, S. Ali Tahir. Reasonable modifications under the B-spline basis with the SSPRK54 for solving the Burgers equation. 4^{th} international conference on analysis and its applications, September-11-14, Kirsehir, Turkey 2018.