Block-Hybrid method for Burgers-Huxley equation

M PHEKO

orcid.org/0000-0002-3876-2030

Dissertation accepted in fulfilment of the requirements for the degree *Master of Science in Applied Mathematics* at the North West University

Supervisor: Dr. LD. Moleleki (NWU)

Co-supervisor: Prof. S.P Goqo (UKZN)

Graduation ceremony: 18 May 2023

Student number: 29298679

Contents

Declaration iv

Dedication v

Acknowledgements vi

Abstract vii

List of Figures viii

List of Tables x

List of Acronyms xi

1 Introduction **1**

Introduction **1**

1.1 Literature review 2

1.2 Numerical methods 3

1.2.1 Finite difference method 4

1.2.2 Spectral collocation methods 6

1.2.2.1 Multidomain methods 7

1.2.3 Multistep methods 8

1.2.4 Block-Hybrid methods 9

1.3 Analytical methods 10

1.3.1 The Tanh method . . . 10
1.3.2 G'/ G-expansion method . . . 11
1.3.3 Lie group method . . . 11
1.4 Aims and Objectives . . . 13
1.5 Dissertation outline . . . 13

2 Lie group and numerical methods for solving NLPDEs 14

2.1 Lie group theory of PDEs . . . 14
2.1.1 Local one-parameter Lie group . . . 14
2.1.2 Infinitesimal transformations . . . 16
2.1.3 Group (Set) invariants . . . 16
2.1.4 Symmetry set construction . . . 17
2.1.4.1 Prolongation of point transformations . . . 17
2.1.4.2 Group (set) admitted by a PDE . . . 19
2.1.5 Lie algebras . . . 20
2.2 Numerical methods . . . 20
2.2.1 Spectral methods . . . 20
2.2.2 Spectral quasi-linearization method . . . 23
2.2.2.1 Quasi-linearization method . . . 24
2.2.2.2 Bivariate spectral quasi-linearization method . . . 24
2.2.2.3 Multi-domain approach . . . 27
2.3 Block-Hybrid method for PDEs . . . 28
2.3.1 Development of the Method . . . 29
2.4 Summary . . . 33

3 A comparative study of a spectral quasilinearization, multidomain spectral quasilinearization and block-hybrid methods for solving Burgers-Huxley equation **34**

3.1 The generalized Burgers-Huxley equation 34

3.2 Numerical solution of Burger-Huxley equation 35

3.2.1 Lie symmetry 36

3.2.2 Spectral Quasi-linearizaation Method 37

3.2.3 Multi-Domain Bivariate Spectral Quasi-linearization Method . 38

3.2.4 Block-Hybrid Method 40

3.3 Results and Discussion 42

3.3.1 Numerical convergence analysis 43

3.4 summary 48

4 Numerical results of evolution problems **49**

4.1 Nonlinear evolution equations 49

4.2 Numerical Experimentations 50

4.2.1 Example 1: The generalized Burgers-Fisher equation 50

4.2.2 Example 2: The FitzHugh-Nagumo equation 52

4.2.3 Example 3: The Coupled Burgers'equation 54

4.3 Summary 56

5 Results and Discussion **57**

5.1 Summary 71

6 Conclusion **73**

Declaration

I, MAPULE PHEKO, student number 29298679, declare that this dissertation for the degree of Master of Science in Applied Mathematics at North West University, Mafikeng Campus, hereby submitted, has not previously been submitted by me for a degree at this or any other University, that this is my own work in design and execution and that all material contained herein has been duly acknowledged.

Signed: ....

Miss M. Pheko

Date: 04 April 2023

This dissertation has been submitted with my approval as a University supervisor and would certify that the requirements applicable for the Master of Science degree rules and regulations have been fulfilled.

Signed:..

Dr. L. D. Moleleki

Date: 04 April 2023

Signed:.....................................

Prof. S. P. Goqo

Date: 04/04/2023

Dedication

I dedicate this dissertation to my caring family and to all of you who helped me with my studies. My former maths teachers, Mr. Mabokela, Mr. Koole, and Mr. Lekwene, are also honored in this dissertation.

Acknowledgements

First and foremost, I would like to thank God, the Almighty, and my ancestors for giving me the strength and life to carry out this research. I thank my family for raising and supporting me during difficult times. I can't pin down what role each person played in the development of who I am today, but they all have a special place in my heart. I am grateful to everyone who prayed for me, they gave me strength and enabled me to persevere in difficult circumstances. I'd like to express my heartfelt gratitude to my supervisors Dr. L. D. Moleleki and Prof. S. P. Goqo for their assistance and encouragement in compiling this project together. Finally, I would also like to acknowledge the financial assistance provided by the North West University Department Fund and the North West University postgraduate bursary scheme, Mafikeng Campus.

Abstract

In this study, Lie symmetries, spectral quasilinearization method, multi-domain bivariate spectral quasilinearization method, and block-hybrid method are utilized in solving the second order nonlinear evolution problems. The spectral quasilinearization method is attained by applying the spectral collocation technique in space derivatives, the multi-domain bivariate spectral quasilinearization method is applied both in time and space derivatives and the block-hybrid method is applied only on the time derivatives. The applicability of these techniques is presented by solving a class of second-order nonlinear evolution PDEs, specifically the Burgers-Huxley equation, Burgers-Fisher equation, FitzHugh-Nagumo equation, and Coupled Burgers' equation. The numerical outcomes are confirmed for accuracy by comparing them with analytical solutions. Tables for the spectral quasilinearization method, multi-domain bivariate spectral quasilinearization method, and Block-Hybrid method with their computational times were produced for comparing the order of precision for each technique, and error graphs are displayed.

Keywords: Lie symmetries, Spectral quasilinearization method, Multi-domain bivariate spectral quasilinearization method, Block-hybrid method, Nonlinear evolution equations

List of Figures

3.1 Exact solution vs Numerical solution of Burgers-Huxley equation utilizing the SQLM at $t = 0$. 44

3.2 Exact solution vs Numerical solution of Burgers-Huxley equation utilizing the MD-BSQLM at $t = 10$. 44

3.3 Exact solution vs Numerical solution of Burgers-Huxley equation at $t = 10$ utilizing the BHM. 44

3.4 Error profile of the numerical results of Burgers-Huxley equation utilizing SQLM at $t = 0$. 45

3.5 Error profile of the numerical results of Burgers-Huxley equation utilizing MD-BSQLM at $t = 10$. 45

3.6 Error profile of the numerical results of Burgers-Huxley equation utilizing BHM at $t = 10$. 45

3.7 Exact solution and Numerical solution of Burgers-Huxley equation . . 48

5.1 Exact VS Numerical of Burgers-Fisher equation using SQLM for $\nu = \eta = n = 1$. 58

5.2 Exact VS Numerical of FitzHugh-Nagumo equation using SQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. 58

5.3 Error profile of Burgers-Fisher equation using SQLM for $\nu = \eta = n = 1$. 58

5.4 Error profile of FitzHugh-Nagumo equation using SQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. 58

5.5 Exact VS Numerical of Burgers-Fisher equation for MD-BSQLM for $\mu = \eta = n = 1$. 61

5.6 Exact VS Numerical of FitzHugh-Nagumo equation using MD-BSQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. 61

5.7 Exact vs Numerical solution of Coupled Burgers' equation using MD-BSQLM. 61

5.8 Error profile of Burgers-Fisher equation using MD-BSQLM $\nu = \eta = n = 1$. 62

5.9 Error profile of MD-BSQLM for FitzHugh-Nagumo equation for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. 62

5.10 Error profile of MD-BSQLM for Coupled Burgers' equation. 62

5.11 Exact vs Numerical solution of Burgers-Fisher equation for BHM for $\nu = \eta = n = 1$. 65

5.12 Exact vs Numerical solution of FitzHugh-Nagumo equation using BHM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. 65

5.13 Exact vs Numerical solution of Coupled Burgers' equation using BHM. 66

5.14 Error profile of Burgers-Fisher equation for BHM for $\alpha = \beta = n = 1$. 66

5.15 Error profile of BHM for FitzHugh-Nagumo equation for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. 66

5.16 Error profile of BHM for Coupled Burgers' equation. 67

5.17 Exact solution and Numerical solution of Burgers-Fisher equation . . 70

5.18 Exact solution and Numerical solution of FitzHugh-Nagumo equation 70

5.19 Exact solution of Coupled Burgers' equation 71

5.20 Numerical solution of Coupled Burgers' equation 71

List of Tables

3.1 Maximum errors for Burgers-Huxley equation using SQLM. 46

3.2 Maximum errors for Burgers-Huxley equation using MD-BSQLM. . . 47

3.3 Maximum errors for Burgers-Huxley equation using BHM. 47

5.1 Maximum errors for Burgers-Fisher equation utilizing SQLM for $\nu = \eta = n = 1$. 59

5.2 Maximum errors for FitzHugh-Nagumo equation utilizing SQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. 60

5.3 Maximum errors for Burgers-Fisher equation utilizing MD-BSQLM for $\nu = \eta = n = 1$. 63

5.4 Maximum errors for FitzHugh-Nagumo equation utilizing MD-BSQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. 64

5.5 Maximum errors for the Coupled Burgers' equation obtained utilizing MD-BSQLM. 64

5.6 Maximum errors for Burgers-Fisher equation utilizing BHM for $\nu = \eta = n = 1$. 68

5.7 Maximum errors for FitzHugh-Nagumo equation utilizing BHM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. 68

5.8 Maximum errors for Coupled Burgers' equation utilizing BHM. 69

List of Acronyms

QLM:	Quasi-Linearization Method
SQLM:	Spectral Quasi-Linearization Method
BSQLM:	Bivariate Spectral Quasi-Linearization Method
MD-BSQLM:	Multi-Domain Bivariate Spectral Quasi-Linearization Method
BHM:	Block-Hybrid Method
SBHM:	Spectral Block-Hybrid Method
ODE:	Ordinary differential equation
PDE:	Partial differential equation
NLPDE:	Nonlinear partial differential equation

Chapter 1

Introduction

Numerous natural phenomena are modelled using differential equations, such as fluid mechanics, mathematical biology and financial mathematics. Analytical techniques such as the Tanh method, G'/G-expansion technique, Lie group method, homotopy perturbation technique and many others are frequently utilized to determine the exact outcomes of nonlinear differential equations. Although many schemes have already been developed by academics, computational efficiency can still be improved significantly. New schemes should be developed, current numerical methods should be improved, and effective techniques should be created to enhance computing capacity. However, some numerical techniques for approximating NLPDEs have been proposed. These comprise, but are not restricted to, the spectral method, the finite element method, the block-hybrid method, as well as the finite difference method. In Section 1, we tackle the literature review of the nonlinear evolution problems. In Section 2 and 3, we look at the advantages and disadvantages of numerical and analytical methods respectively. Section 4 presents the project's aims and objectives. This dissertation's outline is presented in Section 5.

1.1 Literature review

Nonlinear partial differential equations can accurately describe natural phenomena and their dynamics. Nonlinear evolution equations such as the Burgers equation became utilized as mathematical archetypal for agitation [1], Burgers-Huxley equation classifies how convection effects, reaction mechanisms, including the diffusion transports interact [2], FitzHugh-Nagumo equation, on the other hand, is employed in population inheritances and electrical circuit idea to model nerve impulse transmission [3]. In literature, a great number of numerical techniques for approximating a solution of nonlinear evolution problems have been suggested. Inan et al., [4] came up with a new implicit exponential finite difference method for solving the generalized Burgers-Huxley equation. For the numerical solution of the nonlinear generalized Burgers-Huxley equation, the technigue provides high accuracy. The Burgers-Huxley equation was solved by Alinia et al., [5] implementing the B-spline method. When compared to the solutions presented in literature, the obtained results are quite acceptable and stable. The collocation method with radial basis functions was utilized to obtain the numerical result to the Korteweg-De Vries (KdV) equation by Dris et al., [6]. In terms of accuracy in obtaining the KdV equation solution, a contrast of the typical radial basis functions in the collocation technique discloses that the multiquadric radial basis function is far superior, while the collocation technique with the reedy plate spline function has the lowest accuracy. As a result, the multiquadric radial basis function can also be used to compute KdV equation solutions. To investigate the synchronization of two FitzHugh-Nagumo neurons through or devoid of external current, the Laplace transform along with the Adomian decomposition technique was employed by Zhen et al., [7]. The numerical simulations validate that the principle obtainable in this article is effectual. The exact result derived from the Adomian decomposition method, which describes the synchronous dynamics of the two FitzHugh-Nagumo neurons, is sufficiently accurate. FitzHugh-Nagumo equation was solved utilizing Galerkin finite element method by Ali et al., [8]. According to this result, this technique has a higher estimated order of convergence.

As a result, Galerkin finite element method was determined to be an effective, unreservedly steady, highly flexible, and simply expandable method for solving more complex engineering problems. Fernandino et al., [9] used the least-squares spectral element method (LS-SEM) to solve the one-dimensional Cahn-Hilliard equation. The LS-SEM was applied in a time-space coupled formulation, in which the same discretization method was used simultaneously to solve space and time dependencies. Concerning numerical result, mass conservation, energy level, along with dissipation, the solution detected with the LS-SEM was the same as that established by other studies in literature [10–12]. According to the convergence study, when the problem is properly determined, the technique pleases the predictable spectral convergence in the residual. The nonpolynomial B-spline collocation technique is used by Zahra et al., [13] to solve time-dependent problems such as the PHI-Four and Allen-Cahn equations. The findings of the study are reassuring. When compared to the numerical results of time-dependent problems, the presented scheme is not complicated, and is a good substitute for some other techniques. A compact analysis of numerical methods follows next.

1.2 Numerical methods

A numerical method is a computer technique that in most cases is used to solve a mathematical problem that does not have an analytical solution. Other fields, such as science, medicine, economics, and engineering, have made use of numerical methods. Applications frequently generate mathematical problems that are inconvenient to solve using exact solutions [14]. In most cases, developing numerical methods entails employing a small number of general and relatively simple ideas. Iteration is a concept that appears frequently in many contexts. Iteration, in its broadest sense, refers to the repetition of a process or action pattern [14]. This section goes over a few numerical methods for solving nonlinear PDEs. We talk about their characteristics, benefits, and drawbacks.

1.2.1 Finite difference method

The finite difference method, the finite element method, and the finite volume method are the three methods for solving PDEs numerically. The finite difference methods are a type of a numerical technique for solving differential equations by approximating derivatives with finite differences. It has been used to solve a diversity of problems. These consist of linear and nonlinear problems, plus problems that are time-independent and time-dependent. The technique was first used to solve the Dirichlet biharmonic equations in 1930 [15]. This results from a Taylor series that approximates the unknown function derivatives utilizing central difference, backward, including forward schemes [16]. Finite difference method is categorized into many types but for the purpose of this study we chose the following: Crank-Nicolson approximation, which implements the central difference scheme, the explicit finite difference method, that uses the forward difference approximation, and the implicit finite difference method, that utilizes the backward difference approximation. The values at present time interval must be known in order to calculate values at the following time interval, which is why this method is known as an explicit finite difference method. In these circumstances, the time-to-space step ratio determines convergence and stability. The implicit technique is monotonous and detailed when solving a system of equations because each time step entails the result of a system of numerical equations. Although the Crank-Nicolson method combines the forward and backward methods, it is not merely the average of those two approaches because the backward difference method has an implicit dependence on the result. To approximate a function, finite difference methods use a local polynomial interpolant. The function's derivatives are estimated by deriving local polynomial. A local approximation is the utilization of delicately dispersed grid points to approach a function or its derivative at a point.

The finite element technique, like the finite difference technique, is a numerical method that is common and potent in its application to practical problems including multi-physics, complex geometry, and boundary conditions [17]. In the finite element method, a particular domain is regarded as a collection of subdomains, and the key equation is approximated over each subdomain using any of the old variational methods or any method that is appropriate. The key motive for pursuing approximate solutions on a collection of subdomains is that a complex function is clearer to represent than a collection of basic polynomials. The finite element method is made up of dual numerical approaches, the Galerkin and the Ritz approaches. The variables are approached within the component by means of continuous piecewise functions in Galerkin technique. Ritz technique employs calculus of difference to change the problem to integral form [18]. Galerkin technique performs for smooth solutions, whereas the deep Ritz method performs better for low-regularity solutions. Deep Ritz methods can outperform deep Galerkin methods for smooth solutions, and vice versa for low-regularity solutions. However, the two methods are clearly dimensionally dependent.

The Finite Volume Method is a numerical procedure for converting PDEs in place of conservation laws in excess of discrete volumes keen on detached algebraic equations over finite volumes (or elements or cells) [19]. It has been widely used in a range of engineering disciplines, as well as fluid mechanics, heat along with mass transfer, and petroleum engineering. Volume integrals in a PDE with a deviation term are changed to surface integrals in the finite volume method by way of the divergence theorem. Some of the major traits of the finite volume method are like those of the finite element method: it can be applied to arbitrary irregular geometries with planned or unplanned meshes, and yields powerful schemes [20]. The finite-volume method's power is that it only necessitates flux evaluation at the cell boundaries. This holds true for nonlinear problems as well, making it mainly beneficial for finance with nonlinear conservation laws in transportation problems. Local precision of the finite-volume technique, such as near a point of curiosity, can be improved by refining the mesh about that point, as with the finite-element method. The functions that approximate the result, however, cannot be easily made of higher order when using the finite-volume method.

We have a need for numerical methods that allow us to use coarser grids with smaller quantities of computational resources. Spectral techniques are an illustration of such approaches as they employed all obtainable function values to assemble essential estimations. The concise overview of spectral methods is presented below.

1.2.2 Spectral collocation methods

To find an approximate numerical result to a differential equation, the spectral collocation method employs a finite set of differentiable basis functions. The precision of these methods for a given number of unknowns is their main advantage. They provide exponential convergence rates and spectral accuracy for smooth problems in simple geometries. A more recent method for solving nonlinear equation systems is the spectral quasi-linearization method (SQLM). The quasi-linearization method (QLM), which was first proposed by Bellman and Kalaba [21], is a simplification of the Newton-Raphson technique for resolving nonlinear boundary value problems. The iterative technique is achieved by linearizing the nonlinear elements of the differential equations utilizing the addition of the Taylor multivariable sequence. This method works for both ODEs and PDEs. It only takes a few iterations to get converged solutions with a lower number of grid points [**?**], and it provides precise results in few seconds. Spectral local linearization method (SLLM) separates and also linearizes the systems of equations by merging a univariate linearization technique along with a spectral collocation discretization [22]. The SLLM algorithm's key aspect is to divide a huge coupled system of equations to become a series of reduced subsystems that can be dissolved in sequence in an absolute computationally efficient manner [22]. The spectral relaxation method (SRM) is created on direct iteration schemes that are put together by decreasing the order of the first equation and then reordering the subsequent governing nonlinear equation systems [24]. The SRM, unlike most iterative schemes for solving nonlinear systems of equations, does not need any evaluation of derivatives, linearization, as well as perturbation [24]. Motsa et al., [25] used SQLM and SRM to study the unsteady boundary layer flow problems. Dlamini et al., [26] verified the synchronization of two similar chaotic Lorenz structures using the same method. The SLLM was used by Shateyi and Marewo [27] to investigate the movement of warmth over a porous surface that prolongs thermal radiation. Motsa et al., [28] implemented the SLLM to comprehend the conditions that are ideal for a two-dimensional nanofluid stream over a warmed perpendicular surface with a sinusoidal temperature. Motsa et al., [29] developed the BSQLM as an upgrade to the SQLM for solving NLPDEs. Among the problems effectively resolved by the BSQLM are the modified KdV equation, Burger equation, Cahn-Hillard equation, including FitzHugh-Nagumo equation [29]. Researchers came to the conclusion that the technique is precise, and dependable. The acquired outcomes further verified that the technique converges quickly to the exact solution and achieves high accuracy while utilizing a lower number of spatial grid points. Additionally, Shateyi et al., [30], employed the BSQLM to solve the Bratu problem in two dimensions. BSQLM was found to be faster and more computationally efficient. The enhanced SRM is known as bivariate interpolation spectral relaxation method (BI-SRM). Magagula et al., [31] used bivariate interpolation spectral relaxation (BSRM) to solve for magnetohydrodynamic flow in porous media. The major benefit of the BSRM is that it provides enhanced precision on coarser grids, which ominously increases the method's computational rapidity [31]. The bivariate spectral local linearization method (BSLLM) was invented as an addition to the local linearization method, which has beforehand been magnificently useful to systems of ODEs [22]. Motsa et al., [32] solved the problem of unstable heat and mass transmission past a semi-infinite perpendicular plate with diffusion-thermo along with thermophoresis effects in the attendance of resistance and found it to be accurate. This technique can also be modified to solve other types of NLPDE systems with fluid mechanics applications or problems from disparate scientific and engineering specialities that are defined in terms of NLPDE systems.

1.2.2.1 Multidomain methods

The multidomain bivariate spectral quasilinearization method (MD-BSQLM) is an improvement on the BSQLM. The multidomain method can be applied to either space or time domain but not both. MD-BSQLM can be used on equations with time $t > 1$. The domain is divided into nonoverlapping subintervals, and the equations are calculated by means of the Chebyshev spectral collocation method. The MD-BSQLM for NLPDE systems was solved by Magagula et al., [33]. It was established that the method entails nominal calculation time and that its precision does not quickly decline as the time domain is increased. Using MD-BSQLM, Oyelakin et al., [34] solved non-Darcian mixed convection movement in a power-law liquid by viscous dissipation, including the outcomes display that the numerical technique is simple to employ, rapidly converges, and yields precise results. The multidomain bivariate spectral local linearization method (MD-BSLLM) is an addition of the BSLLM that was advanced for computing systems of coupled nonsimilar boundary layer PDEs [35]. Magagula [35] utilized the MD-BSLLM to solve the systems of nonsimilar boundary layer PDEs. It was perceived that this method uses the least calculation time and that its precision does not degrade with time. The multistep methods are briefly described next.

1.2.3 Multistep methods

A broad linear multistep method is established for attaining an approximate solution of a class of useful differential equations comprehending both fixed and variable delays [36]. Linear multistep methods are used to solve ODEs numerically. A numerical technique begins at an initial point and then moves onward in time to discover the next result point. The procedure is repeated in order to map out the solution. To determine the modern value, single-step methods (such as Euler's method) speak of merely one previous point including its derivative. Runge-Kutta methods, for example, take some middle steps (for example, a half-step) to obtain a higher-order technique, even though it sheds all prior data before proceeding to the subsequent step. Multistep techniques try to improve effectiveness by retaining and utilizing data from existing steps rather than negating it. As a result, multistep method makes use of more than a few preceding points and derivatives. The ability to determine the local truncation error and contain a rectification term, which improves the accuracy of the solution at each step, is a required feature of a multistep method. The Adams-Bashforth method, the Adams-Moulton method, and the backward differentiation method are the three utmost common linear multistep methods [37]. The Adams-Bashforth method is an explicit method that only entails one function evaluation per time step [38]. However, Adams-Moulton method and backward differentiation formulas are implicit techniques. Predictor–corrector is attained by mingling Adams–Bashforth and Adams–Moulton methods [39]. The predictor-corrector technique is utilized to solve the implicit linear multistep technique, which has enhanced steadiness conditions than the explicit method [40]. Regardless of the method's success, the pressing drawback is that the predictors are in decreasing order of precision, so the technique does not offer a restored approximation, aside from the computational liability allied with the technique [40]. A brief overview of BHM follows.

1.2.4 Block-Hybrid methods

Hybrid block methods, as opposed to the Runge-Kutta method, are self-starting schemes that yield more accurate results because they are implemented as a block [41]. Block techniques have been established to achieve numerical results at further than one point at a time. The block method was suggested in 1953 as one of the efficient methods for improving the performance of numerical methods [42]. Some benefits of block methods consist of the ability to overcome overlapping pieces of results and the ability to self-start, which eliminates the need for other methods to obtain starting solutions [43]. Following that, [44] pioneered hybrid methods, which involve evaluating functions at off-step points. The use of hybrid (off-step) points in block techniques provides numerous benefits, in the same way as the capability to adjust step size, use information off-step points, and minimize the zero stability barrier condition [42]. Step and off-step node points are unified to structure a single block in the hybrid block strategy for solving ODEs [45]. Academics such as [46, 47] recommended a high-method derivative to overcome difficulty in ODEs in order to increase the precision of numerical approaches even further. They then proposed a second-derivative method of Simpson's type for solving an inflexible system of first-order initial value problems (IVPs). These academics inspired Alkasassbeh et al., [48] to create a new common three-hybrid one-step third-derivative implicit technique for solving second-order ODEs straight utilizing the method of interpolation also collocation for common use in order to increase the effectiveness of the approximate result. Alkasassbeh et al., [48] tried the implicit one-step block hybrid third-derivative method on initial value problems of second-order ODEs. In terms of precision, the numerical results confirm the recent technique's superiority over the prevailing methods. Following is a brief overview of analytical methods.

1.3 Analytical methods

As a result of the enlarged curiosity in these problems, a variety of exact techniques, for instance Hirota's bilinear technique, inverse scattering transform, Painlevé analysis, and so on, have been developed. Furthermore, numerical simulation with more influential computers has changed existing computation methods. On the other hand, those analytical techniques are often challenging to apply and need a detailed understanding of their properties and capabilities before being applied to the problem at hand [49]. This section discusses some analytical techniques for solving nonlinear PDEs. We go over their properties, benefits, and shortcomings.

1.3.1 The Tanh method

As a result of the growing concern in the discovery of exact solutions to those problems, a wide variety of analytical methods are now obtainable [50]. One of these techniques is the Tanh (or hyperbolic tangent). A Tanh technique is an effective tool for locating traveling waves that emerge from one-dimensional nonlinear wave and evolution equations. The Tanh method plays a vital part in problems involving dispersive effects, reaction, diffusion, and/or convection [49]. For an extensive range of nonlinear ODEs and PDEs, exact as well as approximate solutions can be found in a direct and efficient manner [51]. For extensiveness, we ought to indicate that this method is limited to the examination of traveling waves [52]. In consequence, in a moving frame of reference, we basically compact with one-dimensional shock waves (kink type) and solitary wave (pulse type) results.

1.3.2 G'/ G-expansion method

In engineering sciences, mathematical physics, and actual application fields, the G'/G-expansion technique is a simple and effective mathematical instrument for solving nonlinear evolution equations. The pioneering effort Wang et al., [53] proposed the G'/ G-expansion technique for unfailingly treating nonlinear wave equations. Numerous researchers have conducted additional research to launch the competence and firmness of the G'/ G-expansion technique and to expand its variety of applicability. Zhang et al., [54], for example, generalized the G'/G-expansion method for evolution equations with variable coefficients. Zhang et al., [55] also established an enhanced G'/ G-expansion technique for locating new wide-ranging traveling wave solutions. The application of these methods to the KdV equation exemplifies their originality and advantages [56]. The results demonstrate the efficacy of the suggested approach. Furthermore, these methods could be used to deal with advanced dimensional and higher-order nonlinear evolution equations, which are common in a lot of scientific actual application fields.

1.3.3 Lie group method

The Lie group technique is a potent and straight method for obtaining precise results for nonlinear differential equations [57]. The Lie group method can also produce other types of exact solutions for PDEs, for instance traveling wave solutions, soliton solutions, power series solutions, fundamental solutions, and many others [58]. To solve differential equations that arise as mathematical models in financial problems, Lie group theory is used. Lie group analysis is a mathematical concept which unifies the symmetry of differential equations [59]. Sophus Lie (1842-1899), the creator of this concept, was the first to categorize differential equations based on their symmetry groups in order to determine the set of equations that could be integrated or abridged to lower-order equations using group theoretical algorithms [60]. Methods for their integration, in particular, as well as the set arrangement of linear second-order PDEs with double self-reliant variables, were established. As stated by Lie's arrangement, entirely parabolic equations that admit the highest-order symmetry group reduces to the equation of heat conduction. The study and advancement of mathematics and physics are currently heavily reliant on the concept of symmetry. In fact, Lie group and Lie algebra theory is used in a variety of mathematical disciplines, together with differential geometry, algebraic topology, and bifurcation theory, to name a limited. Lie's unique theories had a significant impact on the study of physically significant systems of differential equations in classical and quantum mechanics, fluid dynamics, elasticity, and numerous other applicable fields [61,62]. Lie's infinitesimal transformation approach offers a technique for finding closed-form results for ODEs. Symmetries can be used to describe common solutions to first-order or linear ODEs. Lie was able to locate ODEs that can be entirely integrated using group theoretic methods or reduced to lower-order ones by categorizing them into groups. Lie's approach generates group-invariant solutions and conservation laws [63] to PDEs. Equivalence classes can be created by utilizing the symmetries of PDEs, allowing for the derivation of new solutions from existing sites. Application of Lie group techniques to concrete physical systems necessitates time-consuming computations. When performed by hand, even a simple system of differential equations' continuous symmetry group calculation is prone to errors. Computer algebra systems (CAS) for instance Mathematica and Maple can greatly assist with such calculations.

1.4 Aims and Objectives

The purpose of this research is to implement numerical methods, specifically the SQLM in space and the BHM in time to work out the nonlinear evolution problems. The objective is to analyze the results obtained by the BHM and SQLM and compare them with the MD-BSQLM in space and time when solving nonlinear evolution problems. These techniques will then be applied to work out the Burgers-Huxley equation, Burgers-Fisher equation, FitzHugh-Nagomu eqution. The accuracy, convergence rate, CPU time, and efficiency of all the methods are investigated by analyzing the errors.

1.5 Dissertation outline

The study is in the following order: In chapter 2, we convey the Lie group and numerical methods that are employed to solve the nonlinear problems. A brief literature review of the Burgers-Huxley equation is displayed in chapter 3. For demonstration purposes, we present the numerical solutions of the Burgers-Huxley equation utilizing the SQLM, MD-BSQLM, and the BHM. We also illustrate how Lie symmetries and steady state are used to reduce the Burgers-Huxley equation to an ODE. In chapter 4, an appraisal of the literature on evolution equations is given. These methods will be used to solve a family of evolution problems, specifically the Burgers-Fisher equation, the FitzHugh-Nagumo equation, and the coupled Burgers equation. In chapter 5, we present the results obtained from chapter 4 and discuss them. The conclusion of this dissertation's work is provided in Chapter 6.

Chapter 2

Lie group and numerical methods for solving NLPDEs

2.1 Lie group theory of PDEs

This unit offers a concise overview of the Lie group theory of PDEs. This will entail determining the Lie point symmetries of PDEs utilizing an algorithm.

2.1.1 Local one-parameter Lie group

In this context, a transformations refers to inverting transformations, i.e., a bijective map. Consider t as well as x to be self-governing variables , including u represent the reliant on variable. Examine the following modifications to variables t, x, and u:

$$R_c : \bar{t} = q(t, x, u, c), \ \bar{x} = p(t, x, u, c), \ \bar{u} = k(t, x, u, c), \tag{2.1}$$

with c being an actual limiting factor with values that vary continuously from a neighborhood $\mathcal{D}' \subset \mathcal{D} \subset \mathbb{R}$ of $c = 0$ as well as q, p including k are differentiable functions.

Definition 2.1 A *continuing one-parameter (local) Lie group of transformations* is a set Q of transformations (2.1) that complies with the three requirements listed below:

(i) For R_c, $R_d \in Q$ where $c, d \in \mathcal{D}' \subset \mathcal{D}$ then R_d, $R_c = R_c \in Q$, $c = \phi(c, d) \in \mathcal{D}$ (Closure).

(ii) $R_0 \in Q$ just incase $a = 0$ such that $R_0 \, R_c = R_c \, R_0 = R_c$ (uniqueness).

(iii) For $R_c \in Q$, $c \in \mathcal{D}' \subset \mathcal{D}$, $R_c^{-1} = R_{c^{-1}} \in Q$, $c^{-1} \in \mathcal{D}$ such that $R_c \, R_{c^{-1}} = R_{c^{-1}} \, R_c = R_0$ (Inverse).

We can discern from (i) that, the associativity property is satisfactory. Additionally, if the uniqueness transformation ensues at $c = c_0 \neq 0$ i.e, R_{c_0} is the uniqueness, then shifting the parameter $c = \bar{c} + c_0$ yields R_0 as per above. (i) is a property that may be expressed as

$$
\begin{aligned}
\bar{\bar{t}} &\equiv q(\bar{t}, \bar{x}, \bar{u}, d) = q(t, x, u, \phi(c, d)), \\
\bar{\bar{x}} &\equiv p(\bar{t}, \bar{x}, \bar{u}, d) = p(t, x, u, \phi(c, d)), \\
\bar{\bar{u}} &\equiv k(\bar{t}, \bar{x}, \bar{u}, d) = k(t, x, u, \phi(c, d)). \qquad (2.2)
\end{aligned}
$$

The function ϕ is identified as the *set composition law.* A set parameter c is termed *canonical* if $\phi(c, d) = c + d$.

Theorem 2.1 For every $\phi(c, d)$, there subsists the canonical parameter $\tilde{c}$ identified as

$$
\tilde{c} = \int_0^c \frac{db}{z(b)}, \text{ where } z(b) = \left. \frac{\partial \, \phi(b, d)}{\partial d} \right|_{d=0}.
$$

We now define a symmetry group for the second-order PDE

$$
u_t = H(t, x, u, u_x, u_{xx}), \quad \frac{\partial H}{\partial u_{xx}} \neq 0. \qquad (2.3)
$$

Definition 2.2 (Symmetry group) *symmetry group*

$$
\bar{u}_{\bar{t}} = H(\bar{t}, \bar{x}, \bar{u}, \bar{u}_{\bar{x}}, \bar{u}_{\bar{x}\bar{x}}), \qquad (2.4)
$$

where function H is similar as in Eq. (2.3).

2.1.2 Infinitesimal transformations

Conferring to Lie's concept, assembling a symmetry set Q is equal to determining the equidistant *infinitesimal transformations*:

$$\bar{t} \approx t + c\,\tau(t,x,u),\ \ \bar{x} \approx x + c\,\xi(t,x,u),\ \ \bar{u} \approx u + c\,\eta(t,x,u), \tag{2.5}$$

attained from Eq. (2.1) by extending the functions q, p including k into Taylor series in c about $c = 0$, additionally considering the first conditions

$$q\big|_{c=0} = t, \quad p\big|_{c=0} = x, \quad k\big|_{c=0} = u\,.$$

Hence, we obtain

$$\tau(t,x,u) = \frac{\partial q}{\partial c}\bigg|_{c=0}, \quad \xi(t,x,u) = \frac{\partial p}{\partial c}\bigg|_{c=0}, \quad \eta(t,x,u) = \frac{\partial k}{\partial c}\bigg|_{c=0}. \tag{2.6}$$

Now one can write (2.5) as

$$\bar{t} \approx (1 + c\,J)t, \quad \bar{x} \approx (1 + c\,J)x, \quad \bar{u} \approx (1 + c\,J)u\,,$$

where

$$J = \tau(t,x,u)\frac{\partial}{\partial t} + \xi(t,x,u)\frac{\partial}{\partial x} + \eta(t,x,u)\frac{\partial}{\partial u}\,. \tag{2.7}$$

This differential operator J is notorious as the *infinitesimal operator* of the set Q. If the set Q is permitted by Eq. (2.3), we say that J is an *permitted operator* of Eq. (2.3) alternatively J is an *infinitesimal symmetry* of Eq. (2.3).

2.1.3 Group (Set) invariants

Definition 2.3 function $H(t,x,u)$ is labeled an *invariant of set of transformation* (2.1) if

$$H(\bar{t},\bar{x},\bar{u}) \equiv H(q(t,x,u,c), p(t,x,u,c), k(t,x,u,c)) = H(t,x,u), \tag{2.8}$$

equal within t, x, u including c.

Theorem 2.2 (Infinitesimal criterion of invariance) A necessary and sufficient condition for function $H(t, x, u)$ to be an invariable is that

$$J\,H \equiv \tau(t, x, u)\frac{\partial H}{\partial t} + \xi(t, x, u)\frac{\partial H}{\partial x} + \eta(t, x, u)\frac{\partial H}{\partial u} = 0\,. \tag{2.9}$$

As of the above theorem, it follows that every one-parameter group of point transformations (2.1) has dual functionally independent invariants, which can be taken to be the left-hand side of any first integrals

$$K_1(t, x, u) = b_1, \quad K_2(t, x, u) = b_2,$$

of the characteristic equations

$$\frac{dt}{\tau(t, x, u)} = \frac{dx}{\xi(t, x, u)} = \frac{du}{\eta(t, x, u)}.$$

Theorem 2.3 Provided the infinitesimal transformation (2.5) alternatively its character J, the equivalent one-parameter set Q is attained by working out Lie equations

$$\frac{d\bar{t}}{dc} = \tau(\bar{t}, \bar{x}, \bar{u}), \quad \frac{d\bar{x}}{dc} = \xi(\bar{t}, \bar{x}, \bar{u}), \quad \frac{d\bar{u}}{dc} = \eta(\bar{t}, \bar{x}, \bar{u}), \tag{2.10}$$

subject to the initial conditions

$$\bar{t}\big|_{c=0} = t, \quad \bar{x}\big|_{c=0} = x, \quad \bar{u}\big|_{c=0} = u\,.$$

2.1.4 Symmetry set construction

Procedure for determining a symmetry set for a specified PDE is described here, but first, a handful of denotations are provided.

2.1.4.1 Prolongation of point transformations

Consider a PDE of second order

$$B(t, x, u, u_t, u_x, u_{tt}, u_{xx}, u_{tx}) = 0, \tag{2.11}$$

where t and x are two independent variables and u is a dependent variable. Let

$$J = \tau(t, x, u)\frac{\partial}{\partial t} + \xi(t, x, u)\frac{\partial}{\partial x} + \eta(t, x, u)\frac{\partial}{\partial u}, \tag{2.12}$$

be the infinitesimal generator of the one-parameter group Q of transformation (2.1). The *first prolongation* of the operator J is implied by $J^{[1]}$ and is given by

$$J^{[1]} = J + \zeta_1(t, x, u, u_t, u_x)\frac{\partial}{\partial u_t} + \zeta_2(t, x, u, u_t, u_x)\frac{\partial}{\partial u_x},$$

where

$$\begin{aligned}
\zeta_1 &= E_t(\eta) - u_t E_t(\tau) - u_x E_t(\xi), \\
\zeta_2 &= E_x(\eta) - u_t E_x(\tau) - u_x E_x(\xi),
\end{aligned}$$

and the total derivatives E_t and E_x are given by

$$E_t = \frac{\partial}{\partial t} + u_t\frac{\partial}{\partial u} + u_{tx}\frac{\partial}{\partial u_x} + u_{tt}\frac{\partial}{\partial u_t} + \cdots, \tag{2.13}$$

$$E_x = \frac{\partial}{\partial x} + u_x\frac{\partial}{\partial u} + u_{xx}\frac{\partial}{\partial u_x} + u_{tx}\frac{\partial}{\partial u_t} + \cdots. \tag{2.14}$$

A Lie algebra is a vector space T of operators as an accessory to the property described below: For all $J_1, J_2 \in T$, the commutator $[J_1, J_2] \in T$.

$$J^{[2]} = J + \zeta_1\frac{\partial}{\partial u_t} + \zeta_2\frac{\partial}{\partial u_x} + \zeta_{11}\frac{\partial}{\partial u_{tt}} + \zeta_{12}\frac{\partial}{\partial u_{tx}} + \zeta_{22}\frac{\partial}{\partial u_{xx}}. \tag{2.15}$$

where

$$\begin{aligned}
\zeta_{11} &= E_t(\zeta_1) - u_{tt}E_t(\tau) - u_{tx}E_t(\xi), \\
\zeta_{12} &= E_x(\zeta_1) - u_{tt}E_x(\tau) - u_{tx}E_x(\xi), \\
\zeta_{22} &= E_x(\zeta_2) - u_{tx}E_x(\tau) - u_{xx}E_x(\xi).
\end{aligned}$$

Using the above definitions of E_t and E_x, we get

$$\zeta_1 = \eta_t + u_t\eta_u - u_t\tau_t - u_t^2\tau_u - u_x\xi_t - u_tu_x\xi_u. \tag{2.16}$$

$$\zeta_2 = \eta_x + u_x\eta_u - u_t\tau_x - u_tu_x\tau_u - u_x\xi_x - u_x^2\xi_u. \tag{2.17}$$

$$\begin{aligned}\zeta_{11} = {} & \eta_{tt} + 2u_t\eta_{tu} + u_{tt}\eta_u + (u_t)^2\eta_{uu} - 2u_{tt}\tau_t - u_t\tau_{tt} - 2(u_t)^2\tau_{tu} \\ & -3u_tu_{tt}\tau_u - (u_t)^3\tau_{uu} - 2u_{tx}\xi_t - u_x\xi_{tt} - 2u_tu_x\xi_{tu} - (u_t)^2u_x\xi_{uu} \\ & -(u_xu_{tt} + 2u_tu_{tx})\xi_u.\end{aligned} \tag{2.18}$$

$$\begin{aligned}\zeta_{12} = {} & \eta_{tx} + u_x\eta_{tu} + u_t\eta_{xu} + u_{xt}\eta_u + u_tu_x\eta_{uu} - u_{tx}(\tau_t + \xi_x) - u_t\tau_{tx} - u_{tt}\tau_x \\ & -u_tu_x(\tau_{tu} + \xi_{xu}) - u_t^2\tau_{xu} - (2u_tu_{tx} + u_xu_{tt})\tau_u - (u_t)^2u_x\tau_{uu} - u_x\xi_{tx} \\ & -u_{xx}\xi_t - (u_x)^2\xi_{tu} - (2u_xu_{tx} + u_tu_{xx})\xi_u - u_t(u_x)^2\xi_{uu}.\end{aligned} \tag{2.19}$$

$$\begin{aligned}\zeta_{22} = {} & \eta_{xx} + 2u_x\eta_{xu} + u_{xx}\eta_u + (u_x)^2\eta_{uu} - 2u_{xx}\xi_x - u_x\xi_{xx} - 2(u_x)^2\xi_{xu} \\ & -3u_xu_{xx}\xi_u - (u_x)^3\xi_{uu} - 2u_{tx}\tau_x - u_t\tau_{xx} \\ & -2u_tu_x\tau_{xu} - (u_tu_{xx} + 2u_xu_{tx})\tau_u - u_t(u_x)^2\tau_{uu}.\end{aligned} \tag{2.20}$$

2.1.4.2 Group (set) admitted by a PDE

Operator

$$J = \tau(t,x,u)\frac{\partial}{\partial t} + \xi(t,x,u)\frac{\partial}{\partial x} + \eta(t,x,u)\frac{\partial}{\partial u}, \tag{2.21}$$

is a *symmetry* of second-order PDE

$$B(t,x,u,u_t,u_x,u_{tt},u_{tx},u_{xx}) = 0, \tag{2.22}$$

if

$$J^{[2]}(B) = 0, \tag{2.23}$$

whenever $B = 0$. This is equivalently known as the "symmetry condition"

$$J^{[2]}\,B\big|_{B=0} = 0, \tag{2.24}$$

hence the character $|_{B=0}$ means determined on the equation $B = 0$.

Definition 2.4 Equation (2.24) describes the *determining equation* of (2.22), since it determines each and every one of the infinitesimal symmetries of (2.22).

2.1.5 Lie algebras

Consider the subsequent two operators J_1 and J_2 defined by

$$J_1 = \tau_1(t,x,u)\frac{\partial}{\partial t} + \xi_1(t,x,u)\frac{\partial}{\partial x} + \eta_1(t,x,u)\frac{\partial}{\partial u},$$

and

$$J_2 = \tau_2(t,x,u)\frac{\partial}{\partial t} + \xi_2(t,x,u)\frac{\partial}{\partial x} + \eta_2(t,x,u)\frac{\partial}{\partial u}.$$

Definition 2.5 (Commutator) The *commutator* of J_1 and J_2, written as $[J_1, J_2]$, is defined by $[J_1, J_2] = J_1(J_2) - J_2(J_1)$.

Definition 2.6 (Lie algebra) A Lie algebra is a vector space T of operators with the property described below: For all $J_1, J_2 \in T$, the commutator $[J_1, J_2] \in T$.

The dimension of a Lie algebra is the vector space T dimension .

Theorem 2.4 The set of all solutions of any determining equation forms a Lie algebra.

2.2 Numerical methods

2.2.1 Spectral methods

The spectral methods use a shortened series of orthogonal functions to approximate various functions [64]. The Fourier series, which is employed to solve periodic problems, Chebyshev, Legendre polynomials, etc. are examples of such orthogonal functions. One of the key benefits of spectral methods is the efficiency of convergence. Compared to comparable numerical techniques for instance finite differences and finite elements, they are considerably more accurate. In the 1970s, the methods earned recognition [65, 66].

The Chebyshev spectral method uses reduced series of Chebyshev orthogonal polynomials to approximate functions. The Chebyshev polynomials $P_{N_x}(x)$ of order N_x are given by

$$P_{N_x}(x) = \cos(N_x \cos^{-1}(x)), \;\; N_x \in \mathbb{N}. \tag{2.25}$$

The Chebyshev interpolation $u_{N_x}(x)$ of a function $u(x)$ at $x = x_i$ is expressed by

$$u_{N_x}(x) = \sum_{i=0}^{N_x} u(x_i) M_i(x), \tag{2.26}$$

where the collocation points x_i are selected to be the extrema of $P_{N_x}(x)$:

$$x_i = \left\{ \cos\left(\frac{\pi i}{N_x} \right) \right\}_{i=0}^{N_x}, \tag{2.27}$$

that are the Chebyshev-Gauss-Lobatto points. $M_i(x), \; i = 0, 1, ..., N_x$ are Lagrange polynomials of order N_x based on the Chebyshev-Gauss- Lobatto points described as

$$M_i(x) = \frac{(-1)^{i+1}(1 - x^2) P'_{N_x}(x)}{\bar{b}_i N_x^2 (x - x_i)}, \;\; i = 0, 1, \ldots, N_x \tag{2.28}$$

where $\bar{b}_0 = \bar{b}_{N_x} = 2, \bar{b}_i = 1$ for $i = 1, 2, \ldots, N_x - 1$. The approximate result at the collocation points is approximated by the derivatives, which are calculated as

$$\frac{d^n}{dx^n} u(x_i) = \sum_{j=0}^{N_x} u(x_i) M^{(n)}(x_j) = \sum_{j=0}^{N_x} D_{i\,j}^{(n)} u(x_i), \tag{2.29}$$

where $D_{i\,j}^{(n)} = M^{(n)}(x_i)$ is an $(N_x + 1)(N_x + 1)$ Chebyshev differentiation matrix for $i, j = 0, 1, \ldots, N_x$. At the collocation points, the initial sequence Chebyshev derivative matrix is given by [65, 66]

$$D_{i\,j} = D_{i\,j}^{(1)} = \begin{cases} \frac{c_i(-1)^{j+i}}{c_j(x_j - x_i)}, & i \neq j, \\ -\frac{x_i}{2(1 - x_i^2)}, & (i = j) \neq 0, \; N_x, \\ \frac{2N_x^2 + 1}{6}, & i, \; j = 0, \\ -\frac{2N_x^2 + 1}{6}, & i, \; j = N_x. \end{cases} \tag{2.30}$$

Adding PDEs to the spectral collocation concept (Bivariate Chebyshev spectral method), Chebyshev polynomials $P_{N_x}(\xi)$ of sequence N_x are defined as

$$P_{N_x}(\xi) = \cos(N_x \cos^{-1}(\xi)),\ \ N_x \in \mathbb{N}. \tag{2.31}$$

Chebyshev interpolation $u_{N_x}(\xi, \tau)$ of function $u(\xi, \tau)$ at $\xi = \xi_i$ and $\tau = \tau_j$ is defined by

$$u_{N_x}(\xi, \tau) = \sum_{j=0}^{N_t} \sum_{I=0}^{N_x} u_{j\,i} M_j(\tau) M_i(\xi), \tag{2.32}$$

whilst the collocation points ξ_i and τ_j are selected to be extrema of P_{N_x}:

$$\xi_i = \left\{ \cos\left(\frac{\pi i}{N_x}\right) \right\}_{i=0}^{N_x},\ \ \tau_j = \left\{ \cos\left(\frac{\pi i}{N_t}\right) \right\}_{j=0}^{N_t}, \tag{2.33}$$

which are the Chebyshev-Gauss-Lobatto points. $M_i(\xi), i = 0, 1, \ldots, N_x,$ are Lagrange polynomials of arrangement N_x based on the Chebyshev-Gauss-Lobatto points described as

$$M_i(\xi) = \frac{(-1)^{i+1}(1-\xi^2)P'_{N_x}(\xi)}{\bar{b}_i N_x^2 (\xi - \xi_i)},\ \ i = 0, 1, \ldots, N_x, \tag{2.34}$$

where $\bar{b}_0 = \bar{b}_{N_x} = 2, \bar{b}_i = 1$ for $i = 1, 2, \ldots, N_x - 1$. The polynomials $M_j(\tau)$ are stated similarly.

The approximate solution's nth sequence time and space derivatives at the collocation points are calculated as

$$\frac{\partial^n}{\partial \xi^n} u(\xi_i, \tau_j) = \sum_{q=0}^{N_t} \sum_{p=0}^{N_x} u(\xi_p, \tau_q) M_q(\tau_j) \frac{d^n M_p(\xi_i)}{d\xi^n} = \sum_{p=0}^{N_x} D^n_{i\,p} u(\xi_p, \tau_j) = \mathbf{D}^n \mathbf{U}_j, \tag{2.35}$$

$$\frac{\partial^n}{\partial \tau^n} u(\xi_i, \tau_j) = \sum_{q=0}^{N_t} \sum_{p=0}^{N_x} u(\xi_p, \tau_q) \frac{d^n M_q(\tau_j)}{d\tau^n} M_p(\xi_i) = \sum_{q=0}^{N_t} d^n_{j\,q} u(\xi_i, \tau_q), \tag{2.36}$$

where $D_{i\,p} = \frac{dM_p(\xi_i)}{d\xi}$ is an $(N_x + 1)(N_x + 1)$ Chebyshev differentiation matrix for $i, p = 0, 1, \ldots, N_x$, and $d_{j\,q} = \frac{dM_q(\tau_j)}{d\tau}$ is an $(N_t + 1)(N_t + 1)$ Chebyshev differentiation matrix for $j, q = 0, 1, \ldots, N_t$.

The collocation points' original sequence Chebyshev derivative matrix is given by [65, 66]

$$D_{ip} = D^{(1)}_{ip} = \begin{cases} \frac{c_i(-1)^{p+i}}{c_p(\xi_p-\xi_i)}, & i \neq p, \\ -\frac{\xi_i}{2(1-\xi_i^2)}, & (i = p) \neq 0,\ N_x, \\ \frac{2N_x^2+1}{6}, & i,\ p = 0, \\ -\frac{2N_x^2+1}{6}, & i,\ p = N_x. \end{cases} \tag{2.37}$$

The first order Chebyshev differentiation matrix d_{jq} can be calculated in a similar manner.

2.2.2 Spectral quasi-linearization method

We detail the algorithm in this subsection to exemplify how the spectral quasi-linearization method can be utilized to resolve nonlinear evolution PDEs. We examine the nonlinear PDE of the form

$$\frac{\partial u}{\partial t} = G\Big(u, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}, \dots, \frac{\partial^n u}{\partial x^n}\Big),\ (x, t) \in [a, b] \times [0, T], \tag{2.38}$$

where n is the order of the PDE, $u(x, t)$ is problem variable, also G is a non-linear operator that hold any of u's spatial derivatives. The region $t \in [0, T]$ is transformed into $\tau \in [-1, 1]$ by utilizing the linear transformation $t = T(\tau + 1)/2$, as well as $x \in [a, b]$ is transformed into $\xi \in [-1, 1]$ by utilizing the linear transformation $x = \frac{1}{2}(b - a)\xi + \frac{1}{2}(b + a)$.

Eq. (2.38) becomes

$$\frac{\partial u}{\partial \tau} = G\Big(u, \frac{\partial u}{\partial \xi}, \frac{\partial^2 u}{\partial \xi^2}, \dots, \frac{\partial^n u}{\partial \xi^n}\Big),\ (\xi, \tau) \in [-1, 1] \times [-1, 1]. \tag{2.39}$$

2.2.2.1 Quasi-linearization method

Bellman and Kalaba's quasilinearization (QLM) [21] is an approach that is utilized to simplify nonlinear ODEs and PDEs. The Newton-Raphson method is utilized to establish the QLM, which is composed of linear expressions of Taylor series about a first approach to the result. The QLM presumes that the variation in solution among two consecutive iterations is very small. The QLM determines the approximate result of the linear equation at $(s+1)$

$$G\Big(u_s, u_s', u_s'', \ldots, u_s^{(n)}\Big) + \sum_{p=0}^{n} \frac{\partial G}{\partial u^p}\Big(u_s, u_s', u_s'', \ldots, u_s^{n}\Big)\Big(u_{s+1}^{(p)} - u_s^{(p)}\Big) = 0, \tag{2.40}$$

whilst the Chebyshev spectral collocation method is employed to resolve the now linear Eq. (2.40), it can be written as

$$a_{0,s}(x,t)u_{s+1}^{(n)} + a_{1,s}(x,t)u_{s+1}^{(n-1)} + \ldots, a_{n-1,s}(x,t)u_{s+1}^{'} + a_{n,s}(x,t)u_{s+1} = R_r(x,t), \tag{2.41}$$

where

$$a_{p,s}(x,t) = \frac{\partial G}{\partial u_s^{(p)}}\Big(u_s, u_s', u_s'', \ldots, u_s^{(n)}\Big),\ p = 0, 1, \ldots, n, \tag{2.42}$$

and

$$R_s(x,t) = \sum_{p=0}^{n} a_{p,s}(x,t)u_s^{(n)} - G\Big(u_s, u_s', u_s'', \ldots, u_s^{(n)}\Big). \tag{2.43}$$

2.2.2.2 Bivariate spectral quasi-linearization method

This unit introduces the BSQLM for solving nonlinear PDE solutions. We look at NLPDEs of the form Eq. (2.39). The process of solving the problem presume that the result can be approached by a Chebyshev interpolation of the structure Eq. (2.32), interpolates at $u(\xi,\tau)$ at certain points in ξ and τ directions expressed by Eq. (2.33). The option of Chebyshev-Gauss-Lobatto grid terms Eq. (2.33), certifies the continuing derivatives, in space and time, is easily converted to discrete derivatives at the grid terms. Pre-conveying Eq. (2.39) in linear structure, it is appropriate to divide G to become linear and nonlinear operators also rephrase the equation in the structure,

$$T[u, u', \ldots, u^{(n)}] + K[u, u', \ldots, u^{(n)}] - \dot{u} = 0, \tag{2.44}$$

providing that dot and prime represent the time and space derivatives, respectively, T is a linear operator, and K is a non-linear operator. Supposing the variation $u_{s+1} - u_s$ including each and every one of its space derivatives are minor, we start by approximating the nonlinear operator K with the linear expressions of the Taylor series. Therefore

$$K[u, u', \ldots, u^{(n)}] \approx K[u, u', \ldots, u^{(n)}] + \sum_{p=0}^{n} \frac{\partial K}{\partial u^{(p)}}(u_{s+1}^{(p)} - u_s^{(p)}), \tag{2.45}$$

where s and $s+1$ signify the preceding and the recent iterations, respectively. Eq. (2.45) can be stated as

$$\begin{aligned} K[u, u', \ldots, u^{(n)}] \approx & K[u, u', \ldots, u^{(n)}] + \sum_{p=0}^{n} a_{p,s}[u, u', \ldots, u^{(n)}] u_{s+1}^{(p)} \\ & - a_{p,s}[u, u', \ldots, u^{(n)}] u_s^{(p)}, \end{aligned} \tag{2.46}$$

where

$$a_{p,s}[u, u', \ldots, u^{(n)}] = \frac{\partial K}{\partial u^{(p)}}[u, u', \ldots, u^{(n)}]. \tag{2.47}$$

Substituting equation (2.46) into (2.44), we get

$$T[u_{s+1}, u'_{s+1}, \ldots, u_{s+1}^{(n)}] + \sum_{p=0}^{n} a_{p,s} u_{s+1}^{(p)} - \dot{u}_{s+1} = R_r[u_s, u'_s, \ldots, u_s^{(n)}], \tag{2.48}$$

where

$$R_r[u_s, u'_s, \ldots, u_s^{(n)}] = \sum_{p=0}^{n} a_{p,s} u_s^{(p)} - G[u_s, u'_s, \ldots, u_s^{(n)}].$$

The time derivative values at the Chebyshev-Gauss-Lobatto terms x_i, t_j, are computed utilizing Eq. (2.36) for $j = 0, 1, \ldots, N_t$ and values of space derivatives at chebyshev-Gauss-Lobatto terms x_i, t_j, are calculated employing Eq. (2.35) for $i = 0, 1, \ldots, N_x$. Substituting Eq. (2.35) into Eq. (2.48) we attain

$$T[\mathbf{U}_{s+1,j},\mathbf{U}'_{s+1,j},\ldots,\mathbf{U}^{(n)}_{s+1,j}]+\sum_{p=0}^{n}\mathbf{a}_{p,s}\mathbf{U}^{(p)}_{s+1,j}-\sum_{p=0}^{N_t}d_{j\,p}\mathbf{U}_{s+1,p} \\ =R_r[\mathbf{U}_{s,j},\mathbf{U}'_{s,j},\ldots,\mathbf{U}^{(n)}_{s+,j}], \tag{2.49}$$

for $j=0,1,2,\ldots,N_t$, where

$$\mathbf{U}^{(n)}_{s+1,j}=\mathbf{D}^n\mathbf{U}_{s+1,j},\ \ \mathbf{a}_{p,s}=\begin{bmatrix} a_{p,s}(x_0,t_j) & & & \\ & a_{p,s}(x_1,t_j) & & \\ & & \ddots & \\ & & & a_{p,s}(x_{N_x},t_j)\end{bmatrix}. \tag{2.50}$$

The initial condition for Eq. (2.39) is equivalent to $\tau_{N_t}=-1$ as a result, we articulate Eq. (2.49) as

$$T[\mathbf{U}_{s+1,j},\mathbf{U}'_{s+1,j},\ldots,\mathbf{U}^{(n)}_{s+1,j}]+\sum_{p=0}^{n}\mathbf{a}_{p,s}\mathbf{U}^{(p)}_{s+1,j}-\sum_{p=0}^{N_t-1}d_{j\,p}\mathbf{U}_{s+1,p}=R_j, \tag{2.51}$$

where

$$R_j=R_r[\mathbf{U}_{s,j},\mathbf{U}'_{s,j},\ldots,\mathbf{U}^{(n)}_{s,j}]+d_{j\,N_t}\mathbf{U}_{N_t},\ \ j=0,1,\ldots,N_t-1.$$

Eq. (2.51) could be conveyed as the subsequent $N_t(N_x+1)\times N_t(N_x+1)$ matrix system

$$\begin{bmatrix} B_{0,0} & B_{0,1} & \ldots & B_{0,N_t-1} \\ B_{0,0} & B_{0,1} & \ldots & B_{0,N_t-1} \\ \vdots & \vdots & \ddots & \vdots \\ B_{N_t-1,0} & B_{N_t-1,1} & \vdots & B_{N_t-1,N_t-1}\end{bmatrix}\begin{bmatrix}\mathbf{U}_0\\ \mathbf{U}_1\\ \vdots\\ \mathbf{U}_{N_t-1}\end{bmatrix}=\begin{bmatrix}\mathbf{R}_0\\ \mathbf{R}_1\\ \vdots\\ \mathbf{R}_{N_t-1}\end{bmatrix}, \tag{2.52}$$

where

$$\mathbf{B}_{i,i}=T[\mathbf{I},\mathbf{D},\ldots\mathbf{D}^n]+\sum_{p=0}^{n}\mathbf{a}_{p,s}\mathbf{D}^{(p)}-d_{i,i}\mathbf{I}, \tag{2.53}$$

$$\mathbf{B}_{i,j}=-d_{i,j}\mathbf{I},\ \text{where } i\neq j, \tag{2.54}$$

with $\mathbf{I}$ being the identity matrix size $(N_x+1)\times(N_x+1)$. Desolving Eq. (2.52) produces the approaching value of $u(x,t)$.

2.2.2.3 Multi-domain approach

Multi-domain methods permit preferable-conditioned matrices with bigger step sizes apart from single-domain methods [67]. Another advantage of using a multiple-domain approach is that they improve approximation precision significantly. The problem is solved in each sub-domain using such methods, that also divide the particular domain into two or more sub-domains as well as connect the solutions across the sub-domain boundaries using the appropriate interface conditions. We begin by breaking down the integration interval $\Omega = [0, T]$ into intervals that are not overlapping $\Omega_k = [t_{k-1}, t_k]$ where $k = 1, 2, 3, ..., p$, whereas $t_0 = 0$ with $t_p = T$. The foremost notion behind this technique is to determine the result of the differential equation singly on every sub-interval, in succession, starting with the original condition [68]. The known original condition is utilized to calculate the result in the starting sub-interval and is regarded as the time interval's left boundary. The calculated result at the right side boundary of the starting interval is utilized as original condition in the following sub-interval to compute the result in the following sub-interval. Procedure continues pending the final sub-interval being reached. The procedure of matching results from different intervals along their typical boundary is identified as patching. First, we decompose the interval $0 \leq t \leq T$ before resolving the linearized form of the nonlinear evolution equation per non-overlapping intervals described as

$$\Gamma_k = [t_{k-1}, t_k],\ \ k = 1, 2, \ldots, p,$$

where

$$0 = t_0 \leq t_1 \leq t_2 \leq \ldots \leq t_p = T.$$

This method's objective is to resolve u in Eq. (2.38) within every sub-interval. The first condition is employed to find the result in the original sub-interval $[t_0, t_1]$. Subsequently, we utilize the continuity condition within neighboring sub-intervals to determine the final condition for the following intervals. As a result, the result at the end of every interval gets to be the starting point for the succeeding interval. Patching condition necessitates that

$$u^{(k)}(x, t_{k-1}) = u^{(k-1)}(x, t_{k-1}), x \in [a, b], \tag{2.55}$$

whereas $u^{(k)}(x,t)$ signifies result of Eq. (2.38) for every sub-interval Γ_k and $1 \leq k \leq p$. The domain $t = [t_{k-1}, t_k]$ is changed into $\tau \in [-1,1]$ utilizing linear transformation

$$t = \frac{1}{2}(t_k - t_{k-1})\tau + \frac{1}{2}(t_k + t_{k-1}), \tag{2.56}$$

and similarly the spatial domain $x \in [0,1]$ is changed to $\eta \in [-1,1]$ utilizing linear transformation

$$x = \frac{1}{2}(b-a)\eta + \frac{1}{2}(b+a). \tag{2.57}$$

Thus, in every sub-interval, we are obliged to work out

$$\begin{aligned} &a_{0,s}(x,t)u''^{(k)}_{s+1}(x,t) + a_{1,s}(x,t)u'^{(k)}_{s+1}(x,t) + a_{2,s}(x,t)u^{(k)}_{s+1}(x,t) - \dot{u}^{(k)}_{s+1} \\ &= R^{(k)}_s(x,t), \;\; x \in [a,b], t \in [t_{k-1}, t_k], \end{aligned} \tag{2.58}$$

subjected to

$$u^{(k)}(x, t_{k-1}) = u^{(k-1)}(x, t_{k-1}). \tag{2.59}$$

2.3 Block-Hybrid method for PDEs

The development of block hybrid linear multi-step methods with off-step points for the solution of nonlinear differential equation systems is considered. The extra off-step points improve method accuracy while also ensuring consistency, zero-stability, and convergence. The hybrid method's precision is increased by including more off-step points at the same time keeping the grid size consistent. These methods have an advantage over traditional methods such as Runge-Kutta methods due to their flexibility. Our aim is to approximate the exact solution of general second-order PDE In the nature of,

$$\dot{y} = f(x, t, y, y', y''), \tag{2.60}$$

where the dot denotes the differentiation with respect to time and primes signify differentiation with respect to space variable. Eq. (2.60) is to be solved subject to the intial and boundary conditions

$$y(x,0) = y_0(x), \;\; y(a,t) = y_a(t) \;\; y(b,t) = y_b(t),$$

where $y_0(x)$, $y_a(t)$ and $y_b(t)$ are known functions. The PDE Eq. (2.60) is solved using the BHM in time t and the spectral collocation method in space x following the utilization of a suitable linear transformation to transform $x \in [a, b]$ to $\xi \in [-1, 1]$. To exemplify the spectral method's application, consider linear PDE, which is expressed in general for second order as follows,

$$f(x, t, y, y', y'') = q_2(x, t)y''(x, t) + q_1(x, t)y'(x, t) + q_0(x, t)y(x, t)q + g(x, t),$$

where $g(x, t)$ and $q_k(x, t)$ for $k = 0, 1, 2$ are known fuctions. NLPDEs are linearised before the spectral method is applied. Applying the spectral method with $N_x + 1$ collocation points on (2.60) gives,

$$\dot{\mathbf{y}} = f(t, \mathbf{y}),\ \ \mathbf{y}(0) = \mathbf{y}_0, \tag{2.61}$$

with boundary conditions given by

$$y(\xi_{N_x}, t) = y_a(t),\ \ y(\xi_0, t) = y_b(t), \tag{2.62}$$

and

$$f(t, (y)) = [\mathbf{Q}_2(t)\mathbf{D}^2 + \mathbf{Q}_1(t)\mathbf{D} + \mathbf{Q}_0(t)]\mathbf{y} + \mathbf{G}(\mathbf{t}). \tag{2.63}$$

In the above equations, $\mathbf{D} = 2D/(b - a)$, where D is the chebyshev differentiation matrix including the matrices and vectors are

$$\mathbf{Q}_k = \begin{bmatrix} q_k(\xi_0, t) & & & \\ & q_k(\xi_1, t) & & \\ & & \ddots & \\ & & & q_k(\xi_{N_x}, t) \end{bmatrix},$$

$\mathbf{y} = [y(\xi_0, t), y(\xi_1, t), \ldots, y(\xi_{N_x}, t)]^T$, $\mathbf{G} = [g(\xi_0, t), g(\xi_1, t), \ldots, g(\xi_{N_x}, t)]^T$.

2.3.1 Development of the Method

The continuous method works by approximating the analytical solution $y(t)$ of the non-linear differential equation of Eq. (2.60) by

$$y(t) \approx Y(t) = \sum_{i=0}^{M+1} c_{n,i}(t - t_n)^i, \tag{2.64}$$

where $t \in [a, b]$, $c_{n,i}$ are unknown coefficients to be determined and M is the number of collocation points. The continuous approximation is established by enforcing the following conditions:

$$\begin{aligned}
\dot{Y}_{n+p_i} &= f(t_{n+p_i}, y_{n+p_i}),\ i = 0, 1, 2, \dots M, \\
Y(t_n) &= c_{n,0} = y_n,\ n = 0, 1, \dots N-1,
\end{aligned} \tag{2.65}$$

where the dot points out differentiation with respect to time t. Working out the equations that emerge in Eq. (2.65), gives $c_{n,0} = y_n$ and $c_{n,1} = f_n$ for all values of M. The coefficients $c_{n,k}$ for $k \geq 2$ are

when $M = 2$,

$$c_{n,2} = -\frac{3f_n + f_{n+1} - 4f_{n+\frac{1}{2}}}{2h}, \quad c_{n,3} = \frac{2(f_n + f_{n+1} - 2f_{n+\frac{1}{2}})}{3h^2}.$$

When $M = 3$

$$\begin{aligned}
c_{n,2} &= -\frac{11f_n - 2f_{n+1} - 18f_{n+\frac{1}{3}} + 9f_{n+\frac{2}{3}}}{4h}, \\
c_{n,3} &= \frac{3(2f_n - f_{n+1} - 5f_{n+\frac{1}{3}} + 4f_{n+\frac{2}{3}})}{2h^2}, \\
c_{n,4} &= -\frac{9(2f_n - f_{n+1} - 3f_{n+\frac{1}{3}} + 3f_{n+\frac{2}{3}})}{2h^2}.
\end{aligned}$$

When $M = 4$

$$\begin{aligned}
c_{n,2} &= -\frac{25f_n + 3f_{n+1} + 36f_{n+\frac{1}{2}} - 48f_{n+\frac{1}{4}} - 16f_{n+\frac{3}{4}}}{6h}, \\
c_{n,3} &= \frac{2(35f_n + 11f_{n+1} + 114f_{n+\frac{1}{2}} - 104f_{n+\frac{1}{4}} - 56f_{n+\frac{3}{4}})}{9h^2}, \\
c_{n,4} &= -\frac{2(5f_n + 3f_{n+1} + 24f_{n+\frac{1}{2}} - 18f_{n+\frac{1}{4}} - 14f_{n+\frac{3}{4}})}{3h^3}, \\
c_{n,5} &= \frac{32(f_n + f_{n+1} + 6f_{n+\frac{1}{2}} - 4f_{n+\frac{1}{4}} - 4f_{n+\frac{3}{4}})}{15h^{14}}.
\end{aligned}$$

The BHM equations are then obtained by subtituting the solutions for $c_{n,k}$ in the continuous approximation $Y(t)$ and evaluating the result at the collocation points t_{n+p_i} for $i = 1, 2 \ldots, M$. That is,

$$y_{n+p_i} = Y(t_{n+p_i}), \quad i = 1, 2, \ldots, M,$$

for equally spaced nodes $(0, \frac{1}{2}, 1)$ with $M = 2$, we obtain

$$\begin{aligned}
y_{n+\frac{1}{2}} &= y_n + \frac{1}{24}h(5f_n - f_{n+1} + 8f_{n+\frac{1}{2}}),\\
y_{n+1} &= y_n + \frac{1}{6}h(f_n + f_{n+1} + 4f_{n+\frac{1}{2}}).
\end{aligned}$$

When $M = 2$ with equally spaced nodes $(0, \frac{1}{3}, \frac{2}{3}, 1)$, we attained

$$\begin{aligned}
y_{n+\frac{1}{3}} &= y_n + \frac{1}{72}h(9f_n + f_{n+1} + 19f_{n+\frac{1}{3}} - 5f_{n+\frac{2}{3}}),\\
y_{n+\frac{2}{3}} &= y_n + \frac{1}{9}h(f_n + 4f_{n+\frac{1}{3}} + f_{n+\frac{2}{3}}),\\
y_{n+1} &= y_n + \frac{1}{8}h(f_n + f_{n+1} + 3f_{n+\frac{1}{3}} + 3f_{n+\frac{2}{3}}).
\end{aligned}$$

Applying the BHM on the first order Eq. (2.61) gives

$$\mathbf{y}_{n+p_i} = \mathbf{y}_n + h\sum_{j=1}^{M}[\alpha_{i,j}\mathbf{f}_{n+p_j} + \beta_{i,j}\mathbf{f}_n], \; i = 1, 2, \ldots, M. \tag{2.66}$$

Where $\mathbf{f}_{n+p_j} = f(t_{n+p_j}, \mathbf{y}_{n+p_j})$ and $\mathbf{f}_n = f(t_n, \mathbf{y}_n)$. For general linear PDEs, spectral method decomposed scheme Eq. (2.66) can be expressed as

$$\mathbf{y}_{n+p_i} = \mathbf{y}_n + h\sum_{j=1}^{M}[\alpha_{i,j}(\mathbf{\Phi}_{n+p_j}\mathbf{y}_{n+p_j} + \mathbf{\Psi}_{n+p_j}) + \beta_{i,j}\mathbf{f}_n], \; i = 1, 2, \ldots, M, \tag{2.67}$$

which in matrix form becomes

$$\begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,M} \\ A_{2,1} & A_{2,2} & \dots & A_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ A_{M,1} & A_{M,2} & \dots & A_{M,M} \end{pmatrix} \begin{pmatrix} \mathbf{y}_{n+p_1} \\ \mathbf{y}_{n+p_2} \\ \vdots \\ \mathbf{y}_{n+p_M} \end{pmatrix} = \begin{pmatrix} \mathbf{y}_n \\ \mathbf{y}_n \\ \vdots \\ \mathbf{y}_n \end{pmatrix} + \begin{pmatrix} B_{1,1} & B_{1,2} & \dots & B_{1,M} \\ B_{2,1} & B_{2,2} & \dots & B_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ B_{M,1} & B_{M,2} & \dots & B_{M,M} \end{pmatrix} \begin{pmatrix} \mathbf{f}_n \\ \mathbf{f}_n \\ \vdots \\ \mathbf{f}_n \end{pmatrix}$$
$$+ \begin{pmatrix} C_{1,1} & C_{1,2} & \dots & C_{1,M} \\ C_{2,1} & C_{2,2} & \dots & C_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ C_{M,1} & C_{M,2} & \dots & C_{M,M} \end{pmatrix} \begin{pmatrix} \mathbf{\Psi_{n+p_1}} \\ \mathbf{\Psi_{n+p_2}} \\ \vdots \\ \mathbf{\Psi_{n+p_M}} \end{pmatrix} \tag{2.68}$$

where

$$\mathbf{\Phi}_{n+p_i} = \mathbf{Q}_{v,n+p_i}\mathbf{D}^v + \mathbf{Q}_{v-1,n+p_i}\mathbf{D}^{v-1} + \cdots + \mathbf{Q}_{1,n+p_i}\mathbf{D} + \mathbf{Q}_{0,n+p_i},$$

and

$$\mathbf{A}_{i,j} = \begin{cases} -h\alpha_{i,j}\mathbf{\Phi}_{n+p_j} & i \neq j \\ \mathbf{I} - h\alpha_{i,i}\mathbf{\Phi}_{n+p_i} & i = j \end{cases} \qquad \mathbf{B}_{i,j} = h\beta_{i,j}\mathbf{I}, \quad \mathbf{C}_{i,j} = h\alpha_{i,j}\mathbf{I}.$$

A compact representation of Eq. (2.68) is

$$\mathbf{A}_M\mathbf{Y}_{n+M} = \mathbf{Y}_n + \mathbf{B}_m\mathbf{F}_n + C_M\mathbf{\Psi}_{n+M}\mathbf{f}_{n+M}, \tag{2.69}$$

with $\mathbf{A}_M$, $\mathbf{B}_M$, $\mathbf{C}_M$ being the coefficient matrices from Eq. (2.68) and

$$\begin{aligned} \mathbf{Y}_{n+M} &= [\mathbf{y}_{n+p_1}, \mathbf{y}_{n+p_2}, \dots, \mathbf{y}_{n+p_M}]^T, \\ \mathbf{G}_{n+M} &= [\mathbf{G}_{n+p_1}, \mathbf{G}_{n+p_2}, \dots, \mathbf{G}_{n+p_M}]^T, \\ \mathbf{Y}_n &= [\mathbf{y}_n, \mathbf{y}_n, \dots, \mathbf{y}_n]^T, \quad \mathbf{F}_n = [\mathbf{f}_n, \mathbf{f}_n, \dots, \mathbf{f}_n]^T. \end{aligned}$$

In essence, since the right hand side of Eq. (2.69) is known, the equation can be expressed as

$$\mathbf{A}_M\mathbf{Y}_{n+M} = \mathbf{K}_{n+M}, \tag{2.70}$$

where $\mathbf{K}_{n+M} = \mathbf{Y}_n + \mathbf{B}_m \mathbf{F} F_n + \mathbf{C}_M \mathbf{\Psi} f_{n+M}$. The boundary conditions for the PDE are enforced on the initial and final row of collectively $\mathbf{A}_{i,j}$ with the corresponding row entry of $\mathbf{K}_{n+M}$ in Eq. (2.70).

2.4 Summary

In this chapter, we detailed several fundamental descriptions and outcomes of Lie group analysis of PDEs. The algorithm for determining the Lie symmetries of the PDEs was among them. Furthermore, we discussed how the SQLM, MD-BSQLM, and BHM can be applied to nonlinear evolution problems, beginning with a description of how the QLM and the multidomain approach work on nonlinear evolution PDEs. In the following chapter, we will demonstrate how these techniques are implemented to the Burgers-Huxley equation.

Chapter 3

A comparative study of a spectral quasilinearization, multidomain spectral quasilinearization and block-hybrid methods for solving Burgers-Huxley equation

3.1 The generalized Burgers-Huxley equation

The generalized Burgers-Huxley equation is a NLPDE which implies a comprehensive class of physical nonlinear phenomena. It describes the interaction between reaction mechanisms, convection effects and diffusion transports. It finds its application in plenty of fields such as biology, nonlinear acoustics, metallurgy, chemistry, combustion, mathematics and engineering. Harry Bateman first introduced this equation in 1915 [69,70], succeeding Johannes Martinus Burgers who examined it in 1948 [1], and was named after him. It contains a convection term $u^{\delta}u_{xx}$ and dissipative term u_{xx} [71,72]. The generalized Burgers-Huxley equation with all possible parameters is given by

$$\frac{\partial u}{\partial t} = \xi\frac{\partial^2 u}{\partial x^2} - \nu u^\delta \frac{\partial u}{\partial x} + \eta u(1-u^\delta)(u^\delta - \mu), \tag{3.1}$$

where ξ, ν, δ, η and $\mu \in \mathbb{R}$, $t > 0$.

When a vague condition is enforced on the parameters, the generalized Burgers-Huxley equation simplifies to numerous parabolic evolution equations of physical insight. When $\delta = 1$ and $\eta = 0$ Eq. (3.1) is simplified to the Burgers equation, which can be employed to investigate sound waves in a viscous medium [1]. When $\delta = 0$, $\nu = 0$ and $\mu = -1$, we have the Newell-Whitehead-Segel equation [73] which is utilized to simulate the transmission of nerve impulses. The FitzHugh-Nagumo equation is acquired when $\delta = 1$ and $\nu = 0$. This equation has a variety of applications, including neurophysiology, logistic population growth, and auto-catalytic reactions [74]. The Huxley equation which is obtained, depicts the propagation of nerve pulses, the motion of liquid crystal walls, and nerve fibers when $\nu = 0$ [75]. We note that Eq. (3.1) yields the Burgers-Huxley equation when $\delta = 1$.

3.2 Numerical solution of Burger-Huxley equation

In this section we illustrate how Lie symmetry, steady state case, SQLM, MD-BSQLM and BHM are used to solve the Burgers-Huxley equation.

We solve the Burgers-Huxley equation

$$u_t + \nu u u_x - u_{xx} - \eta(1+\mu)u^2 + \eta\mu u + \eta u^3 = 0, \;\; \forall \;\; (x,t) \in [0,1]\times[0,T], \tag{3.2}$$

subject to the initial condition

$$u(x,0) = \frac{\mu}{2} + \frac{\mu}{2}\tanh(A_1 x), \tag{3.3}$$

and boundary conditions

$$u(0,t) = \frac{\mu}{2} + \frac{\mu}{2}\tanh(-A_1A_2t), \;\; u(1,t) = \frac{\mu}{2} + \frac{\mu}{2}\tanh\left[A_1(1 - A_2t)\right], \tag{3.4}$$

where $\nu, \eta \in \mathbb{R}^+$ and $T = 10$. For $\nu = \eta = 0.5$ as well as $\mu = 0.001$. The analytical result is obtained by Deng [72] as

$$u(x,t) = \frac{\mu}{2} + \frac{\mu}{2}\tanh\left[A_1(x - A_2 t)\right], \tag{3.5}$$

where

$$A_1 = \frac{-\nu + \sqrt{\nu^2 + 8\eta}}{8}\mu, \text{ and } A_2 = \frac{(\nu - \sqrt{\nu^2 + 8\eta})\mu + 2(\nu + \sqrt{\nu^2 + 8\eta})}{4}. \tag{3.6}$$

3.2.1 Lie symmetry

In this subsection, we utilize the Lie symmetry attained by [76] to reduce the Burgers-Huxley equation to an ODE. The translation symmetries are given by,

$$X_1 = \frac{\partial}{\partial x}, \; X_2 = \frac{\partial}{\partial t}, \tag{3.7}$$

and the travelling wave equation is obtained by combining the translation symmetries which is given by,

$$X = A_2\frac{\partial}{\partial x} + \frac{\partial}{\partial t}. \tag{3.8}$$

In this instance, we utilize travelling wave equation to reduce the Burgers-Huxley equation. Thus the characteristic of travelling wave equation is given by,

$$\frac{dt}{1} = \frac{dx}{A_2} = \frac{du}{0}, \tag{3.9}$$

since $du = 0$, then

$$k_1 = u. \tag{3.10}$$

From Eq. (3.8), we get,

$$\frac{dt}{1} = \frac{dx}{A_2}, \tag{3.11}$$

which implies that,

$$k_2 = x - A_2 t, \tag{3.12}$$

where k_1 and k_2 are arbitary constants.

Thus the general solution is expressed by writting $k_1 = f(k_2)$. That being the case, the general solution is given by,

$$u(x,t) = f(x - A_2 t). \tag{3.13}$$

Let $\xi = x - A_2 t$, then our equation becomes,

$$u = f(\xi), \tag{3.14}$$

where $f(\xi)$ is an arbitary function. We differentiate Eq. (3.14) with respect x and t and we obtain the following,

$$u_t = -A_2 f'(\xi),\ u_x = f'(\xi),\ u_{xx} = f''(\xi). \tag{3.15}$$

We now subtitute Eq. (3.15) into Eq. (3.2), therefore an ODE of Burgers-Huxley equation becomes,

$$-A_2 f'(\xi) + \nu f(\xi) f'(\xi) - f''(\xi) - \eta(1+\mu) f(\xi)^2 + \eta\mu f(\xi) + \eta f(\xi)^3 = 0. \tag{3.16}$$

When $t = 0$, $\xi = x$, thus Eq. (3.16) and Eq. (3.3) becomes,

$$-A_2 f'(x) + \nu f(x) f'(x) - f''(x) - \eta(1+\mu) f(x)^2 + \eta\mu f(x) + \eta f(x)^3 = 0. \tag{3.17}$$

and the exact solution

$$f(x) = \frac{\mu}{2} + \frac{\mu}{2}\tanh(A_1 x). \tag{3.18}$$

3.2.2 Spectral Quasi-linearizaation Method

In this sub-division, we illustrate how the implementation of the SQLM is employed to solve the Burgers-Huxley equation. In this study the SQLM is utilized to solve ODE. Consider the Burgers-Huxley equation given by Eq. 3.2. Let

$$F(t, f, f', f'') \equiv A_2 f' - \nu f f' + f'' + \eta(1+\mu) f^2 - \eta\mu f - \eta f^3 = 0. \tag{3.19}$$

We attain the QLM scheme to approximate the result f_{r+1} at $(r+1)$th iteration level as;

$$a_{0,r}(x) f''_{r+1}(x) + a_{1,r}(x) f'_{r+1}(x) + a_{2,r}(x) f_{r+1}(x) = R_r(x), \tag{3.20}$$

where,

$$\begin{aligned} a_{0,r}(x) &= \frac{\partial F}{\partial f''} = 1,\ \ a_{1,r}(x) = \frac{\partial F}{\partial f'} = A_2 - \nu f_r, \\ a_{2,r}(x) &= \frac{\partial F}{\partial f} = -\nu f'_r + 2\eta(1+\mu) f_r - \eta\mu - 3\eta f_r^2, \end{aligned} \tag{3.21}$$

and

$$R_r(x) = -\nu f_r f_r' + \eta(1+\mu)f_r^2 - 2\eta f_r^3. \tag{3.22}$$

The domain $x \in [0,1]$ is trasformed to $\kappa \in [-1,1]$ using linear trasformation,

$$\xi = \frac{b-a}{2}\kappa + \frac{b+a}{2}. \tag{3.23}$$

The boundary conditions becomes,

$$f(\kappa_N) = \frac{\mu}{2}, \;\; f(\kappa_0)\frac{\mu}{2} + \frac{\mu}{2}\tanh(A_1). \tag{3.24}$$

Applying spectral collocation on (3.20) gives,

$$\mathbf{A}\mathbf{f}_{r+1} = \mathbf{R}_r, \text{ where } \mathbf{A} = \mathbf{D}^2 + a_{1,r}\mathbf{D} + a_{2,r} \tag{3.25}$$

where $\mathbf{f}_{r+1} = [f(x_o), f(x_1), \ldots, f(x_N)]^T$, $\mathbf{R}_r = [R_r(x_o), R_r(x_1), \ldots, R_r(x_N)]^T$ and $a_{p,r}$ are diagonal matrices of $[a_{p,r}(x_o), a_{p,r}(x_1), \ldots, a_{p,r}(x_N)]^T$ for $p = 0, 1, 2$. The boundary conditions Eq. (3.24) are applied on the first and the last row of Eq. (3.25) as shown below,

$$\begin{bmatrix} 1 & 0 & 0 & \ldots & 0 & 0 \\ A_{1,0} & A_{1,1} & A_{1,2} & \ldots & A_{1,N-1} & A_{1,N} \\ A_{2,0} & A_{2,1} & A_{2,2} & \ldots & A_{2,N-1} & A_{2,N} \\ \vdots & \vdots & \vdots & \ddots & \ldots & \vdots \\ A_{N-1,0} & A_{N-1,1} & A_{N-1,2} & \ldots & A_{N-1,N-1} & A_{N-1,N} \\ 0 & 0 & 0 & \ldots & 0 & 1 \end{bmatrix} \begin{bmatrix} f(\kappa_0) \\ f(\kappa_1) \\ f(\kappa_2) \\ \vdots \\ f(\kappa_{N-1}) \\ f(\kappa_N) \end{bmatrix} = \begin{bmatrix} \frac{\mu}{2} + \frac{\mu}{2}\tanh(A_1) \\ R_r(\kappa_1) \\ R_r(\kappa_2) \\ \vdots \\ R_r(x\kappa_{N-1}) \\ \frac{\mu}{2} \end{bmatrix}.$$

3.2.3 Multi-Domain Bivariate Spectral Quasi-linearization Method

In this subsection we illustrate the implementation of the MD-BSQLM. The MD-BSQLM is a method that utilizes the spectral collocation method independently on more than one variable. In this case, the time and space variable. . Since our $t > 1$, we use the multidomain technique on the time (t) variable in this study. The multidomain approach is used to decompose the givem domain in the time variable into smaller nonoverlapping subintervals. From Eq. (3.2), we set,

$$F[\dot{u}, u, u', u''] \equiv -\nu u u' + u'' + \eta(1+\mu)u^2 - \eta\mu u - \eta u^3 = 0. \tag{3.26}$$

Applying the QLM and spectral collocation, we obtain the linearized system and k denote solution at the kth subinterval

$$[a_{0,r}(x,t_j)\mathbf{D}^2 + a_{1,r}(x,t_j)\mathbf{D} + a_{2,r}(x,t_j)]\mathbf{U}_i^{(k)} - \sum_{q=0}^{N_t} d_{j,q}\mathbf{U}_q^{(k)} = \mathbf{R}_r^{(k)}(x,t_j), \tag{3.27}$$

where,

$$\left.\begin{aligned} a_{0,r}(x,t_j) &= \frac{\partial F}{\partial u''} = 1,\\ a_{1,r}(x,t_j) &= \frac{\partial F}{\partial u'} = -\nu u_r,\\ a_{2,r}(x,t_j) &= \frac{\partial F}{\partial u} = -\nu u_r' + 2\eta(1+\mu)u_r - \eta\mu - 3\eta u_r^2, \end{aligned}\right\} \tag{3.28}$$

and

$$R_r(x,t_j) = -\nu u_r u_r' + \eta(1+\mu)u_r^2 - 2\eta u_r^3. \tag{3.29}$$

The multidomain approach is implemented on the linearized scheme Eq. (3.27) , which is solely implemented to the time variable t. In each subinterval $k = 1, 2, \ldots, p$, we solve

$$[a_{0,r}(x,t_i)\mathbf{D}^2 + a_{1,r}(x,t_i)\mathbf{D} + a_{2,r}(x,t_i)]\mathbf{U}_i^{(k)} - \sum_{j=0}^{N_t-1} d_{i,j}\mathbf{U}_j^{(k)} = \mathbf{R}_r^{(k)}(x,t_i) + d_{i,N_t}\mathbf{U}_{t_{N_t}}^{(k)}$$

$x \in [0,1],\ t \in [t_{k-1}, t_k]$,

with boundary conditions:

$$\left.\begin{aligned} u^{(k)}(0,t) &= \frac{\mu}{2} + \frac{\mu}{2}\tanh(-A_1A_2t),\\ u^{(k)}(1,t) &= \frac{\mu}{2} + \frac{\mu}{2}\tanh\left[A_1(1 - A_2t)\right], \end{aligned}\right\} \tag{3.30}$$

and initial conditions at each interval given by,

$$\left.\begin{aligned} u^{(1)}(x,0) &= \frac{\mu}{2} + \frac{\mu}{2}\tanh(A_1x),\ k = 1 \text{ and},\\ u^{(k)}(x,t_{k-1}) &= u^{(k-1)}(x,t_{k-1}),\ k = 2, 3, \ldots, p. \end{aligned}\right\} \tag{3.31}$$

Eq. (3.27) results into the subsequent set of coefficient matrices:

$$\left.\begin{aligned}
&\mathbf{A}_{i,i} = \mathbf{D}^2 + a_{1,r}(x,t_i)\mathbf{D} + a_{2,r}(x,t_i) - d_{i,i}\mathbf{I},\\
&\mathbf{A}_{i,j} = -d_{i,j}\mathbf{I}, \text{ when } i \neq j,\\
&\mathbf{R}_i^{(k)} = \mathbf{R}_r^{(k)} + d_{i,N_t}\mathbf{U}_{N_t}^{(k)},
\end{aligned}\right\} \tag{3.32}$$

where $\mathbf{I}$ is an identity matrix as size $(N_x+1)\times(N_x+1)$. The boundary conditions at the collocation points are

$$\left.\begin{aligned}
U^{(k)}(x_N,t_i) &= \frac{\mu}{2} + \frac{\mu}{2}\tanh(-A_1A_2t_i),\\
U^{(k)}(x_0,t_i) &= \frac{\mu}{2} + \frac{\mu}{2}\tanh\left[A_1(1-A_2t_i)\right].
\end{aligned}\right\} \tag{3.33}$$

The first condition at different subintervals is specified by

$$\left.\begin{aligned}
U_{t_{N_t}}^{(1)} &= \frac{\mu}{2} + \frac{\mu}{2}\tanh(A_1x), \quad \text{for } k=1 \text{ and}\\
U_{t_{N_t}}^{(k)} &= U_{t_0}^{(k-1)}, \ for \ \ k=2,3,\ldots,p.
\end{aligned}\right\} \tag{3.34}$$

The boundary conditions are imposed on Eq. (3.33) and we obtain the subsequent $N_t(N_x+1)\times N_t(N_x+1)$ matrix system

$$\begin{bmatrix}
1 & 0 & 0 & \ldots & 0\\
A_{1,0} & A_{1,1} & A_{1,2} & \ldots & A_{1,N_t-1}\\
\vdots & \vdots & \vdots & \ddots & \vdots\\
A_{N-2,0} & A_{N-2,1} & A_{N-2,2} & \ldots & A_{N_t-2,N_t-1}\\
0 & 0 & 0 & \ldots & 1
\end{bmatrix}
\begin{bmatrix}
U_0^{(k)}\\ U_1^{(k)}\\ U_2^{(k)}\\ \vdots\\ U_{N_t-1}^{(k)}
\end{bmatrix}
=
\begin{bmatrix}
\frac{\mu}{2} + \frac{\mu}{2}\tanh\left[A_1(1-A_2t_i)\right]\\
R_1^{(k)}\\
R_2^{(k)}\\
\vdots\\
\frac{\mu}{2} + \frac{\mu}{2}\tanh(-A_1A_2t_i)
\end{bmatrix}.$$

3.2.4 Block-Hybrid Method

In this subsection we illustrate how the BHM with spectral method are utilized to solve the Burgers-Huxley equation. We apply the BHM in time t as well as the spectral collocation method in space x.

From Eq. 3.2, we set,

$$\dot{u} \equiv f(x,u,u',u'') = -\nu uu' + u'' + \eta(1+\mu)u^2 - \eta\mu u - \eta u^3, \tag{3.35}$$

Thus, the QLM scheme becomes,

$$\dot{u} = a_{0,r}(x,t)u''_{r+1} + a_{1,r}(x,t)u'_{r+1} + a_{2,r(x,t)}u_{r+1} + R_r(x,t), \tag{3.36}$$

where,

$$\left.\begin{aligned} a_{0,r}(x,t) &= \frac{\partial f}{\partial u''} = 1, \\ a_{1,r}(x,t) &= \frac{\partial f}{\partial u'} = -\nu u_r, \\ a_{2,r}(x,t) &= \frac{\partial f}{\partial u} = -\nu u'_r + 2\eta(1+\mu)u_r - \eta\mu - 3\eta u_r^2, \end{aligned}\right\} \tag{3.37}$$

and

$$R_r(x,t) = \nu u_r u'_r - \eta(1+\mu)u_r^2 + 2\eta u_r^3. \tag{3.38}$$

The domain $x \in [0,1]$ is trasformed to $\kappa \in [-1,1]$ utilizing linear trasformation,

$$x = \frac{b-a}{2}\kappa + \frac{b+a}{2}. \tag{3.39}$$

Applying the spectral method with $N_x + 1$ collocation points on Eq. (3.36) gives,

$$\dot{u} = f(t,u), \quad u(x,0) = \frac{\mu}{2} + \frac{\mu}{2}\tanh(A_1 x) \tag{3.40}$$

with the boundary conditions given by,

$$\left.\begin{aligned} u(x_{N_x},t) &= \frac{\mu}{2} + \frac{\mu}{2}\tanh(-A_1A_2t), \\ u(x_0,t) &= \frac{\mu}{2} + \frac{\mu}{2}\tanh\left[A_1(1-A_2)t\right], \end{aligned}\right\} \tag{3.41}$$

and

$$\mathbf{f}(t,u) = [a_{0,r}(x,t)\mathbf{D}^2 + a_{1,r}(x,t)\mathbf{D} + a_{2,r}(x,t)\mathbf{I}]\mathbf{u}_{r+1} + \mathbf{G}(t), \tag{3.42}$$

$a_{p,r}$ are diagonal matrices of $[a_{p,r}(x_0,t), a_{p,r}(x_1,t), \ldots, a_{p,r}(x_{N_x},t)]^T$, $p = 0,1,2$, $\mathbf{u} = [u(x_0,t), u(x_1,t), \ldots, u(x_{N_x},t)]^T$, $\mathbf{G} = [g(x_0,t), g(x_1,t), \ldots, g(x_{N_x},t)]^T$.

Applying the BHM on the first order equation Eq. (3.40) yields

$$\mathbf{u}_{n+p_i} = \mathbf{u}_n + h\sum_{j=1}^{M}[\delta_{i,j}(\mathbf{\Phi}_{n+p_j}\mathbf{u}_{n+p_j} + \mathbf{\Psi}_{n+p_j}) + \zeta_{i,j}\mathbf{f}_n], \quad i = 1,2,\ldots,M, \tag{3.43}$$

where

$\mathbf{f}_n = f(t_n, u_n)$, $\boldsymbol{\Phi}_{n+p_j} = a_{0,r}(x,t)\mathbf{D}^2 + a_{1,r}(x,t)\mathbf{D} + a_{2,r}(x,t)\mathbf{I}$,

and

$$\mathbf{A}_{i,j} = \begin{cases} -h\delta_{i,j}\boldsymbol{\Phi}_{n+p_j} & i \neq j \\ \mathbf{I} - h\delta_{i,i}\boldsymbol{\Phi}_{n+p_i} & i = j \end{cases}, \quad B_{i,j} = h\zeta_{i,j}\mathbf{I}, \quad C_{i,j} = h\delta_{i,j}\mathbf{I}.$$

The boundary conditions are imposed on Eq. (3.43) and we attain the succeeding matrix system

$$\begin{pmatrix} 1 & 0 & \dots & 0 \\ A_{2,1} & A_{2,2} & \dots & A_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} \begin{pmatrix} \mathbf{y}_{n+p_1} \\ \mathbf{y}_{n+p_2} \\ \vdots \\ \mathbf{y}_{n+p_M} \end{pmatrix} = \begin{pmatrix} \mathbf{y}_n \\ \mathbf{y}_n \\ \vdots \\ \mathbf{y}_n \end{pmatrix} + \begin{pmatrix} B_{1,1} & B_{1,2} & \dots & B_{1,M} \\ B_{2,1} & B_{2,2} & \dots & B_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ B_{M,1} & B_{M,2} & \dots & B_{M,M} \end{pmatrix} \begin{pmatrix} \mathbf{f}_n \\ \mathbf{f}_n \\ \vdots \\ \mathbf{f}_n \end{pmatrix}$$
$$+ \begin{pmatrix} C_{1,1} & C_{1,2} & \dots & C_{1,M} \\ C_{2,1} & C_{2,2} & \dots & C_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ C_{M,1} & C_{M,2} & \dots & C_{M,M} \end{pmatrix} \begin{pmatrix} \frac{\mu}{2} + \frac{\mu}{2}\tanh\left[A_1(1-A_2)t\right] \\ \mathbf{g}(x_{N-1}, t) \\ \vdots \\ \frac{\mu}{2} + \frac{\mu}{2}\tanh(-A_1A_2t) \end{pmatrix}. \tag{3.44}$$

3.3 Results and Discussion

We display numerical results to Burgers-Huxley equation in this section when $\nu = \eta = 0.5$ and $\mu = 0.001$ is attained utilizing the SQLM, MD-BSQLM, and BHM. The number of collocation points in the spatial x variable results is $N_x = 20$ for the SQLM, MD-BSQLM, and BHM. In the same way, the number of collocations in time t variable is $N_t = 40$ for the MD-BSQLM and BHM. It was instituted that satisfactory precision was reached by utilizing above-mentioned values in all simulations. The results were obtained using MATLAB R2022a.

3.3.1 Numerical convergence analysis

The error norm is employed in order to assess the schemes' convergence and precision. Because the equations possess analytical results, the norm can certainly be determined at every iteration. The level of precision of the methods is calculated by the approximate result at each time level and the equivalent analytical solution. The error norm for any bounded function is well-defined as

$$E_{N_x} = ||u_r - u_r^*||_\infty \ , 0 \leqq r \leqq N_x, \tag{3.45}$$

where u_r exemplifies the approximated solution and u_r^*, denotes the analytical result at time level t. When compared to the analytical result, we use the maximum error to evaluate the precision for SQLM, MD-BSQLM, and BHM approximate solutions at a given time are defined as

$$\left.\begin{array}{l} E_{N_x} = max_r|u(x_r) - u^*(x_r)| \ , 0 \leqq r \leqq N_x, \ \text{for the SQLM,} \\ E_{N_x} = max_r|u(x_r,t) - u^*(x_r,t)| \ , 0 \leqq r \leqq N_x, \\ \text{for the MD-BSQLM and BHM} \end{array}\right\} \tag{3.46}$$

where $u(x_r)$ and $u(x_r,t)$ are the exact solutions at the time level t, as indicated by Motsa et al., [29], and $u^*(x_r)$ and $u^*(x_r,t)$ are the approximate result. These methods will converge if the maximum errors reach zero as the number of collocation points enhances. It also converges when the maximum errors decline as the number of collocation points increases. Maximum and norm errors will be represented graphically and displayed in tabular form.

Figures 3.1, 3.2 and 3.3 provide the comparison of the analytical and approximate solutions to Burgers-Huxley equation utilizing SQLM, MD-BSQLM, and BHM. The approximate solutions agree with the analytical solution very well, proving the precision of the algorithms utilized in this study. Figures 3.4, 3.5 and 3.6 show error analysis graphs for SQLM, MD-BSQLM, and BHM, respectively.

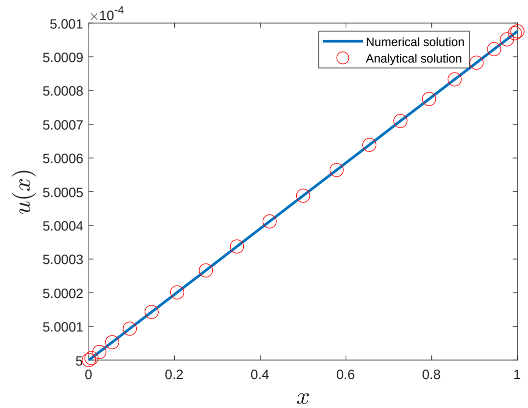


Figure 3.1: Exact solution vs Numerical solution of Burgers-Huxley equation utilizing the SQLM at $t = 0$.

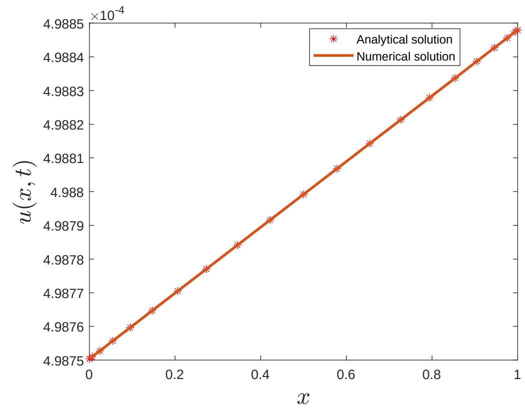


Figure 3.2: Exact solution vs Numerical solution of Burgers-Huxley equation utilizing the MD-BSQLM at $t = 10$.

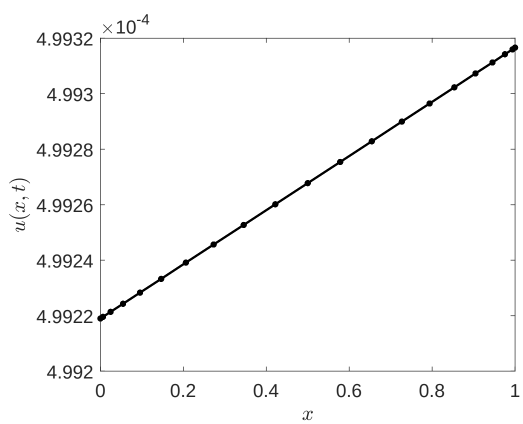


Figure 3.3: Exact solution vs Numerical solution of Burgers-Huxley equation at $t = 10$ utilizing the BHM.

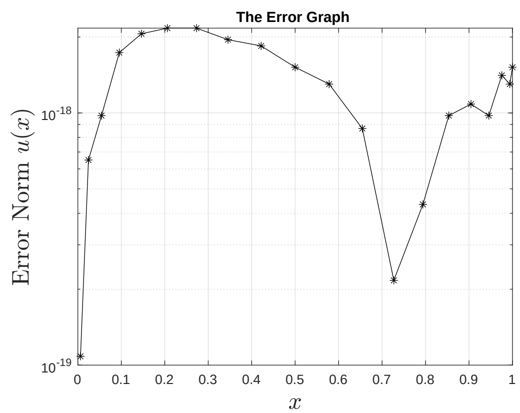


Figure 3.4: Error profile of the numerical results of Burgers-Huxley equation utilizing SQLM at $t = 0$.

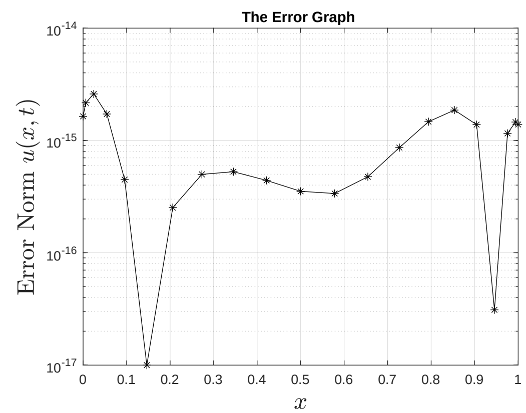


Figure 3.5: Error profile of the numerical results of Burgers-Huxley equation utilizing MD-BSQLM at $t = 10$.

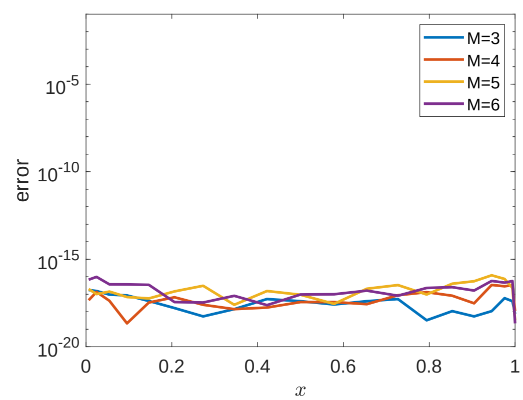


Figure 3.6: Error profile of the numerical results of Burgers-Huxley equation utilizing BHM at $t = 10$.

The maximum errors of the Burgers-Huxley equation obtained using SQLM, MD-BSQLM, and BHM have been given in Tables 3.1, 3.2 and 3.3, respectively, at $t = 0$ for the SQLM and $t \in [0, 10]$ for both MD-BSQLM and BHM. The outcomes were calculated in the $x \in [0, 1]$ space domain. The computational time to produce the outcomes are also acquired in order to give a significance of the methods' computational efficiency. The reliability of these methods is made very clear through tables 3.1, 3.2, and 3.3.

The maximum errors of Burgers-Huxley equation obtained with the SQLM are presented in Table 3.1. The accuracy appears to increase as the number of collocation points N_x significantly increases. It is clear that using a small number of collocation points in the spatial x variable $N_x \leq 20$ yields precise findings with errors of order 10^{-19}. This simply proves that SQLM is an effective technique for solving the Burgers-Huxley equation. We should also mention that SQLM is computationally fast, generating reliable data in split seconds.

$t = 0 \setminus N_x$	14	16	18	20
	3.90e-18	3.04e-18	3.80e-18	1.52e-18
	4.01e-18	2.17e-18	3.58e-18	1.41e-18
	3.90e-18	9.76e-19	4.12e-18	1.08e-18
	3.90e-18	1.08e-18	3.14e-18	4.34e-19
	3.69e-18	1.08e-18	1.63e-18	8.67e-19
	3.36e-18	1.63e-18	7.59e-19	1.52e-18
	2.71e-18	1.41e-18	8.674e-19	1.95e-18
	1.951e-18	4.34e-19	1.084e-19	2.168e-18
CPU time(sec)	0.002250	0.000236	0.000280	0.001727

Table 3.1: Maximum errors for Burgers-Huxley equation using SQLM.

The Burgers-Huxley equation was solved utilized the MD-BSQLM. The maximum errors are displayed in Table 3.2. The accuracy varies as the number of collocation points N_x increases. Notably, precise outcomes with errors of order 10^{-18} are attained with only a small number of collocation points in the x as well as t variables: $N_t \leq 40, N_x \leq 20$. It clearly demonstrates the efficiency of the MD-BSQLM in solving the Burgers-Huxley problem. We should also take into account that the MD-BSQLM is computationally quick and efficient, achieving accurate data in a matter of seconds.

$t \setminus N_x$	14	16	18	20
0.50	4.955e-17	4.987e-18	1.867e-16	6.222e-16
2.50	1.236e-17	7.199e-17	7.698e-17	7.878e-16
5.00	2.840e-16	1.468e-16	5.996e-16	4.505e-16
7.50	1.284e-16	2.585e-16	1.064e-16	1.597e-16
10.00	6.928e-17	3.863e-16	3.618e-16	3.523e-16
CPU time(sec)	0.128877	0.161370	0.168589	0.187094

Table 3.2: Maximum errors for Burgers-Huxley equation using MD-BSQLM.

The maximum error of Burgers-Huxley problem attained utilizing BHM is displayed in Table 3.3. The precision declines as the M increases. Using a small number of collocation points $N_t \leq 40$ and $N_x \leq 20$, precise outcomes with errors of order to 10^{-18} are achieved. This clearly displays that the BHM is a powerful technique suitable for working out the Burgers-Huxley equation, and it is also computationally fast, as precise fallouts are produced in a few seconds.

t	$M = 3$	$M = 4$	$M = 5$	$M = 6$
0.00	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00
2.50	9.9747e-18	1.5613e-17	1.1352e-16	9.0748e-17
5.00	9.7578e-18	2.6021e-17	7.4810e-17	4.4019e-17
7.50	9.9747e-18	3.3827e-17	1.1655e-16	4.9223e-17
10.00	1.7456e-17	3.4044e-17	1.2089e-16	9.8229e-17
CPU time(sec)	0.179115	0.270916	0.342384	0.511625

Table 3.3: Maximum errors for Burgers-Huxley equation using BHM.

Figure 3.7 depicts a comparison of the Burgers-Huxley equation's exact and numerical solutions. The numerical solution admits perfectly with the exact solution, illustrating the precision of the algorithms reported in this research.

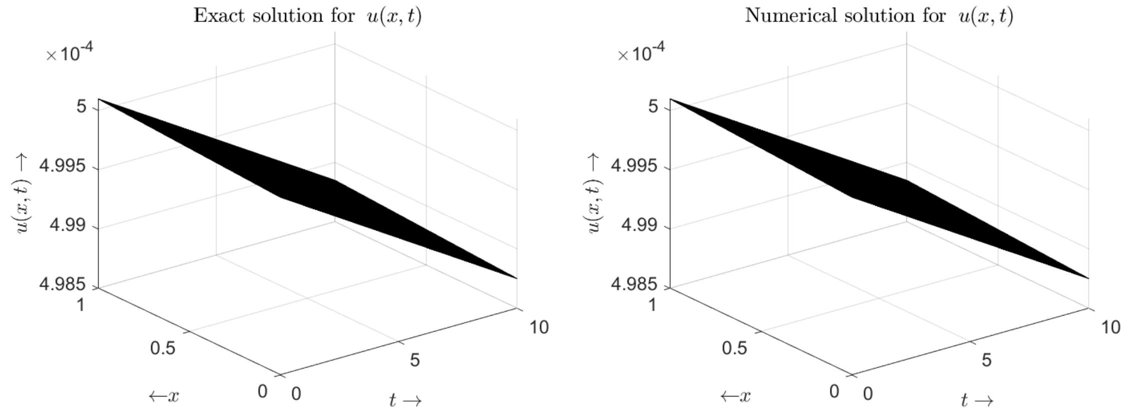


Figure 3.7: Exact solution and Numerical solution of Burgers-Huxley equation

3.4 summary

In this chapter, we reduced the Burgers-Huxley equation to an ODE utilizing Lie symmetries, solved the ODE utilizing SQLM, and solved the Burgers-Huxley problem utilizing MD-BSQLM and BHM. To compare the SQLM, MD-BSQLM, and BHM, the norm error was calculated using both the exact and approximate solutions. The objective of this chapter was to contrast and compare the SQLM, MD-BSQLM, and BHM for solving the Burgers-Huxley equation. It was discovered in this study that MD-BSQLM generates more computationally efficient results than SQLM and BHM. Further to that, evidence suggests that a larger number of nonoverlapping intervals leads to greater precision, more rapid convergence, and quicker time to generate solutions when utilizing the MD-BSQLM. In the next chapter, we solve the examples of nonlinear evolution equations using the four proposed methods.

Chapter 4

Numerical results of evolution problems

In this chapter, we examine the results of the family of the evolution equation which are the Burgers-Fisher equation, Fitzhugh-Nagumo equation, and coupled Burgers' equation. The equations are solved separately using the lie symmetries, SQLM, MD-BSQLM, and BHM. We demonstrate the numerical results against the analytical solution of each equation and compare the three approaches grounded on the error given by each method.

4.1 Nonlinear evolution equations

An evolution equation is a PDE that explains the time evolution of a physical system commencing from given initial data. Evolution equations appear in a range of applied and engineering sciences. Scientists whose study aims are to enhance a better comprehension of such enticing nonlinear problems are still interested in the improvement of numerical and analytical techniques for solving complex, greatly nonlinear evolution PDEs. Countless questions can be raised about evolution equations, such as the existence of specific types of results, such as stability results, traveling waves, self-similar solutions, and time-periodic solutions, the dynamic stability of these solutions, the long time asymptotic behavior of solutions, chaotic dynamics, complete integrability, singular perturbation expressions for solutions, evolution of random solutions, numerical scheme convergence, and etc. [77]. The greatest fundamental query is about the existence and exceptionality of results. Burgers-Huxley equation, Cahn-Allen equation, Burgers-Fisher equation, KdV equation, Newell-Whitehead-Segel equation, also FitzHugh-Nagumo equations, just to name a few are the family of evolution equations. The search for exact traveling wave solutions to nonlinear evolution equations in mathematical physics is critical in a variety of scientific and engineering fields, including soliton theory, plasma physics, nonlinear optics, fluid dynamics, biophysics, and many others. As a result, many prevailing techniques, such as the homogeneous balance method, the tanh method, the tanh-coth method, the modified tanh-coth method, the inverse scattering method, Hirota's bilinear method, the sine-cosine method, the Exp-function method, the F-expansion method, and others, have been developed [78]. Also, evolution equations has being solved using different numerical schemes such as, the Adomain decomposition method, spectral collocation method, finite difference method, and B-spline method [79–82].

4.2 Numerical Experimentations

This section examines the solutions of the families of evolution equations, namely; the Burgers-Fisher equation, Fitzhugh-Nagumo equation, as well as Coupled Burgers' equation. We put forward solutions acquired from the Lie symmetries, SQLM, MD-BSQLM, and BHM for the three examples examined. The key purpose of this study is to prove the exactness and appearance of the applicability of the method utilized. A comparability of the norm error solutions is achieved from the numerical approximation and exact results. The results were obtained using ten iterations.

4.2.1 Example 1: The generalized Burgers-Fisher equation

The generalized Burgers-Fisher [83] equation is considered

$$\frac{\partial u}{\partial t} + \nu u^n \frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial t^2} = \eta u, \quad (x,t) \in [0,1] \times [0,10], \tag{4.1}$$

subject to initial condition

$$u(x,0) = \left[\frac{1}{2} + \frac{1}{2}\tanh(\mathrm{a}_1 x)\right]^{\frac{1}{n}}, \tag{4.2}$$

and exact solution

$$u(x,t) = \left(\frac{1}{2} + \frac{1}{2}\tanh\left[\mathrm{a}_1(x - \mathrm{a}_2 t)\right]\right)^{\frac{1}{n}}, \tag{4.3}$$

where $\mathrm{a}_1 = \dfrac{-\nu n}{2(1+n)}$ and $\mathrm{a}_2 = \dfrac{\nu}{1+n} + \dfrac{\eta(1+n)}{\nu}$. Where ν, η and n are parameters. We chose these parameters to be $\nu = \eta = n = 1$ for illustration purposes. This equation contains numerous applications in plasma physics, fluid physics, capillary gravity waves, nonlinear optics, chemical physics, and other fields [84].

The symmetries of this equation obtained using Mathemetica are

$$X_1 = \frac{\partial}{\partial x}, \; X_2 = \frac{\partial}{\partial t},$$

and combining X_1 and X_2 we obtain the travelling wave equation which is given by

$$X = \mathrm{a}_2 \frac{\partial}{\partial x} + \frac{\partial}{\partial t}. \tag{4.4}$$

Therefore the ODE of the Burger-Fisher equation is given by

$$\nu f(\xi) f'(\xi) + \eta f^2(\xi) - \mathrm{a}_2 f'(\xi) - \eta f(\xi) - f''(\xi) = 0, \text{ where } \xi = x - \mathrm{a}_2 t. \tag{4.5}$$

When $t = 0$ on Eq. (4.5) we obtain $\xi = x$. Thefore the ODE becomes

$$\nu f(x) f'(x) + \eta f^2(x) - \mathrm{a}_2 f'(x) - \eta f(x) - f''(x) = 0, \tag{4.6}$$

and the exact solution of Eq. (4.6) becomes

$$f(x) = \left(\frac{1}{2} + \frac{1}{2}\tanh(\mathrm{a}_1 x)\right)^{\frac{1}{n}}. \tag{4.7}$$

The appropriate nonlinear operator F is preferred in this SQLM example, as

$$F(f, f', f'') = f'' - \nu f f' - \eta f^2 + \mathrm{a}_2 f' + \eta f. \tag{4.8}$$

The appropriate nonlinear operator H for this example using MD-BSQLM is selected as

$$H(u, u', u'') = u'' - \nu u u' - \eta u^2 + \eta u. \tag{4.9}$$

For the purposes of this example using BHM, the suitable nonlinear operator H is taken as

$$\dot{u} \equiv H(u, u', u'') = u'' - \nu u u' - \eta u^2 + \eta u, \tag{4.10}$$

thus

$$\left.\begin{aligned} &\mathbf{\Phi}_{n+p_j} = \mathbf{D}^2 + \mathrm{diag}(-\nu u)\mathbf{D} + \mathrm{diag}(\nu u' + \eta - 2\eta u)\mathbf{I}, \text{ and} \\ &\mathbf{\Psi}_{n+p_j} = \nu u u' - \eta u^2. \end{aligned}\right\} \tag{4.11}$$

We utilise Taylor series expansion to linearize the nonlinear operators F and H.

4.2.2 Example 2: The FitzHugh-Nagumo equation

Consider the FitzHugh-Nagumo equation

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = \eta u(1-u)(u-\mu), \quad (x,t) \in [0,1] \times [0,10] \tag{4.12}$$

with initial condition

$$u(x,0) = \frac{1}{2} - \frac{1}{2}\tanh(\mathrm{A}_1 x), \tag{4.13}$$

and analytical solution [74]

$$u(x,0) = \frac{1}{2} - \frac{1}{2}\tanh\left[\mathrm{A}_1(x - \mathrm{A}_2 t)\right], \tag{4.14}$$

where $\mathrm{A}_1 = \dfrac{\sqrt{\eta}}{2\sqrt{2}}$, $\mathrm{A}_2 = \sqrt{\dfrac{\eta}{2}}(2\mu - 1)$, $\mu \in (0,1)$ and η is a real parameter. For illustration purposes we chose $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$. This equation has numerous applications in the disciplines of neurophysiology, flame propagation, neurophysiology, logistic population growth, nuclear reactor theory, neurophysiology, and branching Brownian motion process [85].

The symmetries [86] of this equation are as follows:

$$X_1 = \frac{\partial}{\partial x},\ X_2 = \frac{\partial}{\partial t},$$

and combining X_1 and X_2 we get the travelling wave equation which is given by

$$X = \mathrm{A}_2\frac{\partial}{\partial x} + \frac{\partial}{\partial t}. \tag{4.15}$$

The ODE of the Fitzhugh-Nagumo equation becomes

$$-\mathrm{A}_2 f'(\xi) - f''(\xi) - \eta f(\xi)(1 - f(\xi))(f(\xi) - \mu) = 0,\ \text{where}\ \xi = x - \mathrm{A}_2 t. \tag{4.16}$$

For $t = 0$ on Eq. (4.16) we obtain $\xi = x$. Thefore the ODE becomes

$$\mathrm{A}_2 f'(x) + f''(x) + \eta f(x)(1 - f(x))(f(x) - \mu) = 0, \tag{4.17}$$

and exact solution becomes

$$f(x) = \frac{1}{2} - \frac{1}{2}\tanh(\mathrm{A}_1 x). \tag{4.18}$$

Using SQLM for this equation, the suitable nonlinear operator F is selected as

$$F(f', f', f'') = f'' + \mathrm{A}_2 f' + \eta f^2(1 + \mu) - \eta\mu f - \eta f^3. \tag{4.19}$$

The appropriate nonlinear operator H for this example using MD-BSQLM is selected as

$$H(u, u', u'') = u'' + \beta u^2(1 + \mu) - \eta\mu u - \eta u^3. \tag{4.20}$$

The appropriate nonlinear operator H for this example using BHM is selected as

$$\dot{u} \equiv H(u, u', u'') = u'' + \eta u^2(1 + \mu) - \eta\mu u - \eta u^3, \tag{4.21}$$

thus

$$\left.\begin{aligned}\mathbf{\Phi}_{n+p_j} &= \mathrm{diag}(1)\mathbf{D}^2 + \mathrm{diag}(2\eta u(1+\mu) - \eta\mu - 3\eta u^2)\mathbf{I},\ \text{and}\\ \mathbf{\Psi}_{n+p_j} &= -\eta u^2(1+\mu) + 2\eta u^3.\end{aligned}\right\} \tag{4.22}$$

We utilize Taylor series expansion to linearize the nonlinear operators F and H.

4.2.3 Example 3: The Coupled Burgers'equation

We consider the Coupled Burgers'equation

$$\left.\begin{aligned} \frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} - 2u\frac{\partial u}{\partial x} + \frac{\partial (uv)}{\partial x} = 0,\ -\pi < x < \pi,\ t > 0, \\ \frac{\partial v}{\partial t} - \frac{\partial^2 v}{\partial x^2} - 2v\frac{\partial v}{\partial x} + \frac{\partial (uv)}{\partial x} = 0,\ -\pi < x < \pi,\ t > 0, \end{aligned}\right\} \tag{4.23}$$

subjected to initial conditions

$$u(x,0) = \sin(x),\ v(x,0) = \sin(x), \tag{4.24}$$

with exact solutions [87]

$$u(x,t) = e^{-t}\sin(x),\ v(x,t) = e^{-t}\sin(x). \tag{4.25}$$

This coupled system Burgers' equation is a simple model of gravity-induced sedimentation or evolution of scaled volume concentrations of two types of particles in fluid suspensions or colloids [87].

Eq. (4.23) can be rewritten as,

$$\left.\begin{aligned} \frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} - 2u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial x} + u\frac{\partial v}{\partial x} = 0,\ -\pi < x < \pi,\ t > 0, \\ \frac{\partial v}{\partial t} - \frac{\partial^2 v}{\partial x^2} - 2v\frac{\partial v}{\partial x} + v\frac{\partial u}{\partial x} + u\frac{\partial v}{\partial x} = 0,\ -\pi < x < \pi,\ t > 0. \end{aligned}\right\} \tag{4.26}$$

Lie symmetries of Eq. (4.26) are given by [88]

$$X_1 = \frac{\partial}{\partial t},\ X_2 = \frac{\partial}{\partial x},\ X_3 = -x\frac{\partial}{\partial x} - 2t\frac{\partial}{\partial t} + u\frac{\partial}{\partial u} + v\frac{\partial}{\partial v},$$

and combining X_1 and X_2 we attein the travelling wave solution given by,

$$X = \frac{\partial}{\partial x} + \frac{\partial}{\partial t}. \tag{4.27}$$

Therefore the ODE of the Coupled Burgers' equation is provided by,

$$\left.\begin{aligned} f''(\xi) + f'(\xi) + 2f(\xi)f'(\xi) - h(\xi)f'(\xi) - f(\xi)h'(\xi) = 0,\ \xi = x - t, \\ h''(\xi) + h'(\xi) + 2h(\xi)h'(\xi) - h(\xi)f'(\xi) - f(\xi)h'(\xi) = 0,\ \xi = x - t. \end{aligned}\right\} \tag{4.28}$$

When $t = 0$, we get $\xi = x$, thus Eq. (4.28) becomes,

$$\left.\begin{aligned} f''(x) + f'(x) + 2f(x)f'(x) - h(x)f'(x) - f(x)h'(x) = 0,\ -\pi < x < \pi, \\ h''(x) + h'(x) + 2h(x)h'(x) - h(x)f'(x) - f(x)h'(x) = 0,\ -\pi < x < \pi, \end{aligned}\right\} \tag{4.29}$$

with exact solutions

$$f(x) = \sin(x),\ \ h(x) = \sin(x). \tag{4.30}$$

For this illustration, the nonlinear appropriate operators F and H are preferred using the SQLM

$$\left.\begin{aligned} F(f, h, f', h', f'', h'') = f'' + f' + 2ff' - hf' - fh', \\ H(f, h, f', h', f'', h'') = h'' + h' + 2hh' - hf' - fh'. \end{aligned}\right\} \tag{4.31}$$

The appropriate nonlinear operators for F and H for this example using MD-BSQLM are given by

$$\left.\begin{aligned} F(u, v, u', v', u'', v'') = u'' + 2uu' - vu' - uv', \\ H(u, v, u', v', u'', v'') = v'' + 2vv' - vu' - uv'. \end{aligned}\right\} \tag{4.32}$$

The appropriate nonlinear operators for F and H for this example using BHM are chosen as

$$\left.\begin{aligned} \dot{u} \approx F(u, v, u', v', u'', v'') = u'' + 2uu' - vu' - uv', \\ \dot{v} \approx H(u, v, u', v', u'', v'') = v'' + 2vv' - vu' - uv', \end{aligned}\right\} \tag{4.33}$$

thus

$$\left.\begin{aligned} &\mathbf{\Phi}^1_{n+p_j} = \Big[\mathrm{diag}(1)\mathrm{D}^2 + (\mathrm{diag}(2u - v)\mathrm{D} + \mathrm{diag}(2u' - v')\mathrm{I}\Big] \\ &+ \Big[\mathrm{diag}(-u)\mathrm{D} + (\mathrm{diag}(-u')\mathrm{I}\Big] \\ &\mathbf{\Psi}^1_{n+p_j} = -2uu' + uv' + vu', \\ &\mathbf{\Phi}^2_{n+p_j} = \Big[\mathrm{diag}(-v)\mathrm{D} + (\mathrm{diag}(-v')\mathrm{I}\Big] \\ &+ \Big[\mathrm{diag}(1)\mathrm{D}^2 + \mathrm{diag}(2v - u)\mathrm{D} + \mathrm{diag}(2v' - u')\mathrm{I}\Big] \\ &\mathbf{\Psi}^2_{n+p_j} = -2vv' + uv' + vu'. \end{aligned}\right\} \tag{4.34}$$

By utilizing the Taylor series expansion, we linearize the nonlinear operators F and H.

4.3 Summary

We used Lie symmetry and the proposed algorithms SQLM, MD-BSQLM, and BHM to solve three examples of evolution equations in this chapter. In the succeeding chapter we discuss the outcomes obtained in this chapter using the three proposed methods to solve the examples.

Chapter 5

Results and Discussion

The approximate solutions attained utilizing the SQLM, MD-BSQLM, and BHM algorithms are presented in this chapter. For SQLM, MD-BSQLM, and BHM, the number of collocation points in the x variable utilized to produce outcomes is $N_x = 20$. For MD-BSQLM and BHM for different M values, the number of collocation points in the time t variable utilized to produced the data is $N_t = 40$. It was discovered that using these values yielded sufficient results in all numerical simulations. MATLAB R2022a was used to produce the results Tables 5.1-5.8 and Figures 5.1-5.20.

Figures 5.1 and 5.2 shows the comparison of analytical and approximate solutions to the Burgers-Fisher equation and FitzHugh-Nagumo equation that used SQLM. The approximate solutions agree well with the exact solution, exemplifying the precision of the algorithms provided in this study. Figures 5.3 and 5.4 of the SQLM for Burgers-Fisher equation and FitzHugh-Nagumo equation show error analysis graphs. The SQLM does not generate numerical results for the Coupled Burgers' equation.

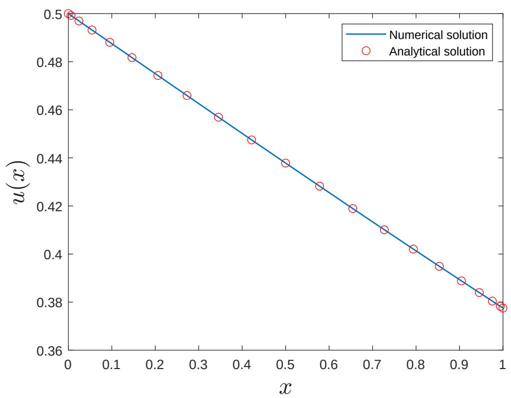


Figure 5.1: Exact VS Numerical of Burgers-Fisher equation using SQLM for $\nu = \eta = n = 1$.

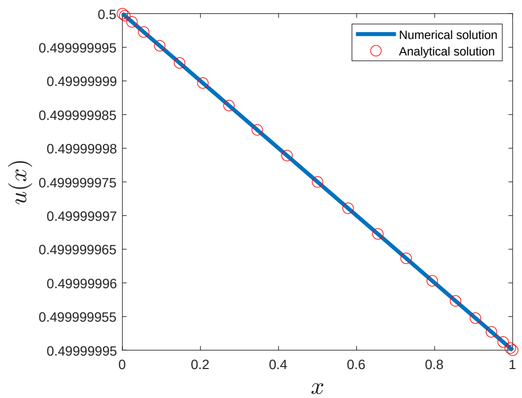


Figure 5.2: Exact VS Numerical of FitzHugh-Nagumo equation using SQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$.

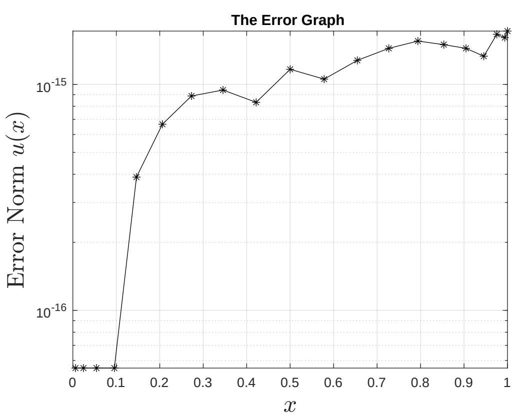


Figure 5.3: Error profile of Burgers-Fisher equation using SQLM for $\nu = \eta = n = 1$.

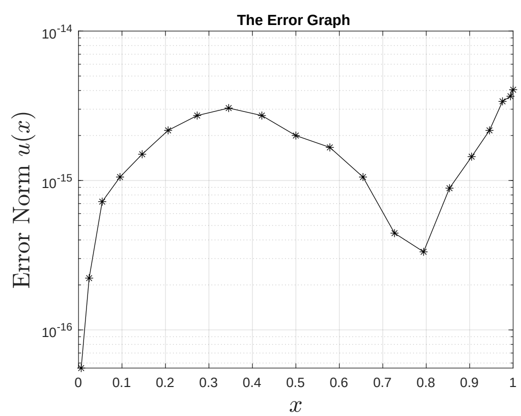


Figure 5.4: Error profile of FitzHugh-Nagumo equation using SQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$.

In Tables 5.1 and 5.2 we provide the maximum errors among the analytical and SQLM at $t = 0$ of Burgers-Fisher equation and FitzHugh-Nagumo equation. The outcomes were calculated in the space domain $x \in [0, 1]$. The computational time to produce the solutions is also given to give a understanding of the methods' computational efficiency. Tables 5.1 and 5.2 show clearly the accuracy of these methods. The maximum errors of the Burgers-Fisher equation and FitzHugh-Nagumo equation obtained with SQLM are presented in Table 5.1 and 5.2. The precision appears to increase with the number of collocation points N_x. It is apparent that precise solutions with errors of order 10^{-17} are attained utilizing a small number of collocation points in the x variable $N_x \leq 20$. This evidently illustrates that SQLM is a powerful technique for solving the Burgers-Fisher equation, and FitzHugh-Nagumo equation, expect for Coupled Burgers' equation. We also note that the SQLM is computationally effective, generating precise solutions in few seconds.

$t = 0 \setminus N_x$	14	16	18	20
	1.254e-02	9.607e-03	7.596e-03	6.156e-03
	1.499e-15	5.551e-17	1.832e-15	1.721e-15
	1.166e-15	3.886e-16	1.499e-15	1.665e-15
	9.437e-16	8.327e-16	1.499e-15	1.443e-15
	8.882e-16	1.055e-15	7.216e-16	9.437e-16
	6.106e-16	4.996e-16	3.886e-16	6.661e-16
	4.996e-16	4.441e-16	1.665e-16	5.551e-17
CPU time(sec)	0.015313	0.000558	0.000593	0.000610

Table 5.1: Maximum errors for Burgers-Fisher equation utilizing SQLM for $\nu = \eta = n = 1$.

$t = 0 \setminus N_x$	14	16	18	20
	7.61e-15	3.28e-15	7.22e-16	2.33e-15
	5.16e-15	2.89e-15	1.89e-15	1.721e-15
	2.22e-16	6.11e-16	3.55e-15	2.220e-16
	3.94e-15	2.78e-15	6.94e-15	1.776e-15
	4.50e-15	5.33e-15	9.10e-15	84.330e-15
	2.89e-15	6.05e-15	6.66e-15	5.274e-15
	1.50e-15	4.05e-15	3.99e-15	5.607e-15
CPU time(sec)	0.015425	0.000371	0.003633	0.001055

Table 5.2: Maximum errors for FitzHugh-Nagumo equation utilizing SQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$.

Figures 5.5, 5.6 and 5.7 compare analytical and approximate solutions to the Burgers-Fisher equation, FitzHugh-Nagumo equation, and coupled Burgers'equation using MD-BSQLM. The approximate solutions agree with the analytical results quite well, exemplifying the precision of the algorithms displayed in this study. Figures 5.8, 5.9, and 5.10 of the MD-BSQLM for Burgers-Fisher equation, FitzHugh-Nagumo equation, also Coupled Burgers'equation show error analysis graphs.

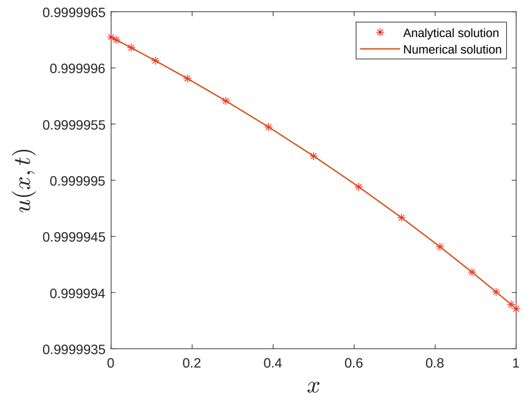


Figure 5.5: Exact VS Numerical of Burgers-Fisher equation for MD-BSQLM for $\mu = \eta = n = 1$.

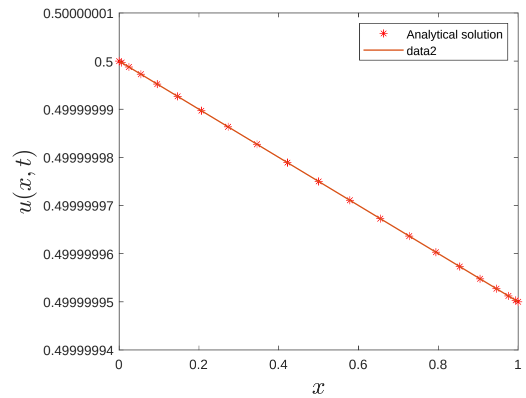
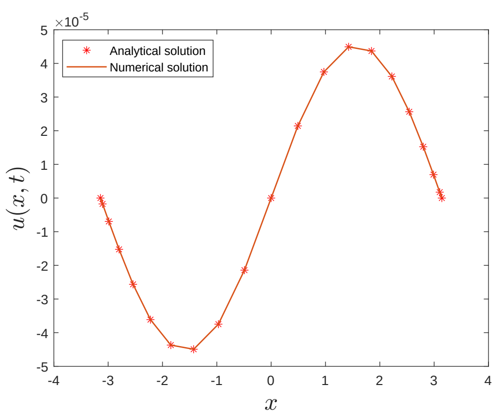
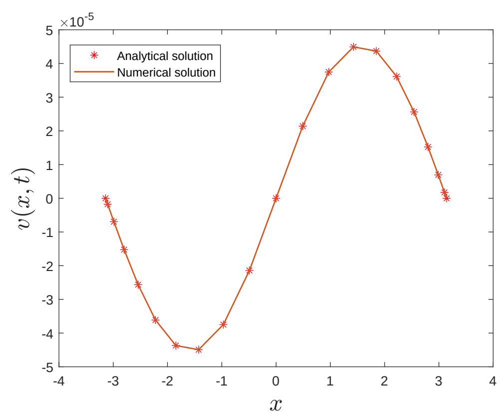


Figure 5.6: Exact VS Numerical of FitzHugh-Nagumo equation using MD-BSQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$.

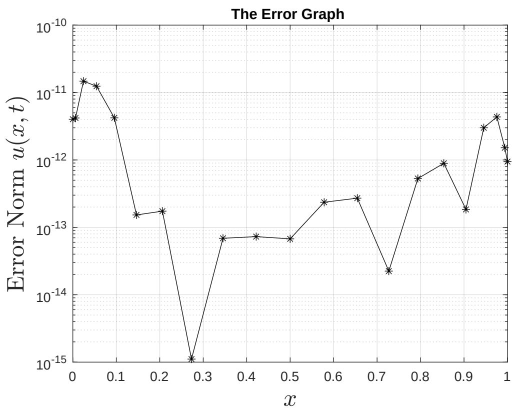


(a) Exact vs Numerical solution of $u(x,t)$.



(b) Exact vs Numerical solution of $v(x,t)$.

Figure 5.7: Exact vs Numerical solution of Coupled Burgers' equation using MD-BSQLM.



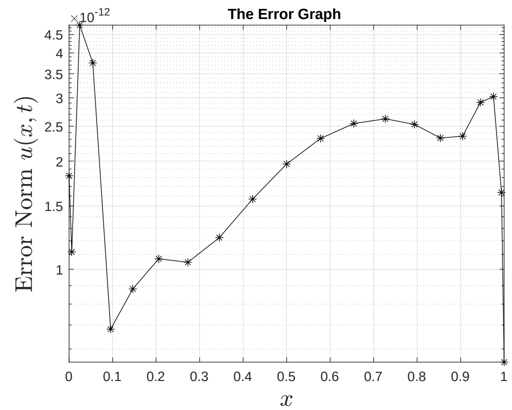
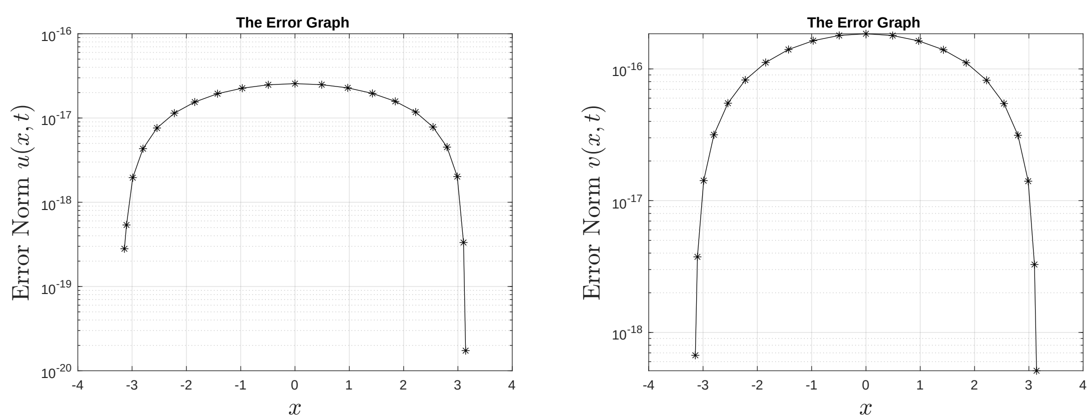


Figure 5.8: Error profile of Burgers-Fisher equation using MD-BSQLM $\nu = \eta = n = 1$.

Figure 5.9: Error profile of MD-BSQLM for FitzHugh-Nagumo equation for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$.



(a) Error profile of $u(x,t)$. **(b)** Error profile of $v(x,t)$.

Figure 5.10: Error profile of MD-BSQLM for Coupled Burgers' equation.

The maximum errors among the analytical as well as MD-BSQLM in time domain $t \in [0, 10]$ of the Burgers-Fisher equation, FitzHugh-Nagumo equation, including

Coupled Burgers' equation are given in Tables 5.3, 5.4, and 5.5. The outcomes were calculated in the $x \in [0, 1]$ space domain. The computational time required to produce the solutions is provided as an indication of the methods' computational efficiency. The accuracy of these methods is demonstrated clearly in 5.3, 5.4, and 5.5. The maximum errors of the Burgers-Fisher equation, FitzHugh-Nagumo equation, and Coupled Burgers' equation obtained with the MD-BSQLM could be seen in tables 5.3, 5.4, and 5.5. As the number of collocation points N_x increases, the accuracy appears to improve. It is noticeable when utilizing a small number of collocation points in x as well as t variables $N_x \leq 20$ and $N_t \leq 40$ achieves precise findings with errors of order 10^{-17}. This simply proves that MD-BSQLM is an effective method for solving the Burgers-Fisher equation, the FitzHugh-Nagumo equation, including the Coupled Burgers' equation. It is also noteworthy to mention that MD-BSQLM is computationally fast, yielding precise findings in split seconds.

$t \setminus N_x$	14	16	18	20
0.50	1.972e-13	7.550e-15	7.849e-13	5.360e-13
2.50	4.663e-14	1.295e-13	1.674e-13	1.132e-14
5.00	2.620e-13	2.398e-14	8.116e-14	3.847e-13
7.50	1.110e-14	2.901e-13	1.927e-12	3.058e-12
10.00	2.224e-13	5.298e-13	1.096e-12	6.750e-14
CPU time(sec)	0.061981	0.072012	0.098814	0.103631

Table 5.3: Maximum errors for Burgers-Fisher equation utilizing MD-BSQLM for $\nu = \eta = n = 1$.

$t \backslash N_x$	14	16	18	20
0.00	3.109e-14	1.328e-12	3.453e-13	1.647e-12
2.50	2.787e-14	7.269e-13	8.582e-13	1.917e-12
5.00	5.163e-15	2.550e-13	8.593e-14	1.260e-12
7.50	4.330e-15	1.218e-13	6.989e-14	1.994e-12
10.00	4.885e-14	3.170e-13	5.884e-14	1.963e-12
CPU time(sec)	0.130587	0.145790	0.160565	0.198814

Table 5.4: Maximum errors for FitzHugh-Nagumo equation utilizing MD-BSQLM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$.

$t \backslash N_x$	14	16	18	20
0.50	4.501e-11	1.334e-12	2.498e-15	3.497e-15
2.50	1.118e-11	2.627e-13	1.089e-15	2.637e-16
5.00	2.598e-12	2.897e-14	1.262e-15	9.059e-17
7.50	3.511e-13	2.988e-15	3.869e-16	1.507e-17
10.00	4.009e-14	2.967e-16	3.3.773e-16	7.813e-18
CPU time(sec)	0.046956	0.066319	0.087233	0.112457

(a) Maximum error for $u(x,t)$.

$t \backslash N_x$	14	16	18	20
0.50	4.501e-11	1.334e-12	2.498e-15	3.497e-15
2.50	1.118e-11	2.627e-13	1.089e-15	2.637e-16
5.00	2.598e-12	2.897e-14	1.262e-15	9.059e-17
7.50	3.511e-13	2.988e-15	3.869e-16	1.507e-17
10.00	4.009e-14	2.967e-16	3.3.773e-16	7.813e-18
CPU time(sec)	0.046956	0.066319	0.087233	0.112457

(b) Maximum error for $v(x,t)$.

Table 5.5: Maximum errors for the Coupled Burgers' equation obtained utilizing MD-BSQLM.

Figures 5.11, 5.12 and 5.13 shows the comparison of analytical and approximate solutions to the Burgers-Fisher equation, FitzHugh-Nagumo equation, and Coupled Burgers'equation using BHM. The approximate solutions agree with the analytical solution quite well, exemplifying the precision of the algorithms presented in this study. Figures 5.14, 5.15, and 5.16 show error analysis graphs of the BHM for Burgers-Fisher equation, FitzHugh-Nagumo equation, including Coupled Burgers'equation.

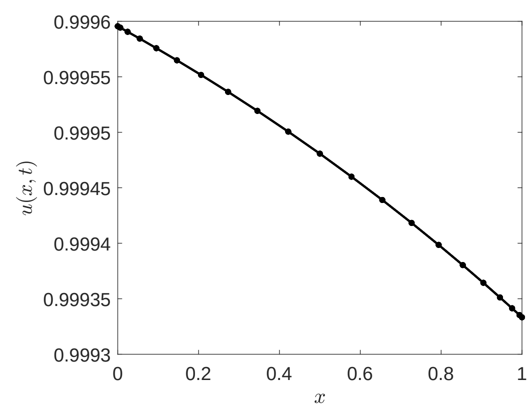


Figure 5.11: Exact vs Numerical solution of Burgers-Fisher equation for BHM for $\nu = \eta = n = 1$.

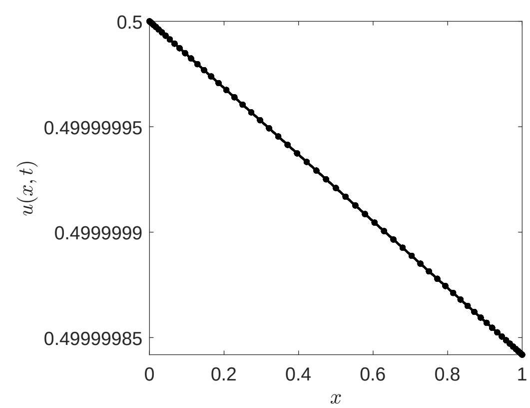
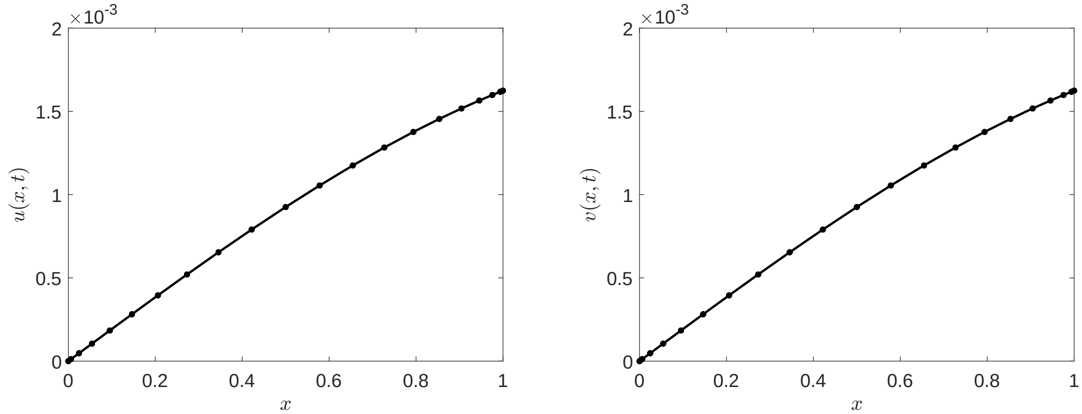


Figure 5.12: Exact vs Numerical solution of FitzHugh-Nagumo equation using BHM for $\mu = 5\times10^{-11}$and $\eta = 2\times10^{-13}$.



(a) Exact vs Numerical solution of $u(x,t)$. **(b)** Exact vs Numerical solution of $v(x,t)$.

Figure 5.13: Exact vs Numerical solution of Coupled Burgers' equation using BHM.

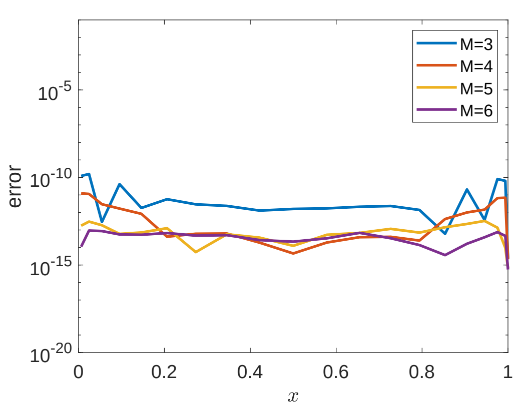


Figure 5.14: Error profile of Burgers-Fisher equation for BHM for $\alpha = \beta = n = 1$.

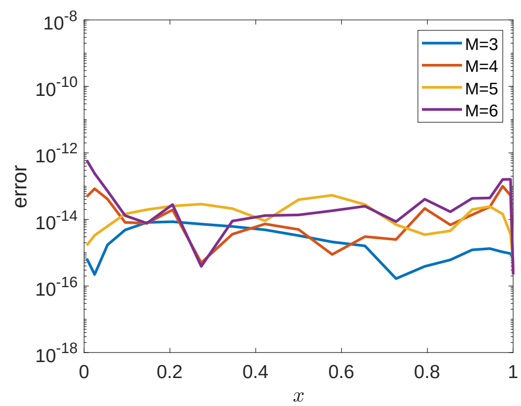
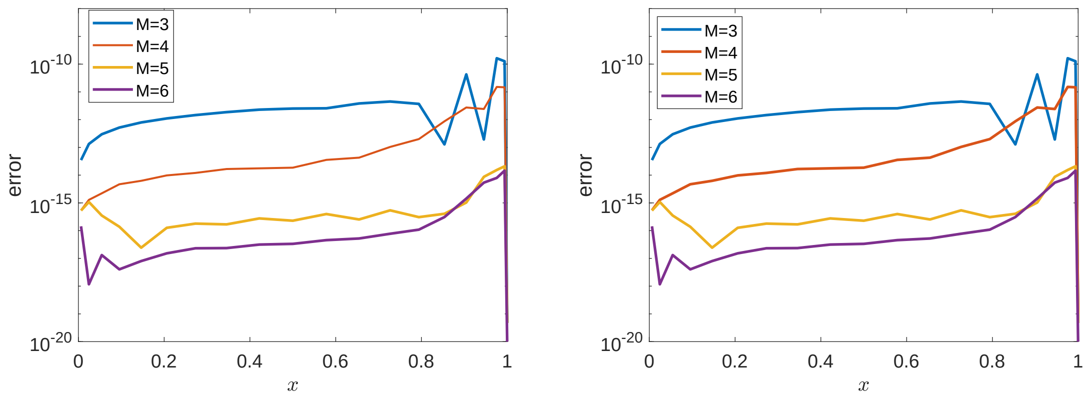


Figure 5.15: Error profile of BHM for FitzHugh-Nagumo equation for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$.



(a) Error profile of $u(x,t)$. **(b)** Error profile of $v(x,t)$.

Figure 5.16: Error profile of BHM for Coupled Burgers' equation.

In Tables 5.6, 5.7 and 5.8, the maximum errors among the exact as well as BHM in the time domain $t \in [0, 10]$ of the Burgers-Fisher equation, FitzHugh-Nagumo equation, including Coupled Burgers' equation are provided. The outcomes were calculated in the space region of $x \in [0, 1]$. The time required to generate the results is also provided as an indicator of the methods' computational efficiency. Tables 5.6, 5.7 and 5.8 amply attest the precision of these techniques. The maximum error of the Burgers-Fisher equation, FitzHugh-Nagumoo equation, including Coupled Burgers' equation attained using BHM can be found in Tables 5.6, 5.7 and 5.8. Accuracy decreases for the FitzHugh-Nagomu equation and increases for the Burgers-Fisher equation and Coupled Burgers'eqaution as M increases. Using a small number of collocation points $N_x \leq 20$ and $N_t \leq 40$, precise solutions with errors of order 10^{-15} are achieved. This clearly illustrates that the BHM is a powerful method for solving the Burgers-Fisher equation, FitzHugh-Nagumo equation, including Coupled Burgers' equation, and it is also computationally reliable, achieving precise data in a matter of seconds.

t	$M=3$	$M=4$	$M=5$	$M=6$
0.00	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00
2.50	7.4933e-09	3.2794e-10	5.0350e-12	1.0514e-13
5.00	8.3324e-10	3.1085e-11	4.4287e-13	6.2395e-14
7.50	2.2348e-10	1.7964e-11	2.2182e-13	7.0055e-14
10.00	1.5796e-10	1.2352e-11	3.2951e-13	9.2371e-14
CPU time(sec)	0.190243	0.186671	0.320958	0.385719

Table 5.6: Maximum errors for Burgers-Fisher equation utilizing BHM for $\nu = \eta = n = 1$.

t	$M=3$	$M=4$	$M=5$	$M=6$
0.00	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00
2.50	9.1038e-15	4.8517e-14	5.3124e-14	1.9962e-13
5.00	8.5487e-15	7.4329e-14	5.3124e-14	3.5677e-13
7.50	8.4377e-15	8.8984e-14	5.3457e-14	4.8933e-13
10.00	8.5487e-15	9.9976e-14	5.3457e-14	6.0485e-13
CPU time(sec)	0.170630	0.268356	0.258403	0.385030

Table 5.7: Maximum errors for FitzHugh-Nagumo equation utilizing BHM for $\mu = 5 \times 10^{-11}$and $\eta = 2 \times 10^{-13}$.

t	$M = 3$	$M = 4$	$M = 5$	$M = 6$
0.00	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00
2.50	5.5749e-09	1.2730e-10	7.1783e-13	4.0037e-14
5.00	4.5357e-10	4.7354e-11	5.8224e-14	2.4936e-14
7.50	2.2915e-10	2.4827e-11	2.5557e-14	1.8284e-14
10.00	1.6314e-10	1.5115e-11	2.0866e-14	1.4523e-14
CPU time(sec)	0.278920	0.405877	0.506662	0.624847

(a) Maximum error for $u(x, t)$.

t	$M = 3$	$M = 4$	$M = 5$	$M = 6$
0.00	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00
2.50	5.5749e-09	1.2730e-10	7.1783e-13	4.0037e-14
5.00	4.5357e-10	4.7354e-11	5.8224e-14	2.4936e-14
7.50	2.2915e-10	2.4827e-11	2.5557e-14	1.8284e-14
10.00	1.6314e-10	1.5115e-11	2.0866e-14	1.4523e-14
CPU time(sec)	0.278920	0.405877	0.506662	0.624847

(b) Maximum error for $v(x, t)$.

Table 5.8: Maximum errors for Coupled Burgers' equation utilizing BHM.

Figures 5.17 - 5.20 compares the exact and numerical solutions to the Burgers-Fisher, FitzHugh-Nagumo, and Coupled Burgers' equations. The numerical solutions agree perfectly with the exact solutions, signifying the precision of the algorithms utilized in this research.

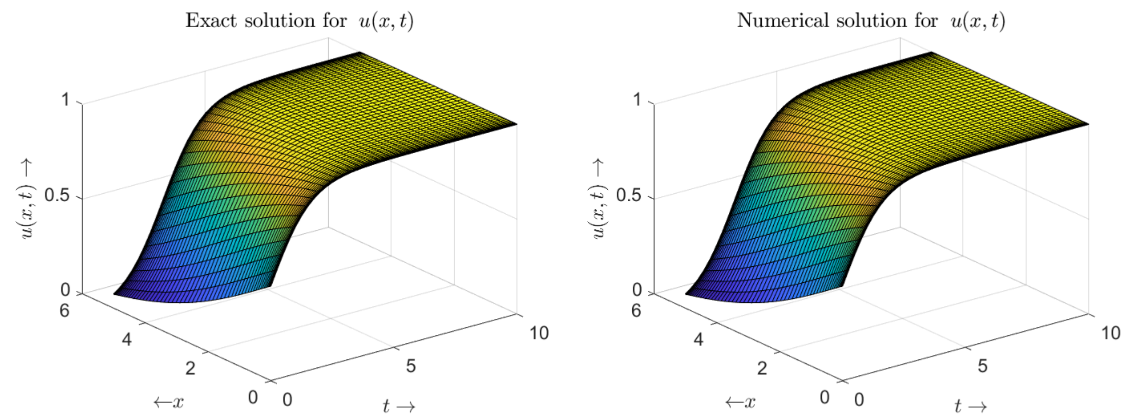


Figure 5.17: Exact solution and Numerical solution of Burgers-Fisher equation

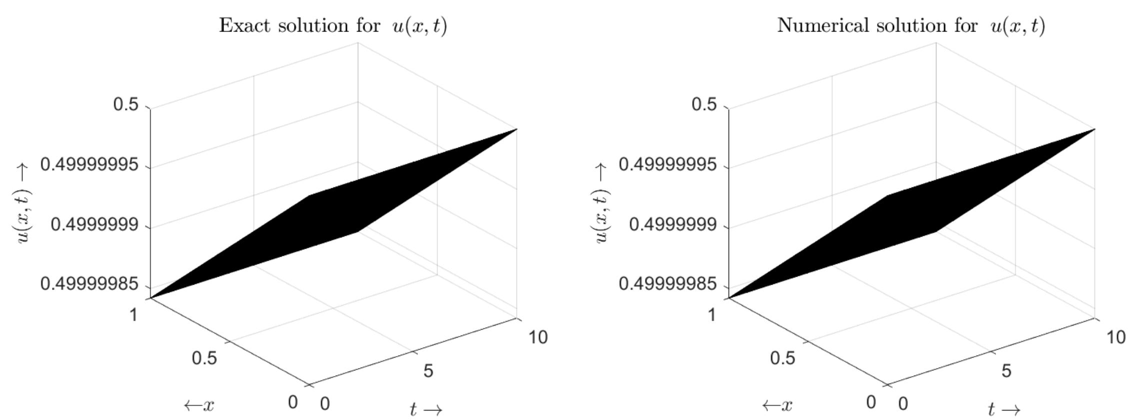
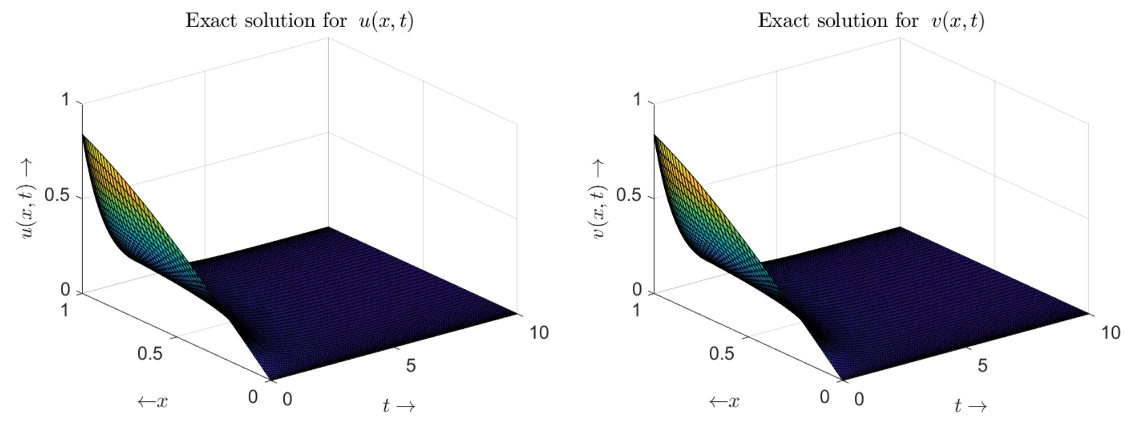
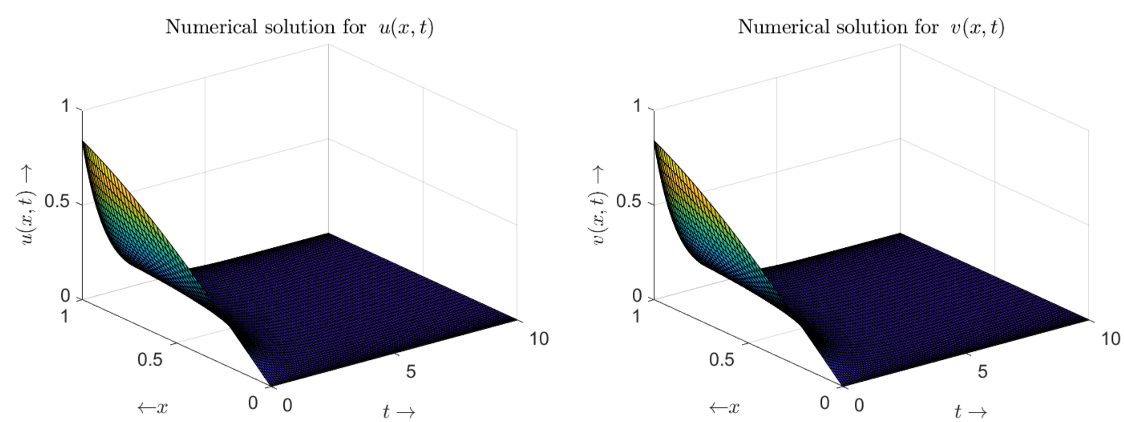


Figure 5.18: Exact solution and Numerical solution of FitzHugh-Nagumo equation



(a) Exact solution of $u(x,t)$ **(b)** Exact solution of $v(x,t)$

Figure 5.19: Exact solution of Coupled Burgers' equation



(a) Numerical solution of $u(x,t)$ **(b)** Numerical solution of $v(x,t)$

Figure 5.20: Numerical solution of Coupled Burgers' equation

5.1 Summary

We discussed the numerical results generated in Chapter 4 that used SQLM, MD-BSQLM, and BHM in this chapter. The norm error was calculated using both the exact and approximate solutions to make comparisons to the SQLM, MD-BSQLM, and BHM. In this study, it was proven that MD-BSQLM outperforms SQLM and BHM in precision, convergence, and CPU time for the three nonlinear equations solved in the preceding chapter. Furthermore, the concept of dividing the major domain into sub-intervals enhances the technique's precision significantly. The greater the number of sub-intervals, the more reliable the technique. Additionally, evidence suggests that a larger number of nonoverlapping intervals results in greater precision, faster convergence, and a relatively short time to achieve solutions when using the MD-BSQLM.

Chapter 6

Conclusion

The purpose of this work was to look into the applicability of three methods: SQLM, MD-BSQLM, and BHM. The most important goal was to carry out an evaluation study and apply the three methods to the Burgers-Huxley equation and other nonlinear evolution PDEs. NLPDEs of the second order that were utilized in this study. First, the discussion focused on the history of the nonlinear evolution of PDEs, in order to gain a perception of what encouraged researchers to seek out solutions to these equations. The Lie group methods, spectral methods, and BHMs were then exhaustively discussed, investigating the benefits of using them as well as the motives for using them as the methods of preference for numerous researchers and also for this study.

Chapter 2 goes over some of the most essential denotations and outcomes of the Lie group analysis of PDEs, as well as the procedure for determining the Lie symmetries of PDEs. Moreover, SQLM, MD-BSQLM, and BHM were discussed on how to be applied to nonlinear evolution problems, together with a description of how QLM and multidomain approach work on nonlinear evolution PDEs.

Chapter 3 discussed the study's main focuses, which include Lie symmetries, the SQLM, MD-BSQLM, and BHM, respectively. The most important objective of this research was to evaluate these techniques for solving the nonlinear Burgers-Huxley equation and analyze the precision, robustness, and efficiency of each approach. The numerical simulations were conducted and the outcomes were discussed. When the results from these three methods are compared, it is comprehensible that the MD-BSQLM produces more precise solutions than the SQLM and the BHM. According to the results, the MD-BSQLM provides effective solutions in spite of the small intervals in both space including time variables, with a CPU time of under a second (0.187094 seconds), whereas the SQLM generates results in the x variable only in the small interval within 0.001727 seconds and the BHM yields solutions in the small interval in both the x and t variables within 0.511625 seconds. The marks of this study show that MD-BSQLM performs computationally better than SQLM and BHM.

In Chapters 4 and 5, numerical simulations were performed and the results were discussed. The equations of Burgers-Fisher, FitzHugh-Nagumo, and Coupled Burgers were solved. When the three proposed methods' results are compared, it is evident that the MD-BSQLM provides more reliable data than the SQLM and BHM. The study revealed that the MD-BSQLM provides better performance including small space and also time intervals, with CPU time of less than a second. In addition, no solution to the Coupled Burgers' equation is provided by the SQLM. As a result, we can conclude that SQLM is inefficient for solving some evolution problems. According to the findings of this study, MDBSQLM outperforms SQLM and BHM significantly.

The SQLM, MD-BSQLM, and BHM were implemented in my dissertation to solve second-order nonlinear evolution equations with analytical solutions. It is evident that the MD-BSQLM is a very effective and also efficient approach for solving broad classes of problems such as the ones examined. In terms of applications, it was proved that the previously described MD-BSQLM was significantly more computationally efficient than the SQLM and BHM, requiring less computational time and producing smaller norm errors. All three methods were linearized using QLM. In this study, MD-BSQLM outperformed SQLM and BHM in solving the examples.

Bibliography

[1] J. M. Burgers, A mathematical model illustrating the theory of turbulence, Advances in applied mechanics 1 (1948) 171–199.

[2] M. Sari, G. Gürarslan, A. Zeytinoğlu, High-order finite difference schemes for numerical solutions of the generalized Burgers–Huxley equation, Numerical Methods for Partial Differential Equations 27 (5) (2011) 1313–1326.

[3] M. Dehghan, J. M. Heris, A. Saadatmandi, Application of semi-analytic methods for the FitzHugh–Nagumo equation, which models the transmission of nerve impulses, Mathematical Methods in the Applied Sciences 33 (11) (2010) 1384–1398.

[4] B. Inan, A. Bahadir, Numerical solutions of the generalized Burgers-Huxley equation by implicit exponential finite difference method, Journal of Applied Mathematics, Statistics and Informatics 11 (2) (2015) 57–67.

[5] N. Alinia, M. Zarebnia, A numerical algorithm based on a new kind of tension b-spline function for solving Burgers-Huxley equation, Numerical Algorithms 82 (4) (2019) 1121–1142.

[6] İ. Dağ, Y. Dereli, Numerical solutions of KdV equation using radial basis functions, Applied Mathematical Modelling 32 (4) (2008) 535–546.

[7] B. Zhen, Z. Song, The study for synchronization between two Coupled FitzHugh-Nagumo neurons based on the laplace transform and the adomian decomposition method, Neural Plasticity (2021).

[8] H. Ali, M. Kamrujjaman, M. S. Islam, Numerical computation of FitzHugh-Nagumo equation: A novel galerkin finite element approach, International Journal of Mathematical Research 9 (1) (2020) 20–27.

[9] M. Fernandino, C. Dorao, The least squares spectral element method for the Cahn–Hilliard equation, Applied mathematical modelling 35 (2) (2011) 797–806.

[10] X. Ye, The fourier collocation method for the Cahn-Hilliard equation, Computers & Mathematics with Applications 44 (1-2) (2002) 213–229.

[11] M. Dehghan, D. Mirzaei, A numerical method based on the boundary integral equation and dual reciprocity methods for one-dimensional Cahn–Hilliard equation, Engineering analysis with boundary elements 33 (4) (2009) 522–528.

[12] K. Park, C. Dorao, M. Fernandino, Numerical solution of Coupled Cahn-Hilliard and Navier-Stokes system using the least-squares spectral element method, (2016).

[13] W. Zahra, Trigonometric b-spline collocation method for solving Phi-Four and Allen–Cahn equations, Mediterranean journal of mathematics 14 (3) (2017) 1–19.

[14] G. Dahlquist, Å. Björck, Numerical methods, Courier Corporation, (2003).

[15] V. Thomée, From finite differences to finite elements a short history of numerical analysis of partial differential equations, in: Numerical analysis: Historical developments in the 20th century, Elsevier, (2001), pp. 361–414.

[16] P. V. Nielsen, F. Allard, H. B. Awbi, L. Davidson, A. Schälin, Computational fluid dynamics (1995).

[17] J. N. Reddy, Introduction to the finite element method, McGraw-Hill Education, (2019).

[18] J. M. Augustine, An application of modern analytical solution techniques to nonlinear partial differential equations., Ph.D. thesis (2014).

[19] M. Darwish, F. Moukalled, The finite volume method in computational fluid dynamics: an advanced introduction with OpenFOAM® and Matlab®, Springer, (2021).

[20] R. Eymard, T. Gallouët, R. Herbin, Finite volume methods, Handbook of numerical analysis 7 (2000) 713–1018.

[21] R. E. Bellman, R. K. Quasilinearization, Nonlinear Boundary-Value Problems, American Elsevier Publishing Co., Inc., New York, (1965).

[22] S. S. Motsa, A new spectral local linearization method for nonlinear boundary layer flow problems, Journal of Applied Mathematics (2013).

[23] L, Leon. An iterative technique for the rectification of observed distributions, The astronomical journal 79 (1974): 745.

[24] S. Motsa, A new spectral relaxation method for similarity variable nonlinear boundary layer flow systems, Chemical Engineering Communications 201 (2) (2014) 241–256.

[25] S. Motsa, P. Dlamini, M. Khumalo, Spectral relaxation method and spectral quasilinearization method for solving unsteady boundary layer flow problems, Advances in Mathematical Physics (2014).

[26] P. G. Dlamini, M. Khumalo, S. S. Motsa, A note on the multi-stage spectral relaxation method for chaos control and synchronization, International Journal of Nonlinear Sciences and Numerical Simulation 15 (5) (2014) 289–298.

[27] S. Shateyi, G. T. Marewo, On a new numerical analysis of the hall effect on mhd flow and heat transfer over an unsteady stretching permeable surface in the presence of thermal radiation and heat source/sink, Boundary Value Problems (1) (2014) 1–17.

[28] S. Motsa, F. Awad, M. Khumalo, Nonlinear nanofluid flow over heated vertical surface with sinusoidal wall temperature variations, in: Abstract and Applied Analysis, Vol. 2014, Hindawi, (2014).

[29] S. Motsa, V. Magagula, P. Sibanda, A bivariate chebyshev spectral collocation quasilinearization method for nonlinear evolution parabolic equations, The Scientific World Journal (2014).

[30] H. Muzara, S. Shateyi, G. T. Marewo, On the bivariate spectral quasi-linearization method for solving the two-dimensional Bratu problem, Open Physics 16 (1) (2018) 554–562.

[31] V. M. Magagula, S. S. Motsa, P. Sibanda, P. G. Dlamini, On a bivariate spectral relaxation method for unsteady magneto-hydrodynamic flow in porous media, SpringerPlus 5 (1) (2016) 1–15.

[32] S. S. Motsa, I. L. Animasaun, A new numerical investigation of some thermo-physical properties on unsteady mhd non-darcian flow past an impulsively started vertical surface, Thermal Science 19 (suppl. 1) (2015) 249–258.

[33] V. M. Magagula, S. S. Motsa, P. Sibanda, Multidomain bivariate pseudo-spectral quasilinearization method for systems of nonlinear partial differential equations, Computational and Mathematical Methods 2 (4) (2020) e1096.

[34] I. Oyelakin, S. Mondal, P. Sibanda, A multi-domain spectral method for non-darcian mixed convection flow in a power-law fluid with viscous dissipation, Physics and Chemistry of Liquids 56 (6) (2018) 771–789.

[35] V. M. Magagula, On the multidomain bivariate spectral local linearisation method for solving systems of nonsimilar boundary layer partial differential equations, International Journal of Mathematics and Mathematical Sciences (2019).

[36] G. A. Kemper, Linear multistep methods for a class of functional differential equations, Numerische Mathematik 19 (5) (1972) 361–372.

[37] R. T. Keller, Q. Du, Discovery of dynamics using linear multistep methods, SIAM Journal on Numerical Analysis 59 (1) (2021) 429–455.

[38] D. R. Durran, The third-order adams-bashforth method: An attractive alternative to leapfrog time differencing, Monthly weather review 119 (3) (1991) 702–720.

[39] T. Allahviranloo, N. Ahmady, E. Ahmady, Numerical solution of fuzzy differential equations by predictor–corrector method, Information Sciences 177 (7) (2007) 1633–1647.

[40] A. O. Adesanya, D. M. Udoh, A. Ajileye, A new hybrid block method for the solution of general third order initial value problems of ordinary differential equations, International journal of pure and applied mathematics 86 (2) (2013) 365–375.

[41] M. Farhan, Z. Omar, F. Mebarek-Oudina, J. Raza, Z. Shah, R. Choudhari, O. Makinde, Implementation of the one-step one-hybrid block method on the nonlinear equation of a circular sector oscillator, Computational Mathematics and Modeling 31 (1) (2020) 116–132.

[42] M. Alkasassbeh, Z. Omar, Generalized hybrid one-step block method involving fifth derivative for solving fourth-order ordinary differential equation directly, Journal of Applied Mathematics (2017).

[43] H. Ramos, Z. Kalogiratou, T. Monovasilis, T. Simos, An optimized two-step hybrid block method for solving general second order initial-value problems, Numerical Algorithms 72 (4) (2016) 1089–1102.

[44] W. B. Gragg, H. J. Stetter, Generalized multistep predictor-corrector methods, Journal of the ACM (JACM) 11 (2) (1964) 188–209.

[45] W. Enright, Second derivative multistep methods for stiff ordinary differential equations, SIAM Journal on Numerical Analysis 11 (2) (1974) 321–331.

[46] F. Ngwane, S. Jator, Block hybrid-second derivative method for stiff systems, International Journal of Pure and Applied Mathematics 80 (4) (2012) 543–559.

[47] R. Sahi, S. Jator, N. Khan, A simpson's-type second derivative method for stiff systems, International journal of pure and applied mathematics 81 (4) (2012) 619–633.

[48] M. Alkasassbeh, Z. Omar, Implicit one-step block hybrid third-derivative method for the direct solution of initial value problems of second-order ordinary differential equations, Journal of Applied Mathematics (2017).

[49] W. Malfliet, The tanh method: a tool for solving certain classes of nonlinear evolution and wave equations, Journal of Computational and Applied Mathematics 164 (2004) 529–541.

[50] W. Hereman, W. Malfliet, The tanh method: a tool to solve nonlinear partial differential equations with symbolic software, in: Proceedings 9th World Multi-Conference on Systemics, Cybernetics and Informatics, Orlando, FL (2005) 165–168.

[51] W. Malfliet, W. Hereman, The tanh method: I. exact solutions of nonlinear evolution and wave equations, Physica Scripta 54 (6) (1996) 563.

[52] W. Malfliet, The tanh method: a tool for solving certain classes of non-linear pdes, Mathematical methods in the applied sciences 28 (17) (2005) 2031–2035.

[53] M. Wang, X. Li, J. Zhang, The (g'/g)-expansion method and travelling wave solutions of nonlinear evolution equations in mathematical physics, Physics Letters A 372 (4) (2008) 417–423.

[54] J. Zhang, X. Wei, Y. Lu, A generalized (g'/g)-expansion method and its applications, Physics Letters A 372 (20) (2008) 3653–3658.

[55] J. Zhang, F. Jiang, X. Zhao, An improved (g'/g)-expansion method for solving nonlinear evolution equations, International Journal of Computer Mathematics 87 (8) (2010) 1716–1725.

[56] H. Naher, F. A. Abdullah, New approach of (g'/g)-expansion method and new approach of generalized (g'/g)-expansion method for nonlinear evolution equation, AIP Advances 3 (3) (2013) 032116.

[57] H. Liu, J. Li, Q. Zhang, Lie symmetry analysis and exact explicit solutions for general Burgers' equation, Journal of Computational and Applied Mathematics 228 (1) (2009) 1–9.

[58] H. Liu, J. Li, Lie symmetry analysis and exact solutions for the short pulse equation, Nonlinear analysis: theory, methods & applications 71 (5-6) (2009) 2126–2133.

[59] R. K. Gazizov, N. H. Ibragimov, Lie symmetry analysis of differential equations in finance, Nonlinear Dynamics 17 (4) (1998) 387–407.

[60] S. Helgason, Sophus lie, the mathematician, in: Contenido en Proceedings of The Sophus Lie Memorial Conference, Oslo (1994) 3–21.

[61] G. Gaeta, Nonlinear symmetries and nonlinear equations, Springer Science & Business Media 299 (2012).

[62] N. H. Ibragimov, CRC handbook of Lie group analysis of differential equations, CRC press 3 (1995).

[63] P. J. Olver, Applications of Lie groups to differential equations, Springer Science & Business Media, 107 (1993).

[64] C. I. Gheorghiu, Spectral methods for differential problems, Casa Cluj-Napoca, (2007).

[65] M. Y. Hussaini, T. A. Zang, Spectral methods in fluid dynamics, Annual review of fluid mechanics 19 (1) (1987) 339–367.

[66] L. N. Trefethen, Spectral methods in MATLAB, SIAM, (2000).

[67] D. Mathale, P. Dlamini, M. Khumalo, Compact finite difference relaxation method for chaotic and hyperchaotic initial value systems, Computational and Applied Mathematics 37 (4) (2018) 5187–5202.

[68] S. S. Motsa, V. M. Magagula, P. Sibanda, The multi-domain spectral relaxation method for chaotic systems of ordinary differential equations, in: Chaotic Systems: Dyn Algo Sync, Nova Science Publishers, Inc (2017) 29–52.

[69] H. Bateman, Some recent researches on the motion of fluids, Monthly Weather Review 43 (4) (1915) 163–170.

[70] G. B. Whitham, Linear and nonlinear waves, John Wiley & Sons (2011).

[71] Y. Kyrychko, M. Bartuccelli, K. Blyuss, Persistence of travelling wave solutions of a fourth order diffusion system, Journal of Computational and Applied Mathematics 176 (2) (2005) 433–443.

[72] X. Deng, Travelling wave solutions for the generalized Burgers–Huxley equation, Applied Mathematics and Computation 204 (2) (2008) 733–737.

[73] S. S. Nourazar, M. Soori, A. Nazari-Golshan, On the exact solution of Newell-Whitehead-Segel equation using the homotopy perturbation method, arXiv preprint arXiv:1502.08016 (2015).

[74] K. M. Agbavon, A. R. Appadu, Construction and analysis of some nonstandard finite difference methods for the FitzHugh–Nagumo equation, Numerical Methods for Partial Differential Equations 36 (5) (2020) 1145–1169.

[75] Y. Bai, T. Chaolu, S. Bilige, Solving Huxley equation using an improved pinn method, Nonlinear Dynamics 105 (4) (2021) 3439–3450.

[76] A. Aksenov, K. Druzhkov, Symmetries and reductions of Burgers–Huxley equation, in: Journal of Physics: Conference Series, IOP Publishing 788 (2017) 012002.

[77] J. K. Hunter, Nonlinear evolution equations, University of California, Davis (1996).

[78] A.-A. Hyder, M. Barakat, General improved kudryashov method for exact solutions of nonlinear evolution equations in mathematical physics, Physica Scripta 95 (4) (2020) 045212.

[79] Z. Waheed, A. Waleed,M El-Azab, B-Spline Soliton Solution of the Fifth Order KdV Type Equations, AIP Conference Proceedings, (2013) 1558. 10.1063–1.4825554.

[80] A. A. M. Arafa, S. Z. Rida, Numerical solutions for some generalized coupled nonlinear evolution equations, Mathematical and Computer Modelling 56.11-12 (2012): 268-277.

[81] T. R Thiab, M. I. Ablowitz. Analytical and numerical aspects of certain nonlinear evolution equations. II. Numerical, nonlinear Schrödinger equation, Journal of computational physics 55.2 (1984): 203-230.

[82] W. J. Andre, A. Cloot. Spectral methods and mappings for evolution equations on the infinite line, Computer Methods in Applied Mechanics and Engineering 80.1-3 (1990): 467-481.

[83] V. Chandraker, A. Awasthi, S. Jayaraj, Numerical treatment of Burger-Fisher equation, Procedia Technology 25 (2016) 1217–1225.

[84] J. Mendoza, C. Muriel, New exact solutions for a generalised Burgers-Fisher equation, Chaos, Solitons Fractals 152 (2021) 111360.

[85] M. Namjoo, S. Zibaei, Numerical solutions of FitzHugh–Nagumo equation by exact finite-difference and nsfd schemes, Computational and Applied Mathematics 37 (2) (2018) 1395–1411.

[86] P. Browne, E. Momoniat, F. Mahomed, A generalized FitzHugh–Nagumo equation, Nonlinear Analysis: Theory, Methods Applications 68 (4) (2008) 1006–1015.

[87] R. Mittal, G. Arora, Numerical solution of the Coupled Viscous Burgers' equation, Communications in Nonlinear Science and Numerical Simulation 16 (3) (2011) 1304–1313.

[88] C. M. Khalique, S. A. Abdallah, Coupled Burgers' equations governing polydispersive sedimentation; a Lie symmetry approach, Results in Physics 16 (2020) 102967.