COMPUTATIONAL AND NUMERICAL ANALYSIS OF DIFFERENTIAL EQUATIONS USING SPECTRAL BASED COLLOCATION METHODS

A THESIS SUBMITTED TO THE UNIVERSITY OF KWAZULU-NATAL FOR THE DEGREE OF DOCTOR OF PHILOSOPHY OF SCIENCE IN THE COLLEGE OF AGRICULTURE, ENGINEERING & SCIENCE

By

Mutua Samuel

School of Mathematics, Statistics & Computer Science

2019

Abstract

In this thesis, we develop accurate and computationally efficient spectral collocation-based methods, both modified and new, and apply them to solve differential equations. Spectral collocation-based methods are the most commonly used methods for approximating smooth solutions of differential equations defined over simple geometries. Procedurally, these methods entail transforming the governing differential equation(s) into a system of linear algebraic equations that can be solved directly. Owing to the complexity of expanding the numerical algorithms to higher dimensions, as reported in the literature, researchers often transform their models to reduce the number of variables or narrow them down to problems with fewer dimensions. Such a process is accomplished by making a series of assumptions that limit the scope of the study. To address this deficiency, the present study explores the development of numerical algorithms for solving ordinary and partial differential equations defined over simple geometries. The solutions of the differential equations considered are approximated using interpolating polynomials that satisfy the given differential equation at selected distinct collocation points preferably the Chebyshev-Gauss-Lobatto points. The size of the computational domain is particularly emphasized as it plays a key role in determining the number of grid points that are used; a feature that dictates the accuracy and the computational expense of the spectral method. To solve differential equations defined on large computational domains much effort is devoted to the development and application of new multidomain approaches, based on decomposing large spatial domain(s) into a sequence of overlapping subintervals and a large time interval into equal non-overlapping subintervals. The rigorous analysis of the numerical results confirms the superiority of these multiple domain techniques in terms of accuracy and computational efficiency over the single domain approach when applied to problems defined over large domains. The structure of the thesis indicates a smooth sequence of constructing spectral collocation method algorithms for problems across different dimensions. The process of switching between dimensions is explained by presenting the work in chronological order from a simple one-dimensional problem to more complex higher-dimensional problems. The preliminary chapter explores solutions of ordinary differential equations. Subsequent chapters then build on solutions to partial differential

equations in order of increasing computational complexity. The transition between intermediate dimensions is demonstrated and reinforced while highlighting the computational complexities involved. Discussions of the numerical methods terminate with development and application of a new method namely; the trivariate spectral collocation method for solving two-dimensional initial-boundary value problems. Finally, the new error bound theorems on polynomial interpolation are presented with rigorous proofs in each chapter to benchmark the adoption of the different numerical algorithms. The numerical results of the study confirm that incorporating domain decomposition techniques in spectral collocation methods work effectively for all dimensions, as we report highly accurate results obtained in a computationally efficient manner for problems defined on large domains. The findings of this study thus lay a solid foundation to overcome major challenges that numerical analysts might encounter.

Dedication

This thesis is wholeheartedly dedicated to my beloved wife Benerdate who has been my source of inspiration and gave me strength when I thought of giving up, who continually provided her moral, spiritual, emotional, and financial support.

To my parents, Mary and Samuel, who had been encouraging me throughout my life and who have actively supported me in my determination to find and realize my potential, and make this contribution to our world.

And lastly, a special feeling of gratitude to my brother, Patrick, sisters, Redemtah and Cecilia, and friends for being my best cheerleaders throughout the entire doctorate program.

Declaration

I am aware of and understand the university's policy on plagiarism and declare that this thesis is composed by myself and it is a presentation of my original research work. Wherever contribution of others are involved, such has been explicitly stated in the text, with due reference to the supporting literature and resources, and acknowledgment of collaborative research and discussions. The work presented in the thesis has not been submitted in support of another degree or qualification from this or any other university or institution of higher learning.

The work presented in this thesis is in the form of journals articles that are jointly co-authored by myself and my supervisor, and which have submitted to various academic journals for consideration for publication. As the student, my contribution was to prepare the entire manuscripts, with my supervisor's role being to proofread, and give comments and suggestions on improving the quality of each manuscript that I incorporated before communicating with the journal.

The work presented in Chapter 2 has been previously submitted to Calcolo Journal for consideration for publication and it is under review.

The work presented in Chapter 3 has been published in the Wave Motion Journal. This work was presented during the 2016 Post Graduate Research and Innovation Symposium organized by the College of Engineering, Agriculture and Science of the University of KwaZulu Natal, held on 1^{st} November 2016 at Howard College Campus, Durban, South Africa where the I won first prize in the oral presentations category. Further, this work was presented at the 35^{th} Southern Africa Mathematical Sciences Association (SAMSA) conference, held between $21^{st} - 24^{th}$ November, 2016 at the University of Pretoria, South Africa, and the 4^{th} Strathmore International Mathematical Conference (SIMC 2017), held at Strathmore University, Nairobi, Kenya, during the week of $19^{th} - 23^{rd}$ June, 2017.

The work presented in Chapter 4 has been previously submitted to Journal of Mathematical Analysis and Applications and it is still under review. The work was presented during the 42^{nd} Southern

Africa Symposium of Numerical and Applied Mathematics (SANUM) conference held between $4^{th} - 6^{th}$ April 2018 at Stellenbosch University, South Africa.

A revised version of the manuscript containing the work presented in Chapter 5 has been submitted for consideration for publication to the Journal of Computers and Mathematics with Applications and it is currently under review. This work was also presented during the 2018 Post Graduate Research and Innovation Symposium organized by the College of Engineering, Agriculture and science of the University of KwaZulu Natal, held on 25^{th} October, 2018 at Westville Campus, Durban, South Africa where I won the first prize under the oral presentations category.

The work presented in Chapter 6 was presented during the Pan African Conference on Science, Computing, and Telecommunications, held at the University of Swaziland between $11^{th} - 13^{th}$ March, 2019. The work has also been accepted for publication in the Springer special issue journal as conference proceedings.

The work contained in this thesis was done under the guidance of Professor Sandile Motsa at the University of KwaZulu Natal, the Republic of South Africa.

Mutua Samuel		21/08/2019
Candidate's Name:	Sign:	Date:

In my capacity as supervisor of the candidate's thesis, I certify that the above statements are true to the best of my knowledge.

Prof. Sandile Motsa		21/08/2019
Supervisor's Name:	Sign:	Date:

Acknowledgements

I wish to thank my supervisor Prof. Sandile Motsa, who was more generous with his expertise and precious time taken on reflecting, reading, encouraging and pointing out areas of improvement and, most importantly, for his patience exercised throughout the entire process. I am very fortunate and grateful to honour the supervisors' ideas that he shared, which were paramount to the realization of this thesis. His excitement over the work and willingness to provide timely feedback made the completion of this research an enjoyable experience.

I would like to acknowledge and thank the staff training committee of Taita Taveta University, Kenya for financial support. Special thanks also go to the University of KwaZulu Natal and particularly the College of Agriculture, Engineering, and Science for sponsorship via doctoral tuition remission and allowing me to conduct my research in the college premises.

I would like to thank the administrator in the School of Mathematics, Statistics, and Computers Science for the assistance granted from time to time, particularly in ensuring that travel and accommodation arrangements were made timeously when I needed to attend research seminars, workshops, and conferences away from the school. The enabling study environment in the research office, with well-maintained furniture and a plentiful supply of stationery, is appreciated.

I am thankful to the editor, for the copy-editing work that improved the flow of the content of this thesis.

Finally, I would like to thank journal reviewers for their comments and suggestions on improving the quality of manuscripts submitted to their academic journals for consideration for publication.

Table of Contents

Chapter 1

Introduction

1.1 General Brief on Differential Equations

Differential equations, both ordinary and partial, are fundamental in modelling problems in science, economics, and engineering [1]. For instance, differential equations find application in the mass-spring-damper system and in checking the stability of the system. The study of simple harmonic motion, population dynamics, the spread of disease, and exponential growth; and decay, all lead to differential equations. In engineering, we often encounter differential equations in the field of fluid dynamics, which has direct application in the design of containers and funnels, in heat conduction analysis applied in the design of heat spreaders in microelectronics, and in combined heat conduction and convection on which the design of heating and cooling chambers is dependent [2]. Rigid-body dynamic analysis, among other physical applications, also relies on solutions to differential equations. In economics, differential equations assist in finding optimal investment strategies. When studying such systems, researchers attempt to find formulas that connect a dependent variable to one or more independent variables to describe the system's behaviour at any given instant, based on prescribed initial and boundary conditions. The complexity involved in obtaining the solution to differential equations increases as one moves from ordinary to the partial differential equations, with well-known difficulties encountered in the case of a nonlinear problem [3].

Solving a differential equation involves, primarily, finding a self-contained formula that satisfies the differential equation. However, this is only possible for a few classes of problems, notably the linear ones [4]. There exist analytical methods for solving differential equations, and the choice of a suitable method depends on the nature of the problem at hand. For linear parabolic partial differential equations (PDEs) with constant coefficients, the method of separation of variables

can be used to solve them; it is based on the assumption that the solution can be expressed as a product of two terms, each depending on a single parameter. The method of characteristics (similarity transformation method), involves finding a characteristic curve that reduces the PDE to an ordinary differential equation (ODE), or simply transforming the PDE to an equivalent PDE that is easier to solve. The integral transform method and method of change of variables seek to transform the original PDE to a simple one [5]. It is, nevertheless, difficult to obtain closed-form solutions of nonlinear differential equations. In most cases, only approximate solutions (either analytical or numerical) can be expected. Over the years, perturbation methods have dominated analytical techniques for studying weakly nonlinear problems in science, engineering, and technology [6]. However, the dependence of perturbation methods on small values for physical parameters makes them inflexible, thereby limiting their applications, [7]. The limitations of over-dependence on the existence of a small parameter were solved by the development and introduction of semi-analytic techniques with the aim of broadening the scope of real-life applicable problems that could be solved.

Examples of semi-analytic techniques developed for solving differential equations are the series expansion methods, which include the Adomian decomposition method (ADM) used by Somali and Gokmen [8] to solve nonlinear Sturm-Liouville problems and the homotopy analysis method (HAM) of Liao [9]. Unlike the perturbation based techniques, the ADM and HAM are independent of a physical parameter being small, so giving such methods more flexibility. Another semi-analytic method that has featured in the solution of nonlinear differential equations is the variational iteration method used by He [10]. The variational iteration method was also applied by Abdou and Soliman [11, 12] to solve a large class of nonlinear problems, which yielded approximations that converged rapidly to accurate solutions. Despite the rigorous research in developing the analytic and semi-analytic methods of solution, their application is also limited in that it excludes the most practical models. Such models are represented by strongly nonlinear differential equations, for which only numerical methods give approximate solutions to a certain degree of accuracy [13].

Traditional numerical methods of solution for differential equations are based on the finite element and the finite difference methods [14]. The finite element methods, on the one hand, seek to eliminate the differential equation completely, in cases where the problem is time independent, or to transform the PDE to a system of ODEs, which are then integrated numerically using well known standard techniques such as Runge-Kutta methods and Euler's method [15]. The finite

difference methods, on the other hand, approximate the derivatives using finite difference equations. Although the finite difference based, methods are applicable to a wide range of problems, including complex nonlinear PDEs where other numerical methods fail, they have the disadvantage that they require many grid points to achieve sufficient accuracy in the results and so are not computationally efficient. The challenge of accuracy and increased computational time are attributed, respectively, to the low order interpolating polynomials used to approximate the solution of the differential equation and to the large-sized matrices associated with the use of many grid points, Ward [16]. Recent advances on the development of numerical methods focus on the use of spectral based collocation techniques, which have been demonstrated by Hussaini et al. [17] to yield very accurate results with few grid points in a computationally efficient manner.

1.2 Spectral Collocation-Based Methods

In the spectral collocation-based methods, the solution of differential equations is approximated by a linear combination of a finite number of basis functions. The basis functions chosen can be of a polynomial kind for non-periodic problems, Fourier basis for periodic problems or rational functions, according to Davis [18]. In the solution process, one seeks to derive the formula that relates the spectral coefficients of the function being approximated with its derivatives. The approximating functions are substituted in the differential equations and then evaluated at selected grid points in a process known as collocation, see Dehghan and Izadi [19]. The equations resulting from the discretized differential equation, coupled with those from prescribed conditions at the collocation points, translate into a linear system that can be solved directly in Mathematica, MatLab or any other mathematics application that supports matrix computation. Many spectral collocation-based methods have been developed to solve differential equations. They differ in several ways. Firstly, they differ in terms of the type of the differential equations they can be applied to solve, either ODEs or PDEs, both linear and nonlinear. They differ in terms of the strategies that are used to simplify the nonlinearity in the differential equations and procedures by which the approximating functions are constructed.

Examples of these spectral based collocation methods for solving ordinary differential equation include, among others; the spectral homotopy analysis method (SHAM), successive local linearization method (SLLM), spectral relaxation method (SRM), and the spectral quasi-linearization method (SQLM). Motsa et al. [20] used the SHAM to solve a system of second-order nonlinear

ODEs that describes Darcy-Brinkman-Forchheimer equation for a fully developed steady fluid flow in a horizontal channel filled with a porous medium. The SHAM was found to be more efficient and converged faster than the standard homotopy analysis method, that had been previously proposed by Liao [21]. The successive linearization method was applied by Motsa and Shateyi [22] to solve second order singular initial and boundary value problems of the Lane-Emdem type. Motsa [23] reported the SLLM and used it to solve a system of highly nonlinear boundary layer flows with exponential profiles. The SQLM and SRM were used by authors in [24] and [25], respectively, to solve systems of nonlinear ordinary differential equations. Both the SRM and SLLM decouple the system of equations and solve them iteratively using the updated solution in any equation immediately when computing the solution in the subsequent equations, whereas the SQLM linearizes the system of equations globally and solves all as a coupled system. It is worth noting that the primary difference between the SRM and the SLLM is that in the SRM the nonlinear terms in the differential equation are evaluated at the previous iteration, whereas in the SLLM the nonlinear terms are linearized using the Gauss-Seidel approach. A new numerical method was proposed by Motsa et al. [26] in order to extend the application of the enumerated spectral based collocation methods to solutions of PDEs. This new method is termed the bivariate spectral collocation method. The method uses the spectral collocation on the time variable to first reduce the problem to an ODE, after which the spectral collocation is applied in the space direction giving systems of linear equations, which are easier to solve. The method was first tested on evolutionary parabolic partial differential equations and it has been adapted by Rezazadeh et al. [27] to solve different classes of partial differential equations involving two independent variables. The spectral collocation-based methods mentioned above rely on interpolating polynomials to approximate the solutions of differential equations. It has been observed that the accuracy of spectral collocation methods depends on the choice of grid points and the type of basis function used in constructing the interpolating polynomial [28]. In addition, when these spectral based methods are applied to PDEs problems defined on a large time interval, the accuracy of the results deteriorates. As will be shown later, the deterioration of accuracy is also noticed in cases of large spatial domains and, in some instances of fluid flow problems where these involve very small or very large values for the Reynolds number. In view of the accuracy of the methods depending on the type of grid points used in interpolation, and the size of computational domains, a better understanding of the underlying theory on polynomial interpolation will be useful in developing superior numerical methods of solution for differential equations. First, we address the problem of polynomial interpolation and thereafter the size of the

computational domain.

1.3 Polynomial Interpolation

Polynomial interpolation is an area of research that dates back many decades. Polynomial interpolation arises from the need to find an appropriate polynomial of a finite degree that closely approximates the underlying function at some given grid points, see Ismail [29]. The interest of numerical analysts is to determine to what extent this approximating function deviates from the exact function. The error in polynomial interpolations depends mainly on the smoothness of interpolated function, as shown by Howell [30]. The regularity of functions plays a crucial role in determining the behaviour of higher order derivatives and, indeed, the literature reveals that the values of the derivatives within the interval of approximation regulate the magnitude of the error in polynomial interpolation, see Narcowich and Ward [31]. Quantifying the sizes of these polynomial interpolation errors becomes very important if such interpolating polynomials are used to approximate solution of differential equations. In consequence, Canuto et al. [32] derived the analytic expression of error bounds for univariate polynomial interpolation. The findings of their study indicate that the errors resulting from polynomial interpolation are influenced by the choice of interpolation nodes. In particular, the symmetrically distributed grid points have been reported to give better interpolation features than the equispaced nodes. The theorems and proofs on analytic expressions of error bounds for univariate polynomial interpolation using equispaced nodes and Chebyshev nodes were given by Ghili and Iaccarino [33] and the results showed that Chebyshev nodes give the least possible polynomial interpolation error. However, this claim was challenged when Fischer and Freund, [34] established that the best approximation is obtained from the set of interpolation nodes which gives a Lebesgue constant close to 1. The study of Lebesgue constants for different sets of grid points had previously been given in [35, 36]. Unfortunately, since it is impossible to obtain the formula that explicitly defines the set of grid points that gives the best approximation, the Chebyshev nodes presently remain the best choice of interpolation nodes [37, 38]. An extension to univariate polynomial interpolation error bound theorems and their proofs are due to the works of Bhrawy [39] and Bhrawy and Zaky [40], who explored the analytic expression for error bounds on bivariate polynomial interpolation. The arguments were analogous to those that had been presented in the case of univariate polynomial interpolation. Close scrutiny of the literature reveals that the analytic expressions for the error bounds when using Chebyshev-Gauss-Lobatto

(CGL) nodes have not yet been developed. However, as explained by Trefethen and Weideman in [41], the Chebyshev-Gauss-Lobatto nodes have consistently been used in discretizing differential equations using spectral collocation-based methods. This raises the question of whether there is any pronounced difference between CGL and Chebyshev nodes and if CGL nodes would be better than Chebyshev nodes in terms of their practical application.

In the present work, we aim to introduce theorems and proofs based on CGL nodes. In addition, analytic expressions of the error bounds for generalized multivariate polynomial interpolation are constructed. To the best of our knowledge, the analytical expressions for the error bound theorems using Chebyshev-Gauss-Lobatto nodes are stated and proved here for the first time in the literature. We remark that when the domain of approximation is large, it is a requirement that many grid points be used to improve the accuracy of numerical approximations. However, the use of many grid points is known to be memory inefficient and requires considerable computer time to invert the large matrices associated with the system of linear algebraic equations being solved. The drawbacks of using a large number of grid points are exacerbated if the underlying function being approximated has unbounded higher ordered derivatives [42]. This feature results in large interpolation errors the magnitude of the approximation error is dependent on the size of these derivatives over the entire domain of interpolation. To resolve this challenge, a key strategy will be to divide the interval of approximation into smaller sub-domains and perform piecewise polynomial interpolation on each subinterval. This approach also creates room to reduce the number of interpolation nodes in each subinterval while retaining excellent interpolating properties. The use of fewer grid points reduces interpolation errors associated with ill-behaved higher ordered derivatives, such as those evident in the Runge function (see Epperson [43]). Based on this idea, several multi-domain based spectral collocation method approaches have been developed to solve differential equations (see [44] and references therein). Research findings have indicated that the multidomain approaches yield very accurate results and are computationally efficient with a small number of grid points. The multi-domain approaches to spectral collocation methods have been applied to ordinary differential equations prescribed as initial value problems [45]. In the multi-domain approach, the solution of the ODE (initial value problem) is sought independently at each subdomain in time and the approximate numerical solution computed by matching solutions along the common boundaries of the non-overlapping subdomains. In the case of ODE formulated as a boundary value problem, non-overlapping subdomains will not suffice due to the presence of boundary conditions that must be

imposed during each solving stage. In such a case, an overlapping grid based spectral collocation method yields promising results for a large interval of the spatial domain [46]. As it will be demonstrated, a non-overlapping subdomains approach is key to reducing the size of associated coefficient matrices, whereas integrating the overlapping grids approach in spectral collocation method introduces sparsity in the matrices, thus rendering a stable numerical scheme.

In this work, domain decomposition techniques are incorporated into existing numerical methods of solving the differential equations to present modified spectral collocation methods, which are applicable in solving ordinary differential equations, one-dimensional partial differential equations of hyperbolic and parabolic type, and two-dimensional PDEs of elliptic type. The thesis is organized in such a way that descriptions of the numerical methods are presented from the simplest to the most complex, with theoretical results of the error bound theorems and proofs being used at each stage to benchmark the findings of numerical simulations. Finally, a new method of solving a partial differential equation involving three independent variables is proposed, described and applied to solve two-dimensional initial-boundary value problems. The method shall be named the trivariate spectral collocation method and its superiority will be improved through the integration of a domain decomposition technique. The new error bound theorems and proofs resulting from the incorporation of domain decomposition techniques and those emanating from trivariate polynomial interpolation are presented here for the first time in the literature. It is worth mentioning that the quasi-linearization method of Bellman and Kalaba [47] will be used to simplify the nonlinear differential equations throughout the thesis owing to its fast quadratic rate of convergence. The Lagrangian form of polynomial interpolation constructed at Chebyshev-Gauss-Lobatto points will be used as approximating functions to the solution of differential equations. The preferable choice of Lagrangian basis functions and Chebyshev-Gauss-Lobatto points is attributed to their convenience in defining the spectral differentiation matrices. For the sake of clarity, we next introduce the set of differential equations considered in this thesis.

1.4 Ordinary Differential Equations

Nonlinear ordinary differential equations that are considered in this study are boundary layer flow problems. Three types of problems are included. The first case considered is that of steady two-dimensional laminar flow of a viscous incompressible electrically conducting fluid over a continuous

shrinking sheet [48]. The second case is the Falkner-Skan equation which describes a two- dimensional flow of incompressible fluid past a stationary semi-infinite impenetrable wedge surface with stream-wise gradient [49], while the third case involves a boundary layer flow over the unsteady stretching sheet in the presence of Hall effect and heat transfer over the stretching surface [50]. The study of boundary layer flow over a stretching sheet has application in industrial manufacturing processes such as paper and glass-fibre production, wire drawing, hot rolling, drawing of plastic films, polymer extrusion, and metal spinning. As reported in [51], the stretching and simultaneous heating or cooling during these processes influence the quality of the final products. The many applications of fluid flow over a stretching sheet have led to growing research interest in it. In his pioneering work, Sakiadis [52] developed the flow model due to a flat surface that is moving with a constant velocity in a quiescent fluid. Sakiadis work was later extended by Crane [53] to the two-dimensional problem of considering the surface velocity that is proportional to the distance from the flat surface. The effect of the transverse magnetic field on the laminar flow over the stretching sheet has also been studied by a number of researchers, such as [54] to [57]. Similarly, the boundary layer flow of an incompressible viscous fluid over a shrinking sheet has received substantial attention, due to its increasing application in engineering systems. Wang [58] first investigated the flow over a shrinking sheet when he was working on the boundary layer flow of a liquid film over an unsteady stretching sheet. Later, Miklavcic and Wang [59] obtained an analytic solution for the steady viscous hydrodynamic flow over a permeable shrinking sheet. We remark that when a strong magnetic field is present, the effect of Hall currents cannot be neglected. The study of magnetohydrodynamics (MHD) viscous flow with Hall currents has important applications in problems of Hall accelerators as well as flight magnetohydrodynamics. With this understanding, and with regard to external hydrodynamic flows, Katagiri [60] and Sato [61] both discussed the effects of Hall current on the boundary layer flow past a semi-infinite plane. Pop and Soundalgekar [62] investigated the Hall effects in the steady hydrodynamic flow past an infinite porous plate. Falkner and Skan [63] first obtained the, so named, Falkner-Skan (FS) equation. In its general form, the Falkner-Skan equation is a third order boundary value problem that is strongly nonlinear with no known closed-form solution. The FS equation has been considered by a number of authors [64] to [66] as a benchmark problem for testing the performance of newly developed solution methods.

It is worth noting that the ordinary differential equations highlighted above are strongly nonlinear and there exists an extensive body of research work invested in the study of methods of

solution for such nonlinear ordinary differential equations. For instance, Elbashbeshy and Bazid [67] presented the similarity solution of the boundary layer equations which describe the unsteady flow and heat transfer over a stretching sheet. An attempt to obtain analytic and semi-analytic solutions for simple cases of boundary layer flow can be found in the work of Liao [68] and Tan and Liao [69], respectively. In the case of flow induced by shrinking sheet, Hayat et al. [70] derived both exact and series solutions describing the magnetohydrodynamic boundary layer flow of a second-grade fluid over a shrinking sheet. The boundary layer flow problem over a shrinking sheet was also solved using the Adomian decomposition method by Noor et al. [71]. A closed form of analytic solution for the flow over shrinking sheet can be found in the works of Fang and Zang [72] and Midya [73]. the homotopy analysis method (HAM) was applied by Sajid and Hayat [74] on boundary layer flows induced by a shrinking sheet. Early attempts to solve the generalized Falker-Skan equation that were based on traditional numerical approaches such as the shooting method and finite differences, can be found in studies by Hartree [75], Weyl [76] and Kuo [77]. The HAM was used by Liao and Campo [78] and Abbasbandy and Hayat [79] to solve the Blasius and Falkner-Skan problems. To counter the limitation of normal HAM, Motsa and Sibanda [80] applied the spectral homotopy analysis method (SHAM) to solve the Falkner-Skan equation. In this thesis, many of the nonlinear ODEs used for numerical experimentation are defined over semi-infinite domains. Before the numerical method is applied, the semi-infinite domain of approximation $[0, \infty)$ must be truncated to a finite domain $[0, L]$. As a rule of thumb, such a truncation should ensure that L is sufficiently large to approximate conditions at infinity. With this in mind, Lakestani [81] truncated the semi-infinite physical domain of the problem to a finite domain and expanded the required approximate solution as the element of Chebyshev cardinal functions. Further, we note the need to use many grid points when the truncated domain of approximation is large, and when the accuracy of the approximate numerical solution must be improved. However, as mentioned earlier, an increase in the number of grid points results in matrices that are large dense. Some large sized matrices are also ill-conditioned, a phenomenon that leads to an unstable numerical scheme. This challenges the extent of the reliability of the numerical results that have been presented in literature so far; a query that we now address. The main concern of the second chapter of this thesis, is, thus, to provide a robust numerical algorithm that can be applied to solve many nonlinear ODEs that model physically relevant boundary layer flow problems defined over a semi-infinite domain. Finer details about the method are contained in Chapter 2.

1.5 One-Dimensional Hyperbolic PDEs

Another class of differential equations that we sought to solve are the hyperbolic partial differential equations, which are initial value problems [82]. These equations are highly applicable in various branches of engineering and science [83], particularly, in mechanics [84]. Therefore their study has generated considerable interest among researchers. With regard to numerical solutions of hyperbolic PDEs by using domain decomposition techniques, close scrutiny of the literature indicates that previous studies focused their approach on the application of overlapping grids on only the space variable. The discretization has widely been performed using the spectral collocation in the space variable and the finite difference in the time variable. Dehghan and Taleei [85] applied a method, based on pseudo-spectral collocation, with a domain decomposition algorithm for approximating the spatial variable in coupled nonlinear Schrödinger equations and demonstrated that such an algorithm reduces the effect of round-off errors. Later, in [86], the same authors considered the long-time behavior of numerical solutions for the Klein-Gordon-Schrödinger equations using a similar approach. Kopriva [87] developed a multidomain Chebyshev spectral method for solving hyperbolic partial differential equations using spectral collocation discretization in space and finite difference discretization in time. A pseudo-spectral method based on the overlapping grids multidomain technique for the numerical solution of the Sine-Gordon equation in one and two spatial dimensions was also applied by Taleei and Dehghan in [88]. In that work, the overlapping domain decomposition technique coupled with spectral collocation discretization was performed on the spatial variable, and an implicit fourth-order Runge-Kutta method was adopted to carry out discretization in the entire time variable. A substantial literature on the numerical methods of solution for two-dimensional hyperbolic PDEs can be found in work by the authors in [89, 90] and the references given therein. Although the overlapping grids approach that had been previously applied to space variable leads to sparse matrices of the resulting system of linear algebraic equations, such matrices are nevertheless large and take considerable large computer time to invert.

In Chapter 3, the bivariate spectral collocation method, that had previously been applied on nonlinear evolutionary parabolic PDEs, is extended to obtain solutions of hyperbolic PDEs defined on large time intervals, by incorporating the non-overlapping subdomains decomposition technique. The multi-domain approach based on decomposing a large time interval into smaller non-overlapping subintervals is adopted to solve one dimensional hyperbolic PDEs. This approach preserves the benefits of dealing with small-sized matrices because the hyperbolic PDE is solved

independently on each subinterval in time, so few grid points can be used in each subinterval. The spectral collocation discretization is applied in both space and time variables and the approximate solution of the differential equation is obtained by matching solutions from different subintervals in time along the common boundary. In the case of nonlinear differential equations, the matching procedure is evoked after the solutions in each subdomain converge. We aim to demonstrate that adopting a multi-domain approach in solving such hyperbolic PDEs yields very accurate results with notably short computer time. We remark that to the best of our knowledge, the non-overlapping based multidomain spectral collocation methods with Chebyshev Gauss-Lobatto points have not been applied to hyperbolic PDEs defined on large time intervals.

The hyperbolic PDEs solved herein are PDEs, both linear and nonlinear, that have been reported in the literature. For simplicity in validating the accuracy of the proposed numerical method, the selected illustrative examples are those which possess exact solutions. We emphasize that the choice of the numerical examples was purely meant to facilitate demonstration on the usefulness of the method and was not tied to the extent of applicability of the problems themselves. However, a similar approach could be adopted to solve important hyperbolic PDEs that emerge in real-life applications. The extent of the discussion of the multidomain approach in this study is limited to only non-overlapping subintervals and is implemented for the time variable. For comparative purposes, the hyperbolic PDEs are also solved using the single domain approach, namely, the bivariate spectral collocation method. The prototypical examples of hyperbolic PDEs that are solved in this thesis include; the damped wave equation, the telegraph equation, the modified Liouville equation, and the 1-D Phi-four equation. The first two are linear, and the latter two are nonlinear. The damped wave equation describes the propagation of disturbances out of the region at a fixed speed in one spatial direction [91, 92]. This wave equation finds application in areas of elasticity, plasma physics, quantum mechanics, and general relativity. The telegraph equation originates from Maxwell's equation, and, for an electrical transmission line, it describes the variation of voltage and current with distance and time [93]. The modified Liouville equation plays an important role in biology and mathematical physics, where it describes the time evolution of the phase space distribution function [94, 94]. Finally, the 1-D Phi-four equation finds application in many areas of physics such as plasma physics, fluid dynamics, solid-state physics, and quantum field theory [96].

1.6 One-Dimensional Parabolic PDEs

Parabolic partial differential equations often arise in many practical problems in fields of mathematical physics and engineering. Parabolic PDEs are useful in describing a wide family of problems in science that include heat diffusion, ocean acoustic propagation and physical or mathematical systems with a time variable [97]. Examples of parabolic PDEs that are considered in this study include the evolutionary equations that are defined as a single equation such as; the reaction-diffusion equation, the generalized Burgers-Fisher equation [98], and the generalized FitzHugh-Nagumo equation [99]. Reaction-diffusion equations model reaction-diffusion processes in chemical kinematics, astrophysics, and biology. The generalized Burgers-Fisher equation describes different mechanisms that arise in the fields of financial mathematics, fluid mechanics, shock wave formation, traffic flows, turbulence, heat conduction and transmission of sound waves through viscous media among other disciplines of applied sciences [100]. The generalized FitzHugh-Nagumo equation [101] arises in genetics, biology, and heat and mass transfer [102, 103]. In biology, it models the activation and deactivation dynamics of a spiking neuron [104]. For systems of parabolic PDEs, we consider fluid flow problems of boundary layer type. In particular, we consider boundary layer flow over a permeable flat plane arising from differences in concentration or material constitution in conjunction with temperature effects. These types of flows have received much attention, by researchers, owing to their great practical importance [105]. For instance, atmospheric flows at all scales, are driven appreciably by both temperature and water concentration differences [106]. In addition, flow in water bodies is driven through equally important effects of temperature [107].

Several methods have been developed and applied to solve parabolic PDEs. They include the semi-analytical methods such as; the Tanh method [108], Adomian decomposition method [109, 110], homotopy perturbation method [111], the variational iteration method [112], and numerical methods such as the finite difference methods [113] and the spectral collocation methods [114]. As for the problems considered in this study, in [115], Javidi studied spectral collocation method for the generalized reaction-diffusion equation and later, working in collaboration with Golbabai [116], the spectral domain decomposition method was applied on the same problem. Babolian [117] gave some analytic approaches to the solution of Burgers-Fisher equation. The explicit solutions and numerical simulations of the generalized Burger-Fisher's equation were introduced by Kaya and Sayed in [118]. Wazwaz [108] presented the Tanh method for the generalized Burgers-Fisher equation which was later solved by Mickens and Gumel [119] and Khao et al. in [120] using a

nonstandard finite difference scheme and Chebyshev-Legendre pseudospectral method, respectively. Numerical approximations of the generalized FitzHugh-Nagumo equation also exist. One such example is, the Hopscotch finite difference scheme, which is a fast second order differential equation solver, that was first proposed by Gordon [121] and further developed by Gourlay [122]. Other numerical methods that have been applied to solve the FitzHugh-Nagumo equation include the pseudospectral method [123] and the polynomial differential quadrature method [124]. In the case of boundary layer flows, the problem of natural convection flow from a vertical permeable plate with variable surface temperature and species concentration was solved by Hussain and Hossain [125] using three methods; finite differences for the entire ξ region (where ξ is the scaled stream-wise variable), the series solution method for small ξ and asymptotic solution method for large ξ. There exists a vast literature on different methods that have been applied for numerical solutions to boundary layer flow problems (see [126]). Close scrutiny of the literature reveals that among the numerical methods that are available for approximating solutions of parabolic PDEs, spectral methods (tau, Galerkin, and collocation-based) are superior, especially where the problem possesses a smooth solution [127]. Spectral methods are advantageous in that they require a few grid points to give accurate results and short computation time is needed to realize the results. When solving initial-boundary value problems, the spectral collocation-based approaches are often used as they offer a simple treatment of boundary conditions [128]. In this regard, the bivariate spectral quasi-linearization method has been successfully applied to solve parabolic PDEs defined over small space and time intervals. However, it has been observed that when the method is applied to solve PDE defined over a large time interval, the accuracy deteriorates even with a large number of grid points [129]. In view of this drawback, the utility of the method has recently improved with a new version that is based on decomposing the time variable domain into smaller non-overlapping subintervals [130]. In the modified approach, the PDE is solved independently at each time subinterval, and the resultant solution to the PDE is obtained by merging the solution from different subintervals along common boundaries. We remark that this approach disregards a large spatial domain, which is an issue that we now address.

In Chapter 4, we introduce a new overlapping grid based multidomain bivariate spectral collocation method for the solution of parabolic PDEs that are defined on over large space and time intervals. We also aim to demonstrate that this multi-domain approach yields very accurate results in an appreciably short computer time. The new method is based on decomposing the space

and time variable domains into smaller equal overlapping and non-overlapping subintervals, respectively. The PDE is then discretized at each of these subintervals using spectral collocation with Chebyshev-Gauss-Lobatto points. The overlapping in the spatial variable is done in such a way that the first and last two nodes of the interior subintervals coincide with those of the neighbouring subintervals and they remain common. Chebyshev differentiation matrices from the different spatial subintervals are incorporated to assemble a new Chebyshev differentiation matrix for the decomposed space variable. The PDE is solved over the entire spatial domain but independently at each time subinterval using the continuity condition to obtain initial conditions at subsequent subintervals. The solution of the PDE in the entire time domain is taken as the union of the solutions from different time subintervals. To the best of our knowledge, the spectral collocation-based methods with Chebyshev-Gauss-Lobatto points have not been applied simultaneously on overlapping grids in space and non-overlapping subintervals in time. To further explain the performance of the proposed method and the preferred choice of Chebyshev-Gauss-Lobatto points over other sets of nodes, we develop new error bound theorems that govern piecewise bivariate polynomial interpolation using CGL nodes and present their rigorous proofs.

1.7 Two-Dimensional Elliptic PDEs

Elliptic partial differential equations describe phenomena that do not change with time. They have numerous applications in almost all areas of mathematics, ranging from harmonic analysis to Lie theory and in physics. As reported by Fedoseyev et al. [131] and in a book on differential equations by Polyanin and Zaitsev [132], many processes in the applied sciences have important features of the real world phenomena which can only be modeled by elliptic equations. The resulting nonlinear partial differential equations present mathematical difficulties [133]. The desire to find and understand the solutions of these equations has inspired mathematicians to develop a diverse spectrum of numerical methods for their solution, thereby giving rigorous answers to important questions of the nonlinear world [134]. Over time, many impressive results in this area have further motivated the introduction of superior methods of solutions. Notably, at the onset of designing a numerical algorithm for nonlinear differential equations, the most remarkable features to consider include the requirements of accuracy, the ease to design the algorithm and computational complexity involved during implementation.

In the literature, there are three basic numerical approaches for solving nonlinear elliptic PDEs;

finite differences, finite elements, and spectral collocation methods [135]. Finite difference methods use many subdomains and calculate the discrete solution on a grid, by expanding to low order in each subdomain. Finite difference methods are simple and the easiest to code, but difficult to work with on complicated domains [136]. Avellaneda et al. [137] applied the finite difference method to solve elliptic PDEs with rapidly oscillating coefficients. Later, Conca and Natesan [138] provided a better approximation to the solution of these elliptic problems using a numerical method based on the Bloch wave approach. Chan et al. [139] solved two-dimensional nonlinear obstacle problems using a generalized finite difference method. There is furthermore a vast literature on the application of finite difference methods to nonlinear elliptic PDEs in work by Jensen [140]. Finite element methods differ in that they are based on partitioning the domain into small finite elements followed by expansion of the solution in basis functions [141]. Finite element methods are particularly well suited to use with irregular domains which appear in many engineering applications: however, they are more complex to set up and analyze. Hou and Wu [142] introduced a multi-scale finite element method for elliptic problems in composite materials and porous media. Matache et al. [143] proposed a p-finite element method for numerically approximating the solution of nonlinear elliptic PDEs. Spectral collocation methods approximate solutions to nonlinear PDEs on a single domain, or a few subdomains, with higher expansion orders of the basis functions. They are generally faster and highly accurate for problems with smooth solutions. [144] presented a pseudo-spectral collocation-based approach to solutions of nonlinear elliptic PDEs using Chebyshev basis functions. Yi and Wangy [145] proposed a Legendre-Gauss-type collocation algorithm for nonlinear partial differential equations and demonstrated the high accuracy and effectiveness of the suggested algorithms.

Multidomain spectral collocation methods originating from pioneering work of Gottlieb and Orszag [146] have been applied to solutions of nonlinear PDEs (see for instance [147]). Despite much attention on this subject, previous studies on the application of spectral collocation methods with domain decomposition techniques have focused on only one-dimensional problems. Where higher dimensional problems are involved, a suitable similarity transformation has been applied to reduce the number of independent variables in an attempt to reduce the computational complexity before the spectral collocation method with domain decomposition is applied [148]. By contrast, direct application of spectral collocation methods with overlapping grids to higher dimensional nonlinear problems and in particular nonlinear elliptic PDEs emanating from heat and mass transfer,

combustion theory and classical fluid flow problems has traditionally been viewed as problematic. The possible reason for this reservation would be the extent of the anticipated numerical complexity involved in dealing with overlapping subintervals in different spatial directions. Indeed, with a few noticeable exceptions [149, 150], very little work has been done on adopting a purely spectral collocation method with overlapping grids for the solution of such an important class of problems.

Chapter 5 focuses on the direct application of a domain decomposition based spectral collocation method on two-dimensional nonlinear PDEs defined on large rectangular domains. Preliminaries of the numerical algorithm involve breaking the large computational domain into smaller equal overlapping subintervals on each space direction. The solution of the PDE is approximated using a bivariate Lagrange interpolating polynomials constructed using Chebyshev-Gauss-Lobatto (CGL) nodes. The linearized PDE is discretized on each subinterval and through careful dislodgement of repeated equations at overlapping regions, a linearly independent system of algebraic equations is obtained. The numerical scheme is tested using three typical nonlinear elliptic PDEs that have been reported in the literature. These include the two dimensional heat and mass transfer equation with a n-th order volume reaction that arise in combustion theory [151], the stationary equation of Khokhlov-Zabolotskaya that describes different acoustic phenomena [152], and a system of equations representing the problem of heat and mass transfer effects of a steady magneto-hydromagnetic flow of viscous, electrically conducting fluid past a semi-infinite inclined porous plate [153]. Numerical simulations confirm that the overlapping grids based approach is highly accurate and computationally efficient when compared to the single domain approach for the considered class of problems.

1.8 Two-Dimensional Initial-Boundary Value Problems

Finally, despite the benefits of the spectral collocation methods, review of the literature indicates that previous application of purely spectral collocation methods has focused on the solutions of either or both of ordinary differential equations and partial differential equations involving two independent variables [154]. With a few noticeable exceptions such as in [155] where spectral collocation methods have been applied to obtain numerical solutions of two-dimensional time-dependent PDEs, such has been achieved through the application of spectral collocation discretization on the space variables and finite difference discretization on the time variable. It is well known that fi-

nite difference methods require many grid points to yield accurate results. Their accuracy cannot match those obtained when spectral collocation methods are applied on PDEs defined on simple geometries if the underlying solutions are smooth. Motivated by the usefulness of incorporating domain decomposition technique in the spectral collocation method, we propose an overlapping grid based multidomain trivariate spectral collocation method for solving nonlinear two-dimensional time-dependent PDEs defined on large rectangular domains over large time intervals.

Typical examples of two-dimensional nonlinear time-dependent PDEs considered in this study include the nonlinear PDEs given as single equations that describe the problem of unsteady two-dimensional heat and mass transfer, firstly, in quiescent media with chemical reaction [156], and secondly, with power-law temperature-dependent thermal conductivity [157]. The problems of heat and mass transfer phenomena are found throughout the physical world and frequently in the industry. For nonlinear PDEs, described as systems of equations, we consider, firstly, a case of the coupled two-dimensional Burgers system [158] and the two-dimensional reaction-diffusion Brusselator system [159]. The Burgers equation is a fundamental PDEs from fluid mechanics; it occurs in various areas of applied mathematics, such as modeling of hydrodynamic turbulence, shock waves, and traffic flow problems. It also describes the sedimentation of particles in fluid suspensions under the effect of gravity, transport, and dispersion of pollutants in rivers [160]. The second system of PDEs considered is the Brusselator system arising in the mathematical modeling of chemical systems such as enzymatic reactions, and in plasma and laser physics in multiple coupling between certain modes [161]. The Brusselator model is also evident in the formation of the ozone layer through a triple collision of oxygen atoms. The problems considered here have exact solutions and have been reported in the literature to be very useful in testing newly developed numerical methods of solution for nonlinear partial differential equations that arise in the modeling of various aspects of the real world. We, therefore, consider them appropriate to demonstrate the effectiveness of the proposed method of solution.

Exact solutions of two-dimensional heat and mass transfer problem in quiescent media with chemical reaction were discussed by Polyanin and Zaitsev [156]. The problem has been solved numerically using an implicit finite-difference method by Chamkha [162]. The problem of two-dimensional heat and mass transfer with power-law temperature-dependent thermal conductivity was examined by Pamuk and Pamuk [157], where they obtained a particular exact solution using the Adomian decomposition method and numerical methods for the solution of this problem can be

found in references they cite. Burgers [158] pioneered the investigation of the mathematical properties defined by the so-named Burgers equation. Analytical solution of the unsteady two dimensional coupled Burgers equation was first given by Fletcher [163] using the Hopf-Cole transformation. In [164], the differential transformation method was applied to obtain the analytical solution of a coupled unsteady Burgers equation. Numerical solution of Burgers equation is a natural and first step towards developing methods for the computation around complex flows. The Burgers equation has been used frequently to test new approaches in computational fluid dynamics by first applying the new approaches to it. For instance, a numerical method based on local discontinuous Galerkin-finite elements was analyzed in [165] to solve the two-dimensional Burgers equation. In another instance, the local radial basis functions collocation method used to approximate the numerical solution of the transient coupled Burgers equation was examined in [166]. The Brusellor model has been extensively studied both numerically and analytically. For instance, Twizell et al. [167] developed a second-order finite difference method for the numerical solution of the initial-boundary value problems of the Brusselator model; Khani et al. [168] found exact solutions of the Brusselator reaction-diffusion model using the exp-function method, while Biazar and Ayati [169] obtained an approximate solution to the Brusselator system by applying the Adomian decomposition method.

In Chapter 6, a purely spectral collocation method, namely, the overlapping grids multidomain trivariate spectral collocation method, is introduced and applied to solve two-dimensional initial-boundary value problems defined on over large space and time intervals. The solution process involves, firstly, the simplification of the PDE using the quasi-linearization method. Next, the spatial domain is decomposed into a sequence of equal overlapping subintervals, and the large time domain is decomposed into equal non-overlapping subintervals. The solution to the linearized PDE is assumed to be a trivariate Lagrangian interpolating polynomial constructed on Chebyshev Gauss-Lobatto points defined on each subinterval. The PDE is discretized in all space variables and time variable using the spectral collocation method to yield a system of linear algebraic equations that are solved iteratively. The solution of the PDE is computed simultaneously across all subintervals in space, and independently at each subinterval in time, applying the continuity condition to obtain the initial condition for the second to last subintervals. The proposed numerical method is then tested using typical examples of initial-boundary value problems reported in the literature. The accuracy of the numerical scheme is assessed by computing the absolute difference between the numerical results with the exact solutions. The differences are presented and discussed in tabular and

graphical form. Numerical results obtained using the new solution approach are compared against those obtained using the standard single domain based trivariate spectral collocation method. Findings from numerical simulations show that the current method yields highly accurate results in a computationally efficient manner when applied to problems defined on a large time interval, large spatial domains, and obtaining numerical approximations of solutions to the differential equation for large parameter values. To the best of our knowledge, the spectral collocation-based method with Chebyshev-Gauss-Lobatto points has not been applied simultaneously on overlapping grids in two-dimensional space and non-overlapping grids in time. The new theoretical results of error bound theorems support the findings of the numerical simulations.

In summary, the diagram given below in Figure 1.1 demonstrates the nature of the differential equations that will be encountered in this thesis. They start with one-dimensional ordinary differential equations and progress through to two-dimensional time-dependent partial differential equations.

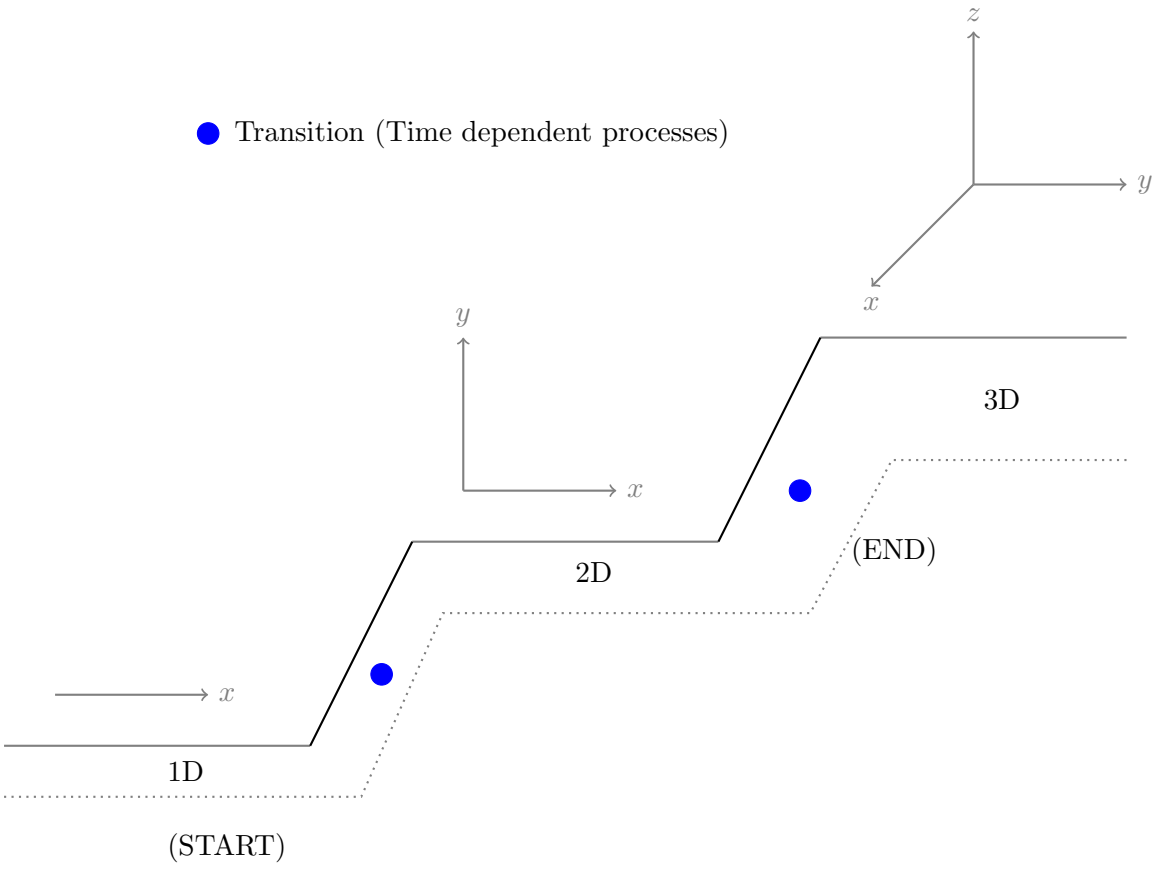


Figure 1.1: Schematic representation of the nature of differential equations solved in this thesis

Chapter 2

A Modified Spectral Collocation Method of Solution for Ordinary Differential Equations

In this chapter, we present an accurate and computationally efficient overlapping grid based spectral collocation method for approximating the solutions of nonlinear ordinary differential equations. The practical applicability of the proposed method is showcased by solving boundary layer flow problems reported in the literature as a single equation or a system of ordinary differential equations. The primary focus is to demonstrate that the proposed solution approach is highly accurate and computationally efficient when applied to solve ordinary differential equations defined over large computational domains. We conclude this chapter with a discussion of new results on error bound theorems and proofs on univariate polynomial interpolation using Chebyshev-Gauss-Lobatto nodes in the single domain and the overlapping grid. The numerical method described in this chapter can be adjusted and applied to many other classes of ordinary differential equations with diverse practical applications.

Highly accurate overlapping grids based spectral collocation method of solution for boundary layer equations

F.M. Samuel[1,a,c], S.S. Motsa[1,a,b]

[a] *University of KwaZulu-Natal, School of Mathematics, Statistics, and Computer Science, Private Bag X01, Scottsville, 3209, South Africa;*
[b] *University of Swaziland, Department of Mathematics, Private Bag 4, Kwaluseni, Swaziland;*
[c] *Taita Taveta University, Department of Mathematics, 635-80300, Voi, Kenya.*

Abstract

In this paper, an accurate, and computationally efficient algorithm for the solution of nonlinear boundary layer flow problems that are defined on semi-infinite domains is presented. In this approach, the quasi-linearization method is used to simplify the nonlinear equations before applying the spectral collocation method on overlapping grids, which are defined on the truncated domain. The practical applicability of the method is demonstrated by solving boundary layer flow problems that have been reported in the literature as a single equation or system of ordinary differential equations (ODEs). The aim is to demonstrate that the proposed approach is highly accurate and computationally efficient when applied on problems defined over large computational domains. The accuracy, efficiency, and robustness of the method are demonstrated by comparing the current results with those obtained using MatLab build-in routine, and a good agreement is observed. The numerical results have also been compared against the exact solution where it exists, and a residual error analysis is conducted to assess the accuracy of the method for problems without exact solutions. We examine the significance of varying the number of overlapping grids and report stable results. Finally, the new error bound theorems and proofs on polynomial interpolation that are presented provide benchmarks the adoption of the present numerical algorithm.

Keywords: Overlapping grids, spectral collocation, polynomial interpolation, boundary layer equations, semi-infinite domains, ODEs.

2010 MSC: 65M15, 65M55, 65M70.

*Corresponding author: felexmutua@gmail.com

1. Introduction

Nonlinear ordinary differential equations that are considered in this paper are boundary layer flow problems and include the cases of, firstly, the steady two-dimensional laminar flow of a viscous incompressible electrically conducting fluid over a continuous shrinking sheet [1], secondly, the Falkner-Skan equation which describes a two-dimensional flow of an incompressible fluid past a stationary semi-infinite impenetrable wedge surface with stream-wise gradient [2] and thirdly, a boundary layer flow over an unsteady stretching sheet in the presence of Hall effect and heat transfer over the stretching surface [3].

The study of boundary layer flow over a stretching sheet has application in the engineering of industrial processes such as the production of paper and glass-fibre, wire drawing, hot rolling, drawing of plastic films, polymer extrusion, and metal spinning. As reported in [4], the stretching and simultaneous heating or cooling during these processes influence the quality of the final products. Accordingly, in view of its many applications, there has been a growing research interest on fluid flow over a stretching sheet. In his pioneering work, Sakiadis [5] developed the flow model due to a flat surface that is moving with a constant velocity in a quiescent fluid. Sakiadis work was later extended by Crane [6] to the two-dimensional problem of considering the surface velocity that is proportional to the distance from the flat surface. The effect of a transverse magnetic field on the laminar flow over a stretching sheet has also been studied by a number of researchers such as [7]-[10]. Paralleling this work, the boundary layer flow of an incompressible viscous fluid over a shrinking sheet has also received considerable attention due to its increasing application in engineering systems. Wang [11] first noted the flow over a shrinking sheet when he was working on the boundary layer flow of a liquid film over an unsteady stretching sheet. Later, Miklavcic and Wang [12] obtained an analytic solution for the steady viscous hydrodynamic flow over a permeable shrinking sheet. We remark that when a strong magnetic field is present, the effect of Hall currents cannot be neglected. The study of magnetohydrodynamics (MHD) viscous flow with Hall currents has important applications in problems of Hall accelerators as well as aeronautical magnetohydrodynamics. With this understanding and with regard to external hydrodynamic flows, Katagiri [13] and Sato [14] discussed the effects of the Hall current on the boundary layer flow past a semi-infinite plane. Pop and Soundalgekar [15] investigated the Hall effects in the steady hydrodynamic flow past an infinite porous plate. The Falkner-Skan (FS) equation was first obtained by Falkner and Skan [16]. In its general form, the equation is a third order boundary value problem that is strongly

nonlinear with no known closed-form solution. The FS equation has been considered by a number of authors [17] to [19] as a benchmark problem for testing the performance of newly developed solution methods.

The examples highlighted above and many others in science and engineering are notably strongly nonlinear. Although an extensive body of research work has been invested in the study of methods of solution of nonlinear systems, exact solutions are limited to special simple cases of these problems. For instance, Elbashbeshy and Bazid [20] presented a similarity solution of the boundary layer equations which describe the unsteady flow and heat transfer over a stretching sheet. Attempts to obtained analytic and semi-analytic solutions for simple cases of boundary layer flow can be found in the work of Liao [21] and Tan and Liao [22], respectively. For the case of flow induced by a shrinking sheet, Hayat et al. [23] derived both exact and series solutions describing the magnetohydrodynamic boundary layer flow of a second-grade fluid over a shrinking sheet. The boundary layer flow problem over shrinking sheet was also solved using Adomian decomposition method by Noor et al. [24]. A closed form of analytic solution for the flow over a shrinking sheet can be found in work of Fang and Zhang [25] and Midya [26]. The homotopy analysis method (HAM) was applied by Sajid and Hayat [27] on boundary layer flows induced by a shrinking sheet. Early attempts to solve the generalized Falker-Skan equation that were based on traditional numerical approaches, such as the shooting method and finite differences, can be found in work by Hartree [28], Weyl [29] and Kuo [30]. The HAM was used by Liao and Campo [31] and Abbasbandy and Hayat [32] to solve the Blasius and Falkner-Skan problems. To counter the limitation of the normal HAM, as explained in studies by Motsa et al. [33], the spectral homotopy analysis method(SHAM) was used to solve the Falkner-Skan equation by Motsa and Sibanda [34].

A close examination of the above class of nonlinear problems reveals that many of the ODEs are defined over semi-infinite domains. Before the numerical method is applied, the semi-infinite domain of approximation $[0, \infty)$ must be truncated to a finite domain $[0, L]$. As a rule of thumb, such a truncation should ensure that L is sufficiently large to approximate conditions at infinity. With this in mind, Lakestani [35] truncated the semi-infinite physical domain of the problem to a finite domain and expanded the required approximate solution as the element of Chebyshev cardinal functions. Further, we note that when the domain of approximation is large, any attempt to improve the accuracy of the approximate numerical solution requires the use of many grid points. However, as reported by Trefethen [36], increasing the number of grid points results in large-sized

and dense matrices. Consequently, the computation time required to invert them is compromised. Furthermore, some large-sized matrices are ill-conditioned; a phenomenon that leads to an unstable numerical scheme. This effect creates some uncertainty concerning the reliability of the numerical results that have been presented in literature: a query that we now address.

The main concern of the present paper is to provide a robust algorithm for the solution of nonlinear ODEs that model physically relevant boundary layer flow problems defined over a semi-infinite domain. The nonlinear ODEs is first linearized using the quasi-linearization method (QLM) of Bellman and Kalaba [37]. The linearized QLM scheme is then solved using the spectral collocation method applied on overlapping grids. In this approach, the truncated computational domain is subdivided into overlapping subintervals of equal length and the approximate solution is computed over the entire interval. The overlapping nature is such that two grid points at the ends of each subinterval coincide with those of neighboring subintervals. We show that the current approach leads to less dense matrices that require short computation time to invert. The well-posed nature of the resulting system of linear equations is demonstrated by displacing the condition number of the coefficient matrices. A well-conditioned system guarantees stability and, consequently, leads to the highly accurate solutions that we claim are reported. To further explain the proposed algorithm, we provide a new error bound theorem with a rigorous proof for error bound expressions emanating from the use of interpolating polynomial to approximate the solution to the ODE for the chosen basis functions and collocation points. It is the result of this theorem that informed us about the usefulness of the present solution approach, for which we now demonstrate the superiority. All calculations were programmed in MatLab and Mathematica and solutions have been presented in tables and graphs.

The outline of the present article is the following. In Section 2 the numerical method is described. Section 3 is devoted for the presentation of the new error bound theorems and proofs for polynomial interpolation. In Section 4 the numerical method is applied to selected boundary layer flow problems and results and discussions are given in Section 5. Finally, concluding remarks are given in Section 6.

2. Method of solution

In this section, we describe the proposed method of solution. The spectral quasi-linearization method of solution has been applied to solve systems of ordinary differential equations on a single

domain, for instance in [38]. In this work, we describe the multi-domain extension of the solution algorithm. The approach is first described for a single nonlinear equation before considering its expansion to a generalized system of nonlinear ODEs. The present approach is reliable for giving highly accurate solutions over large domains.

2.1. The multi-domain spectral collocation method for a single nonlinear ODE

In this subsection, the development of the solution algorithm for a single equation is illustrated by considering an n-th order ordinary differential equation that takes the form;

$$L\left[u^{(0)}(x), u^{(1)}(x), u^{(2)}(x), \ldots, u^{(n)}(x)\right] = h(x), \quad x \in (a, b), \tag{1}$$

where L is a nonlinear operator acting on u and its first n ordinary derivatives, $u^{(0)}(x) = u(x)$, and $h(x)$ is a known function of the independent variable x. The ODE Eq.(1) is to be solved subject to the two point boundary conditions which can be expressed as

$$\sum_{p=0}^{n-1} \alpha_{\nu}^{[p]} u^{(p)}(a) = g_{a,\nu}, \quad \nu = 1, 2, \ldots, n_a, \tag{2}$$

$$\sum_{p=0}^{n-1} \beta_{\sigma}^{[p]} u^{(p)}(b) = g_{b,\sigma}, \quad \sigma = 1, 2, \ldots, n_b, \tag{3}$$

where $g_{a,\nu},\ g_{b,\sigma}$ are constants, $\alpha_{\nu}^{[p]}$ and $\beta_{\sigma}^{[p]}$ are the constant coefficients of p-th ordinary derivative, $u^{(p)}(x)$, in the boundary conditions, and n_a and n_b denote the total number of boundary conditions prescribed at $x = a$ and $x = b$, respectively. The method of solution involves the stages described in the subsections below.

2.1.1. The quasi-linearisation method

The nonlinear ODE is first simplified using the quasi-linearisation method (QLM) of Bellman and Kalaba [37]. The QLM is based on the Newton-Raphson method and is built from the linear terms of a Taylor series expansion about an initial approximation to the solution. The QLM assumes that the difference between solutions at two successive iterations, denoted by $u_{s+1} - u_s$, is very small. In particular, the QLM is comparable to providing the linear approximation of a function of several variables, where, the derivatives of different order and the previous approximation of the solution assume, respectively, the role of independent variables and the functional value at the reference point. Finer details about linear approximation of functions can be found in any

elementary book on differential calculus. Applying the QLM on Eq.(1) we obtain

$$\gamma_{n,s}(x)u_{s+1}^{(n)} + \gamma_{n-1,s}(x)u_{s+1}^{(n-1)} + \ldots + \gamma_{1,s}(x)u_{s+1}^{(1)} + \gamma_{0,s}(x)u_{s+1} = R_s(x), \tag{4}$$

where

$$\begin{aligned} \gamma_{\mu,s}(x) &= \frac{\partial L}{\partial u_s^{(\mu)}}\left[u_s, u_s^{(1)}, u^{(2)}, \ldots, u_s^{(n)}\right], \quad \mu = 0, 1, 2, \ldots, n, \\ R_s(x) &= h(x) + \sum_{\zeta=0}^{n} \gamma_{\zeta,s}(x)u_s^{(\zeta)} - L\left[u_s, u_s^{(1)}, u_s^{(2)}, \ldots, u_s^{(n)}\right]. \end{aligned} \tag{5}$$

Here $s = 1, 2, \ldots,$ denotes the iteration level. Starting with an initial approximation to solution u_0, the QLM scheme Eq.(4) is solved iteratively until a solution with the desired accuracy tolerance is realized.

2.1.2. Domain decomposition and discretization

The spatial domain $x \in [a, b]$ is decomposed into q overlapping subintervals of equal length as

$$\Lambda_l = [x_{l-1}, \bar{x}_l], \quad x_{l-1} < x_l < \bar{x}_l, \quad x_0 = a, \ \bar{x}_q = b, \quad l = 1, 2, \ldots, q, \tag{6}$$

where $x_l < \bar{x}_l$, depicts the overlapping nature. Pictorially, the above decomposition of the x domain can be represented as in [39] by Figure 1.

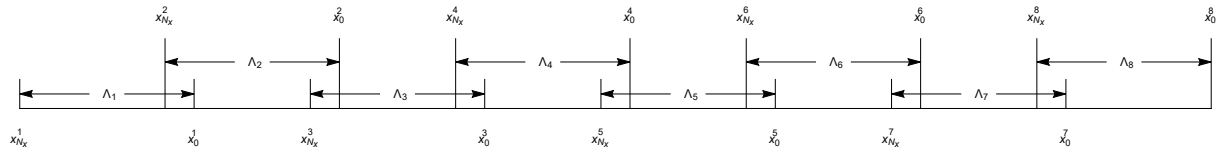


Figure 1: Pictorial illustration of decomposition of the x domain into 8 equal overlapping subintervals

The computational domain $[x_{l-1}, \bar{x}_l]$ in the l^{th} subinterval is transformed into $\hat{x} \in [-1, 1]$ where the standard Chebyshev differentiation matrix is defined by applying the linear map

$$\hat{x}(x) = \frac{2}{\bar{x}_l - x_{l-1}}\left[x - \frac{1}{2}\left(\bar{x}_l + x_{l-1}\right)\right], \quad x \in [x_{l-1}, \bar{x}_l]. \tag{7}$$

Further, each subinterval is discretized into $N_x + 1$ Chebyshev-Gauss-Lobatto points. The subintervals in the decomposed domain are made to overlap in such a way that the last two points in the Λ_l subinterval overlap with the first two points of the Λ_{l+1} subinterval and they remain common.

In general, the set of grid points over the entire x domain can be represented as

$$\{a = x_{N_x}^{(1)}, \ldots, x_1^{(1)} = x_{N_x}^{(2)}, x_0^{(1)} = x_{N_x-1}^{(2)}, \ldots, x_1^{(l-1)} = x_{N_x}^{(l)}, x_0^{(l-1)} = x_{N_x-1}^{(l)}, \ldots, x_0^{(q)} = b,\ 2 \leq l \leq q\}. \tag{8}$$

The ordering of grid points as illustrated in Eq.(8) signifies that the spectral collocation is done from right to left of the subinterval. The collocation nodes in the l^{th} subinterval in x variable are defined in [36] by

$$\{\hat{x}_i\}_{i=0}^{N_x} = \cos\left(\frac{i\pi}{N_x}\right). \tag{9}$$

An explicit expression of the length $L = \bar{x}_l - x_{l-1}$ in terms of the number of overlapping grids q in x is useful in defining the grid points at the interior subintervals. To obtain this, we first observe that overlapping of q subintervals in x results in $(q-1)$ regions of overlaps. Further, using the linear map of Eq.(7) we notice that each of the overlapping space region has the length

$$L\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right). \tag{10}$$

The length L of the subinterval can therefore be obtained from the solution of

$$qL - L\,(q-1)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right) = b - a, \tag{11}$$

which is

$$L = \frac{b-a}{q + (1-q)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right)}. \tag{12}$$

Evidently, x_l and $\bar{x}_l$ are related by

$$\bar{x}_l = x_l + L\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right). \tag{13}$$

We remark that this subdivision of the entire space domain into subintervals of the same length and subsequent discretization of each subinterval into an equal number of grid points is a sufficient condition for the grid points of neighboring subdomains to overlap.

2.1.3. Spectral collocation

The multidomain spectral quasilinearization method is implemented on the linearised QLM scheme Eq.(4) as illustrated below. We assume that the approximate solution that is sought is a Lagrange interpolation polynomial that takes the form

$$u(x) \approx U(x) = \sum_{p=0}^{N_x} U(x_p) L_p(x). \tag{14}$$

The spatial derivatives of the unknown function $u(x)$ at each subinterval, $l = 1, 2, \ldots, q$, are approximated at the collocation nodes, $\hat{x}_j,\ j = 0, 1, \ldots, N_x$, as follows:

$$\frac{du}{dx} \approx \sum_{i=0}^{N_x} U(x_i) L_i'(x_j) = \sum_{j=0}^{N_x} \overset{l}{\mathbf{D}}_{i,j} U(x_j) = \overset{l}{\mathbf{D}}\mathbf{U},\ \ i = 0, 1, \ldots, N_x,\ \ l = 1, 2, \ldots, q, \tag{15}$$

where $\overset{l}{\mathbf{D}} = \left(\frac{2}{L}\right)\hat{\mathbf{D}}$ and $\hat{\mathbf{D}}$ of size $(N_x + 1) \times (N_x + 1)$ is the standard first order Chebyshev differentiation matrix as defined in [36]. The vector $\mathbf{U}$ is defined as

$$\mathbf{U} = \left[u(x_0^l), u(x_1^l), \ldots, u(x_{N_x}^l)\right]^T, \tag{16}$$

where T denotes the matrix transpose. We note that the solution is computed simultaneously across all subintervals and that the multi-domain approach only becomes evident when assembling differentiation matrices in x. Since the last two points in the l^{th} subinterval and the first two points in the $(l+1)^{th}$ subinterval overlap and remain common, we can assemble the differentiation matrix $\mathbf{D}$ for overlapping grids in x by carefully discarding the rows corresponding to the recurrent points as

$$\mathbf{D} = \begin{bmatrix}
\overset{q}{D}_{0,0} & \ldots & \overset{q}{D}_{0,N_x-1} & \overset{q}{D}_{0,N_x} & & & & & & \\
\overset{q}{D}_{1,0} & \ldots & \overset{q}{D}_{1,N_x-1} & \overset{q}{D}_{1,N_x} & & & & & & \\
\ddots & \ddots & \ddots & \ddots & & & & & & \\
\overset{q}{D}_{N_x-1,0} & \ldots & \overset{q}{D}_{N_x-1,N_x-1} & \overset{q}{D}_{N_x-1,N_x} & & & & & & \\
 & & \ddots & \ddots & & & & & & \\
 & & \overset{2}{D}_{1,0} & \overset{2}{D}_{1,1} & \ldots & \overset{2}{D}_{1,N_x-1} & \overset{2}{D}_{1,N_x} & & & \\
 & & \ddots & \ddots & \ddots & \ddots & \ddots & & & \\
 & & \overset{2}{D}_{N_x-1,0} & \overset{2}{D}_{N_x-1,1} & \ldots & \overset{2}{D}_{N_x-1,N_x-1} & \overset{2}{D}_{N_x-1,N_x} & & & \\
 & & & & & \overset{1}{D}_{1,0} & \overset{1}{D}_{1,1} & \ldots & \overset{1}{D}_{1,N_x} \\
 & & & & & \ddots & \ddots & \ddots & \ddots \\
 & & & & & \overset{1}{D}_{N_x-1,0} & \overset{1}{D}_{N_x-1,1} & \ldots & \overset{1}{D}_{N_x-1,N_x} \\
 & & & & & \overset{1}{D}_{N_x,0} & \overset{1}{D}_{N_x,1} & \ldots & \overset{1}{D}_{N_x,N_x}
\end{bmatrix}. \tag{17}$$

Here the empty entries of matrix $\mathbf{D}$ are zeros and $\overset{l}{D}$ represents the Chebyshev differentiation matrix in l^{th} subinterval in space. The size of matrix $\mathbf{D}$ is $(r+1)\times(r+1)$, where $r = N_x + (N_x - 1)\times(q - 1)$.

The x grid corresponding to the assembled differentiation matrix over entire spatial domain in the multi-domain approach is

$$\{a = x_{N_x}^{(1)}, \ldots, x_1^{(1)}, x_{N_x-1}^{(2)}, \ldots, x_1^{(2)}, \ldots, x_{N_x-1}^{(l)}, \ldots, x_1^{(l)}, \ldots, x_{N_x-1}^{(q)} \ldots, x_0^{(q)} = b,\ 3 \leq l \leq q-1\}. \tag{18}$$

The higher order differentiation matrix over the entire domain can be approximated using matrix multiplication as

$$\frac{d^p u}{dx^p} \approx \mathbf{D}^p \bar{\mathbf{U}}, \tag{19}$$

where vector $\bar{\mathbf{U}}$ is defined as

$$\bar{\mathbf{U}} = [u(x_0), u(x_1), \ldots, u(x_r)]^T. \tag{20}$$

We assert that the bar in $\bar{\mathbf{U}}$ at Eq.(20) distinguishes it from the vector $\mathbf{U}$ at Eq.(16). The x grid points at Eq.(18) are those in Eq.(8) when arranged from right to left such that $x_0 = b = x_0^{(q)},\ x_r = a = x_{N_x}^{(1)}$. Using the definition of discrete derivatives, Eq.(4) can be expressed in matrix form as

$$\left[\gamma_{n,s}(\mathbf{x})\mathbf{D}^n + \gamma_{n-1,s}(\mathbf{x})\mathbf{D}^{n-1} + \ldots + \gamma_{1,s}(\mathbf{x})\mathbf{D} + \gamma_{0,s}(\mathbf{x})\right] \bar{\mathbf{U}} = \mathbf{R}_s(\mathbf{x}). \tag{21}$$

Eq.(21) is a $(r+1)$ system of linear equations which can expressed as an $(r+1) \times (r+1)$ matrix system that is given by

$$\begin{bmatrix} A_{0,0} & A_{0,1} & A_{0,2} & \ldots & A_{0,r} \\ A_{1,0} & A_{1,1} & A_{1,2} & \ldots & A_{1,r} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ A_{r-1,0} & A_{r-1,1} & A_{r-1,2} & \ldots & A_{r-1,r} \\ A_{r,0} & A_{r,1} & A_{r,2} & \ldots & A_{r,r} \end{bmatrix} \begin{bmatrix} \bar{U}_0 \\ \bar{U}_1 \\ \vdots \\ \bar{U}_{r-1} \\ \bar{U}_r \end{bmatrix} = \begin{bmatrix} R_0 \\ R_1 \\ \vdots \\ R_{r-1} \\ R_r \end{bmatrix}, \tag{22}$$

where

$$\begin{aligned} &\mathbf{A} = \gamma_{n,s}(\mathbf{x})\mathbf{D}^n + \gamma_{n-1,s}(\mathbf{x})\mathbf{D}^{n-1} + \ldots + \gamma_{1,s}(\mathbf{x})\mathbf{D} + \gamma_{0,s}(\mathbf{x}), \\ &\bar{U}_\delta = u(x_\delta), \quad R_\delta = \mathrm{R}(x_\delta), \quad \delta = 0, 1, 2, \ldots, r. \end{aligned} \tag{23}$$

The boundary conditions are evaluated at the collocation points as

$$\sum_{p=0}^{n-1} \alpha_\nu^{[p]} \sum_{j=0}^{r} \mathbf{D}_{r,j}^p u_{s+1}(x_j) = g_{a,\nu}, \quad \nu = 1, 2, \ldots, n_a, \tag{24}$$

$$\sum_{p=0}^{n-1} \beta_\sigma^{[p]} \sum_{j=0}^{r} \mathbf{D}_{0,j}^p u_{s+1}(x_j) = g_{b,\sigma}, \quad \sigma = 1, 2, \ldots, n_b. \tag{25}$$

The discrete boundary conditions Eq.(24) and Eq.(25) are imposed onto the matrix system Eq.(22).

2.2. Multi-domain spectral collocation method for systems of nonlinear ODEs

In this section, we extend the algorithm described in the previous subsection to the solution of a generalized system of ODEs. We consider a system of m ordinary differential equations in m unknown functions $u_i(x),\ i = 1, 2, \ldots, m$ represented as

$$L_i\left[u_1(x), u_2(x), \ldots, u_m(x)\right] = h_i(x), \quad x \in (a, b),\ i = 1, 2, \ldots, m, \tag{26}$$

where L_i is a nonlinear differential operator acting on unknown functions and $h_i(x)$ is a known function of the independent variable x in the i^{th} equation. For illustrative purpose, we assume that the system Eq.(26) is to be solved subject to the two point boundary conditions given by

$$\sum_{j=0}^{m}\sum_{p=0}^{n_j-1} \alpha_{\nu,j}^{[p]} u_j^{(p)}(a) = g_{a,\nu}, \quad \nu = 1, 2, \ldots, n_a, \tag{27}$$

$$\sum_{j=0}^{m}\sum_{p=0}^{n_j-1} \beta_{\sigma,j}^{[p]} u_j^{(p)}(b) = g_{b,\sigma}, \quad \sigma = 1, 2, \ldots, n_b, \tag{28}$$

where $\alpha_{\nu,j}^{[p]}$ and $\beta_{\sigma,j}^{[p]}$ are the constant coefficients of $u_j^{(p)}(x)$ in the boundary conditions, $n_j,\ (j = 1, 2, \ldots, m)$ is the highest derivative order of the variable u_j appearing in the system of equations, and $n_a,\ n_b$ denote the total number of boundary conditions prescribed at $x = a$ and $x = b$, respectively. Applying the QLM on Eq.(26) we obtain the linearized coupled scheme

$$\sum_{l=1}^{m}\sum_{k=0}^{n_j} \gamma_{i,s}^{k,l}(x) u_{l,s+1}^{(k)} = h_i(x) + \sum_{l=1}^{m}\sum_{k=0}^{n_j} \gamma_{i,s}^{k,l}(x) u_{l,s}^{(k)} - L_i\left[u_{1,s}(x), u_{2,s}(x), \ldots, u_{m,s}(x)\right], \quad i = 1, 2, \ldots, m, \tag{29}$$

where

$$\gamma_{i,s}^{k,l} = \frac{\partial L_i}{\partial u_{l,s}^{(k)}}\left[u_{1,s}(x), u_{2,s}(x), \ldots, u_{m,s}(x)\right], \tag{30}$$

denotes the coefficient of the k^{th} derivative of variable $u_l(x)$ in the i^{th} equation. Applying spectral collocation in Eq.(29) we obtain

$$\sum_{l=1}^{m}\left[\sum_{k=0}^{n_j} \gamma_{i,s}^{k,l}(\mathbf{x})\mathbf{D}^k\right]\bar{\mathbf{U}}_l = \mathbf{R}_{i,s}(\mathbf{x}), \quad i = 1, 2, \ldots, m, \tag{31}$$

where

$$\begin{aligned}\bar{\mathbf{U}}_l &= \left[u_l(x_0), u_l(x_1), \ldots, u_l(x_r)\right]^T, \quad \mathbf{R}_{i,s}(\mathbf{x}) = h_i(\mathbf{x}) \\ &+ \sum_{l=1}^{m}\sum_{k=0}^{n_j} \gamma_{i,s}^{k,l}(\mathbf{x}) u_{l,s}^{(k)} - L_i\left[u_{1,s}(\mathbf{x}), u_{2,s}(\mathbf{x}), \ldots, u_{m,s}(\mathbf{x})\right].\end{aligned} \tag{32}$$

We remark that the highest ordered derivative in the variable $u_l,\ (l = 1, 2, \ldots, m)$ may not appear in every one of the m equations. Consequently, some coefficients may fall to zero. Eq.(31) can be written compactly as

$$\sum_{l=1}^{m} \mathbf{A}^{(l,i)} \bar{\mathbf{U}}_l = \mathbf{R}_{i,s}(\mathbf{x}), \quad i = 1, 2, \ldots, m, \tag{33}$$

where

$$\mathbf{A}^{(l,i)} = \sum_{k=0}^{n_j} \gamma_{i,s}^{k,l}(\mathbf{x}) \mathbf{D}^k. \tag{34}$$

Expansion of Eq.(33) results in an $m(r+1) \times m(r+1)$ matrix system that can be represented as

$$\begin{bmatrix} \mathbf{A}^{(1,1)} & \mathbf{A}^{(1,2)} & \mathbf{A}^{(1,3)} & \ldots & \mathbf{A}^{(1,m)} \\ \mathbf{A}^{(2,1)} & \mathbf{A}^{(2,2)} & \mathbf{A}^{(2,3)} & \ldots & \mathbf{A}^{(2,m)} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ \mathbf{A}^{(m-1,1)} & \mathbf{A}^{(m-1,2)} & \mathbf{A}^{(m-1,m)} & \ldots & \mathbf{A}^{(m-1,m)} \\ \mathbf{A}^{(m,1)} & \mathbf{A}^{(m,2)} & \mathbf{A}^{(m,3)} & \ldots & \mathbf{A}^{(m,m)} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{U}}_1 \\ \bar{\mathbf{U}}_2 \\ \vdots \\ \bar{\mathbf{U}}_{m-1} \\ \bar{\mathbf{U}}_m \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \\ \vdots \\ \mathbf{R}_{m-1} \\ \mathbf{R}_m \end{bmatrix}, \tag{35}$$

where $\mathbf{R}_l = [R_l(x_0), R_l(x_1), \ldots, R_l(x_r)]^T, \ \ l = 1, 2, \ldots, m.$

The matrix system Eq.(35) is solved subject to the discrete boundary conditions

$$\sum_{j=1}^{m} \sum_{p=0}^{n_j - 1} \alpha_{\nu,j}^{[p]} \sum_{k=0}^{r} \mathbf{D}_{r,j}^{p} u_{j,s+1}(x_k) = g_{a,\nu}, \quad \nu = 1, 2, \ldots, n_a, \tag{36}$$

$$\sum_{j=1}^{m} \sum_{p=0}^{n_j - 1} \beta_{\sigma,j}^{[p]} \sum_{k=0}^{r} \mathbf{D}_{0,j}^{p} u_{j,s+1}(x_k) = g_{b,\sigma}, \quad \sigma = 1, 2, \ldots, n_b. \tag{37}$$

The boundary conditions Eq.(36)-(37) are imposed on the diagonal submatrices $\mathbf{A}^{(l,l)},\ l = 1, 2, \ldots, m,$ of Eq.(35).

2.3. Error bounds in univariate polynomial interpolation

Below, we state the theorem that governs error bounds in univariate polynomial interpolation;

Theorem 1. *[40][**Theorem of interpolation errors**] If $U(x)$ is a polynomial of degree at most N that interpolates $u(x)$ at $N+1$ distinct grid points $x_0, x_1, x_2, ..., x_N$, in the interval $[a, b]$, and if the first $(N+1)^{th}$ derivatives of $u(x)$ exists and are continuous, then, $\forall\ x \in [a, b]$ there exist an ξ_x for which*

$$|u(x) - U(x)| \leq \frac{1}{(N+1)!} u^{(N+1)}(\xi_x) \prod_{j=0}^{N} (x - x_j). \tag{38}$$

where $U(x)$ is the $N+1$-th degree interpolating polynomial that approximates $u(x)$.

In practical applications, the function $u(x)$ is not known and therefore there is little that can be done on $u^{(N+1)}(\xi_x)$ in the quest to improve accuracy. However, the values of N should ideally be controlled within some optimal value, because it has implications on the memory requirements and consequently the computation time needed to realize results. if, based on the physics of the problem, it is possible to predict the behaviour of the function some control of N may be possible. Eq.(38) suggests that the only sufficient condition to guarantee reduction in approximation errors as the number of grid points increase is the uniform boundedness of higher order derivatives of u within the interval of approximation $x \in [a, b]$. We remark that if the values of higher ordered derivatives increase indefinitely without bound, then the approximations errors may grow as the number of grid points N increase. It is worth noting that the product term $\prod_{j=0}^{N}(x-x_j)$ can be made as small as possible by making a suitable choice of grid points, [41]. Going forward, we reveal new error bound theorems and proofs emanating from an interpolating polynomial constructed using Chebyshev-Gauss-Lobatto nodes. The ultimate goal is to obtain the upper bound of the product term $\prod_{j=0}^{N}(x-x_j)$ with $x_0,\ x_1,\ ,\ldots,x_N$, being the Chebyshev-Gauss-Lobatto nodes defined on the finite interval $[a, b]$ on both single and multiple domains. The following definition shall be required in understanding the key elements of the proofs.

The Chebyshev-Gauss-Lobatto nodes on the interval $x \in [a, b]$ are defined as

$$\{\hat{x}_j\}_{j=0}^{N} = \cos\left(\frac{j\pi}{N}\right), \quad \hat{x}(x) = \frac{2}{b-a}\left[x - \frac{1}{2}(a+b)\right], \quad x \in [a, b]. \tag{39}$$

and are the relative extremes of the N-th degree Chebyshev polynomial [42], $T_N(\hat{x}),\ \hat{x} \in [-1, 1]$. Thus, to obtain the interior Gauss-Lobatto nodes, we solve the equation $T'_N(\hat{x}) = 0$, where the prime denotes differentiation with respect to x. The exterior nodes, -1 and 1, are solutions of $(1-\hat{x}^2) = 0$. Consequently, the complete set of the Gauss-Lobatto nodes defined by equation (39) can be said to be the roots of $(N+1)$-th degree polynomial,

$$L_{N+1}(\hat{x}) = (1-\hat{x}^2)T'_N(\hat{x}). \tag{40}$$

We formulate the theorem below.

Theorem 2 (**Error bound for Chebyshev-Gauss-Lobatto (CGL) grid points in single domain**). *The resulting error bound when CGL grid points, $\{x_j\}_{j=0}^{N} \in [a, b]$, are used in univariate polynomial interpolation is given by*

$$E(x) \leq \frac{4}{2^N(N+1)!}\left(\frac{b-a}{2}\right)^{N+1} u^{(N+1)}(\xi_x). \tag{41}$$

Proof. To prove the analytic expression for the error bound when Chebyshev-Gauss-Lobatto points are used, we first construct a new $(N+1)$-th degree polynomial, L_{N+1} using the relation stated in [43] as

$$L_{N+1}(\hat{x}) = (1-\hat{x}^2)T'_N(\hat{x}) = -N\hat{x}T_N(\hat{x}) + NT_{N-1}(\hat{x}). \tag{42}$$

Since for $\hat{x} \in [-1,1]$, $|T_N(\hat{x})| \leq 1$, applying the triangle inequality to the absolute value of Eq.(42) we have

$$|L_{N+1}(\hat{x})| = |-N\hat{x}T_N(\hat{x}) + NT_{N-1}(\hat{x})| \leq |-N\hat{x}T_N(\hat{x})| + |NT_{N-1}(\hat{x})| \leq N + N = 2N. \tag{43}$$

The absolute value of the leading coefficient of $L_{N+1}(\hat{x})$ is $2^{N-1}N$. The components 2^{N-1} and N come, respectively, from the leading coefficient of $T_N(\hat{x})$ and the application of N-th rule of differentiation on $T_N(\hat{x})$. The factorized monic polynomial expression appearing as a product term in the error bound expression given at Eq.(38) can therefore be written as

$$\prod_{j=0}^{N}(\hat{x}-\hat{x}_j) = \frac{L_{N+1}(\hat{x})}{2^{N-1}N}. \tag{44}$$

This monic polynomial is bounded above by

$$\left|\prod_{j=0}^{N}(x-x_j)\right| = \left|\frac{L_{N+1}(\hat{x})}{2^{N-1}N}\right| \leq \frac{2N}{2^{N-1}N} = \frac{4}{2^N}. \tag{45}$$

For a general interval $x \in [a,b]$ this product term is bounded above by

$$\begin{aligned}\max_{a\leq x\leq b}\left|\prod_{j=0}^{N}(x-x_j)\right| &= \max_{-1\leq\hat{x}\leq 1}\left|\prod_{j=0}^{N}\frac{(b-a)}{2}(\hat{x}-\hat{x}_j)\right| = \left(\frac{b-a}{2}\right)^{N+1}\max_{-1\leq\hat{x}\leq 1}\left|\prod_{j=0}^{N}(\hat{x}-\hat{x}_j)\right| \\ &= \left(\frac{b-a}{2}\right)^{N+1}\max_{-1\leq\hat{x}\leq 1}\left|\frac{L_{N+1}(\hat{x})}{2^{N-1}N}\right| \leq \frac{4}{2^N}\left(\frac{b-a}{2}\right)^{N+1}.\end{aligned} \tag{46}$$

Using Eq.(46) in Eq.(38) the proof is completed. □

2.3.1. Error bound theorem for CGL nodes in multiple domains

Here, we extend Theorem 2, the univariate interpolation polynomial error bound to obtain its variant on a decomposed domain. In the description, it is assumed that the number of grid points is the same for all subintervals.

Theorem 3 (**Error bound in the decomposed domain**). *The error bound when Chebyshev-Gauss-Lobatto grid points* $\{x_j\}_{j=0}^{N} \in [x_{l-1}, \bar{x}_l]$, $l = 1, 2, \ldots, q$, *for the decomposed domain x-variable are used in univariate polynomial interpolation is given by*

$$E(x) \leq \frac{4}{2^N(N+1)!}\left(\frac{L}{2}\right)^{N+1} u^{(N+1)}(\xi_x). \tag{47}$$

where L is the length of each subinterval.

Proof. In the entire domain $[a, b]$, we have that

$$\left| \prod_{j=0}^{N} (x - x_j) \right| \leq \frac{4}{2^N} \left(\frac{b-a}{2} \right)^{N+1}, \quad x \in [a, b]. \tag{48}$$

This implies that in the decomposed domain and at each subinterval, we should have

$$\left| \prod_{j=0}^{N} (x - x_j) \right| \leq \frac{4}{2^N} \left(\frac{L}{2} \right)^{N+1}, \quad x \in [x_{l-1}, \bar{x}_l]. \tag{49}$$

For smooth u, there exists $\xi_\mu \in [x_{\mu-1}, \bar{x}_\mu]$, $\mu = 1, 2, \ldots, p$, for which the value $u^{(N+1)(\xi_\mu)}$ is the absolute extrema of $u^{(N+1)(x)}$ in $[x_{\mu-1}, \bar{x}_\mu]$. This enables us to define the error bound in component form for each subinterval as

$$\left\{ \frac{4}{2^N (N+1)!} \left(\frac{L}{2} \right)^{N+1} u^{(N+1)}(\xi_\mu) \right\}_{\mu=1}^{q}. \tag{50}$$

We define

$$\| \hat{u}^{(N+1)}(\xi) \|_\infty \equiv \max\{u^{(N+1)}(\xi_1), u^{(N+1)}(\xi_2), \ldots, u^{(N+1)}(\xi_q)\} \tag{51}$$

to denote the maximum absolute value of the $(N+1)$-th derivative of $u(x)$ with respect to x in $[a, b]$. Clearly, $\| \hat{u}^{(N+1)}(\xi) \|_\infty = u^{(N+1)}(\xi_x)$, where $u^{(N+1)}(\xi_x)$ is identical to the one at Eq.(41). To expand the error bound over the entire domain, we shall take the largest possible error across all sub-domains which is

$$\frac{4}{2^N (N+1)!} \left(\frac{L}{2} \right)^{N+1} u^{(N+1)}(\xi_x). \tag{52}$$

Using the results at Eq.(52) in Eq.(38) completes the proof. □

To the best of our knowledge, the above theorem is presented for the first time in this article. We note that $L < (b-a)$ and $\left(\frac{L}{2}\right)^{N+1} \ll \left(\frac{b-a}{2}\right)^{N+1}$. Comparing Eq.(41) and Eq.(47) we see that the error in polynomial interpolation is smaller when interpolation is conducted on multiple domains than on a single domain. We emphasize that when approximating the solution to an initial or a boundary value problem using an interpolating polynomial in spectral collocation methods, for interpolation nodes, using the CGL nodes is preferable to using the well known optimal Chebyshev nodes [44]. This is because the CGL nodes are convenient in constructing differentiation matrices as they contain the nodes at the end of computational domain $[a, b]$, which is advantageous when treating the boundary conditions of the problem as opposed to using Chebyshev nodes. The use

of a multiple domains approach and subsequent reduction in the number of grids points N at each subinterval leads to a good approximation. The multiple domain approach is advantageous if interpolation errors that ought to be propagated into the numerical scheme are attributed to unbounded higher ordered derivatives of the unknown function $u(x)$.

3. Numerical experimentation

In this section, we consider several examples of boundary layer equations that are defined over semi-infinite domains to demonstrate the efficiency and accuracy of our proposed method.

3.1. One equation ODE

Example 1. For a single nonlinear boundary layer equation, we first consider a problem of a steady two dimensional laminar flow of a viscous incompressible electrically conducting fluid over a continuous shrinking sheet. The governing equation for this problem is given in similarity form in [1] as

$$f''' + ff'' - \left(f'\right)^2 - M^2 f' = 0, \tag{53}$$

subject to boundary conditions

$$f(0) = 0, \quad f'(0) = -1, \quad f'(\infty) = 0, \tag{54}$$

where prime denotes differentiation with respect to the similarity variable η and M is the magnetic interaction parameter. The QLM scheme for ODE Eq.(53) is

$$\gamma_{3,s} f'''_{s+1} + \gamma_{2,s} f''_{s+1} + \gamma_{1,s} f'_{s+1} + \gamma_{0,s} f_{s+1} = R_s, \tag{55}$$

where

$$\gamma_{3,s} = 1, \;\; \gamma_{s,2} = f_s, \;\; \gamma_{1,s} = -2f'_s - M^2, \;\; \gamma_{0,s} = f''_s, \;\; R_s = f_s f''_s - \left(f'_s\right)^2, \tag{56}$$

and the coefficient matrix is

$$\mathbf{A} = \gamma_{3,s}(\boldsymbol{\eta})\mathbf{D}^3 + \gamma_{2,s}(\boldsymbol{\eta})\mathbf{D}^2 + \gamma_{1,s}(\boldsymbol{\eta})\mathbf{D} + \gamma_{0,s}(\boldsymbol{\eta}). \tag{57}$$

The boundary conditions at the collocation points are

$$f_{s+1}(\eta_r) = 0, \quad \sum_{j=0}^{r} \mathbf{D}_{r,j} f_{s+1}(\eta_j) = -1, \quad \sum_{j=0}^{r} \mathbf{D}_{0,j} f_{s+1}(\eta_j) = 0. \tag{58}$$

The exact analytical solution of Eq.(53) exists for $M \geq 1$ and is known to be

$$f(\eta) = \frac{1}{\sigma}\left(e^{-\sigma\eta} - 1\right), \;\; \sigma = \sqrt{M^2 - 1}. \tag{59}$$

This problem is thus chosen to ease validation of the accuracy of the proposed solution algorithm. The accuracy is found by computing the absolute difference between the numerical approximations at the current iteration level f_{s+1}, and the exact solution. The absolute error formula shall be defined as the infinity norm

$$E_{s+1} = ||f_{s+1} - f||_\infty, \tag{60}$$

where f is the exact solution evaluated at the collocation points.

Example 2. Secondly, we consider the Falkner-Skan equation [16] given in non-dimensional form as

$$f''' + \alpha f f'' + \beta \left[1 - (f')^2\right] = 0, \tag{61}$$

subject to boundary conditions

$$f(0) = 0, \quad f'(0) = 0, \quad f'(\infty) = 1, \tag{62}$$

where f is the dimensionless stream function of the dimensionless normal co-ordinate η, α, β are constants and $\beta \geq 0$. Physically, the Falkner-Skan equation describes a two dimensional flow of incompressible fluid past a stationary semi-infinite impenetrable wedge surface with stream-wise gradient. The surface is inclined at an angle $\beta\pi$, which in the limit of $\beta \to 0$ gives a flat plate, and consequently the Blasius equation with the same boundary conditions for $\alpha = \frac{1}{2}$.

The QLM scheme for Eq.(61) is

$$\gamma_{3,s} f'''_{s+1} + \gamma_{2,s} f''_{s+1} + \gamma_{1,s} f'_{s+1} + \gamma_{0,s} f_{s+1} = R_s, \tag{63}$$

where

$$\gamma_{3,s} = 1, \;\; \gamma_{s,2} = \alpha f_s, \;\; \gamma_{1,s} = -2\beta f'_s, \;\; \gamma_{0,s} = \alpha f''_s, \;\; R_s = \alpha f_s f''_s - \beta \left(1 + f'^2_s\right). \tag{64}$$

The coefficient matrix becomes

$$\mathbf{A} = \gamma_{3,s}(\boldsymbol{\eta})\mathbf{D}^3 + \gamma_{2,s}(\boldsymbol{\eta})\mathbf{D}^2 + \gamma_{1,s}(\boldsymbol{\eta})\mathbf{D} + \gamma_{0,s}(\boldsymbol{\eta}). \tag{65}$$

The boundary conditions at the collocation points are

$$f_{s+1}(\eta_r) = 0, \quad \sum_{j=0}^{r} \mathbf{D}_{r,j} f_{s+1}(\eta_j) = 0, \quad \sum_{j=0}^{r} \mathbf{D}_{0,j} f_{s+1}(\eta_j) = 1. \tag{66}$$

For this problem, the accuracy of the scheme shall be assessed by considering the residual error at the current iteration level $s+1$, which is defined as

$$Res_F = ||F_{s+1}||_\infty, \quad F_{s+1} = f'''_{s+1} + \alpha f_{s+1} f''_{s+1} + \beta \left[1 - (f'_{s+1})^2\right] = 0. \tag{67}$$

The iterative scheme is said to have converged after ω iterations if the infinity norm $||f_{s+1} - f_s||_\infty \leq \tau$, $\forall s \geq \omega$, for some specified tolerance τ.

3.2. System of ODEs

Example 3. We consider, here, the problem of a boundary layer flow over an unsteady stretching sheet in the presence of Hall effect and heat transfer over the stretching surface. The governing system of equations are given in dimensionless form by El-Aziz [3] as

$$f''' + ff'' - \left(f'\right)^2 - A\left(f' + \frac{\eta}{2}f''\right) - \frac{M}{1+m^2}\left(f' + mh\right) = 0, \tag{68}$$

$$h'' + fh' - f'h - A\left(h + \frac{\eta}{2}h'\right) + \frac{M}{1+m^2}\left(mf' - h\right) = 0, \tag{69}$$

$$\frac{1}{Pr}\theta'' + f\theta' - 2f'\theta - \frac{A}{2}\left(3\theta + \eta\theta'\right) = 0, \tag{70}$$

subject to boundary conditions

$$f(0) = 0, \quad f'(0) = 1, \quad h(0) = 0, \quad \theta(0) = 1, \quad f'(\infty) = 0, \quad h(\infty) = 0, \quad \theta(\infty) = 0. \tag{71}$$

Here, $f'(\eta)$, $h(\eta)$, and $\theta(\eta)$ are unknown functions representing the axial velocity, the transverse velocity, and the non-dimensionless temperature, respectively. The primes denote differentiation with respect to η, A is the unsteadiness parameter, M is the magnetic parameter, m is the Hall parameter and Pr is the Prandtl number.

The QLM scheme for the system of the ODEs Eq.(68) - Eq.(70) is

$$\gamma_{1,s}^{3,1} f'''_{s+1} + \gamma_{1,s}^{2,1} f''_{s+1} + \gamma_{1,s}^{1,1} f'_{s+1} + \gamma_{1,s}^{0,1} f_{s+1} + \gamma_{1,s}^{0,2} h_{s+1} = R_{1,s}, \tag{72}$$

$$\gamma_{2,s}^{1,1} f'_{s+1} + \gamma_{2,s}^{0,1} f_{s+1} + \gamma_{2,s}^{2,2} h''_{s+1} + \gamma_{2,s}^{1,2} h'_{s+1} + \gamma_{2,s}^{0,2} h_{s+1} = R_{2,s}, \tag{73}$$

$$\gamma_{3,s}^{1,1} f'_{s+1} + \gamma_{3,s}^{0,1} f_{s+1} + \gamma_{3,s}^{2,3} \theta''_{s+1} + \gamma_{3,s}^{1,3} \theta'_{s+1} + \gamma_{3,s}^{0,3} \theta_{s+1} = R_{3,s}, \tag{74}$$

where

$$\begin{aligned}
&\gamma_{1,s}^{3,1} = 1, \ \ \gamma_{1,s}^{2,1} = fs - A\frac{\eta}{2}, \ \ \gamma_{1,s}^{1,1} = -2f'_s - A - \frac{M}{1+m^2}, \ \ \gamma_{1,s}^{0,1} = f''_s, \ \ \gamma_{1,s}^{0,2} = -\frac{Mm}{1+m^2}, \\
&\gamma_{2,s}^{1,1} = -h_s + \frac{Mm}{1+m^2}, \ \ \gamma_{2,s}^{0,1} = h'_s, \ \ \gamma_{2,s}^{2,2} = 1, \ \ \gamma_{2,s}^{1,2} = f_s - A\frac{\eta}{2}, \ \ \gamma_{2,s}^{0,2} = -f'_s - A - \frac{M}{1+m^2}, \\
&\gamma_{3,s}^{1,1} = -2\theta_s, \quad \gamma_{3,s}^{1,0} = \theta'_s, \quad \gamma_{3,s}^{2,3} = \frac{1}{Pr}, \quad \gamma_{3,s}^{1,3} = f_s - A\frac{\eta}{2}, \quad \gamma_{3,s}^{0,3} = -2f'_s - \frac{3A}{2}, \\
&R_{1,s} = f'_s f''_s - \left(f'_s\right)^2, \quad R_{2,s} = f_s h'_s - f'_s h_s, \quad R_{3,s} = f_s \theta'_s - 2f'_s \theta_s.
\end{aligned} \tag{75}$$

The coefficient matrices are

$$\begin{aligned}
&\mathbf{A}^{(1,1)} = \gamma_{1,s}^{3,1}(\boldsymbol{\eta})\mathbf{D}^3 + \gamma_{1,s}^{2,1}(\boldsymbol{\eta})\mathbf{D}^2 + \gamma_{1,s}^{1,1}(\boldsymbol{\eta})\mathbf{D} + \gamma_{1,s}^{0,1}(\boldsymbol{\eta}), \qquad \mathbf{A}^{(1,2)} = \gamma_{1,s}^{0,2}(\boldsymbol{\eta}), \qquad \mathbf{A}^{(1,3)} = \mathbf{0} \\
&\mathbf{A}^{(2,1)} = \gamma_{2,s}^{1,1}(\boldsymbol{\eta})\mathbf{D} + \gamma_{2,s}^{0,1}(\boldsymbol{\eta}), \qquad \mathbf{A}^{(2,2)} = \gamma_{2,s}^{2,2}(\boldsymbol{\eta})\mathbf{D}^2 + \gamma_{2,s}^{1,2}(\boldsymbol{\eta})\mathbf{D} + \gamma_{2,s}^{0,1}(\boldsymbol{\eta}), \qquad \mathbf{A}^{(2,3)} = \mathbf{0}, \\
&\mathbf{A}^{(3,1)} = \gamma_{3,s}^{1,1}(\boldsymbol{\eta})\mathbf{D} + \gamma_{3,s}^{0,1}(\boldsymbol{\eta}), \qquad \mathbf{A}^{(3,2)} = \mathbf{0}, \qquad \mathbf{A}^{(3,3)} = \gamma_{3,s}^{2,3}(\boldsymbol{\eta})\mathbf{D}^2 + \gamma_{3,s}^{1,3}(\boldsymbol{\eta})\mathbf{D} + \gamma_{3,s}^{0,3}(\boldsymbol{\eta}),
\end{aligned} \tag{76}$$

with $\mathbf{0}$ being a zero matrix of size $(r+1)\times(r+1)$. The boundary conditions are evaluated at the collocation points as

$$\begin{aligned}
&f_{s+1}(\eta_r) = 0, \;\; \sum_{j=0}^{r} \mathbf{D}_{r,j} f_{s+1}(\eta_j) = 1, \;\; h_{s+1}(\eta_r) = 0, \;\; \theta_{s+1}(\eta_r) = 1, \\
&\sum_{j=0}^{r} \mathbf{D}_{0,j} f_{s+1}(\eta_j) = 0, \;\; h_{s+1}(\eta_0) = 0, \quad \theta_{s+1}(\eta_0) = 0.
\end{aligned} \tag{77}$$

4. Results and discussions

In this section, numerical results obtained by solving the nonlinear boundary layer problems given in Examples 1 to 3 are presented in tables and graphs. The results for each example have been considered separately and they demonstrate various aspects of the proposed method of solution. The validity of the present method is checked by comparing the current results with results existing in literature and good agreement is observed. We emphasize that the numerical results for the first problem, which possesses an exact solution, were meant to validate the accuracy of the algorithm by allowing a comparison between the approximate numerical values and the exact solution. The numerical scheme in Example 1 was implemented on Mathematica and the precision was set at 300 digits. The semi-infinite domain $[0,\infty)$ was truncated to the finite interval $[0,30]$ and 200 grids points were used in both single and multiple domains approaches. The absolute errors given in Table 1 were computed as the difference between the exact and numerical values of the function f at different iteration levels. The data in Table 1 confirms that the algorithm is very accurate, because absolute errors of order 10^{-23} are recorded after only 5 iterations. We affirm that even though the performance of single domain approach is comparable to that of multiple domains approach in terms of accuracy, for Example 1, such is attributed to the advantages of high precision that was set on Mathematica platform where the results were generated. It must be noted that such high precision is computationally expensive. From Table 1 we can infer that the multiple domains approach is advantageous in terms of computation time as opposed to the single domain approach. The problem seems well-posed in the case of the multiple domains approach as signified by the smaller condition number of the coefficient matrix. The numerical results in Table 1 suggest that

the iterative scheme converges after 5 iterations. Figure 2 shows the graph of absolute error values for Example 1, which were recorded after 5 iterations, against the number of subintervals. The number of grid points in the entire domain was maintained at 200. The aim was to determine the number of subintervals that are required to yield the most accurate results, while at the same time saving on the computational time. The figure suggests that the optimal number of subintervals for Example 1 is 6. With 6 subintervals we obtain the most accurate results, within a remarkably short computation time. A further increase in the number of subintervals leads to deterioration of accuracy, because at each further subdivision, the piecewise interpolating polynomial approximating the unknown function will be of lesser degree.

In Example 2, there exists no exact solution and the accuracy of the numerical scheme is assessed by considering the residual error as defined in Eq.(67). The numerical scheme for the Falkner-Skan equation was also implemented in Mathematica. In Table 2, the numerical values of the local skin friction, $f''(0)$, for the Falkner-Skan equation, using the current method, are compared with some of the most accurate results found in literature, where the problem was solved using the MatLab built-in routine bvp4c. A good agreement is observed for different values of the inclination angle, β. From the results in Table 2, we can infer that skin friction increases with an increase in β. This can be explained by an increase in the value of β resulting in an increase in the size of the boundary layer. In Figure 3, the residual errors obtained by varying the number of subintervals q is compared at different iteration levels. The number of grid points in the entire domain was maintained constant. From the graph, we conclude that, in terms of accuracy, the best results for the Falkner-Skan equation are obtained when 2 subintervals are used. This can be explained by, in this particular case, a small value of the parameter $\beta = 0.5$ being considered. The implication is that the thickness of the boundary layer is small. This restriction informed us when truncating the semi-infinite $[0, \infty)$ to the finite interval $[0, 10]$, which is relatively small. For a small interval of approximation, a low number of subintervals are required to yield the most accurate results, showing the advantages of approximating a function using interpolating polynomials of higher degree. It is worth noting that in Example 2, we took advantages of the high precision that was set on Mathematica just as in the case of Example 1.

The numerical scheme of Example 3 was implemented in MatLab. Experimentally, we established that the truncation of the semi-infinite domain to a large finite domain $[0, 70]$ yielded the best results for this problem. In Table 3, the residual errors values for f, h, and θ that are obtained

at different iteration levels using 5 subintervals have been presented. For comparison, residual error values obtained using 2 subintervals have also been presented in Table 4. We observe that the use of 5 subintervals yields more accurate results for a short CPU time than when 2 subintervals are used. This is due to the less dense coefficient matrices associated with the use of many subintervals for a large computational domain, which is the focus in the present approach. Well-posedness is depicted by the small condition number of the coefficient matrix obtained in the case of 5 subintervals. We remark that in this particular example, with a large domain of approximation, the usefulness of the multiple domains approach has been exposed as MatLab platform executes numerals with an accuracy of up to 16 digits. In Figure 4 we display convergence graphs for the unknown functions f, h, and θ. The figure depicts that the numerical scheme is convergent and that convergence is realized after 5 iterations.

Table 1: Spectral approximation of Example 1 on overlapping grids using $N_x = 40,\ q = 5$, for $\eta_\infty = 30,\ M = 5$.

Iteration, s	Single Domain	Multiple Domains
1	3.16939×10^{-2}	3.16939×10^{-2}
2	1.21909×10^{-4}	1.21909×10^{-4}
3	1.47038×10^{-9}	1.47038×10^{-9}
4	1.86137×10^{-19}	1.86137×10^{-19}
5	1.17462×10^{-23}	1.17462×10^{-23}
6	1.17462×10^{-23}	1.17462×10^{-23}
CPU time (sec)	16.6755	4.3368
cond(A)	5.5899×10^{9}	6.4810×10^{7}

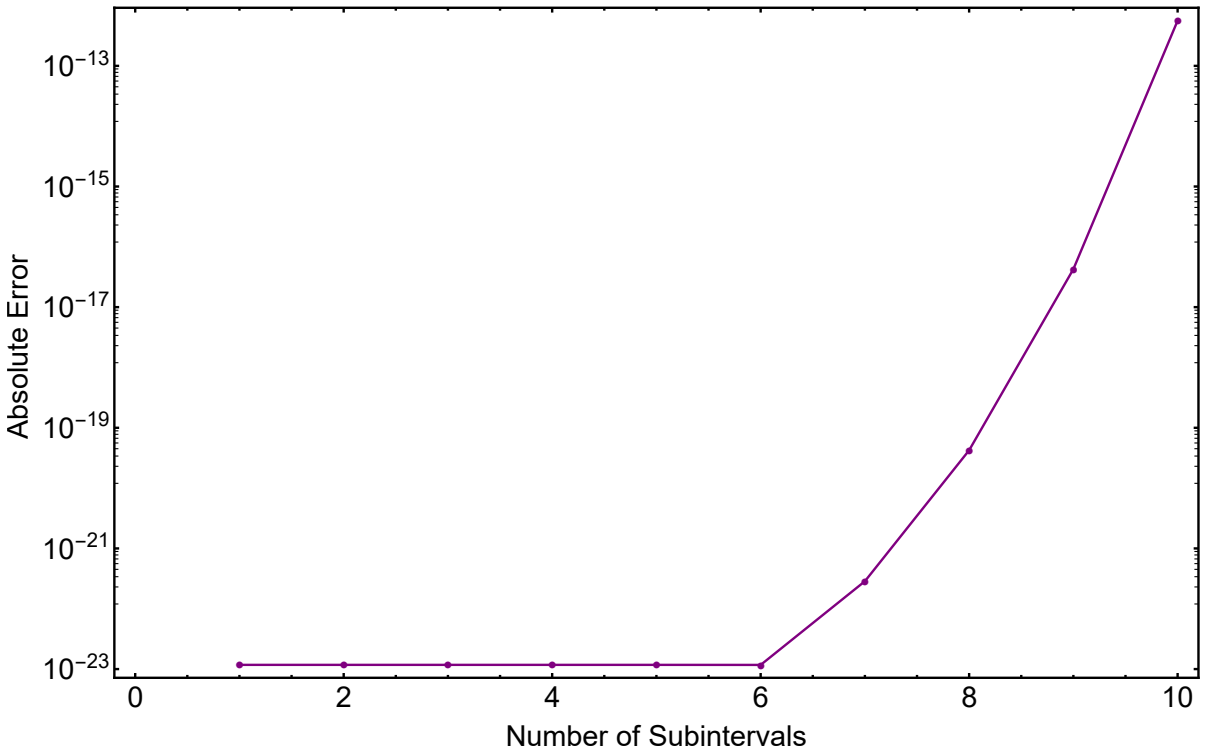


Figure 2: Graph of the absolute error values against the number of subintervals using 200 grid points

Table 2: Comparison between the overlapping grid approach results and the bvp4c numerical results [45] for $f''(0)$ at selected values of β for Falkner-Skan Example 2 using $N_x = 20,\ q = 10$, for $\eta_\infty = 10,\ \alpha = 1$.

β	Current results		Numerical
	s=2	s=3	bvp4c
0.4	0.854422	0.854421	0.854421
0.8	1.120265	1.120268	1.120268
1.2	1.335723	1.335721	1.335721
1.6	1.521514	1.521514	1.521514
2.0	1.687218	1.687218	1.687218

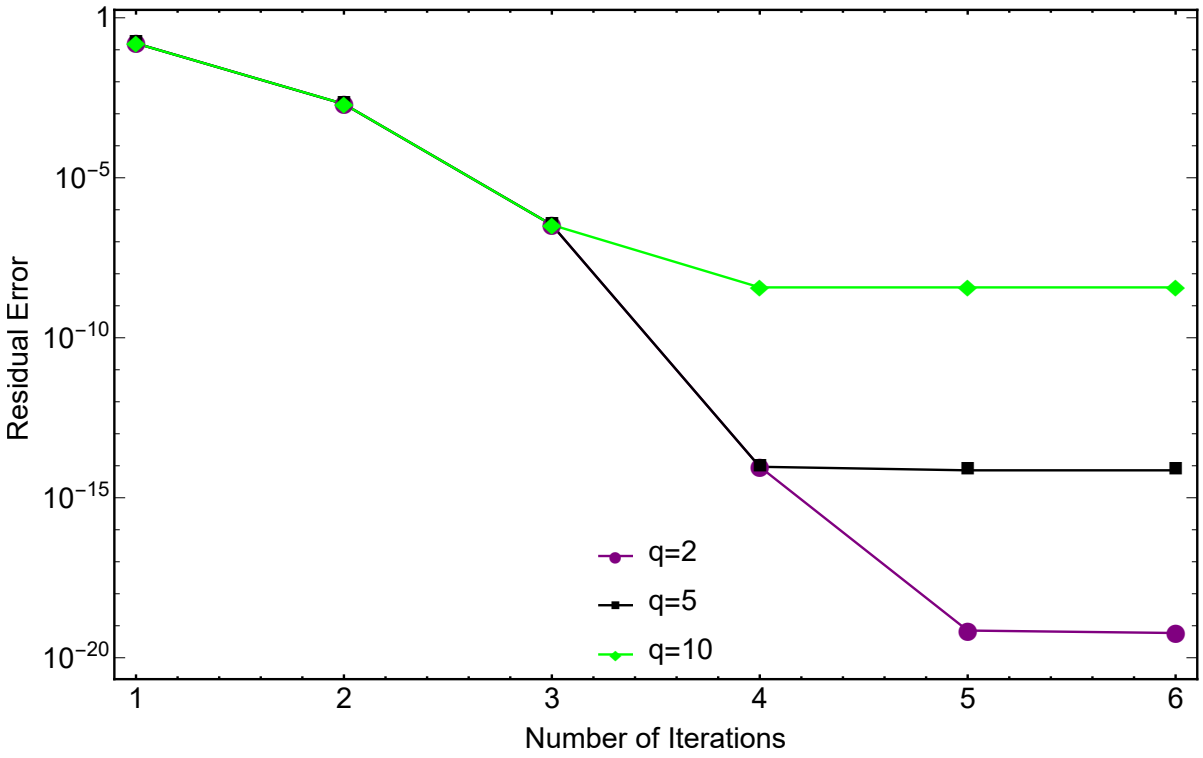


Figure 3: Graph of the residual error against iterations for different domain decomposition

Table 3: Spectral approximation of Example 3 on overlapping grids using $N_x = 40,\ q = 5$, for $\eta_\infty = 70,\ A = 0.5,\ M = 1,\ m = 0.5$.

Iteration	Residual error		
s	$\|\|F_{s+1}\|\|_\infty$	$\|\|H_{s+1}\|\|_\infty$	$\|\|\Theta_{s+1}\|\|_\infty$
1.00	3.75070e-001	5.67287e-002	5.11974e-002
2.00	3.58227e-002	9.17907e-004	1.56959e-002
3.00	2.58387e-004	2.26789e-005	3.87065e-004
4.00	1.01330e-008	1.70642e-009	6.96569e-008
5.00	3.09963e-012	1.16504e-013	5.87419e-013
CPU time (sec)	0.074567		
cond(A)	5.23456e+07		

Table 4: Spectral approximation of Example 3 on overlapping grids using $N_x = 100$, $q = 2$, for $\eta_\infty = 70$, $A = 0.5$, $M = 1$, $m = 0.5$.

Iteration	Residual error		
s	$\|\|F_{s+1}\|\|_\infty$	$\|\|H_{s+1}\|\|_\infty$	$\|\|\Theta_{s+1}\|\|_\infty$
1.00	4.38484e-002	6.07923e-003	1.41794e-002
2.00	2.00334e-004	8.59323e-006	5.16436e-004
3.00	6.24130e-007	8.48203e-010	1.04837e-007
4.00	3.94428e-007	6.67558e-011	1.60723e-010
5.00	1.88953e-007	4.69308e-011	1.25923e-010
CPU time (sec)	0.25463		
cond(A)	7.86324e+09		

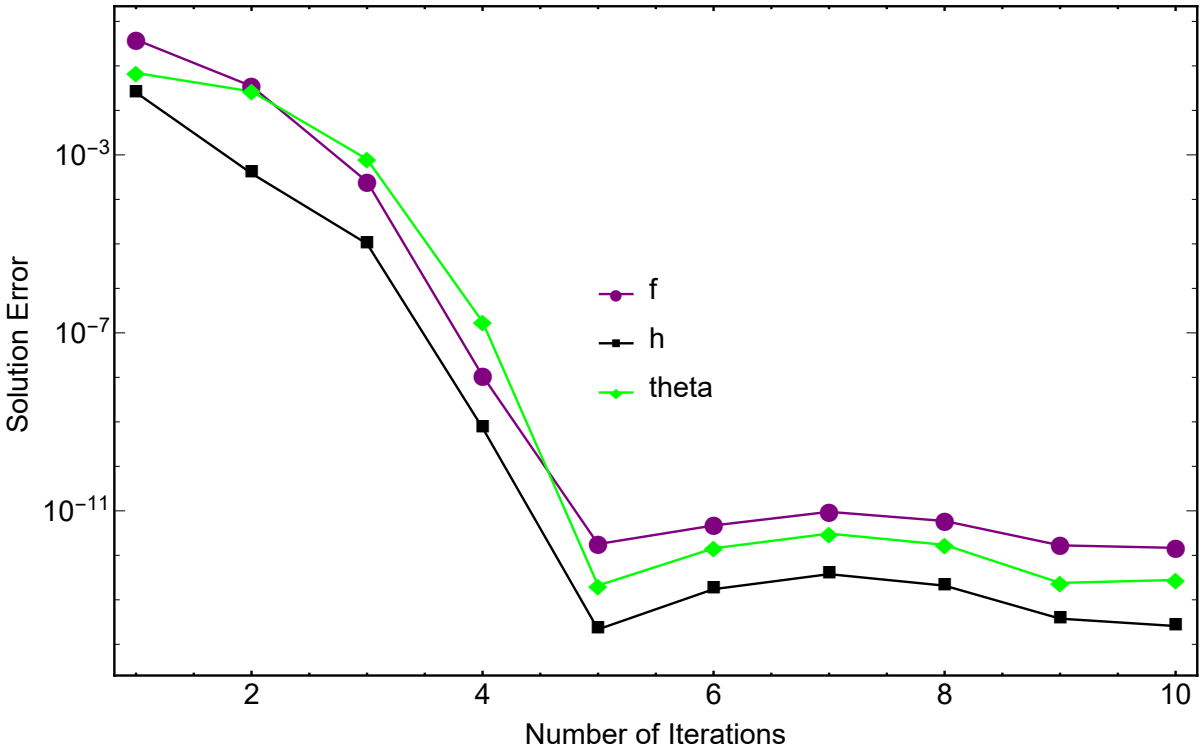


Figure 4: Convergence graphs for f, h, and θ

5. Conclusion

In this study, the spectral collocation method of solution based on overlapping grids has been described and successfully applied to solve nonlinear boundary layer equations characterized as a single or system of ordinary differential equations. It is found that the results of the present work are in agreement with those obtained by other methods. Our aim was to demonstrate that the proposed approach is highly accurate and computationally efficient when applied on problems defined over large computational domains and such has been confirmed. The improved accuracy is attributed to the well-conditioned nature of the matrices resulting from the use of a multiple domains approach for large approximation domains. The new error bound theorems and proofs on polynomial interpolation that have been presented to theoretically form the basis for adopting the current algorithm. The error bounds theorems are based on Chebyshev-Gauss-Lobatto points and Lagrange interpolating polynomials that have been used to construct the approximating functions for the solution of the nonlinear ODEs. The method can be applied to other boundary layer problems that are described as systems of ordinary differential equations.

Acknowledgement

The authors are thankful to the University of KwaZulu-Natal for providing valuable research resources. The first author also thanks, Taita Taveta University, Kenya.

Declaration of interest:

None

References

[1] X. Su, L. Zheng, and X. Zhang, *DTM-BF Method and Dual Solutions for an Unsteady MHD Flow Over a Permeable Shrinking Sheet With Velocity Slip*, Applied Mathematics & Mechanics, 33(2012), pp. 1555-1568.

[2] K. Parand, N. Pakniat, and Z. Delafkar, *Numerical Solution of the Falkner-Skan Equation with Stretching Boundary by Collocation Method*, International Journal of Nonlinear Science, 11(2011), pp. 275-283.

[3] M.A. El-Aziz, *Flow and heat transfer over an unsteady stretching surface with Hall effect*, Meccanica, 45(2010), pp. 97-109.

[4] E. Magyari, and B. Keller, *Heat and Mass Transfer in the Boundary Layers on an Exponentially Stretching Continuous Surface*, Physics D Applied Physics, 32(1999), pp. 577-585.

[5] B.C. Sakiadis, *Boundary-Layer Behavior on Continuous Solid Surfaces: II. The Boundary-Layer on Continuous Flat Surface*, AIChE Journal, 7(1961), pp. 221- 225.

[6] L. Crane, *Flow Past a Stretching Plate*, ZAMP, 21(1970), pp. 645-647.

[7] A. Chakrabarthi, and A.S. Gupta, *Hydromagnetic Flow and Heat Transfer Over a Stretching Sheet*, Quarterly of Applied Mathematics, 37(1979), pp. 73-78.

[8] T. Chiam, *Magneto Hydrodynamic Boundary Layer Flow Due to a Continuous Moving Flat Plate*, Computers & Mathematics with Applications, 26(1993), pp. 1-8.

[9] A.K. Jhankal, and M. Kumar, *MHD Boundary Layer Flow Past a Stretching Plate with Heat Transfer*, International Journal of Engineering and Science, 2(2013), pp. 9-13.

[10] N.F.M, Noor, O. Abdulaziz, and I. Hashim, *MHD Flow and Heat Transfer in a Thin Liquid Film on an Unsteady Stretching Sheet by the Homotopy Analysis Method*, International Journal for Numerical Methods in Fluids, 63(2010), pp. 357-373.

[11] C.Y. Wang, *Liquid Film on an Unsteady Stretching Sheet*, Quarterly of Applied Mathematics, 48(1990), pp. 601- 610.

[12] M. Miklavcic, and C.Y. Wang, *Viscous Flow Due to a Shrinking Sheet*, Quarterly of Applied Mathematics, 64(2006), pp. 283-290.

[13] M. Katagiri, *The Effect Hall Currents on the Viscous Flow Magnetohydrodynamic Boundary Layer Flow past a semi-infinite flat plate*, Physical Society of Japan, 27(2015), pp. 40-51.

[14] H. Sato, *The Hall Effect in the Viscous Flow of Ionized Gas Between Two Parallel Plates Under Transverse Magnetic Field*, Physical Society of Japan, 16(1961), pp. 14-27.

[15] I. Pop, and V.M. Soundalgekar, *Effects of Hall-Current on Hydromagnetic Flow Near a Porous Plate*, Acta Mechanica, 20(1974), pp. 315-318.

[16] V.M. Falkner, and S.W. Skan, *Some Approximate Solutions of the Boundary Layer Equations*, Philosophical Magazine, 12(1931), pp. 865-896.

[17] M.B. Zaturska, and W.H.H. Banks, *A New Solution Branch of the Falkner-Skan Equation*, Acta Mechanica, 152(2001), pp. 197-201.

[18] A. Asaithambi, *Solution of the Falkner-Skan Equation by Recursive Evaluation of Taylor Coefficients*, Computational and Applied Mathematics, 176(2005), pp. 203-214.

[19] N.S. Elgazery, *Numerical Solution for the Falkner-Skan Equation*, Chaos, Solitons & Fractals, 35(2008), pp. 738-746.

[20] E.M.A. Elbashbeshy, and M.A.A. Bazid, *Heat Transfer Over an Unsteady Stretching Surface*, Heat and Mass Transfer, 41(2004), pp. 1-4.

[21] S.J. Liao, *An Analytic Solution of Unsteady Boundary Layer Flows Caused by an Impulsively Stretching Plate*, Communications in Nonlinear Science and Numerical Simulation, 11(2006), pp. 326-339.

[22] Y. Tan, and S.J. Liao, *Series Solution of Three-Dimensional Unsteady Laminar Viscous Flow Due to a Stretching Surface in a Rotating Fluid*, Journal of Applied Mechanics, 74(2007), pp. 1011-1018.

[23] T. Hayat, Z. Abbas, and M. Sajid, *On The Analytic Solution of Magnetohydrodynamic Flow of a Second Grade Fluid Over a Shrinking Sheet*, Journal of Applied Mechanics, Transactions ASM, 74(2007), pp. 1165-1171.

[24] N.F.M. Noor, S.A. Kechilb, and I. Hashim, *Simple Nonperturbative Solution For MHD Viscous Flow Due to a Shrinking Sheet*, Communications in Nonlinear Science and Numerical Simulation, 15(2010), pp. 144-148.

[25] T. Fang, and J. Zhang, *Closed Form Exact Solutions of MHD Viscous Flow Over a Shrinking Sheet*, Communications in Nonlinear Science and Numerical Simulation, 14(2009), pp. 2853-2857.

[26] C. Midya, *Exact Solutions of Chemically Reactive Solute Distribution in MHD Boundary Layer Flow over a Shrinking Surface*, Chinese Physics Letter, 29(2012), pp. 14-27.

[27] M. Sajid, and T. Hayat, *The Application of Homotopy Analysis Method for MHD Viscous Flow Due to a Shrinking Sheet*, Chaos, Solitons & Fractals, 39(2009), pp. 1317-1323.

[28] D.R. Hartree, *On an Equation Occurring in Falkner and Skan's Approximate Treatment of the Equations of the Boundary Layer*, Proceedings of the Cambridge Philosophical Society, 33(1937), pp. 223-239.

[29] H. Weyl, *On the Differential Equations of the Simplest Boundary-Layer Problem*, Annals of Mathematics, 43(1942), pp. 381-407.

[30] B.L. Kuo, *Application of the Differential Transformation Method to the Solutions of Falkner-Skan Wedge Flow*, Acta Mechanica, 164(2003), pp. 161-174.

[31] S.J. Liao , and A. Campo, *Analytic Solutions of the Temperature Distribution in Blasius Viscous Flow Problems*, Journal of Fluid Mechanics, 453(2002), pp. 411-425.

[32] S. Abbasbandy, and T. Hayat, *Solution of the MHD Falkner-Skan Flow by Homotopy Analysis Method*, Communications in Nonlinear Science and Numerical Simulation, 14(2009), pp. 3591-3598.

[33] S.S Motsa, P. Sibanda, and S. Shateyi, *A New Spectral-Homotopy Analysis Method for Solving a Nonlinear Second Order BVP*, Communications in Nonlinear Science and Numerical Simulation, 15(2010), pp. 2293-2302.

[34] S.S Motsa, and P. Sibanda, *An Efficient Numerical Method For Solving Falkner-Skan Boundary Layer Flows*, International Journal for Numerical Methods in Fluids, 69(2012), pp. 499-508.

[35] M. Lakestani, *Numerical Solution For the Falkner-Skan Equation Using Chebyshev Cardinal Functions*, Acta Universitas Apulensis, 27(2011), pp. 229-238.

[36] L.N. Trefethen, *Spectral Methods in MATLAB*, SIAM, Philadelphia, 2000.

[37] R.E. Bellman, and R.E. Kalaba, *Quasilinearization and Nonlinear Boundary-Value Problems*, Elsevier, New York, 1965.

[38] V. Lakshmikantham, *An Extension of the Method of Quasilinearization*, Optimization Theory and Applications, 82(1994), pp. 315-321.

[39] H.H. Yang, B.R. Seymour, and B.D. Shizgal, *A Chebyshev Pseudospectral multi-Domain Method for Steady Flow Past a Cylinder, up to Re=150*, Computers & Fluids, 23(1994), pp. 829-852.

[40] G. Birkhoff, and A. Priver, *Hermite Interpolation Errors for Derivatives*, Mathematical Physics, 46(1967), pp. 440-447.

[41] W. Cheney, and D. Kincaid, *Numerical Mathematics and Computing*, 6th Edition, Belmont, CA, Thomson Higher Education, 2004.

[42] B. Fischer, and R. Freund, *Chebyshev Polynomials Are Not Always Optimal*, Approximation Theory, 65(1991), pp. 261-272.

[43] H.E. Salzer, *Converting Interpolation Series into Chebyshev Series by Recurrence Formulas*, Mathematics of Computation, 30(1976), pp. 295-302.

[44] S. Ghili, and C. Iaccarino, *Reusing Chebyshev Points for Polynomial Interpolation*, Numerical Algorithms, 70(2015), pp. 249-267.

[45] A. Khidir, *A note on the Solution of General Falkner-Skan Problem by two Novel Semi-analytical Techniques*, Propulsion and Power Research, 4(2015), pp. 212-220.

Chapter 3

A Modified Spectral Collocation Method of Solution for One-Dimensional Hyperbolic PDEs

In this chapter, we extend the discussion on spectral collocation-based solution techniques to one-dimensional partial differential equations. The non-overlapping grids based multidomain spectral collocation method is developed and applied to solve hyperbolic PDEs defined over large time intervals. The proposed solution approach involves decomposing the large time interval into smaller non-overlapping subintervals and solving the PDE independently on each of these subintervals. We aim to demonstrate that a reduction in the size of the computational domain in each time subinterval guarantees accurate results within a short computation time. A single domain approach is considered in space to limit the present investigation to the benefits of using non-overlapping subintervals in a large time interval. The new error bound theorems, with proofs in bivariate polynomial interpolation using Chebyshev-Gauss-Lobatto nodes, are given to explain the advantages of the proposed solution algorithm.

Wave Motion 88 (2019) 57–72

Contents lists available at ScienceDirect

Wave Motion

journal homepage: www.elsevier.com/locate/wamot

Solving hyperbolic partial differential equations using a highly accurate multidomain bivariate spectral collocation method

F.M. Samuel [a,c,*], S.S. Motsa [a,b]

[a] University of KwaZulu-Natal, School of Mathematics, Statistics, and Computer Science, Private Bag X01, Scottsville, 3209, South Africa
[b] University of Swaziland, Department of Mathematics, Private Bag 4, Kwaluseni, Swaziland
[c] Taita Taveta University, Department of Mathematics, 635-80300, Voi, Kenya

ARTICLE INFO

Article history:
Received 19 October 2018
Received in revised form 2 December 2018
Accepted 27 January 2019
Available online 6 February 2019

MSC:
65M15
65M55
65M70

Keywords:
Multidomain
Spectral collocation
Bivariate polynomial interpolation
Error bounds
Hyperbolic PDEs

ABSTRACT

In this article, the non-overlapping grids based multidomain bivariate spectral collocation method is applied to solve hyperbolic partial differential equations(PDEs) defined over large time domains. The article is among the very first works which consider the multidomain approach with respect to the time interval for hyperbolic PDEs. The proposed method is based on decomposing the time domain into smaller non-overlapping subintervals and solving the PDE independently on each of these subintervals. In this study, we aim at showing that the reduction in the size of the computational domain at each subinterval guarantees accurate results within a short computational time. In the solution process, the approximate solutions of the PDEs are approximated using bivariate Lagrange interpolating polynomials. The PDEs are discretized in both time and space variables using the spectral collocation, unlike previous studies where spectral collocation method has been applied on space variable only and finite difference based discretization in the time variable and vice versa. The resulting linear systems of algebraic equations are then solved independently at each subinterval with the continuity equation being employed to obtain initial conditions in subsequent subintervals. Finally, the approximate solutions of the PDEs are obtained by matching the solutions on different subintervals along common boundaries. The new error bound theorems and proofs for bivariate polynomial interpolation using Gauss–Lobatto nodes given explain the advantages of the proposed solution algorithm. The effectiveness and accuracy of the proposed method are demonstrated by presenting error analysis and the computational time for the solution of well known hyperbolic PDEs that have been reported in the literature. The method can be adopted and extended to solve problems in real life that are modeled by hyperbolic PDEs.

1. Introduction

Hyperbolic partial differential equations are highly applicable in various branches of engineering and science [1]. In particular, many of the equations encountered in mechanics [2] are hyperbolic and therefore their study has long been a subject of sizeable contemporary interest among researchers. Hyperbolic partial differential equations are initial value

* Corresponding author at: University of KwaZulu-Natal, School of Mathematics, Statistics, and Computer Science, Private Bag X01, Scottsville, 3209, South Africa.
E-mail address: felexmutua@gmail.com (F.M. Samuel).

https://doi.org/10.1016/j.wavemoti.2019.01.014
0165-2125/

problems [3]. One of the important aspects that have been reported in the literature in regards to the development of efficient numerical schemes for solving an initial value problem using spectral collocation-based methods is the size of the computational domain. It has been observed that, when the size of the computational domain is large, many grid points are required to achieve results with stringent accuracy requirements [4]. The use of many grid points poses an ominous problem, requiring a significant amount of computer memory and CPU time. In some instances, the use of large number grid points does not guarantee improved accuracy [5–7]. An increase in the number of grid points increases the size of the resulting coefficient matrix and correspondingly its condition number, a property that deteriorates the accuracy of results of the linear system of equations [8,9]. A large number of grid points also introduces roundoff errors that are associated with the approximation of a function by an interpolating polynomial of a high degree. A similar decay in accuracy is evident in results presented by authors in [10,11]. The most elegant technique that has been used in an attempt to revolve around this challenge is the introduction of multidomain based approaches in the method of solution.

In regards to numerical solutions of hyperbolic PDEs using domain decomposition techniques, close scrutiny of the literature indicates that previous studies focused on the application of overlapping grids approach on the space variable only. The discretization has widely been performed using the spectral collocation in space variable and the finite difference in the time variable. Dehghan et al. [12] applied a method based on the pseudo-spectral collocation method with domain decomposition algorithm for approximating the spatial variable in a coupled nonlinear Schrödinger equations and demonstrated that such an algorithm reduce the effect of round-off errors. Later, in [13], authors considered the long-time behavior of numerical solutions for the Klein–Gordon–Schrödinger equations using a similar approach. Kopriva [14] developed a multidomain Chebyshev spectral method for solving hyperbolic partial differential equations using spectral collocation discretization in space and finite difference discretization in time. A pseudo-spectral method based on the overlapping grids multidomain technique for the numerical solution of the Sine–Gordon equation in one and two spatial dimensions was also applied by Taleei et al. in [15]. In their work, the overlapping domain decomposition technique coupled with spectral collocation discretization was performed on the spatial variable, and an implicit fourth-order Runge–Kutta method was adopted to carry out discretization in the entire time variable. A substantial literature on the numerical methods of solution for two-dimensional hyperbolic PDEs can be found in work by the authors in [16,17] and the references given therein. Although the overlapping grids approach that has been applied previously on space only lead to sparse matrices of the resulting system of linear algebraic equations, such matrices are large and take considerably large CPU time to invert. In the present study, a multi-domain approach based on the decomposition of large time interval into smaller non-overlapping grids is adopted. This approach preserves benefits of dealing with small-sized matrices as the solution is solved independently on each time subinterval. One dimensional hyperbolic PDEs are solved using spectral collocation discretization in both space and time variables.

The non-overlapping grids based multidomain approaches in spectral methods have been applied to solve nonlinear ordinary differential equations that model chaotic systems described as first order systems of equations, [18,19]. In the non-overlapping grids based multidomain spectral collocation method, the differential equation is solved independently on each subinterval and solutions across the different subintervals are matched along the common boundary. In the case of nonlinear differential equations, the matching procedure is evoked after the solution in each subdomain converge. Most recently, Motsa et al. [20] developed and applied the bivariate spectral collocation method to solve nonlinear parabolic evolutionary PDEs that are defined over a small time interval and reported that the method yield very accurate results over a short CPU time. When the same method was applied to parabolic PDEs defined over large time intervals, the accuracy of the numerical results deteriorated. We aim at addressing this downfall in regard to the application of the method to hyperbolic PDEs.

In this work, we aim at extending the bivariate spectral collocation method previously applied on nonlinear evolutionary parabolic PDEs to obtain solutions of hyperbolic PDEs defined on large time interval by in-cooperating the non-overlapping grids domain decomposition technique. We also aim at demonstrating that the adoption of a multi-domain approach in solving such hyperbolic PDEs yields very accurate results at a considerable short CPU time. We remark that to the best of our knowledge, the non-overlapping grids based multidomain spectral collocation method with Chebyshev Gauss–Lobatto points have not been applied on hyperbolic PDEs defined on large time intervals. The hyperbolic PDEs solved herein are linear and nonlinear PDEs that have been reported in the literature. The selected illustrative examples are those which possess exact solutions for simplicity in validating the accuracy of the proposed method. We emphasize that the choice of the numerical examples was purely meant to facilitate demonstration on the usefulness of the method and was not tied to the extent of applicability of the problems themselves. However, a similar approach can be adapted to solve highly important hyperbolic PDEs that emerge in real life. The extent of the discussion of multidomain approach in this study is limited to non-overlapping subintervals only and is implemented on the time variable. For comparative purposes, the hyperbolic PDEs are also solved using the single domain approach namely, the bivariate spectral collocation method.

The prototypical examples of hyperbolic PDEs that are solved in this article include; the damped wave equation, the telegraph equation, the modified Liouville equation, and the 1-D Phi-four equation. The first two are linear, and the latter are nonlinear. The solution of damped wave equation describes the propagation of disturbances out from the region at a fixed speed in one spatial direction [21,22]. This wave equation fits its application in areas of elasticity, plasma physics, quantum mechanics, and general relativity. The telegraph equation originates from Maxwell's equation and it describes the voltage and current on an electrical transmission line with distance and time [23]. On the other hand, the modified Liouville equation plays an important role in biology and mathematical physics where it describes the time evolution of the phase

space distribution function [24,25]. Finally, the 1-D Phi-four equation finds application in many areas of physics such as plasma physics, fluid dynamics, solid state physics, and quantum field theory [26].

In the next section, we describe the multidomain bivariate spectral collocation method for hyperbolic PDEs. We note that hyperbolic PDEs differ from evolutionary parabolic PDEs in terms of constructing the numerical scheme due to the presence of the time derivative initial condition. This is dealt with in this section. In Section 3, we present new error bound theorems and proofs for bivariate polynomial interpolation in single and multiple domains using Gauss–Lobatto nodes. In Section 4, we give the examples that have been chosen for numerical experimentation. In Section 5, we present numerical results for the four example problems. Many practical issues like computational time and conditioning of the matrices are discussed here. A detailed comparison of multi-domain and single domain approaches is also included. Finally, concluding remarks is given in Section 6, followed by Acknowledgments.

2. Method of solution

In this section, we describe an algorithm for the multidomain spectral collocation method and apply it to solve second order partial differential equations of hyperbolic type. To do this, we consider a linear hyperbolic PDE which takes the form;

$$\frac{\partial^2 u}{\partial t^2} + \alpha \frac{\partial u}{\partial t} = \beta^2 \frac{\partial^2 u}{\partial x^2} + \phi(x,t)\frac{\partial u}{\partial x} + \psi(x,t)u + \gamma(x,t), \tag{1}$$

and is defined on the domain $[x,t] \in [a,b] \times [0,T]$, where α, β^2 are known constants and $\phi(x,t)$, $\psi(x,t)$, $\gamma(x,t)$ are known functions. Towards the end of this section, the scheme is extended to nonlinear hyperbolic PDEs. Eq. (1) is to be solved subject to the boundary conditions

$$u(a,t) = g_a(t), \quad u(b,t) = g_b(t), \tag{2}$$

and initial conditions

$$u(x,0) = f(x), \quad u_t(x,0) = h(x), \tag{3}$$

where $g_a(t)$, $g_b(t)$, $f(x)$, and $h(x)$ are known functions. We remark that for the purpose of this paper, the multi-domain approach is implemented on the time-t variable only. Below we give an illustration that demonstrates the implementation of the multidomain approach on the solution of Eq. (1);

Let $t \in \Gamma$ where $\Gamma \in [0,T]$. The domain Γ is decomposed into p non-overlapping intervals as

$$\Gamma_k = [t_{k-1}, t_k], \quad t_{k-1} < t_k, \quad t_0 = 0, \quad t_p = T, \quad k = 1, 2, \ldots, p.$$

The variable $t \in [t_{k-1}, t_k]$ in the kth subdomain is first transformed into $\hat{t} \in [-1,1]$ using the linear transformation

$$t = \frac{1}{2}(t_k - t_{k-1})\,\hat{t} + \frac{1}{2}(t_k + t_{k-1}), \tag{4}$$

before the spectral collocation is applied. Similarly, the spatial domain $x \in [a,b]$ is transformed into $\hat{x} \in [-1,1]$ by applying the linear map

$$x = \frac{1}{2}(b-a)\,\hat{x} + \frac{1}{2}(b+a). \tag{5}$$

The collocation nodes are taken to be the symmetrically distributed Gauss–Lobatto grid points defined in [27] on the interval $[-1,1]$ by

$$\{\hat{x}_i\}_{i=0}^{N_x} = \cos\left(\frac{i\pi}{N_x}\right), \quad \{\hat{t}_j\}_{j=0}^{N_t} = \cos\left(\frac{j\pi}{N_t}\right). \tag{6}$$

The solution at different sub-domains is made distinguishable by adopting the notation $\overset{(k)}{u}(x,t)$, $k = 1, 2, \ldots, p$, to denote solution at the kth subinterval in t. The equation is solved independently at each subinterval. Thus we must solve

$$\frac{\partial^2 \overset{(k)}{u}}{\partial t^2} + \alpha \frac{\partial \overset{(k)}{u}}{\partial t} = \beta^2 \frac{\partial^2 \overset{(k)}{u}}{\partial x^2} + \overset{(k)}{\phi}(x,t)\frac{\partial \overset{(k)}{u}}{\partial x} + \overset{(k)}{\psi}(x,t)\overset{(k)}{u} + \overset{(k)}{\gamma}(x,t), \tag{7}$$

subject to the boundary conditions

$$\overset{(k)}{u}(a,t) = g_a(t), \quad \overset{(k)}{u}(b,t) = g_b(t), \tag{8}$$

and initial conditions

$$\overset{(1)}{u}(x,0) = f(x), \quad \overset{(1)}{u}_t(x,0) = h(x), \tag{9}$$

$$\overset{(k)}{u}(x,t_{k-1}) = \overset{(k-1)}{u}(x,t_{k-1}), \quad \overset{(k)}{u}_t(x,t_{k-1}) = \overset{(k-1)}{u}_t(x,t_{k-1}), \text{ for } k = 2, 3, \ldots, p. \tag{10}$$

We note that the initial conditions for $k = 2, 3, \ldots, p$, Eq. (10) are based on the continuity condition whereby the solution at the right hand boundary of the $(k-1)$th subinterval is taken to be the initial condition when computing the solution in the kth subinterval. The unknown solution, $\overset{(k)}{u}(x,t)$ of Eq. (7) is approximated using a bivariate Lagrange interpolating polynomial defined as

$$\overset{(k)}{U}(x,t) = \sum_{p=0}^{N_x}\sum_{q=0}^{N_t} \overset{(k)}{U}(x_p,t_q)L_p(x)L_q(t), \tag{11}$$

where the functions $L_p(x)$ are the Lagrange cardinal polynomial

$$L_p(x) = \prod_{\substack{p=0\\ p\neq i}}^{N_x} \frac{x-x_p}{x_i-x_p}, \text{ with } l_p(x_i) = \delta_{pi} = \begin{cases} 1 & \text{if } p = i,\\ 0 & \text{if } p \neq i. \end{cases} \tag{12}$$

The functions $L_q(t)$ are defined in a similar manner [28]. The first derivative with respect to x is approximated at the Gauss–Lobatto points (x_i, t_j), for $j = 0, 1, 2, \ldots, N_t$, as

$$\begin{aligned} \left.\frac{\partial \overset{(k)}{U}}{\partial x}\right|_{x=x_i,t=t_j} &= \sum_{p=0}^{N_x}\sum_{q=0}^{N_t} \overset{(k)}{U}(x_p,t_q)L_q(t_j)\left.\frac{dL_p(x)}{dx}\right|_{x=x_i}\\ &= \sum_{p=0}^{N_x} D_{i,p}^{(1,0)}\overset{(k)}{U}(x_p,t_j), \quad i = 0,1,2,\ldots,N_x,\\ &= \mathbf{D}^{(1,0)}\overset{(k)}{\mathbf{U}}_j, \text{ at } t = t_j, \end{aligned} \tag{13}$$

where $\mathbf{D}^{(1,0)}$ is obtained from scaling the standard first derivative Chebyshev differentiation matrix of size $(N_x+1)\times(N_x+1)$ as defined in [27] by multiplying it with $\frac{2}{b-a}$. The vector $\overset{(k)}{\mathbf{U}}_j$ is defined as

$$\overset{(k)}{\mathbf{U}}_j = \left[u(x_0,t_j), u(x_1,t_j), u(x_2,t_j,), \ldots, u(x_{N_x},t_j)\right]^T, \tag{14}$$

where T denotes matrix transpose. The second order partial derivative with respect to x is defined as

$$\left.\frac{\partial^2 \overset{(k)}{U}}{\partial x^2}\right|_{x=x_i,t=t_j} = \mathbf{D}^{(2,0)}\overset{(k)}{\mathbf{U}}_j. \tag{15}$$

The derivatives with respect to t are computed at (x_i, t_j), for $i = 0, 1, 2, \ldots, N_x$, as

$$\begin{aligned} \left.\frac{\partial \overset{(k)}{U}}{\partial t}\right|_{x=x_i,t=t_j} &= \sum_{p=0}^{N_x}\sum_{q=0}^{N_t} \overset{(k)}{U}(x_p,t_q)L_p(x_i)\left.\frac{dL_q(t)}{dt}\right|_{t=t_j}\\ &= \sum_{q=0}^{N_t} D_{j,q}^{(0,1)}\overset{(k)}{U}(x_i,t_q), \quad j = 0,1,2,\ldots,N_t,\\ &= \sum_{q=0}^{N_t} D_{j,q}^{(0,1)}\overset{(k)}{\mathbf{U}}_q, \end{aligned} \tag{16}$$

where $D_{j,q}^{(0,1)}$, $j, q = 0, 1, 2, \ldots, N_t$, is obtained from scaling the standard first derivative Chebyshev differentiation matrix of size $(N_t+1)\times(N_t+1)$ by multiplying it with $\frac{2}{t_k - t_{k-1}}$. Finally, we obtain the second derivative with respect to t as

$$\left.\frac{\partial^2 \overset{(k)}{U}}{\partial t^2}\right|_{x=x_i,t=t_j} = \sum_{q=0}^{N_t} D_{j,q}^{(0,2)}\overset{(k)}{\mathbf{U}}_q. \tag{17}$$

We remark that the temporal differentiation matrices $D_{j,q}^{(0,1)}$ depends on k if the subdomains are not assumed to be of the same length. Substituting the discrete derivatives in the differential equation (7) we obtain

$$\left[\beta^2\mathbf{D}^{(2,0)} + \overset{(k)}{\phi}(\mathbf{x},t_j)\mathbf{D}^{(1,0)} + \overset{(k)}{\psi}(\mathbf{x},t_j)\right]\overset{(k)}{\mathbf{U}}_j - \sum_{q=0}^{N_t}\left[D_{j,q}^{(0,2)} + \alpha D_{j,q}^{(0,1)}\right]\overset{(k)}{\mathbf{U}}_q = -\overset{(k)}{\gamma}(\mathbf{x},t_j). \tag{18}$$

Using the initial condition $\overset{(1)}{u}(x,0)=f(x)$ and noting that the grid points are indexed from right to left of the subinterval in time such that $t=0$ correspond to the grid point t_{N_t}, at the collocation points in the first subinterval in time we have $\overset{(1)}{U}(\mathbf{x},t_{N_t})=f(\mathbf{x})=\overset{(1)}{\mathbf{U}}_{N_t}$ or $\overset{(k)}{U}(\mathbf{x},t_{k-1})=\overset{(k-1)}{U}(\mathbf{x},t_{k-1})=\overset{(k)}{\mathbf{U}}_{N_t}$, for $k=2,3,\ldots,p$. Notation wise, $\mathbf{x}$ denotes the vector of grid points in the entire domain in x. Eq. (18) can be written as

$$\begin{aligned}&\left[\beta^2\mathbf{D}^{(2,0)}+\phi(\mathbf{x},t_j)\mathbf{D}^{(1,0)}+\psi(\mathbf{x},t_j)\right]\overset{(k)}{\mathbf{U}}_j\\&-\sum_{q=0}^{N_t-1}\left[D_{j,q}^{(0,2)}+\alpha D_{j,q}^{(0,1)}\right]\overset{(k)}{\mathbf{U}}_q\qquad=-\overset{(k)}{\gamma}(\mathbf{x},t_j)+\left[D_{j,N_t}^{(0,2)}+\alpha D_{j,N_t}^{(0,1)}\right]\overset{(k)}{\mathbf{U}}_{N_t}.\end{aligned}\tag{19}$$

Expansion of Eq. (19) is a $(N_x+1)(N_t)\times(N_x+1)(N_t)$ matrix system given by

$$\begin{bmatrix}\mathbf{A}_{0,0}&\mathbf{A}_{0,1}&\mathbf{A}_{0,2}&\ldots&\mathbf{A}_{0,N_t-2}&\mathbf{A}_{0,N_t-1}\\\mathbf{A}_{1,0}&\mathbf{A}_{1,1}&\mathbf{A}_{1,2}&\ldots&\mathbf{A}_{1,N_t-2}&\mathbf{A}_{0,N_t-1}\\\mathbf{A}_{2,0}&\mathbf{A}_{2,1}&\mathbf{A}_{2,2}&\ldots&\mathbf{A}_{2,N_t-2}&\mathbf{A}_{0,N_t-1}\\\vdots&\vdots&\vdots&\ddots&\vdots&\vdots\\\mathbf{A}_{N_t-2,0}&\mathbf{A}_{N_t-2,1}&\mathbf{A}_{N_t-2,2}&\ldots&\mathbf{A}_{N_t-2,N_t-2}&\mathbf{A}_{N_t-2,N_t-1}\\\mathbf{A}_{N_t-1,0}&\mathbf{A}_{N_t-1,1}&\mathbf{A}_{N_t-1,2}&\ldots&\mathbf{A}_{N_t-1,N_t-2}&\mathbf{A}_{N_t-1,N_t-1}\end{bmatrix}\begin{bmatrix}\overset{(k)}{\mathbf{U}}_0\\\overset{(k)}{\mathbf{U}}_1\\\overset{(k)}{\mathbf{U}}_2\\\vdots\\\overset{(k)}{\mathbf{U}}_{N_t-2}\\\overset{(k)}{\mathbf{U}}_{N_t-1}\end{bmatrix}=\begin{bmatrix}\overset{(k)}{\mathbf{R}}_0\\\overset{(k)}{\mathbf{R}}_1\\\overset{(k)}{\mathbf{R}}_2\\\vdots\\\overset{(k)}{\mathbf{R}}_{N_t-2}\\\overset{(k)}{\mathbf{R}}_{N_t-1}\end{bmatrix},\tag{20}$$

where

$$\begin{aligned}\mathbf{A}_{i,i}&=\beta^2\mathbf{D}^{(2,0)}+\overset{(k)}{\phi}(\mathbf{x},t_i)\mathbf{D}^{(1,0)}+\overset{(k)}{\psi}(\mathbf{x},t_i)-\left[D_{i,i}^{(0,2)}+\alpha D_{i,i}^{(0,1)}\right]\mathbf{I},\ i=0,1,2\ldots,N_t-1,\\\mathbf{A}_{i,j}&=-\left[D_{i,j}^{(0,2)}+\alpha D_{i,j}^{(0,1)}\right]\mathbf{I},\ i\neq j,\ i,j=0,1,2\ldots,N_t-1,\\\overset{(k)}{\mathbf{R}}_i&=-\overset{(k)}{\gamma}(\mathbf{x},t_i)+\left[D_{i,N_t}^{(0,2)}+\alpha D_{i,N_t}^{(0,1)}\right]\overset{(k)}{\mathbf{U}}_{N_t},\ i=0,1,2\ldots,N_t-1,\end{aligned}$$

and $\mathbf{I}$ is an identity matrix of size $(N_x+1)\times(N_x+1)$. Evaluation of the boundary conditions at the collocation points yields;

$$\overset{(k)}{u}(x_{N_x},t_i)=g_a(t_i),\quad\overset{(k)}{u}(x_0,t_i)=g_b(t_i),\quad i=0,1,2,\ldots,N_t-1.\tag{21}$$

The boundary conditions are imposed by replacing the first and last rows of previously defined $\mathbf{A}_{i,i}$, $\mathbf{A}_{i,j}$ and $\mathbf{R}_i^{(k)}$ blocks as

$$\mathbf{A}_{i,i}=\begin{bmatrix}1&0&\ldots&0&0\\&&\mathbf{A}_{i,i}&&\\0&0&\ldots&0&1\end{bmatrix},\quad\mathbf{A}_{i,j}=\begin{bmatrix}0&0&\ldots&0&0\\&&\mathbf{A}_{i,j}&&\\0&0&\ldots&0&0\end{bmatrix},\ i\neq j,\quad\overset{(k)}{\mathbf{R}}_i=\begin{bmatrix}g_b(t_i)\\\hline\overset{(k)}{\mathbf{R}}_i\\\hline g_a(t_i)\end{bmatrix}.\tag{22}$$

The derivative initial condition at $t=0$, in the first subinterval $k=1$ is resolved at collocation points as follows;

$$\overset{(1)}{u}_t(x_j,t_{N_t})=\sum_{q=0}^{N_t}D_{N_t,q}^{(0,1)}\overset{(1)}{\mathbf{U}}_q=h(x_j),\quad j=0,1,2,\ldots,N_x.\tag{23}$$

Using the condition $\overset{(1)}{u}(\mathbf{x},0)=\overset{(1)}{u}(\mathbf{x},t_{N_t})=\overset{(1)}{\mathbf{U}}_{N_t}=f(\mathbf{x})$, and noting that the values $\overset{(k)}{U}(x_0,t_{N_t})$, $\overset{(k)}{U}(x_{N_x},t_{N_t})$ are known from the boundary conditions, Eq. (23) can be written as

$$\sum_{q=0}^{N_t-1}D_{N_t,q}^{(0,1)}\overset{(1)}{\mathbf{U}}_q=h(x_j)-D_{N_t,N_t}^{(0,1)}f(x_j),\quad j=1,2,\ldots,N_x-1.\tag{24}$$

The conditions reflected on Eq. (24) are imposed on the last block of sub-matrices as follows;

$$\mathbf{A}_{N_t-1,0}=\begin{bmatrix}1&0&0&\ldots&0&0\\\hline&&D_{N_t,0}^{(0,1)}\mathbf{I}(1:N_x-1,0:N_x)&&&\\\hline0&0&0&\ldots&0&1\end{bmatrix},\tag{25}$$

$$\mathbf{A}_{N_t-1,i} = \left[\begin{array}{ccccc} 0 & 0 & 0 & \dots & 0 \quad 0 \\ \hline & & D^{(0,1)}_{N_t,i}\mathbf{I}(1:N_x-1,0:N_x) & & \\ \hline 0 & 0 & 0 & \dots & 0 \quad 0 \end{array}\right],\ i = 1, 2, \dots, N_t - 1. \tag{26}$$

Consequently, the right hand side of Eq. (20) is modified as

$$\mathbf{R}_{(N_t-1)} = \left[\begin{array}{c} g_b(t_{N_t-1}) \\ \hline h(x(1:N_x-1)) - D^{(0,1)}_{N_t,N_t} f(x(1:N_x-1)) \\ \hline g_a(t_{N_t-1}) \end{array}\right]. \tag{27}$$

We remark that in Eqs. (25)–(27), we exerted MatLab syntax " :" for representation of matrices. For $k = 2, 3, \dots, p$, Eq. (24) becomes

$$\sum_{q=0}^{N_t-1} D^{(0,1)}_{N_t,q}\overset{(k)}{\mathbf{U}}_q = \sum_{q=0}^{N_t-1} D^{(0,1)}_{0,q}\overset{(k-1)}{\mathbf{U}}_q = \overset{(k-1)}{\mathbf{U}}\mathbf{U}_0 - D^{(0,1)}_{N_t,N_t}\overset{(k-1)}{\mathbf{U}}_0\,. \tag{28}$$

Here $\overset{(k-1)}{\mathbf{U}}\mathbf{U}_0$ denotes the value of the first derivative with respect to t evaluated at the right hand boundary of the $(k-1)$th subinterval which acts as the derivative initial condition in the kth subinterval. We remark that Eqs. (23)–(28) strikes the major difference between hyperbolic and parabolic PDEs in terms of solution procedure. An extension of the numerical scheme to the solution of nonlinear PDEs is illustrated by considering the solution of a second-order nonlinear hyperbolic PDE of the form

$$\frac{\partial^2 u}{\partial t^2} + \alpha\frac{\partial u}{\partial t} = \beta^2\frac{\partial^2 u}{\partial x^2} + \phi(x,t)\frac{\partial u}{\partial x} + F(u) + \gamma(x,t), \tag{29}$$

where F is a nonlinear operator acting on the unknown function $u(x,t)$. Eq. (29) is to be solved subject to the boundary conditions

$$u(a,t) = g_a(t), \quad u(b,t) = g_b(t), \tag{30}$$

and initial conditions

$$u(x,0) = f(x), \quad u_t(x,0) = h(x). \tag{31}$$

To solve Eq. (29), we first linearize it using the quasi-linearization method (QLM) of Bellman and Kalaba [29]. The QLM is based on the Newton–Raphson method and is constructed from the linear terms of Taylor series expansion about an initial approximation to the solution. The QLM assumes that the difference between solutions at two successive iterations denoted by u_r and u_{r+1} is very small. Applying the QLM on Eq. (29) yields

$$\beta^2\frac{\partial^2 u_{r+1}}{\partial x^2} + \phi(x,t)\frac{\partial u_{r+1}}{\partial x} + \omega_r u_{r+1} - \frac{\partial^2 u_{r+1}}{\partial t^2} - \alpha\frac{\partial u_{r+1}}{\partial t} = -\gamma(x,t) + \omega_r u_r - F(u_r), \tag{32}$$

where $\omega = \frac{dF(u)}{du}$ and $r = 1, 2, 3, \dots$ denotes the level of iteration. The QLM scheme Eq. (32) is solved iteratively subject to the boundary conditions

$$u_{r+1}(a,t) = g_a(t), \quad u_{r+1}(b,t) = g_b(t), \tag{33}$$

and initial conditions

$$u_{r+1}(x,0) = f(x), \quad \frac{\partial u_{r+1}}{\partial t}(x,0) = h(x). \tag{34}$$

The matrix system resulting from spectral collocation are

$$\mathbf{A}_{i,i} = \beta^2\mathbf{D}^{(2,0)} + \overset{(k)}{\phi}(\mathbf{x},t_i)\mathbf{D}^{(1,0)} + \overset{(k)}{\boldsymbol{\omega}}_{r,i} - \left[D^{(0,2)}_{i,i} + \alpha D^{(0,1)}_{i,i}\right]\mathbf{I},\ i = 0, 1, 2 \dots, N_t - 1, \tag{35}$$

$$\mathbf{A}_{i,j} = -\left[D^{(0,2)}_{i,j} + \alpha D^{(0,1)}_{i,j}\right]\mathbf{I},\ i \neq j,\ i,j = 0, 1, 2 \dots, N_t - 1, \tag{36}$$

$$\overset{(k)}{\mathbf{R}}_{r,i} = -\overset{(k)}{\gamma}(\mathbf{x},t_i) + \left[D^{(0,2)}_{i,N_t} + \alpha D^{(0,1)}_{i,N_t}\right]\overset{(k)}{\mathbf{U}}_{N_t} + \overset{(k)}{\boldsymbol{\omega}}_{r,i}\overset{(k)}{\mathbf{U}}_{r,i} - F(\overset{(k)}{\mathbf{U}}_{r,i}),\ 0 \le i \le N_t - 1. \tag{37}$$

3. Error bounds in bivariate polynomial interpolation

In this section, we present the error bound theorems that emanate from bivariate polynomial interpolation using Gauss–Lobatto nodes. To the best of our knowledge, the error bound theorems given in this section are new and are been presented for the first time in literature. The error bound theorems given herein acts a theoretical basis of argument as to why

multidomain approaches are most suitable when approximating the solutions of differential equations that are defined over large interval domains. We note that Gauss–Lobatto nodes are relative extremes of the N_xth degree Chebyshev polynomial of the first kind $T_{N_x}(\hat{x})$, $\hat{x} \in [-1, 1]$. There does not exist a well known explicitly defined polynomial whose roots are the GL nodes. However, if we think of interior Gauss–Lobatto nodes as the solution of $T'_{N_x}(\hat{x}) = 0$, where the prime denotes differentiation with respect to $\hat{x}$, the complete set of the Gauss–Lobatto nodes can be said to be the roots of $(N_x + 1)$th degree polynomial;

$$L_{N_x+1}(\hat{x}) = (1 - \hat{x}^2)T'_{N_x}(\hat{x}). \tag{38}$$

Below, we state a theorem that acts as a benchmark for stating subsequent error bound theorems on bivariate polynomial interpolation and devising their proofs;

Theorem 1 ([30]). *Let $u(x, t) \in C^{N_x+N_t+2}([a, b] \times [0, T])$ be sufficiently smooth such that at least the $(N_x + 1)$th partial derivative with respect to x, $(N_t + 1)$th partial derivative with respect to t and $(N_x + N_t + 2)$th mixed partial derivative with respect to x and t exists and are all continuous, then there exists values $\xi_x, \xi'_x \in (a, b)$, and $\xi_t, \xi'_t \in (0, T)$, such that*

$$\begin{aligned} u(x, t) - U(x, t) = &\frac{\partial^{N_x+1}u(\xi_x, t)}{\partial x^{N_x+1}(N_x + 1)!}\prod_{i=0}^{N_x}(x - x_i) + \frac{\partial^{N_t+1}u(x, \xi_t)}{\partial t^{N_t+1}(N_t + 1)!}\prod_{j=0}^{N_t}(t - t_j) \\ &- \frac{\partial^{N_x+N_t+2}u(\xi'_x, \xi'_t)}{\partial x^{N_x+1}\partial t^{N_t+1}(N_x + 1)!(N_t + 1)!}\prod_{i=0}^{N_x}(x - x_i)\prod_{j=0}^{N_t}(t - t_j), \end{aligned} \tag{39}$$

where $U(x, t)$ is a polynomial interpolant of $u(x, t)$ at $\{x_i\}_{i=0}^{N_x}$ grid points in x-variable and $\{t_j\}_{j=0}^{N_t}$ grid points in t-variable.

A similar result was later reproduced by authors in [31]. Taking the absolute value of Eq. (39) we obtain

$$\begin{aligned} E(x, t) \le &\max_{(x,t)\in\Omega}\left|\frac{\partial^{N_x+1}u(\xi_x, t)}{\partial x^{N_x+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x - x_i)\right|}{(N_x + 1)!} + \max_{(x,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x, \xi_t)}{\partial t^{N_t+1}}\right|\frac{\left|\prod_{j=0}^{N_t}(t - t_j)\right|}{(N_t + 1)!} \\ &+ \max_{(x,t)\in\Omega}\left|\frac{\partial^{N_x+N_t+2}u(\xi'_x, \xi'_t)}{\partial x^{N_x+1}\partial t^{N_t+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x - x_i)\right|\left|\prod_{j=0}^{N_t}(t - t_j)\right|}{(N_x + 1)!(N_t + 1)!}. \end{aligned} \tag{40}$$

Here $E(x, t) = |u(x, t) - U(x, t)|$. Since the function $u(x, t)$ is assumed to be smooth on Ω, it follows that its derivatives are bounded and thus $\exists$ constants C_1, C_2 and C_3, such that

$$\max_{(x,t)\in\Omega}\left|\frac{\partial^{N_x+1}u(x, t)}{\partial x^{N_x+1}}\right| \le C_1, \ \max_{(x,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x, t)}{\partial t^{N_t+1}}\right| \le C_2, \ \max_{(x,t)\in\Omega}\left|\frac{\partial^{N_x+N_t+2}u(x, t)}{\partial x^{N_x+1}\partial t^{N_t+1}}\right| \le C_3. \tag{41}$$

3.1. Error bound theorem on a single domain

In this subsection, we state and prove the error bound theorem that govern bivariate polynomial interpolation on a single domain using Gauss–Lobatto nodes.

Theorem 2 (*Error Bound on a Single Domain*). *The resulting error bound when GL grid points $\{x_i\}_{i=0}^{N_x} \in [a, b]$, in x-variable and $\{t_j\}_{j=0}^{N_t} \in [0, T]$, in t-variable are used in bivariate polynomial interpolation is given by*

$$E(x, t) \le C_1\frac{8\left(\frac{b-a}{4}\right)^{N_x+1}}{(N_x + 1)!} + C_2\frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t + 1)!} + C_3\frac{8^2\left(\frac{b-a}{4}\right)^{N_x+1}\left(\frac{T}{4}\right)^{N_t+1}}{(N_x + 1)!(N_t + 1)!}, \tag{42}$$

where C_1, C_2 and C_3, are as defined in Eq. (41).

Proof. To prove the analytic expression for the error bound when Gauss–Lobatto are used in bivariate polynomial interpolation, we first apply the relation stated in [32] to write the $(N_x + 1)$th degree polynomial $L_{N_x+1}(\hat{x})$ at Eq. (38) as

$$L_{N_x+1}(\hat{x}) = (1 - \hat{x}^2)T'_{N_x}(\hat{x}) = -N_x\hat{x}T_{N_x}(\hat{x}) + N_xT_{N_x-1}(\hat{x}). \tag{43}$$

Noting that $|T_{N_x}(\hat{x})| \le 1$ for $\hat{x} \in [-1, 1]$, we have

$$|L_{N_x+1}(\hat{x})| = |-N_x\hat{x}T_{N_x}(\hat{x}) + N_xT_{N_x-1}(\hat{x})| \le |-N_x\hat{x}T_{N_x}(\hat{x})| + |N_xT_{N_x-1}(\hat{x})| \le 2N_x. \tag{44}$$

The leading coefficient of $L_{N_x+1}(\hat{x})$ is $2^{N_x-1}N_x$. The components 2^{N_x-1} and N_x comes from the leading coefficient of $T_{N_x}(\hat{x})$ and the application of N_x-th rule of differentiation on $T_{N_x}(\hat{x})$, respectively. The product factor in the error bound expression given in Eq. (40) can therefore be taken to be the factorized form of monic polynomial $\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}$. We write

$$\prod_{i=0}^{N_x}(\hat{x}-\hat{x}_i)=\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}. \tag{45}$$

Using Eq. (44) we observe that this monic polynomial Eq. (45) is bounded above by

$$\left|\prod_{j=0}^{N_x}(\hat{x}-\hat{x}_i)\right|=\left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right|\leq\frac{2N_x}{2^{N_x-1}N_x}=\frac{4}{2^{N_x}}. \tag{46}$$

Considering a general interval $x\in[a,b]$, we can show that the first product factor in Eq. (40) is bounded above by

$$\begin{aligned}\max_{a\leq x\leq b}\left|\prod_{i=0}^{N_x}(x-x_i)\right|&=\max_{-1\leq\hat{x}\leq1}\left|\prod_{i=0}^{N_x}\frac{(b-a)}{2}(\hat{x}-\hat{x}_i)\right|=\left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\leq\hat{x}\leq1}\left|\prod_{j=0}^{N_x}(\hat{x}-\hat{x}_i)\right|\\&=\left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\leq\hat{x}\leq1}\left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right|\\&\leq\frac{4\left(\frac{b-a}{2}\right)^{N_x+1}}{2^{N_x}}=8\left(\frac{b-a}{4}\right)^{N_x+1}.\end{aligned} \tag{47}$$

In a similar manner, it can be shown that for the second product term

$$\max_{0\leq t\leq T}\left|\prod_{j=0}^{N_t}(t-t_j)\right|=\left(\frac{T}{2}\right)^{N_t+1}\max_{-1\leq\hat{t}\leq1}\left|\frac{L_{N_t+1}(\hat{t})}{2^{N_t-1}N_t}\right|\leq\frac{4\left(\frac{T}{2}\right)^{N_t+1}}{2^{N_t}}=8\left(\frac{T}{4}\right)^{N_t+1}. \tag{48}$$

Using Eqs. (47), (48), and (41) in Eq. (40) completes the proof. □

3.2. Error bound theorem on multidomain

In this subsection, we extend the bivariate interpolation polynomial error bound theorem presented in the previous subsection to obtain its variant on a decomposed domain. In the description, it is assumed that the number of grid points is the same across all subintervals.

Theorem 3 (*Error Bound in the Decomposed Domain*)**.** *The error bound when Gauss–Lobatto grid points* $\{x_i\}_{i=0}^{N_x}\in[a,b]$ *for x-variable and* $\{t_j\}_{j=0}^{N_t}\in[t_{k-1},t_k]$, $k=1,2,\ldots,p$, *for the decomposed domain in t-variable (expanded over the entire interval* $[0,T]$*) are used in bivariate polynomial interpolation is given by*

$$E(x,t)\leq C_1\frac{8\left(\frac{b-a}{4}\right)^{N_x+1}}{(N_x+1)!}+C_2\frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1}+C_3\frac{8^2\left(\frac{b-a}{4}\right)^{N_x+1}\left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1}. \tag{49}$$

Proof. Let define the Gauss–Lobatto grid points in the t- variable at the kth subinterval as

$$\{\tau_j^{(k)}\}_{j=0}^{N_t}=\cos\left(\frac{j\pi}{N_t}\right),\quad t_j^{(k)}=\frac{1}{2}\left(\frac{t_k+t_{k-1}}{N_t}\right)+\frac{1}{2}\left(\frac{t_k-t_{k-1}}{N_t}\right)\tau_j^{(k)}. \tag{50}$$

In the entire domain $[0,T]$, we have that

$$\left|\prod_{j=0}^{N_t}(t-t_j)\right|\leq\frac{4\left(\frac{T}{2}\right)^{N_t+1}}{2^{N_t}}=8\left(\frac{T}{4}\right)^{N_t+1},$$

which implies that in the decomposed domain, at each subinterval, we should have

$$\left|\prod_{j=0}^{N_t}(t-t_j^{(k)})\right|\leq8\left(\frac{T}{4p}\right)^{N_t+1}=8\left(\frac{T}{4}\right)^{N_t+1}\left(\frac{1}{p}\right)^{N_t+1}. \tag{51}$$

We remark that there exists t values $\xi_\mu\in(t_{\mu-1},t_\mu)$, $\mu=1,2,\ldots,p$, for which the values of the (N_t+1)th partial derivatives with respect to t in each subinterval is the absolute maximum. This enables us to break the second term $C_2\frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}$ in Eq. (42)

into different components that are not necessarily equal in the decomposed domain as

$$\left\{C_2^{(k)}\frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1}\right\}_{k=1}^{p}, \tag{52}$$

where

$$\max_{(x,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x,t)}{\partial t^{N_t+1}}\right| = \left|\frac{\partial^{N_t+1}u(x,\xi_k)}{\partial t^{N_t+1}}\right| \le C_2^{(k)}, \quad t\in[t_{k-1},t_k]. \tag{53}$$

We define

$$\|\hat{C}_2\|_\infty \equiv \max\{C_2^{(1)}, C_2^{(2)}, \ldots, C_2^{(p)}\}, \tag{54}$$

to denote the maximum absolute value of (N_t+1)th partial derivatives of u with respect to t in $[0,T]$. Clearly, $\|\hat{C}_2\|_\infty = C_2$. To expand the error bound over the entire domain, we shall take the largest possible error across all sub-domains which is

$$C_2\frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1}. \tag{55}$$

In a similar manner, the third term $C_3\frac{8^2\left(\frac{b-a}{4}\right)^{N_x+1}\left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_t+1)!}$ in Eq. (42) becomes

$$C_3\frac{8^2\left(\frac{b-a}{4}\right)^{N_x+1}\left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1}, \tag{56}$$

in the case of multi-domain approach. Using Eqs. (55) and (56) in Eq. (42) completes the proof. □

Comparing Eqs. (42) and (49) it is clear that errors associated with bivariate polynomial interpolation are smaller when using multi-domain approach than the single domain approach as the second and third terms in Eq. (42) have been multiplied by $\left(\frac{1}{p}\right)^{N_t+1} \ll 1$ for large p in Eq. (49). The assumption that is made here is that the spatial interval is small and thus the first term in the error bound expression has less influence on the size of interpolation error. This multidomain approach that is described here enhances significant reduction in the size of interpolations errors if such errors are mainly contributed by time, (t) variable. The use of the multi-domain approach allows for a reduction in the number of grids points, N_t on the time variable at each subinterval. Consequently, small-sized matrices that are well-conditioned arises and the approximation errors are minimized. This is especially the case if the enormous errors are caused by large valued higher order time derivatives of the function $u(x,t)$ within the domain of approximation. Reducing the size of interpolation error ensures that more accurate numerical results are arrived at when such interpolating polynomials are used to approximate solutions of differential equations. Polynomial interpolation error bound theorems using the close variant of Gauss–Lobatto nodes namely; the equispaced nodes and Chebyshev nodes are well known in the literature and their proofs can be found in any elementary book in numerical analysis. The equispaced grids points have poor interpolating features as they result in Runge phenomena [33]. The Chebyshev grids points which upon comparison with the theorems presented in this work yields a slightly smaller interpolating error (almost half that of Gauss–Lobatto nodes) are not used in spectral based collocation methods. The reason behind the preferable choice of GL nodes to Chebyshev nodes as collocation nodes is their convenience in generating differentiation matrices. The Gauss–Lobatto nodes also include the boundary nodes of the computational domain -1 and 1 which is advantageous when treating the boundary conditions of the problem in question. This is not the case with Chebyshev nodes.

4. Numerical experimentation

In this section, the applicability of the multidomain bivariate spectral collocation method to solutions of hyperbolic partial differential equations is illustrated by considering the solution of well known linear and non-linear PDEs that have been reported in the literature;

Example 1. Consider the linear damped wave equation

$$\frac{\partial^2 u}{\partial t^2} + 2c\frac{\partial u}{\partial t} = \beta^2\frac{\partial^2 u}{\partial x^2}, \tag{57}$$

where c is a small positive constant. The term $2c\frac{\partial u}{\partial t}$ represents a damping force proportional to the velocity $\frac{\partial u}{\partial t}$. For simplicity, we will assume that the length of the string is $L=\pi$, and the constant $\beta^2 = c = 1$. These substitution yields

$$\frac{\partial^2 u}{\partial t^2} + 2\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}. \tag{58}$$

Eq. (58) is solved subject to boundary conditions

$$u(0,t)=0,\quad u(\pi,t)=0,\quad \text{for all } t>0, \tag{59}$$

and initial conditions

$$u(x,0)=\sin x,\quad u_t(x,0)=-\sin x. \tag{60}$$

The exact solution is given in [22] as

$$u(x,t)=e^{-t}\sin x. \tag{61}$$

Example 2. Consider the linear Telegraph equation

$$\frac{\partial^2 u}{\partial t^2}+\upsilon\frac{\partial u}{\partial t}=\beta^2\frac{\partial^2 u}{\partial x^2}-\kappa u. \tag{62}$$

Eq. (62) describes the voltage $u(x,t)$ inside a piece of telegraph/transmission wire, whose electrical properties per unit length are resistance R, inductance I, capacitance C and conductance of leakage current G, where $\beta^2=\frac{1}{IC}$, $\upsilon=\frac{G}{C}+\frac{R}{L}$, and $\kappa=\frac{GR}{CL}$. Setting all electrical properties quantities to unity Eq. (62) reduces to

$$\frac{\partial^2 u}{\partial t^2}+2\frac{\partial u}{\partial t}=\frac{\partial^2 u}{\partial x^2}-u. \tag{63}$$

Eq. (63) is solved subject to boundary conditions

$$u(0,t)=e^{-2t},\quad u(1,t)=e^{1-2t},\quad \text{for all } t>0, \tag{64}$$

and initial conditions

$$u(x,0)=e^{x},\quad u_t(x,0)=-2e^{x}. \tag{65}$$

The exact solution is given in [23] as

$$u(x,t)=e^{x-2t}. \tag{66}$$

Example 3. Consider the nonlinear modified Liouville equation

$$\frac{\partial^2 u}{\partial t^2}=\beta^2\frac{\partial^2 u}{\partial x^2}+\kappa e^{\lambda u}. \tag{67}$$

Setting $\beta^2=\kappa=\lambda=1$ Eq. (67) reduces to

$$\frac{\partial^2 u}{\partial t^2}=\frac{\partial^2 u}{\partial x^2}+e^{u}. \tag{68}$$

Eq. (68) is to be solved subject to boundary conditions

$$u(1,t)=\ln\left[\frac{6}{(1+2t)^2}\right],\quad u(2,t)=\ln\left[\frac{6}{(2+2t)^2}\right],\quad \text{for all } t>0, \tag{69}$$

with the initial conditions

$$u(x,0)=\ln\left[\frac{6}{x^2}\right],\quad u_t(x,0)=-\frac{4}{x}. \tag{70}$$

The exact solution is given in [24] as

$$u(x,t)=\ln\left[\frac{6}{(x+2t)^2}\right]. \tag{71}$$

Example 4. Consider the nonlinear 1- dimensional phi-four equation

$$\frac{\partial^2 u}{\partial t^2}=\beta\frac{\partial^2 u}{\partial x^2}-\delta u-\lambda u^3. \tag{72}$$

Setting $\beta=\delta=\lambda=-1$ Eq. (72) reduces to

$$\frac{\partial^2 u}{\partial t^2}=-\frac{\partial^2 u}{\partial x^2}+u+u^3. \tag{73}$$

Eq. (73) is to be solved subject to boundary conditions

$$u(0,t)=\tan\left(\frac{t}{4}\right),\quad u\left(\frac{\pi}{2},t\right)=\tan\left(\frac{\frac{\pi}{2}+t}{4}\right),\quad \text{for all } t>0, \tag{74}$$

Table 1
Relative error values obtained when Example Example 1 is solved using $N_x = 20,\ N_t = 100,\ p = 1$.

		t		
x	1.0	2.0	5.0	10.0
0.0193	4.66290e−008	7.44967e−008	4.09623e−007	8.70789e−007
0.4601	1.06465e−009	1.30864e−009	1.58031e−009	5.13471e−008
1.3251	3.23212e−010	2.69150e−009	3.21242e−009	1.75804e−008
2.2839	5.15519e−009	2.53688e−009	7.62498e−009	2.60394e−008
2.9704	1.95880e−008	1.54940e−008	4.31666e−008	7.89615e−008
CPU time (s)	1.423548	Cond number	7.0462e+0011	

with the initial conditions

$$u(x,0) = \tan\left(\frac{x}{4}\right), \quad u_t(x,0) = \frac{1}{4}\sec^2\left(\frac{x}{4}\right). \tag{75}$$

The exact solution is given in [26] as

$$u(x,t) = \tan\left(\frac{x+t}{4}\right). \tag{76}$$

Example 5. Determine the interpolation error in approximating the function

$$u(x,t) = e^{x+t},\ x \in [-1,1],\ t \in [-2,2]. \tag{77}$$

We note that Examples 1–4 are used to demonstrate the applicability of multidomain spectral collocation method in solving hyperbolic PDEs whereas Example 5 is used to test the error bound theorems.

5. Results and discussions

In this section, numerical results in terms absolute error and relative error values obtained at selected values of x and t when solving Examples 1–4 are presented. The bivariate spectral collocation method (single domain approach) and the multi-domain bivariate spectral collocation method have both been applied in solving each Example. The computation time that is taken to produce results and the condition number of the coefficient matrix resulting from collocation is also presented. The CPU time recorded here is the time taken to execute instructions contained in the algorithm right away from compilation to the instant when results are realized. The condition number of the matrix $\mathbf{A}$ is evaluated as $\|\mathbf{A}\|\|\mathbf{A}^{-1}\|$ [34] and is invoked using the MatLab command *cond*($\mathbf{A}$). In the case of the multidomain approach, the largest condition number across all the subintervals is presented. The absolute and relative errors are evaluated as

$$Abs_{i,j} = \left|u_e(x_i,t_j) - U_a(x_i,t_j)\right|, \quad \text{and} \quad Rel_{i,j} = \left|\frac{Abs_{i,j}}{u_e(x_i,t_j)}\right|, \ \text{respectively}, \tag{78}$$

where $u_e(x_i,t_j)$ is the exact solution and $U_a(x_i,t_j)$ is the approximate solution at the collocation points (x_i,t_j). The results for each example have been presented in two separate tables. The first table is a representation of the results obtained using the single domain approach whereas the second one depicts those obtained from the multi-domain approach in the quest to compare their performance. We remark that the bivariate spectral collocation method which is the single domain approach is invoked by setting the number of subintervals $p = 1$. The results in Tables 1–8 were obtained using MatLab software where as those in Table 9 were computed on Mathematica platform.

The results obtained from approximating the solution of Eq. (58) are presented in Tables 1 and 2. Table 1 shows the results obtained when bivariate spectral collocation method is applied. Relative errors of order 10^{-7} are obtained. Table 2 displays similar results when the multi-domain bivariate spectral collocation is applied and smaller absolute errors than those in case of the single domain approach are obtained. The relative errors are of order 10^{-10}. In addition, Tables 1 and 2 indicate that a shorter computation time is taken to generate results and the condition number of the coefficient matrix is smaller when the multidomain approach is adopted. Small condition number is responsible for improved accuracy in the multidomain approach.

The results obtained from approximating the solution of Eq. (63) are presented in Tables 3 and 4. Table 3 shows the results obtained when a single domain approach is applied. The relative errors obtained here are of order 10^{-5}. On the other hand, Table 4 shows similar results that are obtained when the multi-domain approach is applied in the method of solution. The relative errors (of order 10^{-8}) are smaller in the multi-domain approach than those obtained in case of the single domain approach. A shorter computation time and a smaller condition number of the coefficient matrix is evident in the case of the multi-domain approach, Table 4, than that recorded when the single domain approach is adopted, Table 3.

The results obtained from approximating the solution of Eq. (68) are presented in Tables 5 and 6. The results given herein are those obtained after the 5th iteration. Table 5 shows the results obtained when a single domain approach is used whereas

Table 2
Relative error values obtained when Example Example 1 is solved using $N_x = 20,\ N_t = 10,\ p = 10$.

		t		
x	1.0	2.0	5.0	10.0
0.0193	5.33321e−012	3.49871e−011	1.95661e−010	7.62712e−010
0.4601	1.74861e−011	5.05352e−011	2.44538e−010	8.89471e−010
1.3251	1.62381e−011	4.84196e−011	2.42403e−010	8.89054e−010
2.2839	1.40134e−011	4.69972e−011	2.40025e−010	8.88724e−010
2.9704	5.11146e−012	3.92272e−011	2.31320e−010	8.73026e−010
CPU time (s)	0.114274	Cond number	8.7192e+006	

Table 3
Relative error values obtained when Example Example 2 is solved using $N_x = 20,\ N_t = 100,\ p = 1$.

		t		
x	1.0	2.0	5.0	10.0
0.0062	2.18419e−008	8.27964e−008	2.53078e−007	2.34678e−006
0.1464	5.15727e−008	1.80808e−009	7.05926e−007	1.24505e−005
0.4218	1.11300e−008	5.67612e−008	1.74194e−007	7.61562e−006
0.7270	2.86742e−008	3.67432e−008	2.96557e−007	3.60689e−005
0.9455	2.96515e−008	1.35708e−007	2.21656e−007	8.10334e−005
CPU time (s)	2.846143	Cond number	6.4327e+011	

Table 4
Relative error values obtained when Example Example 2 is solved using $N_x = 20,\ N_t = 10,\ p = 10$.

		t		
x	1.0	2.0	5.0	10.0
0.0062	3.95601e−012	2.02862e−012	3.39103e−010	6.39796e−008
0.1464	1.50557e−012	3.41663e−012	3.49983e−010	6.77700e−008
0.4218	4.88940e−012	7.25118e−011	5.85170e−010	9.00861e−008
0.7270	3.79695e−012	9.79213e−012	4.54850e−010	9.87302e−008
0.9455	1.87760e−012	2.69463e−011	7.11001e−011	9.78759e−008
CPU time (s)	0.146381	Cond number	8.4572e+006	

Table 5
Absolute error values obtained when Example Example 3 is solved using $N_x = 20,\ N_t = 50,\ p = 1$, It=5.

		t		
x	0.2	0.4	0.6	0.8
1.0062	3.47944e−009	3.77384e−009	8.2309e−008	2.18613e−009
1.1464	7.45887e−010	3.02654e−009	3.06068e−009	2.18613e−009
1.4218	3.52577e−010	3.89464e−009	6.36734e−010	4.19664e−010
1.7270	7.02645e−011	9.92337e−011	1.09369e−009	1.09697e−010
1.9455	1.78821e−009	2.73041e−009	1.88447e−009	5.45717e−010
CPU time (s)	1.519801	Cond number	7.8367e+009	

Table 6 shows similar results that are obtained when the multi-domain approach is applied in the method of solution. The absolute errors of order 10^{-11} are achieved in the case of the multi-domain approach and are much smaller than those obtained in the case of the single domain approach (order 10^{-8}). The trend on the amount of computation time and the value of the condition number of the coefficient matrix is analogous to those presented in preceding examples.

The results obtained from approximating the solution of Eq. (73) are presented in Tables 7 and 8. Table 7 displays the results obtained when bivariate spectral collocation method is applied in solving Eq. (73). The absolute errors of order 10^{-5} are obtained. Numerical results that are obtained when the multi-domain approach is applied are shown in Table 8. The absolute errors of order 10^{-10} are obtained in the case of the multi-domain approach. In regards to the amount of computation time and the condition number of the coefficient matrix, a trend analogous to that presented in preceding tables is evident across the two approaches.

In Table 9, the results obtained from piece-wise polynomial interpolation on multiple domains have been bolded to distinguish them from those obtained when interpolation performed on a single domain. The column label, Bound, denotes the maximum possible theoretical value of interpolation error evaluated from the error bound theorems. The t domain was subdivided into 2 subintervals of equal length. The number of grid points was maintained the same over the entire domain for comparison purposes. The Table shows that the size of interpolation errors decrease as the number of grid points increase.

Table 6
Absolute error values obtained when Example Example 3 is solved using $N_x = 20,\ N_t = 5,\ p = 10$, It=5.

		t		
x	0.2	0.4	0.6	0.8
1.0062	6.36269e−011	9.95849e−011	5.44590e−011	1.95023e−011
1.1464	1.69966e−011	2.91311e−011	2.86388e−011	1.75647e−011
1.4218	1.97931e−012	5.97371e−011	2.04829e−011	4.18210e−012
1.7270	8.39218e−013	1.21388e−011	9.29423e−013	4.77738e−011
1.9455	1.54289e−011	3.12200e−011	4.69424e−012	8.03992e−011
CPU time (s)	0.105744	Cond number	2.8509e+006	

Table 7
Absolute error values obtained when Example Example 4 is solved using $N_x = 20,\ N_t = 50,\ p = 1$, It=5.

		t		
x	0.02	0.04	0.06	0.08
0.0097	8.79971e−009	4.08907e−008	1.05963e−008	3.71353e−006
0.2300	2.50317e−010	2.50693e−009	2.53192e−008	2.52560e−007
0.6625	2.66273e−010	2.80803e−009	2.75705e−008	2.62539e−007
1.1420	8.84794e−010	1.01925e−008	1.02061e−007	9.73155e−007
1.4852	2.47955e−008	3.31860e−007	3.49922e−006	3.39059e−005
CPU time (s)	1.539690	Cond number	2.3494e+015	

Table 8
Absolute error values obtained when Example Example 4 is solved using $N_x = 20,\ N_t = 5,\ p = 10$, It=5.

		t		
x	0.02	0.04	0.06	0.08
0.0097	3.79300e−012	3.65525e−011	3.78129e−010	3.85786e−009
0.2300	6.13676e−014	4.73760e−013	4.12961e−012	3.94241e−011
0.6625	3.66929e−014	3.03868e−013	2.95070e−012	2.78306e−011
1.1420	1.79856e−013	1.17273e−012	1.06025e−011	9.83894e−011
1.4852	2.71216e−012	3.41522e−011	3.52029e−010	3.38485e−009
CPU time (s)	0.350592	Cond number	1.5534e+007	

Table 9
Numerical values of interpolation errors obtained when interpolating Eq.(77) in a single and multiple domains.

N_x	N_t	P	CPU Time(s)	Bound	Gauss–Lobatto
10	10	1	1.5760	1.4874×10^{-6}	2.8320×10^{-7}
10	**5**	**2**	**1.1437**	$\mathbf{8.2877 \times 10^{-5}}$	$\mathbf{3.8721 \times 10^{-5}}$
20	20	1	6.7860	2.0460×10^{-8}	6.4615×10^{-14}
20	**10**	**2**	**4.7495**	$\mathbf{1.7551 \times 10^{-8}}$	$\mathbf{1.8046 \times 10^{-13}}$
40	40	1	38.0700	6.3882×10^{-9}	1.7955×10^{-13}
40	**20**	**2**	**15.3842**	$\mathbf{5.3520 \times 10^{-10}}$	$\mathbf{2.8411 \times 10^{-16}}$

It also demonstrates that numerical values of absolute errors are always smaller than the theoretical error bound values verifying that the function considered here obeys the bivariate polynomial interpolation error bound theorems presented in Section 3. We observe that performing interpolation over many smaller domains recorded significantly shorter CPU time than the single domain. Further, an improved accuracy is registered when 40 grid points are used in t on multiple domains than the single domain which registers a drop in accuracy.

To further discuss the results, the numerical solution of Examples 1–4 is compared with the exact solution through figures. The numerical solution depicted here is that obtained using the multidomain approach with $N_x = 20,\ N_t = 5,\ p = 10$. Figs. 1–4 shows that the numerical solution is in close agreement with the exact solution confirming the accuracy of the proposed approach. Fig. 5 compares the absolute errors obtained when approximating the polynomial given in Example 5 by an interpolating polynomial using $N_x = 40,\ N_t = 40$ for the single and multidomain approach. The figure confirms that the multidomain approach records smaller interpolation errors as compared to single domain approach.

6. Conclusion

We have successfully proposed the multi-domain bivariate spectral collocation method for solving hyperbolic PDEs. Secondly, we have demonstrated that this method yields very accurate approximate solution when it is applied in solving hyperbolic PDEs that are defined over larger time domains and that results are generated over little time. The reduction in

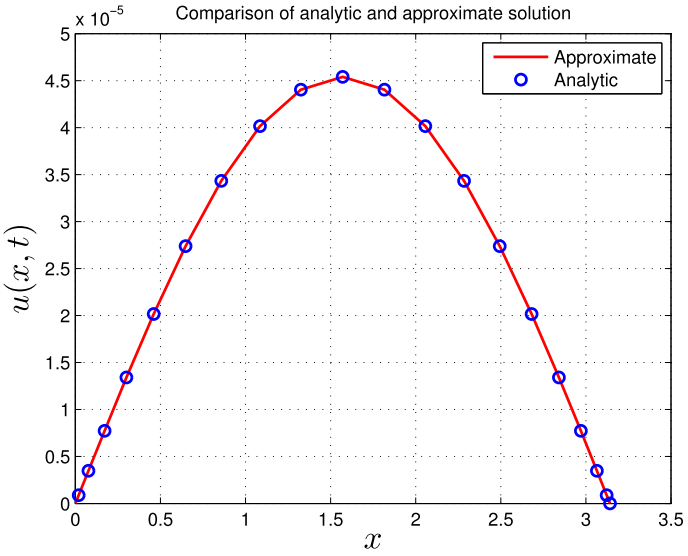


Fig. 1. Comparison of exact and numerical solution for Example 1 evaluated at $t = 10$.

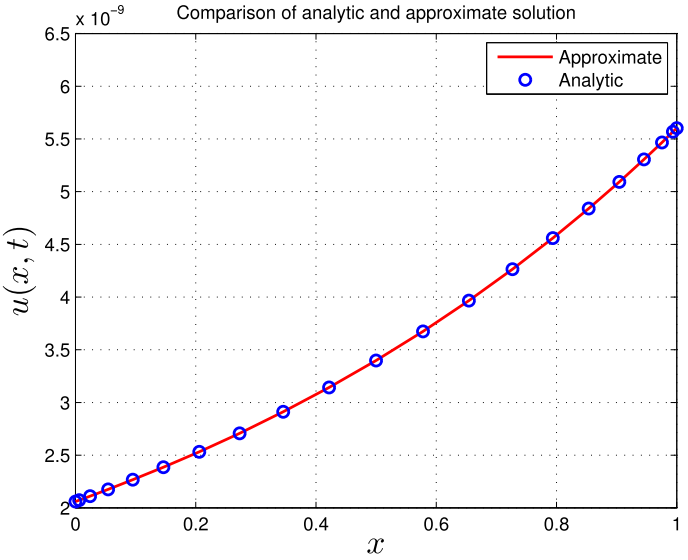


Fig. 2. Comparison of exact and numerical solution for Example 2 evaluated at $t = 10$.

the size of the computation domain and consequently the use of fewer grid points at each sub-domain reduces the size of the coefficient matrix. The small-sized matrices are easy to invert, and this translates to short computation time that is required to generate results. The more accurate results are attributed to small condition numbers of the coefficient matrices obtained when using the multi-domain approach. The performance of the method is reliable, and it gives a more general scheme that is easily adaptable. The multi-domain bivariate spectral collocation method can thus be extended to solve many other problems in real life that are modeled by hyperbolic PDEs. We have also successfully proved and tested new error bound theorems for bivariate Lagrange interpolating polynomial which is used to approximate the solution of the hyperbolic PDEs.

Acknowledgments

The authors are thankful to the University of KwaZulu-Natal for providing valuable research resources. The first author also thanks, Taita Taveta University, Kenya.

Declaration of interest

None.

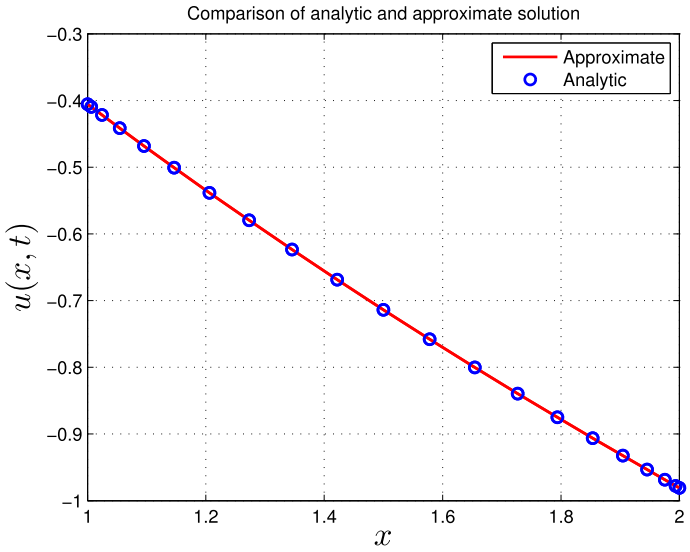


Fig. 3. Comparison of exact and numerical solution for Example 3 evaluated at $t = 1$.

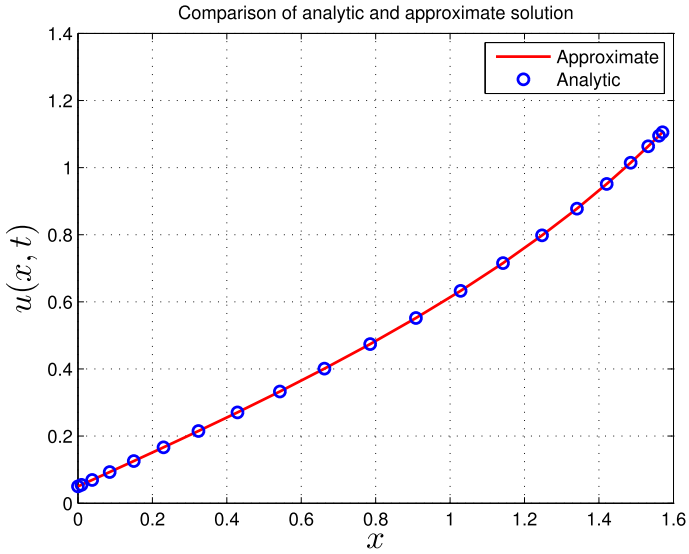


Fig. 4. Comparison of exact and numerical solution for Example 4 evaluated at $t = 0.1$.

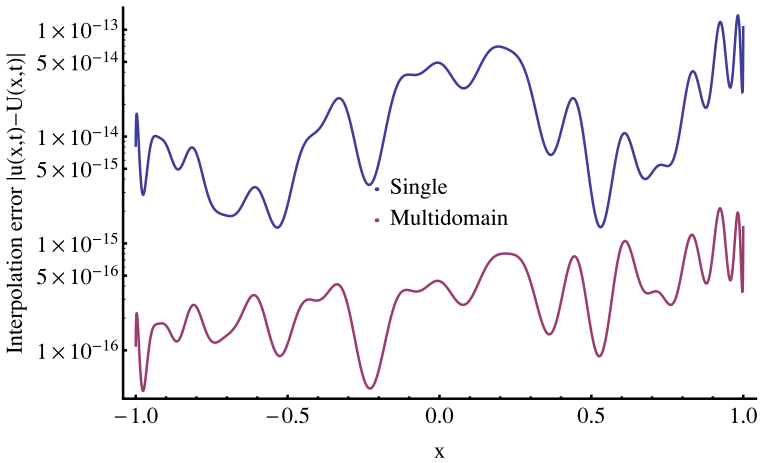


Fig. 5. Comparison of interpolation errors for Example 5 using the single and multidomain approach.

References

[1] M. Witten, Hyperbolic Partial Differential Equation, Pergamon Press, New York, 1983.
[2] A.P.S. Selvadurai, Partial Differential Equations in Mechanics 1, Springer, Stuttgart, Germany, 2001.
[3] H.O. Kreiss, Initial boundary value problems for hyperbolic systems, Comm. Pure Appl. Math. 23 (1970) 277–298.
[4] M.G. Macaraeg, C.L. Street, Improvements in spectral collocation discretization through a multiple domain technique, Appl. Numer. Math. 2 (1986) 95–108.
[5] H. Guillard, R. Peyret, On the use of spectral methods for the numerical solution of stiff problems, Comput. Methods Appl. Mech. Engrg. 66 (1988) 17–43.
[6] T. Kato, Blow-up of solutions of some nonlinear hyperbolic equations, Comm. Pure Appl. Math. 33 (1980) 501–505.
[7] J.B. Marsha, O. Joseph, Adaptive mesh refinement for hyperbolic partial differential equations, Comput. Phys. 53 (1984) 484–512.
[8] J.N. Francis, D.W. Joseph, Norms of inverses and condition numbers of matrices associated with scattered data, Approx. Theory 64 (1991) 69–94.
[9] W.S. Don, Accuracy and speed of computing the Chebyshev collocation derivative, SIAM, Sci. Comput. 16 (1995) 1253–1268.
[10] P. Demaret, M.O. Deville, Chebyshev collocation solutions of the Navier–Stokes equations using multi-domain decomposition and finite element preconditioning, Comput. Phys. 95 (1991) 359–386.
[11] A. Pinelli, A. Vacca, A. Quarteroni, A spectral multi-domain method for the numerical simulation of turbulent flows, Comput. Phys. 136 (1997) 546–558.
[12] M. Dehghan, A. Taleei, A Chebyshev pseudospectral multidomain method for the soliton solution of coupled nonlinear Schrodinger equations, Comput. Phys. Comm. 182 (2011) 2519–2529.
[13] M. Dehghan, A. Taleei, Numerical solution of the Yukawa-Coupled Klein–Gordon–Schrödinger equations via a Chebyshev pseudospectral multidomain method, Appl. Math. Model. 36 (2012) 2340–2349.
[14] D.A. Kopriva, A spectral multidomain method for the solution of hyperbolic systems, Appl. Numer. Math. 2 (1986) 221–241.
[15] A. Taleei, M. Dehghan, A Pseudo-spectral method that uses an overlapping multidomain technique for the numerical solution of sine-gordon equation in one and two spatial dimensions, Math. Methods Appl. Sci. 37 (2014) 1909–1923.
[16] M. Dehghan, R. Salehi, A method based on meshless approach for the numerical solution of the two-space dimensional hyperbolic telegraph equation, Math. Methods Appl. Sci. 35 (2012) 1220–1233.
[17] M. Dehghan, A. Mohebbi, High order implicit collocation method for the solution of two-dimensional linear hyperbolic equation, Numer. Methods Partial Differential Equations 25 (2009) 232–243.
[18] S.S. Motsa, P. Dlamini, M. Khumalo, A new multi-stage spectral relaxation method for solving chaotic initial value systems, Nonlinear Dynam. 72 (2013) 265–283.
[19] S.S. Motsa, A new piecewise-quasilinearization method for solving chaotic systems of initial value problems, Cent. Eur. J. Phys. 10 (2012) 936–946.
[20] S.S. Motsa, V.M. Magagula, P. Sibanda, A bivariate Chebyshev spectral collocation quasilinearization methods for nonlinear evolution parabolic equations, Sci. World J. 20 (2014) 251–260.
[21] F. Stephen, O.R. John, O.S. Colm, T. Patrick, How reflected wave fronts dynamically establish Hooke's law in a spring, Eur. J. Phys. 33 (2012) 417–427.
[22] J.A. Elaf, Modified treatment of initial boundary value problems for one-dimensional heat-like and wave-like equations using adomian decomposition method, Basrah J. Sci. 30 (2012) 86–105.
[23] M.E. Tariq, M.A.H. Eman, Analytical solution for telegraph equation by modified of Sumudu transform "Elzaki transform", Math. Theory Model. 2 (2012) 104–112.
[24] A.E.A. Mahmoud, H.M.Z. Emad, M.A.K. Mostafa, Exact traveling wave solutions for modified Liouville equation arising in mathematical physics and biology, Int. J. Appl. Math. 112 (2015) 75–87.
[25] M.A. Salam, Traveling-wave solution of modified Liouville equation by means of modified simple equation method, Int. Sch. Res. Netw. 47 (2012) 52–56.
[26] E. Farshad, E. Farzad, H. Amin, H. Nahid, Analytical solution of phi-four equation, Tech. J. Eng. Appl. Sci. 14 (2013) 1378–1388.
[27] L.N. Trefethen, Spectral Methods in MATLAB, SIAM, Philadelphia, 2000.
[28] M. Revers, On the approximation of certain functions by interpolating polynomials, Bull. Aust. Math. Soc. 58 (1998) 505–512.
[29] R.E. Bellman, R.E. Kalaba, Quasilinearization and Nonlinear Boundary-Value Problems, Elsevier Publishing Company, New York, 1965.
[30] G. Mariano, S. Thomas, On the history of multivariate polynomial interpolation, Comput. Appl. Math. 122 (2000) 23–35.
[31] H.B. Ali, A highly accurate collocation algorithm for $1+1$ and $2+1$ fractional percolation equations, Vib. Control 5 (2015) 1–23.
[32] E.S. Herbert, Converting interpolation series into Chebyshev series by recurrence formulas, Math. Comp. 30 (1976) 295–302.
[33] J.F. Epperson, On the runge example, J. Amer. Math. Soc. 94 (1987) 329–341.
[34] D.L. Peter, Development of singularities of solutions of nonlinear hyperbolic partial differential equations, Math. Phys. 5 (1964) 6–11.

Chapter 4

A Modified Spectral Collocation Method of Solution for One-Dimensional Parabolic PDEs

In this chapter, we aim to take full advantages of the combined benefits of incorporating both the overlapping grid approach in space and the non-overlapping subdomains approach in time. An effective overlapping grid based multidomain spectral collocation method is developed and applied to solve parabolic partial differential equations defined on large spatial and time domains. Numerical experimentation to illustrate the applicability of the modified solution approach is conducted using typical examples of parabolic PDEs that have been reported in the literature. We demonstrate that concurrent decomposition of large spatial and time computational domain into smaller subintervals guarantees improved accuracy and reduced computational time as opposed to further increase in the number of grid points in a single large domain. To benchmark, the current method of solution, new versions of the error bound theorems and proofs for bivariate polynomial interpolation emerge as a consequence of incorporating the overlapping grid approach and non-overlapping subdomains approach in space and time variables, respectively. The current method of solution could handle other classes of parabolic PDEs modeling different real-life scenarios.

Efficient overlapping grids based multidomain spectral collocation method for solving parabolic partial differential equations

F.M. Samuel[1,a,c], S.S. Motsa[1,a,b]

[a] *University of KwaZulu-Natal, School of Mathematics, Statistics, and Computer Science, Private Bag X01, Scottsville, 3209, South Africa;*
[b] *University of Swaziland, Department of Mathematics, Private Bag 4, Kwaluseni, Swaziland;*
[c] *Taita Taveta University, Department of Mathematics, 635-80300, Voi, Kenya.*

Abstract

In this article, an efficient multidomain spectral collocation method is introduced and applied to solve parabolic partial differential equations (PDEs) that are defined on large intervals. In the literature, there exist multidomain spectral collocation approaches that have been applied in either space or time variables, but not simultaneously on both. In this work, we demonstrate that rather than incrementing the number of grid points in a single domain, concurrent decomposition of large spatial and time computational domains into smaller subintervals ensured better accuracy of results and reduced computational time. In the solution process, the domain of approximation is decomposed into smaller, equal overlapping and non-overlapping subintervals in space and time directions, respectively. The PDE is discretized on each subinterval using spectral collocation at Chebyshev-Gauss-Lobatto (CGL) points. The PDE is then solved simultaneously across all subintervals in space. In the time direction, the solution is computed independently at each subinterval and the approximate solution is obtained by matching solutions at different subintervals along the common boundaries. We demonstrate the accuracy and effectiveness of the proposed method by presenting numerical results of well known parabolic PDEs that have been reported literature as a single equation or systems of equations. New error bounds theorems and proofs have been presented to benchmark the current method of solution.

Keywords: Parabolic PDEs, Spectral collocation, Multidomain, Overlapping grids, Error bounds, CGL points.

2010 MSC: 65M15, 65M55, 65M70.

*Corresponding author: felexmutua@gmail.com

1. Introduction

Partial differential equations often arise in many practical problems in fields of mathematical physics and engineering. In particular, parabolic PDEs are useful in describing a wide family of problems in science and engineering that include physical and mathematical systems with a time variable such as heat diffusion and ocean acoustics propagation [1]. Many highly nonlinear PDEs that are encountered in engineering are difficult to solve analytically; the few that may possess analytic solutions, finding the solution is cumbersome, and the solution process may depend on making a number of essential assumptions [2]. Consequently, owing to the potential applicability of the nonlinear PDEs, the quest to seek the best methods that approximate the solutions of nonlinear models has remained an active area of research in recent decades.

Examples of parabolic PDEs that are considered in this article include the evolutionary equations that are single equations such as the reaction-diffusion equation, the generalized Burgers-Fisher equation [3], and the generalized FitzHugh-Nagumo equation [4]. Reaction-diffusion equations model such processes in chemical kinematics, astrophysics, and biology. The generalized Burgers-Fisher equation describes different mechanisms that arise in fields of financial mathematics, fluid mechanics, shock wave formation, traffic flows, turbulence, heat conduction and transmission of sound waves through viscous media, among other disciplines of applied sciences [5]. The generalized FitzHugh-Nagumo equation [6] arises in genetics, biology, and heat and mass transfer [7, 8]. In biology, the FitzHugh-Nagumo equation models the activation and deactivation dynamics of a spiking neuron [9]. For systems of parabolic PDEs, we consider fluid flow problems of boundary layer type, in particular, the boundary layer flow over a permeable flat plate arising from differences in concentration or material constitution, in conjunction with temperature effects. These type of flows have received much attention from researchers owing to their great practical importance [10]. For instance, atmospheric flows at all scales, are driven appreciably by both temperature and water concentration differences [11]. In addition, flow in water bodies are driven through equally important effects of temperature [12].

Several methods have been developed and applied to solve parabolic PDEs. They include the semi-analytical methods such as; the Tanh method [13], Adomian decomposition method [14, 15], homotopy perturbation method [16], the variational iteration method [17], and numerical methods such as the finite difference methods [18] and the spectral collocation methods [19]. As for the problems considered in this study, Javidi [20] studied the spectral collocation method for the

generalized reaction-diffusion equation and later, working in collaboration with Golbabai [21], they applied the spectral domain decomposition method on the same problem. Babalian and Saeidian [22] gave some analytic approaches to the solution of the Burgers-Fisher equation. The explicit solutions and numerical simulations of the generalized Burgers-Fisher equation were introduced by Kaya and El-Sayed [23]. Wazwaz [13] presented the Tanh method for the generalized Burgers-Fisher equation which was later solved by Mickens and Gumel [24] and Zhao et al. [25] using a nonstandard finite difference scheme and the Chebyshev Legendre pseudospectral method, respectively. Numerical approximations of the generalized FitzHugh-Nagumo equation also exist. For instance, the Hopscotch finite difference scheme, which is a fast second order differential equation solver first proposed by Gordon [26], was further developed by Gourlay [27] and applied to solve the FitzHugh-Nagumo equation. Other numerical methods that have been applied to solve the FitzHugh-Nagumo equation include the pseudospectral method [28] and the polynomial differential quadrature method [29]. For the case, of boundary layer flows, the problem of natural convection flow from a vertical permeable flat plate with variable surface temperature and species concentration was solved by Hussain et al. [30] using finite differences for the entire ξ region (where ξ is the scaled stream-wise variable), series solution method for small ξ and asymptotic solution method for large ξ. There exists a vast literature on different methods that have been applied to numerically solve boundary layer flow problems (see [31]).

Among the numerical methods that are available in the literature for approximating solutions of parabolic PDEs, spectral methods (Galerkin, tau, and collocation-based) have been shown to be superior, especially where the problem possesses a smooth solution [32]. Spectral methods are advantageous in that they require low numbers of grid points to give accurate results and short computation time is needed to realize the results [33]. When solving initial-boundary value problems, the spectral collocation-based approaches are often used because they offer a simple treatment of boundary conditions [34, 35]. Motsa et al. [36] used spectral and collocation techniques to develop the bivariate spectral quasi-linearization method for solving nonlinear parabolic PDEs. In this method, the spectral approximation of the PDE that is sought is a bivariate Lagrange interpolating polynomial. The quasi-linearization (QLM) method of Bellman and Kalaba [37] is first applied to simplify a nonlinear PDE, which is then discretized using spectral collocation in both space and time variables. The method has since been successfully applied to solve problems that are defined over small space and time intervals. However, it has been observed that when the method is

applied to solve a PDE defined over a large time interval, the accuracy deteriorates even with a large number of grid points [38]. In view of this drawback, the applicability of the method has recently been widened in a new version that is based on decomposing the time variable domain into smaller non-overlapping subintervals [39]. In the modified approach, the PDE is solved independently on each time subinterval and the resultant solution of the PDE is obtained by merging the solution from different subintervals along common boundaries. We remark that, nevertheless, this approach still disregards a large spatial domain an issue that we now address.

In this study, we aim to introduce a new multidomain spectral collocation method for the solution of parabolic PDEs that are defined over large space and time intervals. We also aim to demonstrate that the current multi-domain approach yields very accurate results in a remarkably short CPU time. The new method is based on decomposing the space and time variable domains into smaller equal overlapping and non-overlapping subintervals, respectively. The PDE is then discretized for each of these subintervals using spectral collocation with Chebyshev-Gauss-Lobatto points. The spatial variable domain overlapping is carried out such that the first and last two nodes of the interior subintervals coincide with those of the neighboring subintervals and they remain common. Chebyshev differentiation matrices from the different spatial subintervals are incorporated to assemble a new Chebyshev differentiation matrix for the decomposed space variable. The PDE is solved over the entire spatial domain, but independently at each time subinterval. For the subintervals in time, the continuity condition to obtain initial conditions at subsequent subintervals. The solution of the PDE in the entire time domain is taken as the union of the solutions at the different time subintervals. To the best of our knowledge, such a spectral collocation-based method with Chebyshev-Gauss-Lobatto points has not been applied simultaneously on overlapping grids in space and non-overlapping grids in time. To further explain the performance of the proposed method and our preferred choice of Chebyshev-Gauss-Lobatto points over other sets of nodes, we develop new error bound theorems that govern piece-wise bivariate polynomial interpolation using CGL nodes and present their rigorous proofs.

The rest of this article is organized as follows; In Section 2, we briefly introduce the multidomain spectral collocation method and establish numerical discretization for a single nonlinear PDE. This numerical scheme is then extended in Section 3 to solve systems of nonlinear parabolic PDEs of boundary layer flow type. Results that show the performance of the proposed method are given and discussed in Section 4. That section ends with an exploration of error bounds theorems in

piece-wise bivariate polynomial interpolation using CGL grid points. Finally, the conclusion of our work and acknowledgment are given in Sections 5 and 6, respectively.

2. Method of solution for a single nonlinear equation

In this section, we describe an algorithm to show how the multi-domain spectral collocation method can be applied to solve a single equation parabolic PDEs defined on the domain $(x,t) \in [a,b]\times[0,T]$ for large $(b-a)$ and T. The method is illustrated by considering a general second-order nonlinear parabolic PDE;

$$\frac{\partial u}{\partial t} = F\left(u, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right), \quad x \in (a,b),\ t \in (0,T], \tag{1}$$

where F is a nonlinear operator operating on the unknown function $u(x,t)$ and its first two spatial derivatives. The PDE Eq.(1) is solved subject to the boundary conditions

$$\alpha_1 \frac{\partial u}{\partial x}(a,t) + \alpha_0 u(a,t) = g_a(t), \quad \beta_1 \frac{\partial u}{\partial x}(b,t) + \beta_0 u(b,t) = g_b(t), \quad t \in (0,T], \tag{2}$$

and initial condition

$$u(x,0) = f(x), \quad x \in [a,b]. \tag{3}$$

The components $\alpha_0, \alpha_1, \beta_0, \beta_1$ are known constants and $f(x)$, $g_a(t)$ and $g_b(t)$ are known functions. The method of solution involves the stages given in the subsections below.

2.1. The quasi-linearisation method

The PDE is first simplified using the quasi-linearisation method (QLM) of Bellman and Kalaba [37]. The QLM is based on the Newton-Raphson method and is built from the linear terms of a Taylor series expansion about an initial approximation to the solution. The QLM assumes that the difference between solutions at two successive iterations, denoted by $u_{s+1} - u_s$, is small. Applying the QLM on Eq.(1) we obtain

$$\delta_{2,s}(x,t)u''_{s+1} + \delta_{1,s}(x,t)u'_{s+1} + \delta_{0,s}(x,t)u_{s+1} - \dot{u}_{s+1} = R_s(x,t), \tag{4}$$

where

$$R_s(x,t) = \sum_{\zeta=0}^{2} \delta_{\zeta,s}(x,t)u_s^{(\zeta)} - F(u_s, u'_s, u''_s), \quad \delta_{\mu,s}(x,t) = \frac{\partial F}{\partial u_s^{(\mu)}}(u_s, u'_s, u''_s), \quad \mu = 0,1,2. \tag{5}$$

Here the prime and dot denotes differentiation with respect to x and t, respectively, s is the iteration level and $u^{(0)} = u$. Starting with an initial approximation for solution u_0, the QLM scheme Eq.(4) is solved iteratively until a solution with the desired accuracy is realized.

2.2. Domain decomposition and discretization

The domain of approximation is decomposed in both space and time direction into smaller subintervals. First, we consider the fragmentation of time variable domain. If we let $t \in \Gamma$ where $\Gamma \in [0, T]$. The domain Γ is decomposed into p equal non-overlapping subintervals as

$$\Gamma_k = [t_{k-1}, t_k], \quad t_{k-1} < t_k, \quad t_0 = 0, \quad t_p = T, \quad k = 1, 2, \ldots, p. \tag{6}$$

The domain $t \in [t_{k-1}, t_k]$ in each of the k^{th} subintervals is transformed into $\hat{t} \in [-1, 1]$ using the linear map

$$\hat{t}(t) = \frac{2}{t_k - t_{k-1}} \left[t - \frac{1}{2} (t_k + t_{k-1}) \right], \quad t \in [t_{k-1}, t_k]. \tag{7}$$

In the next step, each subinterval is further discretized into N_t+1 Chebyshev-Gauss-Lobatto points, before the spectral collocation is applied. The entire grid in the t variable can be represented as

$$\{0 = t_{N_t}^{(1)}, \ldots, t_0^{(1)} = t_{N_t}^{(2)}, \ldots, t_0^{(k-1)} = t_{N_t}^{(k)}, \ldots, t_0^{(p)} = T, \ldots, \ 2 \leq k \leq p\}. \tag{8}$$

The superscripts and subscripts are the subinterval and grid point indices, respectively. The spatial domain $x \in [a, b]$ is then decomposed into q equal overlapping subintervals as

$$\Lambda_l = [x_{l-1}, \bar{x}_l], \quad x_{l-1} < x_l < \bar{x}_l, \quad x_0 = a, \ \bar{x}_q = b, \quad l = 1, 2, \ldots, q, \tag{9}$$

where $x_l < \bar{x}_l$, depicts the overlapping nature. The computational domain $[x_{l-1}, \bar{x}_l]$ in the l^{th} subinterval is next transformed into $\hat{x} \in [-1, 1]$ by applying the linear map

$$\hat{x}(x) = \frac{2}{\bar{x}_l - x_{l-1}} \left[x - \frac{1}{2} (\bar{x}_l + x_{l-1}) \right], \quad x \in [x_{l-1}, \bar{x}_l]. \tag{10}$$

Finally, each subinterval is further discretized into $N_x + 1$ Chebyshev-Gauss-Lobatto points. The subintervals are made to overlap in such a way that the first two grids points in the Λ_{l+1} subinterval coincide with the last two grid points in the Λ_l subinterval. In general, the set of grid points over the entire x domain can be represented as

$$\{a = x_{N_x}^{(1)}, \ldots, x_1^{(1)} = x_{N_x}^{(2)}, x_0^{(1)} = x_{N_x-1}^{(2)}, \ldots, x_1^{(l-1)} = x_{N_x}^{(l)}, x_0^{(l-1)} = x_{N_x-1}^{(l)}, \ldots, x_0^{(q)} = b, \ 2 \leq l \leq q\}. \tag{11}$$

The ordering of grid points as illustrated in Eq.(8) and Eq.(11) signifies that the spectral collocation is done from right to left of the subinterval. The collocation nodes in the k^{th} subinterval in t variable and l^{th} subinterval in x variable are defined in [32] by

$$\{\hat{t}_j\}_{j=0}^{N_t} = \cos\left(\frac{j\pi}{N_t}\right), \quad \text{and} \quad \{\hat{x}_i\}_{i=0}^{N_x} = \cos\left(\frac{i\pi}{N_x}\right). \tag{12}$$

An explicit expression of the length $L = \bar{x}_l - x_{l-1}$ in terms of the number of subintervals q in x is useful in defining the grid points at the interior subintervals in space. To obtain this, we first observe that overlapping of q subintervals in x results into $(q-1)$ regions of overlaps. Further, using the linear map Eq.(10) we notice that each of the overlapping space regions has the length

$$L\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right).$$

The length L of the subinterval can, therefore, be obtained from the solution of

$$qL - L\,(q-1)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right) = b - a, \tag{13}$$

which is

$$L = \frac{b-a}{q + (1-q)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right)}. \tag{14}$$

Evidently, x_l and $\bar{x}_l$ are related by

$$\bar{x}_l = x_l + L\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right). \tag{15}$$

We remark that this subdivision of the entire space domain into subintervals of the same length and subsequent discretization of each subinterval into an equal number of grid points is a sufficient condition for the grid points of neighboring subdomains to overlap.

2.3. Spectral collocation

The multidomain spectral collocation method is implemented on the linearized QLM scheme Eq.(4) as illustrated below. The PDE is solved independently over each t subinterval; with the label $\overset{(k)}{u}$, $k = 1, 2, \ldots, p$, being used to distinguish solutions at different t subintervals. We note that the solution in the spatial direction is computed simultaneously across all subintervals and that the multidomain approach is evident only when assembling differentiation matrices in x. We must solve

$$\delta_{2,s}(x,t)\frac{\partial^2 \overset{(k)}{u}_{s+1}}{\partial x^2} + \delta_{1,s}(x,t)\frac{\partial \overset{(k)}{u}_{s+1}}{\partial x} + \delta_{0,s}(x,t)\overset{(k)}{u}_{s+1} - \frac{\partial \overset{(k)}{u}_{s+1}}{\partial t} = \overset{(k)}{R}_s(x,t),\ x \in [a,b],\ t \in [t_{k-1}, t_k], \tag{16}$$

subject to boundary conditions

$$\alpha_1 \frac{\partial \overset{(k)}{u}}{\partial x}(a,t) + \alpha_0 \overset{(k)}{u}(a,t) = g_a(t), \quad \beta_1 \frac{\partial \overset{(k)}{u}}{\partial x}(b,t) + \beta_0 \overset{(k)}{u}(b,t) = g_b(t),\ \ t \in (0,T], \tag{17}$$

and initial condition

$$\overset{(1)}{u}(x,0) = f(x), \quad \overset{(k)}{u}(x,t_{k-1}) = \overset{(k-1)}{u}(x,t_{k-1}),\ k = 2,\ldots,p, \quad x \in [a,b]. \tag{18}$$

In the solution process, the approximate solution that is sought is a bivariate Lagrange interpolating polynomial that takes the form

$$\overset{(k)}{u}(x,t) \approx \overset{(k)}{U}(x,t) = \sum_{p=0}^{N_x}\sum_{q=0}^{N_t} \overset{(k)}{U}(x_p,t_q)L_p(x)L_q(t). \tag{19}$$

The spatial derivatives at each subinterval, $l = 1,2,\ldots,q$, are evaluated at the collocation nodes $(\hat{x}_i,\hat{t}_j)$, for $j = 0,1,2,\ldots,N_t$, as follows

$$\frac{\partial \overset{(k)}{u}}{\partial x}(\hat{x}_i,\hat{t}_j) = \mathbf{D}\overset{(k)}{\mathbf{U}}_j = \left(\frac{2}{L}\right)\hat{\mathbf{D}}\overset{(k)}{\mathbf{U}}_j, \quad \frac{\partial^2 \overset{(k)}{u}}{\partial x^2}(\hat{x}_i,\hat{t}_j) = \mathbf{D}^2\overset{(k)}{\mathbf{U}}_j, \quad \frac{\partial^n \overset{(k)}{u}}{\partial x^n}(\hat{x}_i,\hat{t}_j) = \mathbf{D}^n\overset{(k)}{\mathbf{U}}_j \tag{20}$$

where $\hat{\mathbf{D}} = \left(\frac{L}{2}\right)\mathbf{D}$ of size $(N_x+1)\times(N_x+1)$ is the standard first order Chebyshev differentiation matrix as defined in [32]. The vector $\overset{(k)}{\mathbf{U}}_j$ is defined as

$$\overset{(k)}{\mathbf{U}}_j = \left[\overset{(k)}{u}(x_0^{(l)},t_j), \overset{(k)}{u}(x_1^{(l)},t_j), \overset{(k)}{u}(x_2^{(l)},t_j,), \ldots, \overset{(k)}{u}(x_{N_x}^{(l)},t_j)\right]^T, \quad l = 1,2,\ldots,q, \tag{21}$$

where T denotes matrix transpose. Since the last two points in the l^{th} subinterval and the first two points in the $(l+1)^{th}$ subinterval overlap and remain common, we can assemble the Chebyshev differentiation matrix $\overset{(1,0)}{\mathbf{D}}$ for the multidomain approach in x by carefully discarding the recurrent points as

$$\overset{(1,0)}{\mathbf{D}} = \begin{bmatrix}
\overset{q}{D}_{0,0} & \ldots & \overset{q}{D}_{0,N_x-1} & \overset{q}{D}_{0,N_x} & & & & & & \\
\overset{q}{D}_{1,0} & \ldots & \overset{q}{D}_{1,N_x-1} & \overset{q}{D}_{1,N_x} & & & & & & \\
\ddots & \ddots & \ddots & \ddots & & & & & & \\
\overset{q}{D}_{N_x-1,0} & \ldots & \overset{q}{D}_{N_x-1,N_x-1} & \overset{q}{D}_{N_x-1,N_x} & & & & & & \\
 & & \ddots & \ddots & & & & & & \\
 & & \overset{2}{D}_{1,0} & \overset{2}{D}_{1,1} & \ldots & \overset{2}{D}_{1,N_x-1} & \overset{2}{D}_{1,N_x} & & & \\
 & & \ddots & \ddots & \ddots & \ddots & \ddots & & & \\
 & & \overset{2}{D}_{N_x-1,0} & \overset{2}{D}_{N_x-1,1} & \ldots & \overset{2}{D}_{N_x-1,N_x-1} & \overset{2}{D}_{N_x-1,N_x} & & & \\
 & & & & & \overset{1}{D}_{1,0} & \overset{1}{D}_{1,1} & \ldots & \overset{1}{D}_{1,N_x} \\
 & & & & & \ddots & \ddots & \ddots & \ddots \\
 & & & & & \overset{1}{D}_{N_x-1,0} & \overset{1}{D}_{N_x-1,1} & \ldots & \overset{1}{D}_{N_x-1,N_x} \\
 & & & & & \overset{1}{D}_{N_x,0} & \overset{1}{D}_{N_x,1} & \ldots & \overset{1}{D}_{N_x,N_x}
\end{bmatrix}.$$

(22)

Here the empty entries of matrix $\overset{(1,0)}{\mathbf{D}}$ are zeros and $\overset{l}{D}$ represents the Chebyshev differentiation matrix in the l^{th} subinterval in space. The size of matrix $\overset{(1,0)}{\mathbf{D}}$ is $(r+1)\times(r+1)$, where $r = N_x+(N_x-1)\times(q-1)$. The higher order differentiation matrices with respect to x can be obtained using matrix multiplication. The x grid corresponding to the assembled differentiation matrix over entire spatial domain in the multidomain approach is

$$\{a = x_{N_x}^{(1)},\ldots,x_1^{(1)},x_{N_x-1}^{(2)},\ldots,x_1^{(2)},\ldots,x_{N_x-1}^{(l)},\ldots,x_1^{(l)},\ldots,x_{N_x-1}^{(q)}\ldots,x_0^{(q)} = b,\ 3\le l\le q-1\}. \tag{23}$$

The time derivative is approximated at the collocation nodes $(\hat{x}_i,\hat{t}_j)$, for $i = 0,1,2,\ldots,r$, as

$$\frac{\partial \overset{(k)}{u}}{\partial t}(\hat{x}_i,\hat{t}_j) = \sum_{q=0}^{N_t} d_{j,q}\overset{(k)}{\bar{\mathbf{U}}}_q = \sum_{q=0}^{N_t}\left(\frac{2}{t_k-t_{k-1}}\right)\hat{d}_{j,q}\overset{(k)}{\bar{\mathbf{U}}}_q, \tag{24}$$

where $\hat{d}_{j,q} = \left(\frac{t_k-t_{k-1}}{2}\right)d_{j,q},\ \ j,q = 0,1,2,\ldots,N_t$, are entries of the standard first order Chebyshev differentiation matrix of size $(N_t+1)\times(N_t+1)$. In Eq.(24), q is used in the grid points sense and does not refer to the number of subintervals in x. The vector $\overset{(k)}{\bar{\mathbf{U}}}_j$ is defined as

$$\overset{(k)}{\bar{\mathbf{U}}}_j = \left[\overset{(k)}{u}(x_0,t_j),\overset{(k)}{u}(x_1,t_j),\ldots,\overset{(k)}{u}(x_r,t_j)\right]^T, \tag{25}$$

where T denotes matrix transpose. We recognize that the bar in $\overset{(k)}{\bar{\mathbf{U}}}_j$ at Eq.(25) distinguishes it from the vector $\overset{(k)}{\mathbf{U}}_j$ at Eq.(21). The x grid points at Eq.(25) are those in Eq.(23) when arranged from right to left such that $x_0 = b = x_0^{(q)},\ x_r = a = x_{N_x}^{(1)}$. Using the definition of discrete derivatives and the initial condition, Eq.(16) can be expressed in matrix form as

$$\left[\boldsymbol{\delta}_{2,s}(\mathbf{x},t_i)\overset{(2,0)}{\mathbf{D}} + \boldsymbol{\delta}_{1,s}(\mathbf{x},t_i)\overset{(1,0)}{\mathbf{D}} + \boldsymbol{\delta}_{0,s}(\mathbf{x},t_i)\right]\overset{(k)}{\bar{\mathbf{U}}}_i - \sum_{j=0}^{N_t-1} d_{i,j}\overset{(k)}{\bar{\mathbf{U}}}_j = \overset{(k)}{\mathbf{R}}_s(\mathbf{x},t_i) + d_{i,N_t}\overset{(k)}{\bar{\mathbf{U}}}_{t_{N_t}}. \tag{26}$$

Eq.(26) is a system of $N_t(r+1)$ linear equations in $N_t(r+1)$ unknowns which can be expressed as the $N_t(r+1)\times N_t(r+1)$ matrix system given by

$$\begin{bmatrix} \mathbf{A}_{0,0} & \mathbf{A}_{0,1} & \mathbf{A}_{0,2} & \ldots & \mathbf{A}_{0,N_t-1} \\ \\ \mathbf{A}_{1,0} & \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \ldots & \mathbf{A}_{1,N_t-1} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ \mathbf{A}_{N_t-1,0} & \mathbf{A}_{N_t-1,1} & \mathbf{A}_{N_t-1,2} & \ldots & \mathbf{A}_{N_t-1,N_t-1} \end{bmatrix}\begin{bmatrix} \overset{(k)}{\bar{\mathbf{U}}}_0 \\ \overset{(k)}{\bar{\mathbf{U}}}_1 \\ \vdots \\ \overset{(k)}{\bar{\mathbf{U}}}_{N_t-1} \end{bmatrix} = \begin{bmatrix} \overset{(k)}{\mathbf{R}}_0 \\ \overset{(k)}{\mathbf{R}}_1 \\ \vdots \\ \overset{(k)}{\mathbf{R}}_{N_t-1} \end{bmatrix}, \tag{27}$$

where

$$A_{i,i} = \boldsymbol{\delta}_{2,s}(\mathbf{x}, t_i)\overset{(2,0)}{\mathbf{D}} + \boldsymbol{\delta}_{1,s}(\mathbf{x}, t_i)\overset{(1,0)}{\mathbf{D}} + \boldsymbol{\delta}_{0,s}(\mathbf{x}, t_i) - d_{i,i}\mathbf{I}, \;\; A_{i,j} = -d_{i,j}\mathbf{I}, \; i \neq j,$$

$$\mathbf{R}_i^{(k)} = \mathbf{R}_s^{(k)}(\mathbf{x}, t_i) + d_{i,N_t}\mathbf{U}_{N_t}^{(k)},$$

$$\boldsymbol{\delta}_{\mu,s}(\mathbf{x}, t_i) = \begin{bmatrix} \delta_{\mu,s}(x_0, t_i) & & & \\ & \delta_{\mu,s}(x_1, t_i) & & \\ & & \ddots & \\ & & & \delta_{\mu,s}(x_r, t_i) \end{bmatrix}, \quad \mu = 0, 1, 2,$$

and $\mathbf{I}$ is an identity matrix of size $(r+1) \times (r+1)$. The boundary conditions when evaluated at the collocation points give

$$\alpha_1 \sum_{p=0}^{r} D_{r,p}^{(1,0)} \overset{(k)}{U}(x_p, t_i) + \alpha_0 \overset{(k)}{U}(x_r, t_i) = g_a(t_i), \quad \beta_1 \sum_{p=0}^{r} D_{0,p}^{(1,0)} \overset{(k)}{U}(x_p, t_i) + \beta_0 \overset{(k)}{U}(x_0, t_i) = g_b(t_i). \quad (28)$$

The boundary conditions are imposed on the first and last row of the diagonal sub-matrices in Eq.(27).

The practical applicability of the multidomain spectral collocation method to solutions of parabolic PDEs that are described as a single equation is illustrated by considering the solution of well known linear and nonlinear evolutionary PDEs that have been reported in the literature. We notice that the scheme for the nonlinear PDEs described in this section can be adapted to solve linear PDEs and in this case the coefficients $\boldsymbol{\delta}_{\mu,s}$, $\mu = 0, 1, 2$, and $\overset{(k)}{\mathbf{R}}$ will be free of the dependent variable u. The selected test examples below are parabolic PDEs that possess an exact solution for easy validation.

Example 1. Consider the following linear reaction-diffusion equation [14]

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} - u, \;\; x \in (0, 20), \;\; t \in (0, 10]. \quad (29)$$

Eq.(29) is solved subject to the boundary conditions

$$u(0, t) = 1, \quad u(20, t) = e^{-20} + 20e^{-t}, \;\; t > 0, \quad (30)$$

and initial conditions

$$u(x, 0) = e^{-x} + x. \quad (31)$$

The exact solution is

$$u(x, t) = e^{-x} + xe^{-t}. \quad (32)$$

Example 2. Examine the solution of a generalized Burgers-Fisher equation [5]

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{\partial^2 u}{\partial x^2} + u(1-u), \quad x \in (0,20), \quad t \in (0,10], \tag{33}$$

solved subject to the boundary conditions

$$u(0,t) = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{5t}{8}\right), \quad u(20,t) = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{5t}{8} - \frac{20}{4}\right), \quad t > 0, \tag{34}$$

and initial condition

$$u(x,0) = \frac{1}{2} - \frac{1}{2}\tanh\left(\frac{x}{4}\right). \tag{35}$$

The exact solution is given in [5] as

$$u(x,t) = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{5t}{8} - \frac{x}{4}\right). \tag{36}$$

Example 3. Inspect the solution of the generalized FitzHugh-Nagumo equation [4]

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + u(u-1)(1-u), \quad x \in (1,5), \quad t \in (0,1], \tag{37}$$

that is solved subject to the boundary conditions

$$u(1,t) = \frac{1}{2}\left[1 - \coth\left(\frac{-1}{2\sqrt{2}} + \frac{t}{4}\right)\right], \quad u(5,t) = \frac{1}{2}\left[1 - \coth\left(\frac{-5}{2\sqrt{2}} + \frac{t}{4}\right)\right], \quad t > 0, \tag{38}$$

and initial condition

$$u(x,0) = \frac{1}{2}\left[1 - \coth\left(\frac{-x}{2\sqrt{2}}\right)\right]. \tag{39}$$

The exact solution is given in [8] as

$$u(x,t) = \frac{1}{2}\left[1 - \coth\left(\frac{-x}{2\sqrt{2}} + \frac{t}{4}\right)\right]. \tag{40}$$

We remark that Example 3 is chosen to emphasize the fact that the benefits of the proposed method may not be manifested in problems that are defined over small computational domains.

3. Method of solution for systems of parabolic PDEs

In this section, we present an extension of the numerical scheme that was described in the previous section to now include systems of nonlinear partial differential equations of boundary layer flow type. The numerical scheme is illustrated in the examples given below.

Example 4. Consider a system of two equations that describe the problem of steady two-dimensional laminar free convection flow past a non-isothermal porous cone with variable surface temperature. The governing non-similarity system of partial differential equations are given in the dimensionless form by Hussain et al. [30] as

$$f''' + \frac{n+7}{4} f f'' - \frac{n+1}{2} f'^2 + \theta + \xi f'' = \frac{1-n}{4} \xi \left(f' \frac{\partial f'}{\partial \xi} - f'' \frac{\partial f}{\partial \xi} \right), \tag{41}$$

$$\frac{1}{Pr} \theta'' + \frac{n+7}{4} f \theta' - n f' \theta + \xi \theta' = \frac{1-n}{4} \xi \left(f' \frac{\partial \theta}{\partial \xi} - \theta' \frac{\partial f}{\partial \xi} \right), \tag{42}$$

where $Pr = \frac{\nu}{\alpha}$ is the Prandtl number, n is the temperature gradient, ξ is the dimensionless suction parameter and the primes denote differentiation with respect to the similarity variable η. The appropriate corresponding boundary conditions are

$$f(0,\xi) = 0, \quad f'(0,\xi) = 0, \quad \theta(0,\xi) = 1, \quad f'(\infty,\xi) = 0, \quad \theta(\infty,\xi) = 0. \tag{43}$$

The skin friction coefficient C_{fx} and the Nusset number N_{ux} describe the shear-stress and the heat flux rate at the surface, respectively, and are defined in [40] as

$$C_{fx} Gr_x^{1/4} = f''(\xi, 0), \quad \frac{Nu_x}{Gr_x^{1/4}} = -\theta'(\xi, 0). \tag{44}$$

For convenience, Eq.(41) and Eq.(42) are first expressed in the form

$$G = G\left(f''', f'', f', f, \frac{\partial f'}{\partial \xi}, \frac{\partial f}{\partial \xi}, \theta \right) = 0, \text{ and } H = H\left(f', f, \frac{\partial f}{\partial \xi}, \theta'', \theta', \theta, \frac{\partial \theta}{\partial \xi} \right) = 0, \tag{45}$$

respectively, where G and H are nonlinear operators acting on the entities inside the parenthesis. The linearized QLM scheme that can be used to approximate the solution at the $(s+1)$-th iteration level is given by the coupled system

$$\alpha_{0,s} f'''_{s+1} + \alpha_{1,s} f''_{s+1} + \alpha_{2,s} f'_{s+1} + \alpha_{3,s} f_{r+1} + \alpha_{4,s} \frac{\partial f'_{s+1}}{\partial \xi} + \alpha_{5,s} \frac{\partial f_{s+1}}{\partial \xi} + \alpha_{6,s} \theta_{s+1} = R_{1,s}, \tag{46}$$

$$\beta_{0,s} f'_{s+1} + \beta_{1,s} f_{s+1} + \beta_{2,s} \frac{\partial f_{s+1}}{\partial \xi} + \beta_{3,s} \theta''_{s+1} + \beta_{4,s} \theta'_{s+1} + \beta_{5,s} \theta_{s+1} + \beta_{6,s} \frac{\partial \theta_{s+1}}{\partial \xi} = R_{2,s}, \tag{47}$$

where

$$\alpha_{0,s} = \frac{\partial G}{\partial f_s'''} = 1, \quad \alpha_{1,s} = \frac{\partial G}{\partial f_s''} = \frac{n+7}{4} f_s + \xi + \frac{1-n}{4}\xi \frac{\partial f_s}{\partial \xi}, \quad \alpha_{2,s} = \frac{\partial G}{\partial f_s'} = -(n+1) f_s' - \frac{1-n}{4}\xi \frac{\partial f_s'}{\partial \xi},$$

$$\alpha_{3,s} = \frac{\partial G}{\partial f_s} = \frac{n+7}{4} f_s'', \quad \alpha_{4,s} = \frac{\partial G}{\partial \left(\frac{f_s'}{\partial \xi}\right)} = -\frac{1-n}{4}\xi f_s', \quad \alpha_{5,s} = \frac{\partial G}{\partial \left(\frac{f_s}{\partial \xi}\right)} = \frac{1-n}{4}\xi f_s'', \quad \alpha_{6,s} = \frac{\partial G}{\partial \theta_s} = 1,$$

$$\beta_{0,s} = \frac{\partial H}{\partial f_s'} = -n\theta_s - \frac{1-n}{4}\xi \frac{\partial \theta_s}{\partial \xi}, \quad \beta_{1,s} = \frac{\partial H}{\partial f_s} = \frac{n+7}{4}\theta_s', \quad \beta_{2,s} = \frac{\partial H}{\partial \frac{\partial f_s}{\partial \xi}} = \frac{1-n}{4}\xi \theta_s', \quad \beta_{3,s} = \frac{\partial H}{\partial \theta_s''} = \frac{1}{Pr},$$

$$\beta_{4,s} = \frac{\partial H}{\partial \theta_s'} = \frac{n+7}{4} f_s + \xi + \frac{1-n}{4}\xi \frac{\partial f_s}{\partial \xi}, \quad \beta_{5,s} = \frac{\partial H}{\partial \theta_s} = -n f_s', \quad \beta_{6,s} = \frac{H}{\partial \left(\frac{\partial \theta_s}{\partial \xi}\right)} = -\frac{1-n}{4}\xi f_s',$$

$$R_{1,s} = \frac{n+7}{4} f_s f_s'' - \frac{n+1}{2} f'^2_s - \frac{1-n}{4}\xi \left(f' \frac{\partial f_s'}{\partial \xi} - f_s'' \frac{\partial f_s}{\partial \xi} \right),$$

$$R_{2,s} = \frac{n+7}{4} f_s \theta_s' - n f_s' \theta_s - \frac{1-n}{4}\xi \left(f' \frac{\partial \theta_s}{\partial \xi} - \theta_s' \frac{\partial f_s}{\partial \xi} \right).$$

In the solution process, the semi-infinite domain $[0, \infty)$ in η is first truncated into a finite domain $[0, L_x]$ where L_x is taken to be large enough to approximate conditions at infinity. The truncated domain in η is then decomposed into smaller overlapping subintervals as illustrated in variable x, Eq.9, for the case of a single nonlinear equation. The domain in ξ is decomposed into smaller non-overlapping grids, just as demonstrated in t, Eq.6. The initial conditions for this problem at $\xi = 0$ can be found by solving the equations

$$f''' + \frac{n+7}{4} f f'' - \frac{n+1}{2} f'^2 + \theta = 0, \tag{48}$$

$$\frac{1}{Pr}\theta'' + \frac{n+7}{4} f \theta' - n f' \theta = 0, \tag{49}$$

subject to the boundary conditions

$$f(0,0) = 0, \quad f'(0,0) = 0, \quad \theta(0,0) = 1, \quad f'(\infty, 0) = 0, \quad \theta(\infty, 0) = 0. \tag{50}$$

We denote the solutions at the k-th subinterval in ξ by $\overset{(k)}{f}(\eta, \xi)$ and $\overset{(k)}{\theta}(\eta, \xi)$. Applying the continuity conditions

$$\overset{(k)}{f}(\eta, \xi_{k-1}) = \overset{(k-1)}{f}(\eta, \xi_{k-1}), \quad \overset{(k)}{\theta}(\eta, \xi_{k-1}) = \overset{(k-1)}{\theta}(\eta, \xi_{k-1}),$$

to obtain the initial conditions for $k = 2, 3, \ldots, p$, subintervals in ξ, the multidomain approach is applied on the QLM scheme Eq.(46) and Eq.(47). Thus we must solve

$$\overset{(k)}{\alpha}_{0,s} \overset{(k)}{f'''_{s+1}} + \overset{(k)}{\alpha}_{1,s} \overset{(k)}{f''_{s+1}} + \overset{(k)}{\alpha}_{2,s} \overset{(k)}{f'_{s+1}} + \overset{(k)}{\alpha}_{3,s} \overset{(k)}{f_{s+1}} + \overset{(k)}{\alpha}_{4,s} \frac{\partial \overset{(k)}{f'_{s+1}}}{\partial \xi} + \overset{(k)}{\alpha}_{5,s} \frac{\partial \overset{(k)}{f_{s+1}}}{\partial \xi} + \overset{(k)}{\alpha}_{6,s} \overset{(k)}{\theta}_{s+1} = \overset{(k)}{R}_{1,s}, \tag{51}$$

$$\overset{(k)}{\beta}_{0,s}\overset{(k)}{f'}_{s+1}+\overset{(k)}{\beta}_{1,s}\overset{(k)}{f}_{s+1}+\overset{(k)}{\beta}_{2,s}\frac{\partial \overset{(k)}{f}_{s+1}}{\partial\xi}+\overset{(k)}{\beta}_{3,s}\overset{(k)}{\theta''}_{s+1}+\overset{(k)}{\beta}_{4,s}\overset{(k)}{\theta'}_{s+1}+\overset{(k)}{\beta}_{5,s}\overset{(k)}{\theta}_{s+1}+\overset{(k)}{\beta}_{6,s}\frac{\partial \overset{(k)}{\theta}_{s+1}}{\partial\xi}=\overset{(k)}{R}_{s,s}, \tag{52}$$

subject to the boundary conditions

$$\overset{(k)}{f}_{s+1}(0,\xi)=0,\quad \overset{(k)}{f'}_{s+1}(0,\xi)=0,\quad \overset{(k)}{\theta}_{s+1}(0,\xi)=1,\quad \overset{(k)}{f'}_{s+1}(\infty,\xi)=0,\quad \overset{(k)}{\theta}_{s+1}(\infty,\xi)=0. \tag{53}$$

Applying spectral collocation, Eq.(51) and Eq.(52) can be converted into a $2N_t(r+1)$ system of linear equation in $2N_t(r+1)$ unknowns as

$$\begin{aligned}&\left[\overset{(k)}{\boldsymbol{\alpha}}_{0,s}\mathbf{D}^3+\overset{(k)}{\boldsymbol{\alpha}}_{1,s}\mathbf{D}^2+\overset{(k)}{\boldsymbol{\alpha}}_{2,s}\mathbf{D}+\overset{(k)}{\boldsymbol{\alpha}}_{3,s}\right]\overset{(k)}{\mathbf{F}}_{i,s+1}+\overset{(k)}{\boldsymbol{\alpha}}_{4,s}\sum_{j=0}^{N_t-1}d_{i,j}\mathbf{D}\overset{(k)}{\mathbf{F}}_{j,s+1}+\overset{(k)}{\boldsymbol{\alpha}}_{5,s}\sum_{j=0}^{N_t-1}d_{i,j}\overset{(k)}{\mathbf{F}}_{j,s+1}\\&+\overset{(k)}{\boldsymbol{\alpha}}_{6,s}\overset{(k)}{\boldsymbol{\Theta}}_{i,s+1}=\overset{(k)}{\mathbf{R}}_{1,i,s}-\overset{(k)}{\boldsymbol{\alpha}}_{4,s}d_{i,N_t}\mathbf{D}\overset{(k)}{\mathbf{F}}_{N_t,s+1}-\overset{(k)}{\boldsymbol{\alpha}}_{5,s}d_{i,N_t}\overset{(k)}{\mathbf{F}}_{N_t,s+1},\end{aligned} \tag{54}$$

$$\begin{aligned}&\left[\overset{(k)}{\boldsymbol{\beta}}_{0,s}\mathbf{D}+\overset{(k)}{\boldsymbol{\beta}}_{1,s}\right]\overset{(k)}{\mathbf{F}}_{i,s+1}+\overset{(k)}{\boldsymbol{\beta}}_{2,s}\sum_{j=0}^{N_t-1}d_{i,j}\overset{(k)}{\mathbf{F}}_{j,s+1}+\left[\overset{(k)}{\boldsymbol{\beta}}_{3,s}\mathbf{D}^2+\overset{(k)}{\boldsymbol{\beta}}_{4,s}\mathbf{D}+\overset{(k)}{\boldsymbol{\beta}}_{5,s}\right]\overset{(k)}{\boldsymbol{\Theta}}_{i,s+1}\\&+\overset{(k)}{\boldsymbol{\beta}}_{6,r}\sum_{j=0}^{N_t-1}d_{i,j}\overset{(k)}{\boldsymbol{\Theta}}_{j,s+1}=\overset{(k)}{\mathbf{R}}_{2,i,s}-\overset{(k)}{\boldsymbol{\beta}}_{2,s}d_{i,N_t}\overset{(k)}{\mathbf{F}}_{N_t,s+1}-\overset{(k)}{\boldsymbol{\beta}}_{6,s}d_{i,N_t}\overset{(k)}{\boldsymbol{\Theta}}_{N_t,s+1},\end{aligned} \tag{55}$$

or compactly as

$$\overset{(i)}{\mathbf{A}}_{1,1}\overset{(k)}{\mathbf{F}}_{i,s+1}+\overset{(k)}{\boldsymbol{\alpha}}_{4,s}\sum_{j=0}^{N_t}d_{i,j}\mathbf{D}\overset{(k)}{\mathbf{F}}_{j,s+1}+\overset{(k)}{\boldsymbol{\alpha}}_{5,s}\sum_{j=0}^{N_t}d_{i,j}\overset{(k)}{\mathbf{F}}_{j,s+1}+\overset{(i)}{\mathbf{A}}_{1,2}\overset{(k)}{\boldsymbol{\Theta}}_{i,s+1}=\overset{(k)}{\mathbf{B}}_{1,i,s}, \tag{56}$$

$$\overset{(i)}{\mathbf{A}}_{2,1}\overset{(k)}{\mathbf{F}}_{i,s+1}+\overset{(k)}{\boldsymbol{\alpha}}_{5,s}\sum_{j=0}^{N_t}d_{i,j}\overset{(k)}{\mathbf{F}}_{j,s+1}+\overset{(i)}{\mathbf{A}}_{2,2}\overset{(k)}{\boldsymbol{\Theta}}_{i,s+1}+\overset{(k)}{\boldsymbol{\beta}}_{6,s}\sum_{j=0}^{N_t}d_{i,j}\overset{(k)}{\boldsymbol{\Theta}}_{j,s+1}=\overset{(k)}{\mathbf{B}}_{2,i,s}, \tag{57}$$

where

$$\begin{aligned}&\overset{(i)}{\mathbf{A}}_{1,1}=\overset{(k)}{\boldsymbol{\alpha}}_{0,s}\mathbf{D}^3+\overset{(k)}{\boldsymbol{\alpha}}_{1,s}\mathbf{D}^2+\overset{(k)}{\boldsymbol{\alpha}}_{2,s}\mathbf{D}+\overset{(k)}{\boldsymbol{\alpha}}_{3,s},\quad \overset{(i)}{\mathbf{A}}_{1,2}=\overset{(k)}{\boldsymbol{\alpha}}_{6,s}\mathbf{I},\\&\overset{(i)}{\mathbf{A}}_{2,1}=\overset{(k)}{\boldsymbol{\beta}}_{0,s}\mathbf{D}+\overset{(k)}{\boldsymbol{\beta}}_{1,s},\quad \overset{(i)}{\mathbf{A}}_{2,2}=\overset{(k)}{\boldsymbol{\beta}}_{3,s}\mathbf{D}^2+\overset{(k)}{\boldsymbol{\beta}}_{4,s}\mathbf{D}+\overset{(k)}{\boldsymbol{\beta}}_{5,s},\\&\overset{(k)}{\mathbf{B}}_{1,i,s}=\overset{(k)}{\mathbf{R}}_{1,i,s}-\overset{(k)}{\boldsymbol{\alpha}}_{4,s}d_{i,N_t}\mathbf{D}\overset{(k)}{\mathbf{F}}_{N_t,s+1}-\overset{(k)}{\boldsymbol{\alpha}}_{5,s}d_{i,N_t}\overset{(k)}{\mathbf{F}}_{N_t,s+1},\\&\overset{(k)}{\mathbf{B}}_{2,i,s}=\overset{(k)}{\mathbf{R}}_{2,i,s}-\overset{(k)}{\boldsymbol{\beta}}_{2,s}d_{i,N_t}\overset{(k)}{\mathbf{F}}_{N_t,s+1}-\overset{(k)}{\boldsymbol{\beta}}_{6,s}d_{i,N_t}\overset{(k)}{\boldsymbol{\Theta}}_{N_t,s+1}.\end{aligned} \tag{58}$$

The boundary conditions given in Eq.(53) when evaluated at the collocation points, give

$$\begin{aligned}&\overset{(k)}{f}_{s+1}(\eta_r,\xi_i)=0,\quad \sum_{p=0}^{r}\mathbf{D}_{r,p}\overset{(k)}{f}_{s+1}(\eta_p,\xi_i)=0,\quad \overset{(k)}{\theta}_{s+1}(\eta_r,\xi_i)=1,\\&\sum_{p=0}^{r}\mathbf{D}_{0,p}\overset{(k)}{f}_{s+1}(\eta_p,\xi_i)=0,\quad \overset{(k)}{\theta}_{s+1}(\eta_0,\xi_i)=0.\end{aligned} \tag{59}$$

Here $\mathbf{F}$ and $\mathbf{\Theta}$ are vectors of size $r+1$ that are defined just as vector $\bar{\mathbf{U}}$ in Eq.(25) . The scheme described above can be extended to a system of three or more equations in a straightforward manner.

Example 5. For a system of three equations, we consider the problem of natural convection boundary layer flow, influenced by the combined buoyancy forces of mass and thermal diffusion from a permeable vertical flat surface with non-uniform surface temperature and non-uniform surface species concentration, but with a uniform rate of suction of fluid through the permeable surface. The problem is governed by the non-similarity system of equations given in dimensionless form in [40] as

$$f''' + \frac{n+3}{4} f f'' - \frac{n+1}{2} f'^2 + \xi f'' + (1-w)g + wh = \frac{1-n}{4}\xi \left[f' \frac{\partial f'}{\partial \xi} - f'' \frac{\partial f}{\partial \xi} \right], \tag{60}$$

$$\frac{1}{Pr} g'' + \frac{n+3}{4} f g' + \xi g' = \frac{1-n}{4}\xi \left[f' \frac{\partial g}{\partial \xi} - g' \frac{\partial f}{\partial \xi} \right], \tag{61}$$

$$\frac{1}{Sc} h'' + \frac{n+3}{4} f h' + \xi h' = \frac{1-n}{4}\xi \left[f' \frac{\partial h}{\partial \xi} - h' \frac{\partial f}{\partial \xi} \right]. \tag{62}$$

Here, f, g, h, are the non-dimensional stream function, temperature function and concentration function, respectively; η is the pseudo-similarity variable and ξ is the transpiration parameter and the primes denote differentiation with respect to η. The boundary conditions appropriate to the above equations are;

$$\begin{aligned} &f(0,\xi) = f'(0,\xi) = 0, \quad g(0,\xi) = h(0,\xi) = 1, \\ &f'(\infty,\xi) = g(\infty,\xi) = h(\infty,\xi) = 0. \end{aligned} \tag{63}$$

In Eq.(60), w is the measure of relative importance of solutal and thermal diffusion in causing the density changes that drive the flow.

4. Results and discussion

In this section, numerical results that are obtained after solving Examples 1 to 5 using the multidomain spectral collocation method are presented and discussed in tabular and graphical forms. The numerical schemes were implemented in MatLab 2017b and 5 iterations were used for the nonlinear problems. To demonstrate the superiority of the proposed method in terms of accuracy, absolute errors obtained with the multidomain approach are compared with the equivalent

absolute errors obtained when the spectral collocation method is applied on a single domain. The absolute errors are computed at selected grid points in x and t for Examples 1 to 3 and are defined by

$$E_{i,j} = \left| \overset{a}{U}(x_i, t_j) - U(x_i, t_j) \right|. \tag{64}$$

Here $\overset{a}{U}$ and U denote the spectral approximation and the analytic solution, respectively. Further, to show the advantages of adopting the multidomain approach, we determine the effect of increasing the number of subintervals in both x and t directions, as opposed to incrementing the number of grid points on a single large domain. This investigation is aimed at relating the size of the resulting linear system of equations to the condition numbers of the associated coefficient matrices and thereby predicting the magnitude of the CPU time that the proposed method of solution would take to realize results. The condition number of a matrix $\mathbf{A}$ is invoked using the MatLab built-in command $cond\,(\mathbf{A})$.

The numerical results obtained from spectral approximation of Eq.(29) using the multidomain spectral collocation method are given in Table 1. Results indicate that the method is very accurate, as shown by the absolute errors of order 10^{-14} being registered. The algorithm is executed over a short CPU time as results are generated within a fraction of a second.

The results obtained from approximating the solution of Eq.(33) are given in Tables 2 and 3. Table 2 shows the results generated when multidomain spectral collocation method is applied whereas Table 3 presents the results obtained when the single domain based approach namely, the bivariate spectral collocation method, is used. The number of grid points over the entire interval is kept constant. It is evident that the results obtained when using the multidomain approach are more accurate than those realized with the single domain approach, as approximation errors of order 10^{-12} and 10^{-08}, respectively, are recorded. Additionally, we notice that a shorter CPU time is required when the multidomain approach is adopted. This is attributed to the fact that decomposing the domain into smaller subintervals and consequently reducing the number of grid points per subinterval shrinks the size of the coefficient matrix at each subinterval and thus shorter time is needed to invert them. The use of a few numbers of grid points at each subinterval also reduces the effects of roundoff errors that are associated with approximating functions with an interpolating polynomial of higher degrees. Furthermore, small-sized matrices are also advantageous as they are well conditioned and this renders a well-posed problem ensuring that stable results are obtained [41].

Tables 4 and 5 give the numerical results obtained when approximating the solution of Eq.(37) using the proposed multidomain spectral collocation method. In Table 4, there are 5 subintervals and 20 grid points are used in the space variable x; for Table 5, we used 10 subintervals with 10 grid points each. Contrary to the results displayed in Table 2 and 3, in this particular case, the use of many subintervals deteriorates the accuracy of the spectral approximation. This can be attributed to the fact that the spatial domain in Example 3 is small, $b - a = 4$. In this particular case, increasing the number of subintervals at the expense of the number of grid points is not advantageous as it translates into approximating a function using an interpolating polynomial of a very low degree. However, we observe that in both cases the difference in sizes of the coefficient matrix is insignificant. This is because the size of the matrix is a function of both the number of spatial subintervals q and the number of grid points N_x at each of these subintervals. We remark that this unforeseen difference in matrix dimension is linked to the number of common nodes that are discarded when assembling the spatial differentiation matrix in the multidomain overlapping grids based approach.

For the boundary layer flow problem in Example 4, the results are presented in terms of the local skin friction and local Nusselt number for different values of ξ. The numerical values of local skin friction and local Nusselt number are depicted in Table 6 for $Pr = 0.7$, $n = 0$ and they show a similar trend to those obtained in [30]. We observe that the values of skin friction values decrease as ξ increases. From Table 6 it can be seen that the values of the local Nusselt number increase with the increasing values of the suction parameter ξ. The accuracy of the numerical scheme for a system of PDEs is assessed by considering the residual error profiles at different iteration levels. The residual error for f and θ equations is evaluated, respectively, as

$$|\mathbf{G}_{s+1}|_\infty, \quad \text{and} \quad |\mathbf{H}_{s+1}|_\infty, \tag{65}$$

where G and H are as defined in Eq.(45) when evaluated at the collocation points. The residual error profiles, as displayed in Figure 1, suggest that the numerical scheme is very accurate and that convergence occurs after the 4th iteration.

Finally, to dissect the numerical results for the boundary layer flow problem given in Example 5, the effect of the transpiration parameter ξ on the dimensionless velocity, $f'(\eta, \xi)$, the dimensionless temperature $g(\eta, \xi)$, and the dimensionless concentration $h(\eta, \xi)$ in the flow field is shown graphically. Figures 2 and 4 show, the values of the dimensionless velocity, temperature and concentration, respectively, against the similarity variable η for different values of the transpiration

parameter $\xi = 5, 10, 15, 20, 25$, and 30 with $Pr = 0.72$, $w = 1/3$, $n = 0$ and $Sc = 0.94$. We remark that the value $Sc = 0.94$ signifies the presence of carbon dioxide as the chemical species. From Figure 2 it can be observed that the velocity profiles decrease with an increase of transpiration parameter ξ. It can also be seen that for each value of the transpiration parameter ξ there exist local maximum values of velocity in the boundary layer region. From Figure 3 and Figure 4, we see that owing to increase of the values of transpiration parameter ξ both the temperature and concentration decrease. These results are in agreement with those of Hussain and Paul [40].

Table 1: Absolute error values obtained when Example 1 is solved on multiple domains: $N_x = 20,\ q = 5, N_t = 10, p = 10$.

		t			
x		2.0	4.0	6.0	8.0
0.0247		1.33227e-015	1.01030e-014	1.05471e-014	5.77316e-015
4.0934		4.25215e-014	1.15602e-014	2.21004e-015	8.50015e-016
8.2092		8.63754e-014	2.30649e-014	4.78437e-015	8.36570e-016
12.3690		1.29452e-014	3.40006e-014	6.84175e-015	1.22732e-015
16.5689		1.92735e-014	4.61853e-014	8.52096e-015	1.31492e-015
CPU time (sec)	0.612198	Matrix Size	970×970	Cond Number	6.6254e+003

Table 2: Absolute error values obtained when Example 2 is solved on multiple domains: $N_x = 20,\ q = 5, N_t = 10, p = 10$.

		t			
x		2.0	4.0	6.0	8.0
0.0247		1.27898e-013	1.55431e-014	3.65263e-014	4.66294e-015
4.0934		3.17268e-012	2.87881e-013	8.21565e-015	1.08802e-014
8.2092		2.57208e-012	1.17450e-012	6.35048e-014	6.88338e-015
12.3690		3.05943e-013	4.66849e-014	1.12266e-012	2.50910e-014
16.5689		1.95508e-014	5.72674e-013	3.03307e-012	1.06271e-012
CPU time (sec)	7.304909	Matrix Size	970×970	Cond Number	7.2791e+003

Table 3: Absolute error values obtained when Example 2 is solved on a single domain: $N_x = 100,\ q = 1, N_t = 100, p = 1$.

		t			
x		2.0	4.0	6.0	8.0
0.0247		6.41265e-009	5.62239e-009	1.34481e-009	2.44027e-009
4.0934		8.37108e-011	4.13558e-010	1.26998e-012	3.33289e-010
8.2092		3.05311e-012	4.10783e-012	8.32667e-012	1.10125e-008
12.3690		4.75314e-012	2.85327e-011	6.15064e-011	1.43219e-011
16.5689		3.45154e-011	9.77239e-011	2.90989e-010	7.44960e-011
CPU time (sec)	453.126036	Matrix Size	10100×10100	Cond Number	5.4326e+008

Table 4: Absolute error values obtained when Example 3 is solved on multiple domains: $N_x = 20,\ q = 5, N_t = 10, p = 5$.

		t			
x		0.2	0.4	0.6	0.8
1.4649		9.87432e-013	4.31388e-012	1.21436e-012	4.47731e-012
2.3252		4.10338e-013	3.31291e-013	2.75113e-013	5.81757e-014
3.1825		5.83977e-014	4.68514e-014	1.26343e-013	3.93685e-013
4.0353		7.18536e-013	9.76996e-015	1.01830e-012	2.28217e-012
4.8823		3.25540e-012	1.29252e-012	9.20575e-012	3.28981e-012
CPU time (sec)	3.919302	Matrix Size	970×970	Cond Number	2.1610e+005

Table 5: Absolute error values obtained when Example 3 is solved on multiple domain: $N_x = 10,\ q = 10, N_t = 10, p = 5$.

		t			
x		0.2	0.4	0.6	0.8
1.4649		4.80942e-011	2.47637e-010	1.61166e-009	1.50184e-008
2.3252		7.77756e-012	4.67542e-011	2.63619e-010	2.00697e-009
3.1825		2.47358e-013	3.48788e-012	2.01479e-011	1.24099e-010
4.0353		7.79377e-014	7.28306e-014	1.23390e-012	8.24474e-012
4.8823		3.01759e-013	1.22102e-012	1.64313e-013	4.37206e-013
CPU time (sec)	3.379178	Matrix Size	920×920	Cond Number	5.1185e+004

Table 6: Numerical values of local skin friction $f''(0,\xi)$, and local Nusselt number $-\theta'(0,\xi)$ for different values of ξ for Example 4 using $N_x = 50,\ \ q = 5,\ \ N_t = 5,\ \ p = 40,\ \ Lx = 20,\ \ , Pr = 0.7,\ \ n = 0.5,$ Iterations$= 5$

	Multiple Domains Solution	
ξ	$f''(0,\xi)$	$-\theta'(0,\xi)$
5	0.2842719	3.5066677
10	0.1428115	7.0008399
15	0.0952321	10.5002490
20	0.0714271	14.0001050
25	0.0571424	17.5000538
30	0.0476189	21.0000311

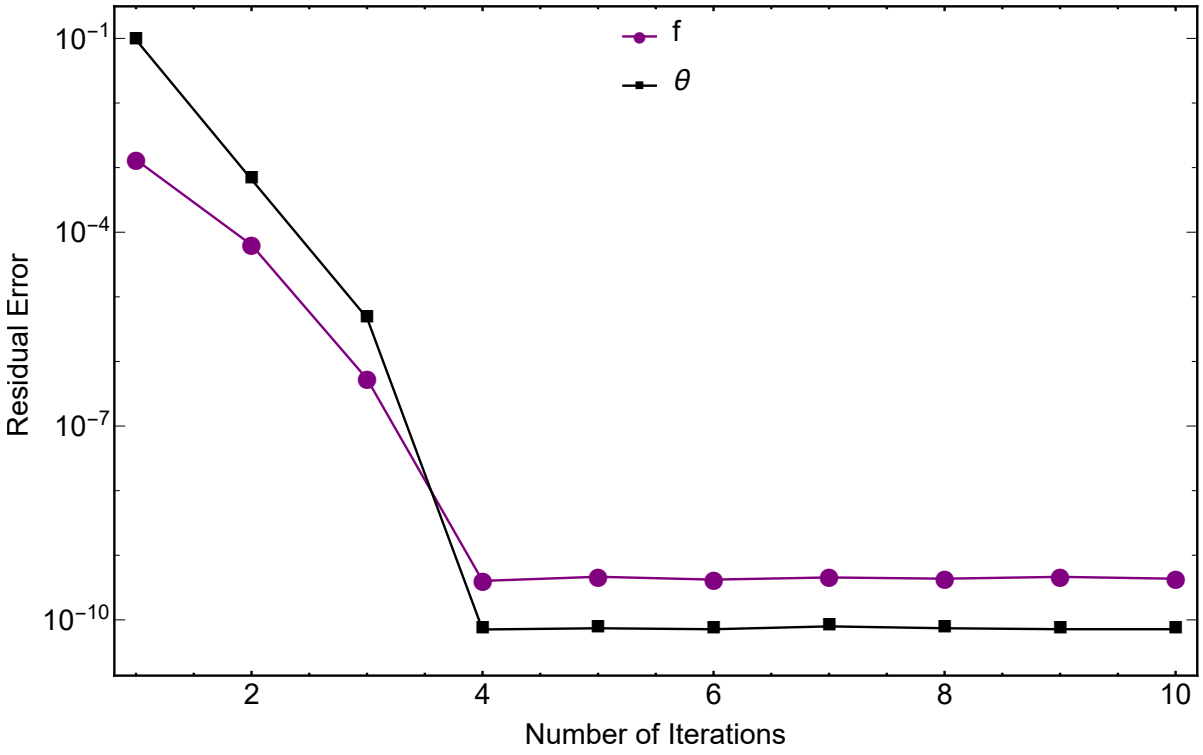


Figure 1: Residual error profile against the number of iterations

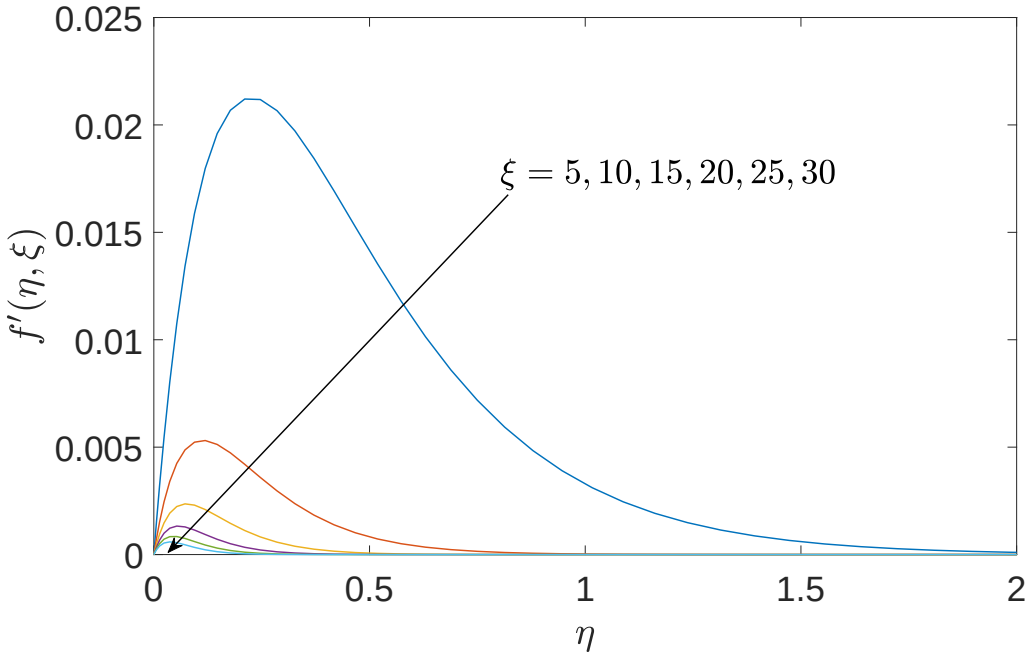


Figure 2: Velocity profile at different values of ξ

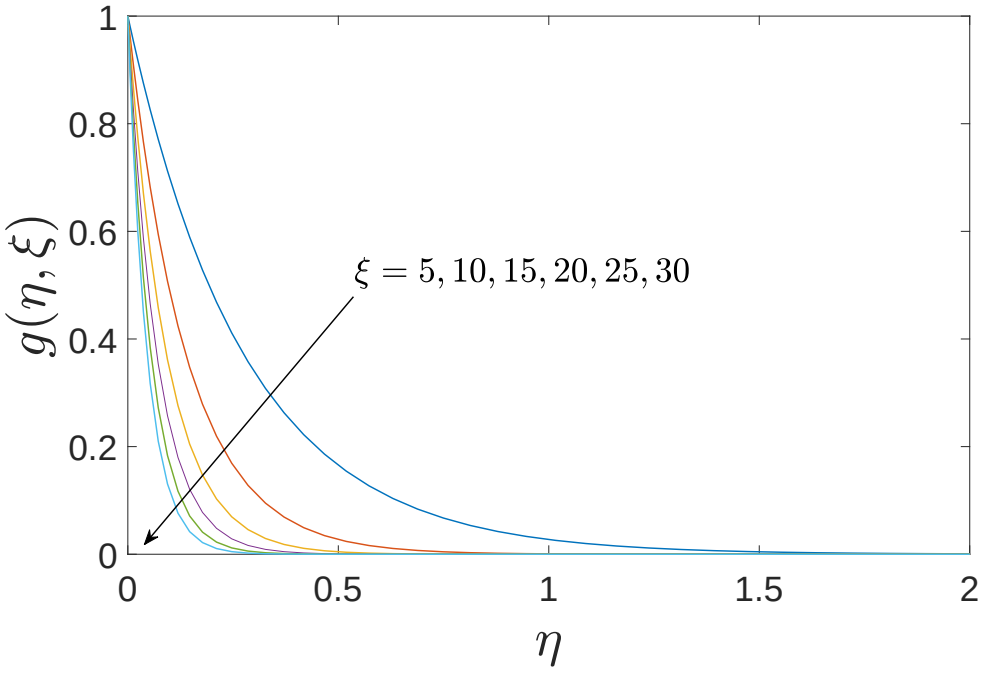


Figure 3: Temperature profile at different values of ξ

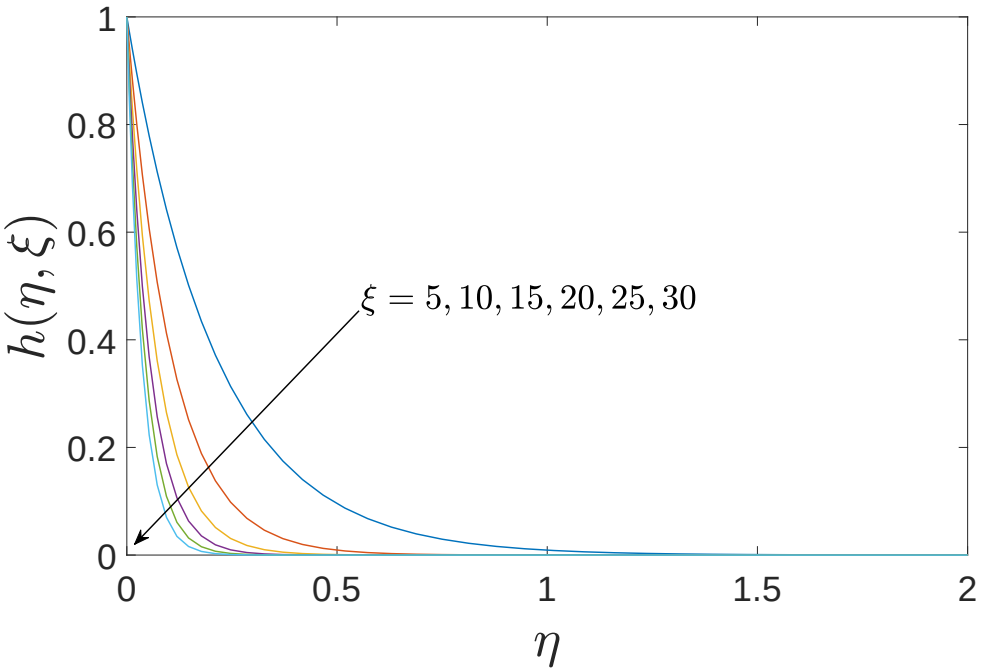


Figure 4: Concentration profile at different values of ξ

4.1. Error bounds theorems in bivariate polynomial interpolation

In this subsection, we present new error bound theorems and proofs emanating from bivariate polynomial interpolation using Chebyshev-Gauss-Lobatto nodes on single and decomposed domains. The error bound theorems given herein form the basis for theoretical argument as to why multidomain approaches are most suitable when approximating the solutions of partial differential equations that are defined over large domains. We note that CGL nodes are the relative extremes of the N_x-th degree Chebyshev polynomial of the first kind, $T_{N_x}(\hat{x}) = \cos\left[N_x \arccos(\hat{x})\right]$, $\hat{x} \in [-1, 1]$. To the best of our knowledge, there does not exist a well-known family of polynomials whose roots are the CGL nodes. However, if we think of the interior CGL nodes as the roots of $T'_{N_x}(\hat{x}) = 0$, one would recognize that a complete set of the CGL nodes can be obtained from the roots of the $(N_x + 1)^{th}$ degree polynomial given by

$$L_{N_x+1}(\hat{x}) = (1 - \hat{x}^2)T'_{N_x}(\hat{x}). \tag{66}$$

Below, we state a theorem that acts as a benchmark in formulating error bound theorems on bivariate polynomial interpolation.

Theorem 1. *[42] Let $u(x,t) \in C^{N_x+N_t+2}([a,b]\times[0,T])$ be sufficiently smooth such that at least the $(N_x+1)^{th}$ partial derivative with respect to x, $(N_t+1)^{th}$ partial derivative with respect to t and $(N_x+N_t+2)^{th}$ mixed partial derivative with respect to x and t exists and are all continuous, then there exist values $\xi_x, \xi_x' \in (a,b)$, and $\xi_t, \xi_t' \in (0,T)$, such that*

$$\begin{aligned} u(x,t) - U(x,t) = &\frac{\partial^{N_x+1}u(\xi_x,t)}{\partial x^{N_x+1}(N_x+1)!}\prod_{i=0}^{N_x}(x-x_i) + \frac{\partial^{N_t+1}u(x,\xi_t)}{\partial t^{N_t+1}(N_t+1)!}\prod_{j=0}^{N_t}(t-t_j) \\ &- \frac{\partial^{N_x+N_t+2}u(\xi_x',\xi_t')}{\partial x^{N_x+1}\partial t^{N_t+1}(N_x+1)!(N_t+1)!}\prod_{i=0}^{N_x}(x-x_i)\prod_{j=0}^{N_t}(t-t_j), \end{aligned} \tag{67}$$

where $U(x,t)$ is a bivariate interpolating polynomial of $u(x,t)$ at $\{x_i\}_{i=0}^{N_x}$ grid points in x-variable and $\{t_j\}_{j=0}^{N_t}$ grid points in t-variable.

A result similar to Eq.(67) was reproduced by Bhrawy in [43]. Taking the absolute value of Eq.(67) we obtain

$$\begin{aligned} |u(x,t) - U(x,t)| \leq &\max_{(x,t)\in\Omega}\left|\frac{\partial^{N_x+1}u(\xi_x,t)}{\partial x^{N_x+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x-x_i)\right|}{(N_x+1)!} + \max_{(x,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x,\xi_t)}{\partial t^{N_t+1}}\right|\frac{\left|\prod_{j=0}^{N_t}(t-t_j)\right|}{(N_t+1)!} \\ &+ \max_{(x,t)\in\Omega}\left|\frac{\partial^{N_x+N_t+2}u(\xi_x',\xi_t')}{\partial x^{N_x+1}\partial t^{N_t+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x-x_i)\right|\left|\prod_{j=0}^{N_t}(t-t_j)\right|}{(N_x+1)!(N_t+1)!}, \end{aligned} \tag{68}$$

where $\Omega = [a,b]\times[0,T]$. Since the function $u(x,t)$ is assumed to be smooth on the interval of approximation, it follows that its derivatives are bounded and thus $\exists$ constants C_1, C_2 and C_3, such that

$$\max_{(x,t)\in\Omega}\left|\frac{\partial^{N_x+1}u(x,t)}{\partial x^{N_x+1}}\right| \leq C_1,\ \max_{(x,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x,t)}{\partial t^{N_t+1}}\right| \leq C_2,\ \max_{(x,t)\in\Omega}\left|\frac{\partial^{N_x+N_t+2}u(x,t)}{\partial x^{N_x+1}\partial t^{N_t+1}}\right| \leq C_3. \tag{69}$$

4.1.1. Error bound theorem on a single domain

The error bound for bivariate polynomial interpolation using Gauss-Lobatto nodes on a single domain is governed by the theorem below,

Theorem 2 (**Error bound in a single domain**). *The resulting error bound when CGL grid points $\{x_i\}_{i=0}^{N_x} \in [a,b]$, in x-variable, and $\{t_j\}_{j=0}^{N_t} \in [0,T]$, in t-variable, are used in bivariate polynomial interpolation is given by*

$$E(x,t) \leq C_1\frac{8\left(\frac{b-a}{4}\right)^{N_x+1}}{(N_x+1)!} + C_2\frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!} + C_3\frac{8^2\left(\frac{b-a}{4}\right)^{N_x+1}\left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_t+1)!}. \tag{70}$$

Proof. First, using the relation stated in [44] we express Eq.(66) as

$$L_{N_x+1}(\hat{x}) = (1-\hat{x}^2)T'_{N_x}(\hat{x}) = -N_x\hat{x}T_{N_x}(\hat{x}) + N_xT_{N_x-1}(\hat{x}). \tag{71}$$

Using the triangle inequality and noting that $|T_{N_x}(\hat{x})| \leq 1,\ \forall\ \hat{x} \in [-1,1]$, we have

$$|L_{N_x+1}(\hat{x})| = |-N_x\hat{x}T_{N_x}(\hat{x}) + N_xT_{N_x-1}(\hat{x})| \leq |-N_x\hat{x}T_{N_x}(\hat{x})| + |N_xT_{N_x-1}(\hat{x})| \leq 2N_x. \tag{72}$$

The leading coefficient of $L_{N_x+1}(\hat{x})$ is $2^{N_x-1}N_x$, where the components 2^{N_x-1} and N_x come, respectively, from the leading coefficient of $T_{N_x}(\hat{x})$ and the application of N_x-th rule of differentiation on $T_{N_x}(\hat{x})$. The product factor in the first term of the error bound expression given at Eq.(68) can therefore be taken as the factorized form of monic polynomial $\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}$. We write,

$$\prod_{i=0}^{N_x}(\hat{x}-\hat{x}_i) = \frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x},\ \ \hat{x} \in [-1,1]. \tag{73}$$

Using Eq.(72), it is easy to establish that the monic polynomial Eq.(73) is bounded by

$$\left|\prod_{j=0}^{N_x}(x-\hat{x}_i)\right| = \left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right| \leq \frac{2N_x}{2^{N_x-1}N_x} = \frac{4}{2^{N_x}}. \tag{74}$$

Considering a general interval $x \in [a,b]$, we can show that the first product factor in Eq.(68) is bounded by

$$\begin{aligned}\max_{a\leq x\leq b}\left|\prod_{i=0}^{N_x}(x-x_i)\right| &= \max_{-1\leq\hat{x}\leq 1}\left|\prod_{i=0}^{N_x}\frac{(b-a)}{2}(\hat{x}-\hat{x}_i)\right| = \left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\leq\hat{x}\leq 1}\left|\prod_{i=0}^{N_x}(\hat{x}-\hat{x}_i)\right| \\ &= \left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\leq\hat{x}\leq 1}\left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right| \leq \frac{4\left(\frac{b-a}{2}\right)^{N_x+1}}{2^{N_x}} = 8\left(\frac{b-a}{4}\right)^{N_x+1}.\end{aligned} \tag{75}$$

Similarly, the second product factor is bounded above by

$$\max_{0\leq t\leq T}\left|\prod_{j=0}^{N_t}(t-t_j)\right| = \left(\frac{T}{2}\right)^{N_t+1}\max_{-1\leq\hat{t}\leq 1}\left|\frac{L_{N_t+1}(\hat{t})}{2^{N_t-1}N_t}\right| \leq \frac{4\left(\frac{T}{2}\right)^{N_t+1}}{2^{N_t}} = 8\left(\frac{T}{4}\right)^{N_t+1}. \tag{76}$$

Using Eq.(75), Eq.(76), and Eq.(69) in Eq.(68) the proof is completed. □

4.1.2. Error bound theorem on multiple domains

Here, we extend the bivariate polynomial interpolation error bound Theorem 2 to obtain its variant on a decomposed domain. In the description, it is assumed that the number of grid points is the same for all subintervals in either space or time direction.

Theorem 3 (**Error bound in the decomposed domain**). *The error bound when Chebyshev Gauss-Lobatto grid points* $\{x_i\}_{i=0}^{N_x} \in [x_{l-1}, \bar{x}_l],\ l = 1, 2, \ldots, q$, *for the decomposed domain in* x*-variable, and* $\{t_j\}_{j=0}^{N_t} \in [t_{k-1}, t_k],\ k = 1, 2, \ldots, p$, *for the decomposed domain in* t*-variable, are used in bivariate polynomial interpolation is given by*

$$E(x,t) \leq C_1 \frac{8\left(\frac{L}{4}\right)^{N_x+1}}{(N_x+1)!} + C_2 \frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1} + C_3 \frac{8^2\left(\frac{L}{4}\right)^{N_x+1}\left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1}. \tag{77}$$

Proof. First, we consider the t variable. In the entire domain $[0, T]$, we have that

$$\left|\prod_{j=0}^{N_t}(t-t_j)\right| \leq 8\left(\frac{T}{4}\right)^{N_t+1},\ \ t \in [0,T]. \tag{78}$$

The implication is that in the decomposed domain and at each subinterval, we must have

$$\left|\prod_{j=0}^{N_t}(t-t_j)\right| \leq 8\left(\frac{T}{4p}\right)^{N_t+1} = 8\left(\frac{T}{4}\right)^{N_t+1}\left(\frac{1}{p}\right)^{N_t+1},\ \ t \in [t_{k-1}, t_k]. \tag{79}$$

For smooth u, there exists $\xi_\mu \in (t_{\mu-1}, t_\mu)$, $\mu = 1, 2, \ldots, p$, for which the values of $(N_t+1)^{th}$ partial derivative of u with respect to t in each subinterval is the absolute extrema. This enables us to break the second term $C_2 \frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}$, which appears in the error bound expression at Eq.(70), into different components that are necessarily not equal in the decomposed domain as

$$\left\{ \overset{(k)}{C}_2 \frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1} \right\}_{k=1}^{p}, \tag{80}$$

where

$$\max_{(x,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x,t)}{\partial t^{N_t+1}}\right| = \left|\frac{\partial^{N_t+1}u(x,\xi_k)}{\partial t^{N_t+1}}\right| \leq \overset{(k)}{C}_2,\ \ t \in [t_{k-1}, t_k].$$

We define

$$\| \hat{C}_2 \|_\infty \equiv \max\{\overset{(1)}{C}_2, \overset{(2)}{C}_2, \ldots, \overset{(p)}{C}_2\}, \tag{81}$$

to denote the maximum absolute value of $(N_t+1)^{th}$ partial derivative of u with respect to t in $[0, T]$. Clearly, $\| \hat{C}_2 \|_\infty = C_2$, where C_2 is identical to the one given at Eq.(70). To expand the error bound over the entire t domain, we shall take the largest possible error across all subintervals in t, which is

$$C_2 \frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1}. \tag{82}$$

In a similar manner, we can show that the first component in the error bound Eq.(70) in the decomposed domain translates to

$$C_1 \frac{8\left(\frac{L}{4}\right)^{N_x+1}}{(N_x+1)!}, \tag{83}$$

where L is the length of each subinterval in space variable, as defined at Eq.(14), and $\left(\frac{L}{4}\right)^{N_x+1} << \left(\frac{b-a}{4}\right)^{N_x+1}$ for a large number of subintervals q. Consequently, the third component in Eq.(70) becomes

$$C_3 \frac{8^2 \left(\frac{L}{4}\right)^{N_x+1} \left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_t+1)!} \left(\frac{1}{p}\right)^{N_t+1}, \tag{84}$$

in the multi-domain approach. Using the results at Eq.(82), Eq.(83), and Eq.(84) in Eq.(70) completes the proof. □

Comparing Eq.(70) and Eq.(77) we note that the error in bivariate polynomial interpolation is smaller when interpolation is conducted on multiple domains than when conducted on a single domain. We reveal that when using spectral collocation methods to solve initial-boundary value problems, CGL nodes are preferable candidates for interpolation, because they are convenient in constructing differentiation matrices due to their containing the boundary nodes. This is advantageous when treating the boundary conditions of the problem.

5. Conclusion

In this work, we have described the multidomain spectral collocation method for solving parabolic PDEs that are defined on large space and time intervals. The superiority of the current method has been demonstrated by presenting numerical results for well known evolutionary parabolic PDEs that are defined over large domains. It has been confirmed that the results obtained using the multidomain spectral collocation method are very accurate and such accuracy is realized with short CPU time. This contrasts with, a single domain approach, which leads to accuracy deterioration when the computational domain is large. The short CPU time that is required in the multidomain approach is linked to the small-sized matrices, which are easy to invert. Improved accuracy is attributed to the small condition numbers of the coefficient matrices. The present results have also been compared against results in the literature and a good agreement is observed. Further, to show that the proposed method is well suited for problems defined over large domains, we have applied the multidomain spectral collocation to a problem defined over a small interval and the results suggest that the multidomain approaches are not fit for such applications. Owing to the performance reliability of the proposed method, the general numerical scheme can be extended to solve many practical problems in real life that are modeled by parabolic PDEs defined on large intervals, as demonstrated by the case of boundary layer flow problems. The preference for the

CGL nodes as the choice of interpolation nodes have been backed up by rigorous proofs of new error bound theorems.

6. Acknowledgement

The authors are thankful to the University of KwaZulu-Natal for providing essential research resources.

Declaration of interest:

None

References

[1] G.B. Folland, *Introduction to Partial Differential Equations*, Princeton: Princeton University Press, 1995.

[2] A.H. Khater, and R.S. Temsah, *Numerical Solutions of Some Nonlinear Evolution Equations by Chebyshev Spectral Collocation Methods*, Computer Mathematics, 84(2007), pp. 326-339.

[3] V. Chandraker, A. Awasthi, and S. Jayaraj, *Numerical Treatment of Burgers-Fisher equation*, Procedia Technology, 25(2016), pp. 1217-1225.

[4] T. Kawahara, and M. Tanaka, *Interactions of Traveling Fronts: an Exact Solution of a Nonlinear Diffusion Equations*, Physics Letters, 97(1983), pp. 3-11.

[5] W. Wang, and A.J. Roberts, *Diffusion Approximation for Self-similarity of Stochastic Advection in Burger's Equation*, Communications in Mathematical Physics, 5(2014), pp. 37-48.

[6] R. FitzHugh, *Mathematical Models of Threshold Phenomena in the Nerve Membrane*, Bulletin of Mathematical Biophysics, 17(1955), pp. 257-278.

[7] H. Li, and Y. Guo, *New Exact Solutions to the Fitzhugh-Nagumo Equation*, Applied Mathematics and Computation, 18(2006), pp. 524-528.

[8] M.C. Nucci, and P.A. Clarkson, *The Nonclassical Method is More General Than the Direct Method for Symmetry Reductions. An Example of the Fitzhugh-Nagumo Equation*, Physics Letters A, 164(1992), pp. 49-56.

[9] R. FitzHugh, *Impulse and the Physiological States in Models of Nerve Membrane*, Biophysical Journal, 1(1961), pp. 445-466.

[10] H.T. Lin, and W.S. Yu, *Combined Heat and Mass Transfer by Laminar Natural Convection From a Vertical Plate With Uniform Heat Flux and Concentration*, Heat & Mass Transfer, 32(1997), pp. 293-299.

[11] K.R. Khair, and A. Bejan, *Mass Transfer to Natural Convection Boundary Layer Flow Driven by Heat Transfer*, Journal of Heat Transfer, 107(1985), pp. 979-986.

[12] A. Mongruel, M. Cloitre, and C. Allain, *Scaling of Boundary-Layer Flows Driven by Double Diffusive Convection*, International Journal of Heat & Mass Transfer, 39(1996), pp. 3899-3910.

[13] A.M. Wazwaz, *The Tanh Method for Generalized Forms of Nonlinear Heat Conduction and Burgers-Fisher Equations*, Applied Mathematics and Computation, 169(2005), pp. 321-338.

[14] H.N.A. Ismail, K. Raslan, and A.A.A. Rabboh, *Adomian Decomposition Method for Burger's-Huxley and Burger's-Fisher's Equations*, Applied Mathematics and Computation, 159(2004), pp. 291-301.

[15] I. Hashim, M.S.M. Noorani, and M.R.S. Al-Hadidi, *Solving the Generalized Burger's-Huxley Equation Using the Adomian Decomposition Method*, Mathematical and Computer Modelling, 16(2006), pp. 11-19.

[16] J.H. He, *Homotopy Perturbation Method for Bifurcation of Nonlinear Problems*, Nonlinear Sciences and Numerical Simulation, 6(2005), pp. 27-33.

[17] A.A. Soliman, and M.A. Abdou, *Numerical Solutions of Nonlinear Evolution Equations Using Variational Iteration Method*, Computational and Applied Mathematics, 207(2007), pp. 111-120.

[18] S.A. Manaa, F.H. Easif, and A.S. Faris, *The Finite Difference Methods for FitzHugh-Nagumo Equation*, IOSR Journal of Mathematics, 11(2015), pp. 51-55.

[19] M.Y. Hussain, and T.A. Zang, *Spectral Methods in Fluid Dynamics*, Annual Review of Fluid Mechanics, 19(1987), pp. 339-367.

[20] M. Javidi, *Spectral Collocation Method for the Solution of the Generalized Burger's-Fisher's equation*, Applied Mathematics and Computation, 174(2006), pp. 345-352.

[21] A. Golbabai, and M. Javidi, *A Spectral Domain Decomposition Approach for the Generalized Burger's-Fisher equation*, Chaos, Solitons & Fractals, 39(2009), pp. 385-392.

[22] E. Babolian, and J. Saeidian, *Analytic Approximate Solutions to Burgers, Fisher, Huxley Equations and Two Combined Forms of these Equations*, Communications in Nonlinear Science and Numerical Simulation, 14(2009), pp. 1984-1992.

[23] D. Kaya, and S.M. El-Sayed, *A Numerical Simulation and Explicit Solutions of the Generalized Burger's-Fishers Equation*, Applied Mathematics and Computation, 152(2004), pp. 403-413.

[24] R.E. Mickens, and A.B. Gumel, *Construction and Analysis of a Non-standard Finite Difference Scheme for the Burger's-Fisher's Equation*, Journal of Sound and Vibration, 257(2002), pp. 791-797.

[25] T. Zhao, C. Li, Z. Zang, and Y. Wua, *Chebyshed Legendre Pseudospectral Method for the Generalised Burger's-Fisher's Equation*, Applied Mathematical Modelling, 36(2012), pp. 1046-1056.

[26] P. Gordon, *Nonsymmetric Difference Equations*, Journal of the Society For Industrial and Applied Mathematics, 13(1965), pp. 667-673.

[27] A.R. Gourlay, *"Hopscotch": A Fast Second Order Partial Differential Equation Solver*, Journal of the Institute of Mathematics and its Applications, 6(1970), pp. 375-390.

[28] D. Olmos, and B.D. Shizgal, *Pseudospectral Method of Solution of the Fitzhugh-Nagumo Equation*, Mathematics and Computers in Simulation, 77(2009), pp. 2258-2278.

[29] R. Jiwari, R.K. Gupta, and V. Kumar, *Polynomial Differential Quadrature Method for Numerical Solutions of the Generalized Fitzhugh-Nagumo Equation with Time-Dependent Coefficients*, Ain Shams Engineering Journal, 5(2014), pp. 1343-1350.

[30] S. Hussain, M.A. Hossain and M. Wilson, *Natural Convection Flow From a Vertical Permeable Flat Plate With Variable Surface Temperature and Species Concentration*, Engineering Computations, 17(2000), pp. 789-812.

[31] H.T. Lin, and W.S. Yu, *Combined Heat and Mass Transfer by Laminar Natural Nonvection Flow From a Vertical Plate*, Heat & Mass Transfer, 30(1995), pp. 369-376.

[32] L.N. Trefethen, *Spectral Methods in MATLAB*, SIAM, Philadelphia, 2000.

[33] E. Tadmor, *Convergence of Spectral Methods for Conservative Laws*, SIAM, Journal of Numerical Analysis, 26(1989), pp. 30-44.

[34] M.R. Malik, T.A. Zang, and M.Y. Hussaini, *A Spectral Collocation Method for the Navier-Stokes Equations*, Computational Physics, 61(1985), pp. 64-88.

[35] B. Costa, *Spectral Methods for Partial Differential Equations*, CUBO: A Mathematical Journal, 6(2004), pp. 1-32.

[36] S.S. Motsa, V.M. Magagula, and P. Sibanda, *A Bivariate Chebyshev Spectral Collocation Quasilinearization Methods for Nonlinear Evolution Parabolic Equations*, The Scientific World Journal, 20(2014), pp. 251-260.

[37] R.E. Bellman, and R.E. Kalaba, *Quasilinearization and Nonlinear Boundary-Value Problems*, Elsevier Publishing Company, New York, 1965.

[38] R. Baltensperger, and J.P. Berrut, *The Errors in Calculating the Pseudospectral Differentiation Matrices for Chebyshev-Gauss-Lobatto points*, Computers and Mathematics with Applications, 37(1999), pp. 41-48.

[39] S.S. Motsa, F.M. Samuel, and S. Shateyi (2016), *Solving Nonlinear Parabolic Partial Differential Equations Using Multidomain Bivariate Spectral Collocation Method*, Nonlinear Systems - Design, Analysis, Estimation and Control, In D. Lee (Ed.), InTech, DOI: 10.5772/64600

[40] M.A. Hossain, and S. C. Paul, *Free Convection From a Vertical Permeable Circular Cone With Non-uniform Surface Temperature*, Acta Mechanica, 151(2001), pp. 103-114.

[41] J.F. Narcowich, and J.D. Ward, *Norms of Inverses and Condition Numbers of Matrices Associated with Scattered Data*, Journal of Approximation Theory, 64(1991), pp. 69-94.

[42] [42] M. Gasca and T. Sauer, *On the History of Multivariate Polynomial Interpolation*, Computational and Applied Mathematics, 122(2001), pp. 135-147.

[43] [43]A.H. Bhrawy, *A Highly Accurate Collocation Algorithm for* $1+1$ *and* $2+1$ *Fractional Percolation Equations*, Journal of Vibration and Control, 5(2015), pp. 1-23.

[44] H.E. Salzer, *Converting Interpolation Series into Chebyshev Series by Recurrence Formulas*, Mathematics of Computation, 30(1976), pp. 295-302.

Chapter 5

A Modified Spectral Collocation Method of Solution for Two-Dimensional Elliptic PDEs

In this chapter, we develop a spectral collocation solution-based approach for solving two-dimensional elliptic PDEs. This is an extension of the discussion of the solution techniques for one-dimensional PDEs considered in Chapters 3 and 4. The solution algorithm involves the decomposition of a large computational domain in each space variable into a sequence of overlapping grids. We remark that the current method of solution is applicable to solving many classes of two-dimensional elliptic PDEs, including those in two-dimensional steady fluid dynamics problems formulated on rectangular cavities. However, in the present study, the proposed method is tested on typical examples of elliptic PDEs that have been reported in the literature. The numerical results show a good agreement. Error bound theorems presented in this chapter with proofs, due to the decomposition of the spatial domains confirm the theory explaining the benefit of the proposed numerical method of solution. Numerical results show that the proposed method of solution is accurate and computationally efficient when applied to solve nonlinear elliptic PDEs defined on large spatial domains.

A powerful overlapping grids based spectral collocation solution technique for two-dimensional nonlinear elliptic partial differential equations

F.M. Samuel[1,a,c], S.S. Motsa[1,a,b]

[a] *University of KwaZulu-Natal, School of Mathematics, Statistics, and Computer Science, Private Bag X01, Scottsville, 3209, South Africa;*
[b] *University of Swaziland, Department of Mathematics, Private Bag 4, Kwaluseni, Swaziland;*
[c] *Taita Taveta University, Department of Mathematics, 635-80300, Voi, Kenya.*

Abstract

Spectral collocation methods on a single domain have been proven effective for solving nonlinear partial differential equations (PDEs) defined on small domains but are inefficient when applied to solve problems defined on large domains. In this article, an overlapping grid based spectral collocation method, which addresses this limitation, is presented and applied to solve two-dimensional nonlinear PDEs. The problems considered are nonlinear elliptic PDEs defined on large rectangular domains. The aim is to demonstrate that solving this class of problems on overlapping grids yields highly accurate results in a more computationally efficient manner than would the application of a similar approach on a single domain. The current solution approach involves decomposing the domain of approximation into smaller overlapping subintervals of equal length in each space variable. The PDE is linearized using the quasi-linearization method and bivariate Lagrange interpolating polynomials are employed to represent the solution in each subinterval. Subsequently, discretization is performed on each subinterval using Chebyshev-Gauss-Lobatto (CGL) nodes. Removal of repeated equations at overlapping regions leads to a linearly independent system of algebraic of equations that are solved iteratively. We compare our numerical solutions with those obtained using a single domain spectral collocation method and display results in the tabular and graphical form. Current numerical results show a good agreement with results published in the literature. Finally, error bound theorems and proofs have been presented to unfold the theory behind the benefits of the proposed numerical method of solution.

Keywords: 2D Nonlinear elliptic PDEs, rectangular domains, spectral collocation, overlapping grids, error bounds, CGL points.

2010 MSC: 65M15, 65M55, 65M70.

*Corresponding author: felexmutua@gmail.com

1. Introduction

Elliptic partial differential equations have numerous applications in almost all areas of mathematics ranging from harmonic analysis to Lie theory, as well as numerous applications in physics. Elliptic PDEs describe phenomena that do not change with time. Many processes in the applied sciences that are modeled by elliptic equations present nonlinear problems. However, as reported by Fedoseyev et al. [1] and in the book on differential equations by Polyanin and Zaitsev [2], nonlinear partial differential equations, and systems of them, exhibit a number of properties associated with an essential to important features of real-world phenomena, which the mathematical model is supposed to describe; these complexities lead to new difficulties in the mathematical treatments of the equations [3]. When dealing with differential equations, the most challenging task is to find explicit formulas for their solution. However, this can only be done in the simplest situations, which excludes the strongly nonlinear elliptic PDEs encountered in many real-life applications. The desire to understand the solutions of these equations has inspired mathematicians to successfully develop a diverse spectrum of numerical methods for solutions that give sufficiently rigorous answers to important questions of the nonlinear world [4]. Over time, new methods have given some impressive results in this area, which has motivated the quest for even more superior methods of solutions. Notably, at the onset of designing a numerical algorithm for nonlinear differential equations, the most remarkable features to consider include; the accuracy requirements, the ease of design for the algorithm and the computational complexity involved during implementation.

In the literature, there are three basic numerical approaches for solving nonlinear elliptic PDEs; namely, finite differences, finite elements, and the spectral collocation methods [5]. Finite difference methods use many subdomains and find the discrete solution on a grid by expanding the solution to low order in each subdomain. Finite difference methods are simple and the coding is easy, but they are difficult to work with on complicated domains [6]. Avellaneda et al. [7] applied the finite difference method to solve elliptic PDEs with rapidly oscillating coefficients. Later, Conca and Natesan [8] provided a better approximation to the solution of these elliptic problems using a numerical method based on the Bloch wave approach while Chan et al. [9] solved two-dimensional nonlinear obstacle problems using a generalized finite difference method and there exists a vast literature on the application of finite difference methods to nonlinear elliptic PDEs in work by Jensen [10]. Finite element methods, by contrast, are based on partitioning the domain into small

finite elements followed by expansion of the solution in basis functions [11]. Finite elements methods are particularly well suited to the irregular domains that appear in many engineering applications; however, they are more complex to set up and analyze. Hou and Wu [12] introduced a multi-scale finite element method for elliptic problems in composite materials and porous media. Matatche et al. [13] proposed a p-finite element method for numerically approximating the solution of nonlinear elliptic PDEs. Spectral collocation methods approximate solutions to nonlinear PDEs on a single domain or few subdomains with higher expansion orders of the basis functions. They are generally fast and highly accurate for problems with smooth solutions. They are, however, not useful on irregular domains or for problems with discontinuities. Ghimire et al. [14] presented a pseudo-spectral collocation-based approach to solutions of nonlinear elliptic PDEs using Chebyshev basis functions. In [15], Yi and Wang proposed a Legendre-Gauss-type collocation algorithm for solving nonlinear partial differential equations and demonstrated its high accuracy and effectiveness.

Since their inception, spectral collocation methods have gained popularity for the numerical approximation of the solution of nonlinear PDEs, owing to their superior accuracy. One important aspect that has been reported in the literature regarding the development of efficient numerical schemes for solving the boundary value problem using spectral collocation-based methods is the size of the computational domain. It has been observed that, when the size of the computational domain is large, many grid points are required to achieve stringently accurate results [16]. The use of many grid points requires a significant amount of computer memory and CPU time. However, as highlighted by Don and Solomonoff [17], the use of a large number of grid points does not always guarantee improved accuracy. Indeed, the opposite happens because increasing the number of grid points increases the size of the resulting coefficient matrix and, correspondingly, its condition number, which leads to deteriorating accuracy of the results of the linear system of equations [18]. The most elegant technique that has been used in an attempt to remedy this challenge is the introduction of multidomain based approaches in the method of solution.

Multidomain spectral collocation methods originate from the pioneering work of Gotltlieb and Orszag [19] and have been applied to solutions of nonlinear PDEs (see for instance [20]). Despite much attention on this subject, previous studies on the application of spectral collocation methods with domain decomposition techniques limited their focus to one-dimensional problems. Where higher dimensional problems are involved, in an attempt to reduce the computational complexity, a suitable similarity transformation has been applied to reduce the number of independent variables before the spectral collocation method with domain decomposition is applied [21]. As an alterna-

tive, direct application of spectral collocation methods with overlapping grids to higher dimensional nonlinear problems, and in particular to nonlinear elliptic PDEs emanating from heat and mass transfer, combustion theory and classical fluid flow problems has traditionally, been viewed as problematic, due, possibly, to the extent of the anticipated numerical complexity involved in dealing with overlapping subintervals in different spatial directions. Indeed, with a few noticeable exceptions [22, 23], very little work has been done on adopting a purely spectral collocation method with overlapping grids for the solution of this class of such important problems.

This paper focuses on the direct application of a domain decomposition based spectral collocation method on two-dimensional nonlinear PDEs defined on large rectangular domains. Preliminaries of the numerical algorithm involve breaking the large computational domain into smaller equal overlapping subintervals on each space direction. The solution of the PDE is then approximated using a bivariate Lagrange interpolating polynomial constructed using Chebyshev-Gauss-Lobatto (CGL) nodes. The linearized PDE is discretized on each subinterval and through careful dislodgement of repeated equations at overlapping regions, a system of linearly independent algebraic equations is obtained. The numerical scheme is tested using typical nonlinear elliptic PDEs, which have been reported in the literature and which includes the two dimensional heat and mass transfer equation with a n-th order volume reaction that arise in combustion theory [24], the stationary equation of Zabolotskaya and Khokhlov that describes different phenomena in acoustics [25], and a system of equations representing the problem of heat and mass transfer effects of a steady magneto-hydromagnetic flow of viscous, electrically conducting fluid past a semi-infinite inclined porous plate [26]. Numerical simulations confirm that the overlapping grids based approach is highly accurate and computationally efficient when compared to a single domain approach for the considered class of problems.

The rest of this paper is organized as follows. In Section 2, we give a detailed description of the numerical algorithm for the solution to nonlinear elliptic PDEs defined as a single equation. This novel approach is adapted and extended to systems of nonlinear elliptic equations in Section 3. then Section 4, is devoted to presenting new error bound theorems, and their rigorous proofs for bivariate interpolating polynomials on overlapping subdomains. These new theorems unveil the theory behind the benefits of the proposed numerical method of solution. In Section 5, we present and discuss numerical results for the selected examples. Finally, concluding remarks and acknowledgment are given in Sections 6 and 7, respectively.

2. Method of solution for a single nonlinear elliptic PDE

In this section, we describe the solution algorithm for a two-dimensional problem described by a single nonlinear equation. To do so, we consider a second-order nonlinear elliptic PDE that takes the form;

$$F\left(\frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial^2 u}{\partial y \partial x}, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, u\right) = h(x, y), \tag{1}$$

where F is a nonlinear operator operating on u, and its first and second order spatial derivatives and h is a known function. Eq.(1) is to be solved subject to the Robin boundary conditions

$$\begin{aligned} \alpha_1^a \frac{\partial u}{\partial x}(a, y) + \alpha_0^a u(a, y) = f_a(y), \quad \alpha_1^b \frac{\partial u}{\partial x}(b, y) + \alpha_0^b u(b, y) = f_b(y), \\ \beta_1^c \frac{\partial u}{\partial y}(x, c) + \beta_0^c u(x, c) = g_c(x), \quad \beta_1^d \frac{\partial u}{\partial y}(x, d) + \alpha_0^d u(x, d) = g_d(x), \end{aligned} \tag{2}$$

where $\alpha_1^a,\ \alpha_0^a,\ \alpha_1^b,\ \alpha_0^b,\ \beta_1^c,\ \beta_0^c,\ \beta_1^d,\ \beta_0^d$ are known constants and $f_a(y),\ f_b(y),\ g_c(x),\ g_d(x)$, are known functions. We note that the Robin boundary conditions in Eq.(2) can be reduced to either Neumann boundary conditions by setting $\alpha_0^a,\ \alpha_0^b,\ \beta_0^c,\ \beta_0^d$ to zero or to Dirichlet boundary conditions by setting $\alpha_1^a,\ \alpha_1^b,\ \beta_1^c,\ \beta_1^d$ to zero.

The PDE Eq.(1) is first simplified using the quasi-linearization method (QLM) of Bellman and Kalaba [27]. The QLM is based on the Newton-Raphson method. It is constructed using linear terms of Taylor series expansion about an initial approximation to the solution, on the assumption that the difference between solutions at two successive iterations, denoted by $u_{s+1} - u_s$, is very small. In particular, the QLM is comparable to the linear approximation of a function of several variables where the derivatives of the different orders and the previous approximation to the solution assume the two respective roles of independent variables and of the functional value at the reference point. Finer details about the linear approximation of functions can be found in any elementary book on differential calculus. Applying the QLM on Eq.(1) we obtain

$$\beta_{0,s}\frac{\partial^2 u_{s+1}}{\partial x^2} + \beta_{1,s}\frac{\partial u_{s+1}}{\partial x} + \beta_{2,s}u_{s+1} + \beta_{3,s}\frac{\partial^2 u_{s+1}}{\partial y \partial x} + \beta_{4,s}\frac{\partial^2 u_{s+1}}{\partial y^2} + \beta_{5,s}\frac{\partial u_{s+1}}{\partial y} = R_s + h(x, y), \tag{3}$$

where

$$\begin{aligned} &\beta_{0,s} = \frac{\partial F_s}{\partial (u_{xx})_s}, \quad \beta_{1,s} = \frac{\partial F_s}{\partial (u_x)_s}, \quad \beta_{2,s} = \frac{\partial F_s}{\partial u_s}, \quad \beta_{3,s} = \frac{\partial F_s}{\partial (u_{xy})_s}, \quad \beta_{4,s} = \frac{\partial F_s}{\partial (u_{yy})_s}, \quad \beta_{5,s} = \frac{\partial F_s}{\partial (u_y)_s}, \\ &R_s = \beta_{0,s}(u_{xx})_s + \beta_{1,s}(u_x)_s + \beta_{2,s}u_s + \beta_{3,s}(u_{xy})_s + \beta_{4,s}(u_{yy})_s + \beta_{5,s}(u_y)_s - F_s. \end{aligned} \tag{4}$$

The spatial domain $x \in [a, b]$ is decomposed into q overlapping subintervals of equal length as

$$\Lambda_l = [x_{l-1}, \bar{x}_l], \;\; x_{l-1} < x_l < \bar{x}_l, \;\; x_0 = a, \; \bar{x}_q = b, \;\; l = 1, 2, \ldots, q, \tag{5}$$

where $x_l < \bar{x}_l$, depicts the overlapping nature. Pictorially, the above decomposition of the x domain can be represented as in [28] by

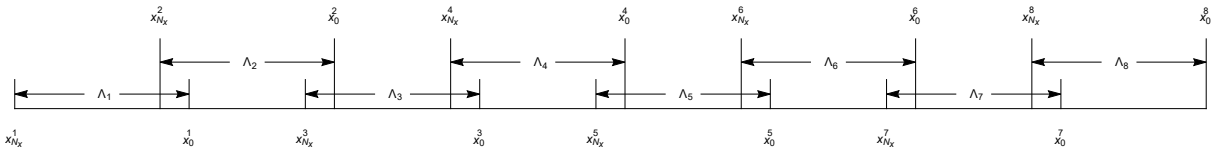


Figure 1: Pictorial illustration of decomposition of the x domain into 8 equal overlapping subintervals

The computational domain $[x_{l-1}, \bar{x}_l]$ in the l^{th} subinterval is then transformed into $\hat{x} \in [-1, 1]$ where the standard Chebyshev differentiation matrix is defined by applying the linear map

$$\hat{x}(x) = \frac{2}{\bar{x}_l - x_{l-1}} \left[x - \frac{1}{2} \left(\bar{x}_l + x_{l-1} \right) \right], \;\; x \in [x_{l-1}, \bar{x}_l]. \tag{6}$$

Further, each subinterval is discretized into $N_x + 1$ Chebyshev-Gauss-Lobatto points. The subintervals in the decomposed domain overlap in such a way that the last two points in the Λ_l subinterval overlap with the first two points in the Λ_{l+1} subinterval and they remain common. In general, the set of grid points over the entire x domain can be represented as

$$\{a = x^{(1)}_{N_x}, \ldots, x^{(1)}_1 = x^{(2)}_{N_x}, x^{(1)}_0 = x^{(2)}_{N_x - 1}, \ldots, x^{(l-1)}_1 = x^{(l)}_{N_x}, x^{(l-1)}_0 = x^{(l)}_{N_x - 1}, \ldots, x^{(q)}_0 = b, \; 2 \leq l \leq q\}. \tag{7}$$

Similarly, the spatial domain in y variable $[c, d]$ is decomposed into p equal overlapping subintervals which are further discretized into $N_y + 1$ collocation points. The subdivision of the computational domain in y is illustrated by

$$\Gamma_k = [y_{k-1}, \bar{y}_k], \;\; y_{k-1} < y_k < \bar{y}_k, \;\; y_0 = c, \; \bar{y}_p = d, \;\; k = 1, 2, \ldots, p, \tag{8}$$

and the overlapping nature is analogous to that in x variable. The computational domain $[y_{k-1}, \bar{y}_k]$ in the k^{th} subinterval is transformed into $\hat{y} \in [-1, 1]$ by applying the linear transformation

$$\hat{y}(y) = \frac{2}{\bar{y}_k - y_{k-1}} \left[y - \frac{1}{2} \left(\bar{y}_k + y_{k-1} \right) \right], \;\; y \in [y_{k-1}, \bar{y}_k], \tag{9}$$

before spectral collocation is applied. The points in the entire y domain can be represented as

$$\{c = y_{N_y}^{(1)}, \ldots, y_1^{(1)} = y_{N_y}^{(2)}, y_0^{(1)} = y_{N_y-1}^{(2)}, \ldots, y_1^{(k-1)} = y_{N_y}^{(k)}, y_0^{(k-1)} = y_{N_y-1}^{(k)}, \ldots, y_0^{(p)} = d,\ 2 \leq k \leq p\}. \tag{10}$$

We remark that the number of subintervals q in x need not to be equal to the number of subintervals p in y. The ordering of grid points as illustrated in Eq.(7) and Eq.(10) signifies that the spectral collocation is done from right to left of the subinterval. The grid points in the k^{th} subinterval in y and the l^{th} subinterval in x variable are defined in [29] by

$$\{\hat{y}_j\}_{j=0}^{N_y} = \cos\left(\frac{j\pi}{N_y}\right), \quad \text{and} \quad \{\hat{x}_i\}_{i=0}^{N_x} = \cos\left(\frac{i\pi}{N_x}\right). \tag{11}$$

To obtain the explicit expression of the length of each subinterval $L = \bar{x}_l - x_{l-1}$ in x and $K = \bar{y}_k - y_{k-1}$ in y, in terms of the number of subintervals q and p, respectively, we solve

$$qL - L\,(q-1)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right) = b - a, \quad pK - K\,(p-1)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_y}\right\}\right) = d - c, \tag{12}$$

to obtain

$$L = \frac{b-a}{q + (1-q)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right)}, \text{ and } K = \frac{d-c}{p + (1-p)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_y}\right\}\right)}. \tag{13}$$

Consequently, we obtain the following relations:

$$\bar{x}_l = x_l + L\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right), \quad \bar{y}_k = y_k + K\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_y}\right\}\right). \tag{14}$$

Eq.(14) is used in defining the boundaries of the overlapping subintervals when performing discretization.

In the solution process, the approximate solution of the nonlinear elliptic PDE Eq.(1) is assumed to be a bivariate Lagrange interpolating polynomial that takes the form;

$$u(x,y) \approx U(x,y) = \sum_{p=0}^{N_x}\sum_{q=0}^{N_y} U(x_p, y_q) L_p(x) L_q(y). \tag{15}$$

The derivative of u with respect to x in the l^{th} subinterval for $l = 1, 2, \ldots, q$, is approximated at the collocation nodes $(\hat{x}_i, \hat{y}_j)$, for $j = 0, 1, 2, \ldots, N_y$, as follows

$$\frac{\partial u}{\partial x}(\hat{x}_i, \hat{y}_j) \approx \overset{l}{\mathbf{DU}}_j = \left(\frac{2}{L}\right)\overset{l}{\hat{\mathbf{D}}}\mathbf{U}_j, \tag{16}$$

where $\overset{l}{\hat{\mathbf{D}}} = \left(\frac{L}{2}\right)\overset{l}{\mathbf{D}}$ is the standard first order Chebyshev differentiation matrix of size $(N_x+1)\times(N_x+1)$ as defined in [29]. The vector $\mathbf{U}_j$ is defined as

$$\mathbf{U}_j = [u(x_0,y_j), u(x_1,y_j), u(x_2,y_j,), \ldots, u(x_{N_x},y_j)]^T, \tag{17}$$

where T denotes matrix transpose. The spatial derivative in y in the k^{th} subinterval $k=1,2,\ldots,p$, is approximated at the collocation nodes $(\hat{x}_i,\hat{y}_j)$, for $i=0,1,2,\ldots,N_x$, as follows

$$\frac{\partial u}{\partial y}(\hat{x}_i,\hat{y}_j) \approx \sum_{q=0}^{N_y} \overset{k}{\bar{D}}_{j,q}\mathbf{U}_q = \sum_{q=0}^{N_y}\left(\frac{2}{K}\right)\overset{k}{\hat{\bar{D}}}_{j,q}\mathbf{U}_q, \tag{18}$$

where $\overset{k}{\hat{\bar{D}}}_{j,q} = \left(\frac{K}{2}\right)\overset{k}{\bar{D}}_{j,q}$, $j,q = 0,1,2,\ldots,N_y$, are entries of a standard first order Chebyshev differentiation matrix of size $(N_y+1)\times(N_y+1)$. We recognize that the bar in $\overset{k}{\bar{\boldsymbol{D}}}$ at Eq.(18) distinguishes the differentiation matrix in y from that in x, $\overset{l}{\mathbf{D}}$. The higher order differentiation matrices with respect to x and y can be obtained using matrix multiplication in Eq.(16) and Eq.(18), respectively. It is worth noting that L and K appearing in Eq.(16) and Eq.(18), are the usual lengths of the overlapping subintervals in x and y, respectively, and the simplification arrived therein employs properties of Lagrange interpolating polynomials. Finally, the mixed second order derivative of u is approximated as

$$\frac{\partial^2 u}{\partial x \partial y}(\hat{x}_i,\hat{y}_j) \approx \sum_{q=0}^{N_y}\overset{k}{\bar{D}}_{j,q}\left[\overset{l}{\mathbf{D}}\mathbf{U}_q\right], \quad j,q=0,1,2,\ldots,N_y. \tag{19}$$

The solution in each spatial direction is computed simultaneously across all subintervals and the multidomain approach is noticeable only when assembling the differentiation matrices. The multidomain overlapping grids based spectral collocation method is implemented on the linearized QLM scheme Eq.(3) as detailed below; the first step is to assemble the differentiation matrices in both space variables. To achieve this, we take the x variable as an example and since the last two points in the l^{th} subinterval and the first two points in the $(l+1)^{th}$ subinterval overlap and remain common, we discard the rows corresponding to the recurrent grid points and assemble the Chebyshev differentiation matrix $\mathbf{D}$ (without the superscript l) for the multidomain approach in x

as

$$\mathbf{D} = \begin{bmatrix}
\overset{q}{D}_{0,0} & \dots & \overset{q}{D}_{0,N_x-1} & \overset{q}{D}_{0,N_x} & & & & & & \\
\overset{q}{D}_{1,0} & \dots & \overset{q}{D}_{1,N_x-1} & \overset{q}{D}_{1,N_x} & & & & & & \\
\ddots & \ddots & \ddots & \ddots & & & & & & \\
\overset{q}{D}_{N_x-1,0} & \dots & \overset{q}{D}_{N_x-1,N_x-1} & \overset{q}{D}_{N_x-1,N_x} & & & & & & \\
 & & \ddots & \ddots & & & & & & \\
 & & \overset{2}{D}_{1,0} & \overset{2}{D}_{1,1} & \dots & \overset{2}{D}_{1,N_x-1} & \overset{2}{D}_{1,N_x} & & & \\
 & & \ddots & \ddots & \ddots & \ddots & \ddots & & & \\
 & & \overset{2}{D}_{N_x-1,0} & \overset{2}{D}_{N_x-1,1} & \dots & \overset{2}{D}_{N_x-1,N_x-1} & \overset{2}{D}_{N_x-1,N_x} & & & \\
 & & & & & \overset{1}{D}_{1,0} & \overset{1}{D}_{1,1} & \dots & \overset{1}{D}_{1,N_x} \\
 & & & & & \ddots & \ddots & \ddots & \ddots \\
 & & & & & \overset{1}{D}_{N_x-1,0} & \overset{1}{D}_{N_x-1,1} & \dots & \overset{1}{D}_{N_x-1,N_x} \\
 & & & & & \overset{1}{D}_{N_x,0} & \overset{1}{D}_{N_x,1} & \dots & \overset{1}{D}_{N_x,N_x}
\end{bmatrix}. \tag{20}$$

Here the empty entries of matrix $\mathbf{D}$ are zeros and $\overset{l}{D}$ represents Chebyshev differentiation matrix in the l^{th} subinterval in space variable x. The size of matrix $\mathbf{D}$ is $(\delta+1)\times(\delta+1)$, where $\delta = N_x + (N_x-1)\times(q-1)$. The differentiation matrix in y is assembled using $\overset{k}{\bar{\mathbf{D}}}$ in a similar manner to obtain a matrix $\bar{\mathbf{D}}$ of size $(\sigma+1)\times(\sigma+1)$, where $\sigma = N_y + (N_y-1)\times(p-1)$. The second order differentiation matrices with respect to x and y, $\mathbf{D}^2$ and $\bar{\mathbf{D}}^2$, respectively, can be obtained using matrix multiplication. Notation wise, the bar in $\bar{\mathbf{D}}$ has been used to distinguish the assembled differentiation matrices in y from those in x. The mixed second order derivative of u in the multiple domains case is approximated as

$$\frac{\partial^2 u}{\partial x \partial y}(\hat{x}_i, \hat{y}_j) \approx \sum_{q=0}^{\sigma} \bar{D}_{j,q} \left[\mathbf{D}\bar{\mathbf{U}}_q\right], \quad j,q = 0,1,2,\dots,\sigma. \tag{21}$$

In the multiple domains approach, the vector $\bar{\mathbf{U}}_j$, of size $(\delta+1)$, corresponds to vector $\mathbf{U}_j$ given in Eq.(17), and is defined as

$$\bar{\mathbf{U}}_j = [u(x_0, y_j), u(x_1, y_j), \dots, u(x_\delta, y_j)]^T, \quad j = 0,1,2,\dots,\sigma, \tag{22}$$

where T denotes matrix transpose. The x grid points at Eq.(22) are those in Eq.(7) when arranged from right to left such that $x_0 = b = x_0^{(q)}$, $x_\delta = a = x_{N_x}^{(1)}$. The bar in $\bar{\mathbf{U}}_j$ at Eq.(22) is meant to distinguish it from the vector $\mathbf{U}_j$ at Eq.(17). Using the assembled differentiation matrices in both

x and y directions in Eq.(3) we obtain a $(\delta+1)(\sigma+1)$ system of linear algebraic equations given by

$$\left[\boldsymbol{\beta_{0,s}}\mathbf{D}^2+\boldsymbol{\beta_{1,s}}\mathbf{D}+\boldsymbol{\beta_{2,s}}\mathbf{I}\right]\mathbf{U}_i+\boldsymbol{\beta_{3,s}}\sum_{j=0}^{\sigma}\bar{D}_{i,j}\left[\mathbf{DU}\right]_j+\sum_{j=0}^{\sigma}\left[\boldsymbol{\beta_{4,s}}\bar{D}_{i,j}^2+\boldsymbol{\beta_{5,s}}\bar{D}_{i,j}\right]\mathbf{U}_j=\mathbf{R}_i, \quad (23)$$

where $\mathbf{R}_i$ is a vector of size $(\delta+1)$ defined by $\mathbf{R}_i=\mathbf{R}_{i,s}+h(\mathbf{x},y_i),\ \ i=0,1,2,\ldots,\sigma$. The linear system of equations Eq.(23) can be expanded into a $(\delta+1)(\sigma+1)\times(\delta+1)(\sigma+1)$ matrix system that takes the form

$$\begin{bmatrix}\mathbf{A}_{0,0} & \mathbf{A}_{0,1} & \mathbf{A}_{0,2} & \ldots & \mathbf{A}_{0,\sigma}\\ \mathbf{A}_{1,0} & \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \ldots & \mathbf{A}_{1,\sigma}\\ \vdots & \vdots & \vdots & \ldots & \vdots\\ \mathbf{A}_{\sigma,0} & \mathbf{A}_{\sigma,1} & \mathbf{A}_{\sigma,2} & \ldots & \mathbf{A}_{\sigma,\sigma}\end{bmatrix}\begin{bmatrix}\bar{\mathbf{U}}_0\\ \bar{\mathbf{U}}_1\\ \vdots\\ \bar{\mathbf{U}}_\sigma\end{bmatrix}=\begin{bmatrix}\mathbf{R}_0\\ \mathbf{R}_1\\ \vdots\\ \mathbf{R}_\sigma\end{bmatrix}, \quad (24)$$

where

$$\begin{aligned}A_{i,i}&=\boldsymbol{\beta_{0,s}}\mathbf{D}^2+\boldsymbol{\beta_{1,s}}\mathbf{D}+\boldsymbol{\beta_{2,s}}\mathbf{I}+\boldsymbol{\beta_{3,s}}\mathbf{D}\bar{D}_{i,i}+\left[\boldsymbol{\beta_{4,s}}\bar{D}_{i,i}^2+\boldsymbol{\beta_{5,s}}\bar{D}_{i,i}\right]\mathbf{I},\\ A_{i,j}&=\boldsymbol{\beta_{3,s}}\mathbf{D}\bar{D}_{i,j}+\left[\boldsymbol{\beta_{4,s}}\bar{D}_{i,j}^2+\boldsymbol{\beta_{5,s}}\bar{D}_{i,j}\right]\mathbf{I},\ i\neq j,\ \ i,j=0,1,2,\ldots,\sigma,\end{aligned} \quad (25)$$

and $\mathbf{I}$ is an identity matrix of size $(\delta+1)\times(\delta+1)$. The boundary conditions are evaluated at the collocation nodes as

$$\alpha_1^a\sum_{p=0}^{\delta}D_{\delta,p}u(x_p,y_i)+\alpha_0^a u(x_\delta,y_i)=f_a(y_i),\ \ \alpha_1^b\sum_{p=0}^{\delta}D_{0,p}u(x_p,y_i)+\alpha_0^b u(x_0,y_i)=f_b(y_i),\ \ i=0,1,2,\ldots,\sigma, \quad (26)$$

and

$$\beta_1^c\sum_{q=0}^{\sigma}\bar{D}_{\sigma,q}u(x_j,y_q)+\beta_0^c u(x_j,y_\sigma)=g_c(x_j),\ \ \beta_1^d\sum_{q=0}^{\sigma}\bar{D}_{0,q}u(x_j,y_q)+\beta_0^d u(x_j,y_0)=g_d(x_j),\ \ j=0,1,2,\ldots,\delta. \quad (27)$$

The boundary conditions are imposed on the matrix system Eq.(24) to yield a different matrix system that takes the form

$$\begin{bmatrix}\beta_1^d\bar{D}_{0,0}\mathbf{I}+\beta_0^d\mathbf{I} & \beta_1^d\bar{D}_{0,1}\mathbf{I} & \beta_1^d\bar{D}_{0,2}\mathbf{I} & \ldots & \beta_1^d\bar{D}_{0,\sigma}\mathbf{I}\\ \bar{\mathbf{A}}_{1,0} & \bar{\mathbf{A}}_{1,1} & \bar{\mathbf{A}}_{1,2} & \ldots & \bar{\mathbf{A}}_{1,\sigma}\\ \bar{\mathbf{A}}_{2,0} & \bar{\mathbf{A}}_{2,1} & \bar{\mathbf{A}}_{2,2} & \ldots & \bar{\mathbf{A}}_{2,\sigma}\\ \vdots & \vdots & \vdots & \ldots & \vdots\\ \bar{\mathbf{A}}_{\sigma-1,0} & \bar{\mathbf{A}}_{\sigma-1,1} & \bar{\mathbf{A}}_{\sigma-1,2} & \ldots & \bar{\mathbf{A}}_{\sigma-1,\sigma}\\ \beta_1^c\bar{D}_{\sigma,0}\mathbf{I} & \beta_1^c\bar{D}_{\sigma,1}\mathbf{I} & \beta_1^c\bar{D}_{\sigma,2}\mathbf{I} & \beta_1^c\ldots & \beta_1^c\bar{D}_{\sigma,\sigma}\mathbf{I}+\beta_0^c\mathbf{I}\end{bmatrix}\begin{bmatrix}\bar{\mathbf{U}}_0\\ \bar{\mathbf{U}}_1\\ \bar{\mathbf{U}}_2\\ \vdots\\ \bar{\mathbf{U}}_{\sigma-1}\\ \bar{\mathbf{U}}_\sigma\end{bmatrix}=\begin{bmatrix}\bar{\mathbf{G}}_d\\ \bar{\mathbf{R}}_1\\ \bar{\mathbf{R}}_2\\ \vdots\\ \bar{\mathbf{R}}_{\sigma-1}\\ \bar{\mathbf{G}}_c\end{bmatrix}, \quad (28)$$

where

$$\bar{\mathbf{A}}_{i,i} = \begin{bmatrix} \alpha_1^b D_{0,0} + \alpha_0^b & \alpha_1^b D_{0,1} & \alpha_1^b D_{0,2} & \dots & \alpha_1^b D_{0,\delta} \\ & & \mathbf{A}_{i,i}(1:\delta-1, 0:\delta) & & \\ \alpha_1^a D_{\delta,0} & \alpha_1^a D_{\delta,1} & \alpha_1^a D_{\delta,2} & \dots & \alpha_1^a D_{\delta,\delta} + \alpha_0^a \end{bmatrix}, \quad i = 1, 2, \dots, \sigma - 1,$$

$$\bar{\mathbf{A}}_{i,j} = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ & & \mathbf{A}_{i,j}(1:\delta-1, 0:\delta) & & \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \; i \neq j, \; i = 1, 2, \dots, \sigma - 1, \; j = 0, 1, 2, \dots, \sigma, \tag{29}$$

and

$$\bar{\mathbf{R}}_i = \begin{bmatrix} f_b(y_i) \\ \mathbf{R}_i(1:\delta-1) \\ f_a(y_i) \end{bmatrix}, \quad i = 1, 2, \dots, \sigma - 1, \tag{30}$$

$$\bar{\mathbf{G}}_d = [g_d(x_0), g_d(x_1), g_d(x_2), \dots, g_d(x_\delta)]^T, \quad \bar{\mathbf{G}}_c = [g_c(x_0), g_c(x_1), g_c(x_2), \dots, g_c(x_\delta)]^T.$$

We remark that we have employed the MatLab syntax for representing a matrix in Eq.(29) and Eq.(30). Starting with an initial approximation to the solution, the system Eq.(28) is solved iteratively until a solution with the desired accuracy is realized.

The practical applicability of the multidomain overlapping grid based spectral collocation method to solutions of nonlinear elliptic PDEs given as a single equation is illustrated by considering the solution of typical examples of nonlinear elliptic PDEs that have been reported in the literature. For easy validation of the method, the selected test examples are nonlinear elliptic PDEs with exact solutions.

Example 1 (Steady heat and mass transfer equation with an n^{th} order volume reaction in two dimensions [24])**.** This equation arises in combustion theory and is given by

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = k u^n. \tag{31}$$

We analyze the nonlinear PDE for the case $k = 8$ and $n = 2$, which has the exact solution

$$u(x, y) = \left(x + \frac{y}{\sqrt{3}} + 1 \right)^{-2}. \tag{32}$$

Eq.(31) is solved subject to the boundary conditions

$$u(0, y) = \left(\frac{y}{\sqrt{3}} + 1 \right)^{-2},\ u(10, y) = \left(11 + \frac{y}{\sqrt{3}} \right)^{-2},\ u(x, 0) = (x + 1)^{-2},\ u(x, 10) = \left(x + \frac{10}{\sqrt{3}} + 1 \right)^{-2}. \tag{33}$$

Example 2 (Stationary PDE of Zabolotskaya and Khokhlov [25])**.** The stationary equation of Zabolotskaya and Khokhlov describes different phenomena in acoustic, nonlinear mechanics, heat and mass transfer theory. Its general form is given by

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial}{\partial y} \left[(\alpha u + \beta) \frac{\partial u}{\partial y} \right] = 0. \tag{34}$$

The analytical solution for $\alpha = 1,\ \beta = 0$ is

$$u(x, y) = - \left(\frac{y + 1}{x + 1} \right)^2. \tag{35}$$

Eq.(34) is solved subject to the boundary conditions

$$u(0, y) = - (y + 1)^2,\ u(5, x) = - \left(\frac{y + 1}{6} \right)^2\ u(x, 0) = - \left(\frac{1}{x + 1} \right)^2,\ u(x, 10) = - \left(\frac{11}{x + 1} \right)^2. \tag{36}$$

3. Method of solution for a system of nonlinear elliptic PDEs

In this section, we extend the multidomain overlapping grids based spectral collocation method described in the previous section to solutions of a system of two-dimensional nonlinear PDEs. This is demonstrated by considering the problem of heat and mass transfer effects on a steady magneto-hydromagnetic flow of a viscous, incompressible, electrically conducting and dissipating fluid past a semi-infinite porous plate inclined at an angle of α to the vertical. The flow is assumed to be in the x direction, taken as being along the semi-infinite inclined porous plate, and the y-axis is normal to it.

Example 3. The equations governing this flow were given in dimensionless form in [26] as

$$\frac{\partial \psi}{\partial y} \frac{\partial^2 \psi}{\partial x \partial y} - \frac{\partial \psi}{\partial x} \frac{\partial^2 \psi}{\partial y^2} = \frac{\partial^3 \psi}{\partial y^3} + Gr\theta \cos\alpha + Gc\phi \cos\alpha - \left(M + \frac{1}{K} \right) \frac{\partial \psi}{\partial y}, \tag{37}$$

$$\frac{\partial \psi}{\partial y} \frac{\partial \theta}{\partial x} - \frac{\partial \psi}{\partial x} \frac{\partial \theta}{\partial y} = \frac{1}{Pr} \frac{\partial^2 \theta}{\partial y^2} + Ec \left(\frac{\partial^2 \psi}{\partial y^2} \right)^2 \tag{38}$$

$$\frac{\partial \psi}{\partial y}\frac{\partial \phi}{\partial x}-\frac{\partial \psi}{\partial x}\frac{\partial \phi}{\partial y}=\frac{1}{Sc}\frac{\partial^2 \phi}{\partial y^2}-Kr\phi, \tag{39}$$

where ψ, θ, and ϕ are unknown functions of x and y representing the stream function, the temperature distribution, and concentration distribution, respectively. The governing parameters are, namely, thermal Grashof number Gr, mass Grashof number Gc, magnetic field number M, permeability parameter K, Prandtl number Pr, Eckert number Ec, Schmidt number Sc, and chemical reaction parameter Kr. The appropriate boundary conditions for this problem are

$$\begin{aligned}&\frac{\partial \psi}{\partial x}=\frac{\partial \psi}{\partial y}=0, \quad \theta=1, \quad \psi=1, \quad \text{at} \quad y=0\\ &\frac{\partial \psi}{\partial y}\to 0, \quad \theta\to 0, \quad \phi\to 0, \quad \text{as} \quad y\to\infty.\end{aligned} \tag{40}$$

As in the case of a single nonlinear equation, the system of PDEs Eqs.(37)-(39) is first simplified using the quasi-linearization method. Applying the QLM on Eqs.(37)-(39) we obtain

$$\beta^{(1)}_{0,s}\left(\psi_x\right)_{s+1}+\beta^{(1)}_{1,s}\left(\psi_{xy}\right)_{s+1}+\beta^{(1)}_{2,s}\left(\psi_{yyy}\right)_{s+1}+\beta^{(1)}_{3,s}\left(\psi_{yy}\right)_{s+1}+\beta^{(1)}_{4,s}\left(\psi_y\right)_{s+1}+\beta^{(1)}_{5,s}\theta_{s+1}+\beta^{(1)}_{6,s}\theta_{s+1}=R_{1,s} \tag{41}$$

$$\beta^{(2)}_{0,s}\left(\psi_x\right)_{s+1}+\beta^{(2)}_{1,s}\left(\psi_{yy}\right)_{s+1}+\beta^{(2)}_{2,s}\left(\psi_y\right)_{s+1}+\beta^{(2)}_{3,s}\left(\theta_x\right)_{s+1}+\beta^{(2)}_{4,s}\left(\theta_{yy}\right)_{s+1}+\beta^{(2)}_{5,s}\left(\theta_y\right)_{s+1}=R_{2,s} \tag{42}$$

$$\beta^{(3)}_{0,s}\left(\psi_x\right)_{s+1}+\beta^{(3)}_{1,s}\left(\psi_y\right)_{s+1}+\beta^{(3)}_{2,s}\left(\phi_x\right)_{s+1}+\beta^{(3)}_{3,s}\phi_{s+1}+\beta^{(3)}_{4,s}\left(\phi_{yy}\right)_{s+1}\beta^{(3)}_{5,s}\left(\phi_y\right)_{s+1}+=R_{3,s} \tag{43}$$

where

$$\begin{aligned}&\beta^{(1)}_{(0,s)}=\left(\psi_{yy}\right)_s, \quad \beta^{(1)}_{(1,s)}=-\left(\psi_y\right)_s, \quad \beta^{(1)}_{2,s}=1, \quad \beta^{(1)}_{3,s}=\left(\psi_x\right)_s, \quad \beta^{(1)}_{4,s}=-\left(\psi_{xy}\right)_s-\left(M+\frac{1}{k}\right),\\ &\beta^{(1)}_{(5,s)}=Gr\cos\alpha, \quad \beta^{(1)}_{(6,s)}=Gc\cos\alpha, \quad R_{1,s}=\left(\psi_x\right)_s\left(\psi_{yy}\right)_s-\left(\psi_y\right)_s\left(\psi_{xy}\right),\\ &\beta^{(2)}_{0,s}=\left(\theta_y\right)_s, \quad \beta^{(2)}_{1,s}=2Ec\left(\psi_{yy}\right)_s, \quad \beta^{(2)}_{2,s}=-\left(\theta_x\right)_s, \quad \beta^{(2)}_{(3,s)}=-\left(\psi_y\right)_s, \quad \beta^{(3)}_{(4,s)}=\frac{1}{Pr},\\ &\beta^{(2)}_{(5,s)}=\left(\psi_x\right)_s, \quad R_{2,s}=Ec\left(\psi_{yy}\right)^2_s+\left(\psi_x\right)_s\left(\theta_y\right)_s-\left(\psi_y\right)_s\left(\theta_x\right)_s,\\ &\beta^{(3)}_{0,s}=\left(\phi_y\right)_s, \quad \beta^{(3)}_{1,s}=-\left(\phi_x\right)_s, \quad \beta^{(3)}_{(2,s)}=-\left(\psi_y\right)_s, \quad \beta^{(3)}_{(3,s)}=-Kr, \quad \beta^{(3)}_{4,s}=\frac{1}{Sc},\\ &\beta^{(3)}_{(5,s)}=\left(\psi_x\right)_s, \quad R_{3,s}=\left(\psi_x\right)_s\left(\phi_y\right)_s-\left(\psi_y\right)_s\left(\phi_x\right)_s.\end{aligned} \tag{44}$$

Here, the subscripts inside parenthesis denote the partial derivatives of the various functions with respect to the indicated independent variable. For instance, (ψ_x) is the first derivative of ψ with

respect to x. Using an initial approximations ψ_0, θ_0, and ϕ_0 that satisfies the given boundary conditions, the linearized QLM scheme in Eqs.(41)-(43) is solved iteratively until a solution of the desired accuracy is obtained. Before applying the spectral collocation, the semi-infinite domains of $[0,\infty)$ in x and $[0,\infty)$ in y are first truncated into the finite domains $[0,L_x]$ and $[0,L_y]$, respectively, where Lx and L_y are taken to be large enough to approximate conditions at infinity in their respective directions. The finite domains of approximation are then decomposed into smaller subintervals, exactly as demonstrated in Eq.(5) and Eq.(8) in the previous section. The smaller subintervals are further discretized into Chebyshev-Gauss-Lobatto points as illustrated in Eq.(11). In the solution process, the approximate solutions sought for the system of PDEs Eqs.(37)-(39) are bivariate Lagrange interpolating polynomials that take the form

$$\begin{aligned}
\psi(x,y) \approx \Psi(x,y) &= \sum_{p=0}^{N_x}\sum_{q=0}^{N_y} \Psi(x_p,y_q)L_p(x)L_q(y),\\
\theta(x,y) \approx \Theta(x,y) &= \sum_{p=0}^{N_x}\sum_{q=0}^{N_y} \Theta(x_p,y_q)L_p(x)L_q(y),\\
\phi(x,y) \approx \Phi(x,y) &= \sum_{p=0}^{N_x}\sum_{q=0}^{N_y} \Phi(x_p,y_q)L_p(x)L_q(y).
\end{aligned} \tag{45}$$

The spectral approximation of derivatives for the unknown functions $\psi(x,y)$, $\theta(x,y)$ and $\phi(x,y)$ is achieved by adopting the strategy described in the previous section. By making an appropriate substitution of the discrete derivatives, the QLM scheme Eqs.(41)-(43) is converted into a $3(\sigma+1)(\delta+1)$ system of linear algebraic equations given by

$$\boldsymbol{\beta}^{(1)}_{0,s}\mathbf{D}\boldsymbol{\Psi}_i+\boldsymbol{\beta}^{(1)}_{1,s}\sum_{j=0}^{\sigma}\bar{D}_{i,j}\left[\mathbf{D}\boldsymbol{\Psi}_q\right]+\sum_{j=0}^{\sigma}\left[\boldsymbol{\beta}^{(1)}_{2,s}\bar{D}^3_{i,j}+\boldsymbol{\beta}^{(1)}_{3,s}\bar{D}^2_{i,j}+\boldsymbol{\beta}^{(1)}_{4,s}\bar{D}_{i,j}\right]\boldsymbol{\Psi}_j+\boldsymbol{\beta}^{(1)}_{5,s}\boldsymbol{\Theta}_i+\boldsymbol{\beta}^{(1)}_{6,s}\boldsymbol{\Phi}_i=\mathbf{R}_{1,i}, \tag{46}$$

$$\boldsymbol{\beta}^{(2)}_{0,s}\mathbf{D}\boldsymbol{\Psi}_i+\sum_{j=0}^{\sigma}\left[\boldsymbol{\beta}^{(2)}_{1,s}\bar{D}^2_{i,j}+\boldsymbol{\beta}^{(2)}_{2,s}\bar{D}_{i,j}\right]\boldsymbol{\Psi}_j+\boldsymbol{\beta}^{(2)}_{3,s}\mathbf{D}\boldsymbol{\Theta}_i+\sum_{j=0}^{\sigma}\left[\boldsymbol{\beta}^{(2)}_{4,s}\bar{D}^2_{i,j}+\boldsymbol{\beta}^{(3)}_{5,s}\bar{D}_{i,j}\right]\boldsymbol{\Theta}_q=\mathbf{R}_{2,i}, \tag{47}$$

$$\boldsymbol{\beta}^{(3)}_{0,s}\mathbf{D}\boldsymbol{\Psi}_i+\sum_{j=0}^{\sigma}\boldsymbol{\beta}^{(3)}_{1,s}\bar{D}_{i,j}\boldsymbol{\Psi}_q+\left[\boldsymbol{\beta}^{(3)}_{2,s}\mathbf{D}+\boldsymbol{\beta}^{(3)}_{3,s}\right]\boldsymbol{\Phi}_i+\sum_{j=0}^{\sigma}\left[\boldsymbol{\beta}^{(3)}_{4,s}\bar{D}^2_{i,j}+\boldsymbol{\beta}^{(3)}_{5,s}\bar{D}_{i,j}\right]\boldsymbol{\Phi}_q=\mathbf{R}_{3,i}, \tag{48}$$

where $i = 0, 1, 2, \ldots, \sigma$. The boundary conditions are evaluated at the collocation nodes as

$$\begin{aligned} &\psi'_{s+1}(x_j, y_\sigma) = 0, \quad \sum_{q=0}^{N_y} \bar{D}_{\sigma,q}\psi_{s+1}(x_j, y_q) = 0, \quad \theta_{s+1}(x_j, y_\sigma) = 1, \;\; \phi_{s+1}(x_j, y_\sigma) = 1, \\ &\sum_{q=0}^{\sigma} \bar{D}_{0,q}\psi_{s+1}(x_j, y_0) = 0, \quad \theta_{s+1}(x_j, y_0) = 0, \;\; \phi_{s+1}(x_j, y_0) = 0, \quad j = 0, 1, 2, \ldots, \delta. \end{aligned} \tag{49}$$

In Eq.(49) the prime in ψ' denotes the vector of discrete derivative, $\psi' = \mathbf{D}\boldsymbol{\Psi}$. The vectors $\boldsymbol{\Psi}_i$, $\boldsymbol{\Theta}_i$, and $\boldsymbol{\Phi}_i$ for $i = 0, 1, 2, \ldots, \sigma$ are defined analogouslyt o $\bar{\mathbf{U}}_j$ in Eq.(22). The boundary conditions at Eq.(49) are imposed by replacing the corresponding equations in Eqs.(46)-(48) with Eq.(49) to yield a new system of linear algebraic equations, that can be expressed as a $3(\sigma+1)(\delta+1)\times 3(\sigma+1)(\delta+1)$ matrix system. As will be shown in the results, we notice that, in the single domain approach, when enlarging the size of the truncated intervals $[0, L_x]$ and $[0, L_y]$ in x and y directions, respectively, in an attempt to properly approximate conditions at infinity, leads to deteriorating accuracy even with increase in the number of collocation points. This is the challenge that the overlapping grids based algorithm described in this paper seeks to address.

4. Error bounds theorems in a bivariate polynomial interpolation

In this section, we present new error bound theorems and their proofs which emanate from a bivariate Lagrange interpolating polynomial that is constructed using Chebyshev Gauss-Lobatto nodes on a domain that has been decomposed into smaller overlapping grids of equal length. The error bound theorems given herein form the foundation for the reasoning behind the theoretical argument as to why multidomain approaches are most suitable for approximating the solutions to partial differential equations defined over large domains. Fundamental components related to the construction of proofs of the theorems include understanding, firstly, that CGL nodes are the relative extremes of the N_x-th degree Chebyshev polynomial of the first kind $T_{N_x}(\hat{x}) = \cos\left[N_x \arccos(\hat{x})\right]$, $\hat{x} \in [-1, 1]$, and, secondly, the general properties of Chebyshev polynomials. Although to the best of our knowledge, there does not exist a well-known family of polynomials whose roots are the CGL nodes, it is easy to discern that the interior CGL nodes are roots of $T'_{N_x}(\hat{x}) = 0$. This fact leads to the discovery of a complete set of the CGL nodes as the roots of the $N_x + 1$-th degree polynomial given by

$$L_{N_x+1}(\hat{x}) = (1 - \hat{x}^2)T'_{N_x}(\hat{x}). \tag{50}$$

Below, is the theorem that benchmarks formulation of the error bound theorems on bivariate polynomial interpolation.

Theorem 1. *[30] Let $u(x,t) \in C^{N_x+N_y+2}([a,b]\times[c,d])$ be sufficiently smooth such that at least the $(N_x+1)^{th}$ partial derivative with respect to x, the $(N_y+1)^{th}$ partial derivative with respect to t and $(N_x+N_y+2)^{th}$ mixed partial derivative with respect to x and y all exist and are all continuous, then there exist values $\xi_x, \xi_x' \in (a,b)$, and $\xi_y, \xi_y' \in (c,d)$, such that*

$$\begin{aligned} u(x,y) - U(x,y) = &\frac{\partial^{N_x+1}u(\xi_x,y)}{\partial x^{N_x+1}(N_x+1)!}\prod_{i=0}^{N_x}(x-x_i) + \frac{\partial^{N_y+1}u(x,\xi_y)}{\partial y^{N_y+1}(N_y+1)!}\prod_{j=0}^{N_y}(y-y_j) \\ &- \frac{\partial^{N_x+N_y+2}u(\xi_x',\xi_y')}{\partial x^{N_x+1}\partial y^{N_y+1}(N_x+1)!(N_y+1)!}\prod_{i=0}^{N_x}(x-x_i)\prod_{j=0}^{N_y}(y-y_j), \end{aligned} \tag{51}$$

where $U(x,y)$ is a bivariate interpolating polynomial of $u(x,y)$ at $\{x_i\}_{i=0}^{N_x}$ grid points in the x-variable and $\{y_j\}_{j=0}^{N_y}$ grid points in the y-variable.

A result similar to Eq.(51) was reproduced by Bhrawy in [31]. Taking the absolute value of Eq.(51) we obtain

$$\begin{aligned} |u(x,y) - U(x,y)| \leq &\max_{(x,y)\in\Omega}\left|\frac{\partial^{N_x+1}u(\xi_x,y)}{\partial x^{N_x+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x-x_i)\right|}{(N_x+1)!} + \max_{(x,y)\in\Omega}\left|\frac{\partial^{N_y+1}u(x,\xi_y)}{\partial y^{N_y+1}}\right|\frac{\left|\prod_{j=0}^{N_y}(y-y_j)\right|}{(N_y+1)!} \\ &+ \max_{(x,y)\in\Omega}\left|\frac{\partial^{N_x+N_y+2}u(\xi_x',\xi_y')}{\partial x^{N_x+1}\partial y^{N_y+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x-x_i)\right|\left|\prod_{j=0}^{N_y}(y-y_j)\right|}{(N_x+1)!(N_y+1)!}, \end{aligned} \tag{52}$$

where $\Omega = [a,b]\times[c,d]$. Since the function $u(x,y)$ is assumed to be smooth on the interval of approximation, it follows that its derivatives are bounded and thus $\exists$ constants C_1, C_2 and C_3, such that

$$\max_{(xyt)\in\Omega}\left|\frac{\partial^{N_x+1}u(x,y)}{\partial x^{N_x+1}}\right| \leq C_1,\ \max_{(x,y)\in\Omega}\left|\frac{\partial^{N_y+1}u(x,y)}{\partial y^{N_y+1}}\right| \leq C_2,\ \max_{(x,y)\in\Omega}\left|\frac{\partial^{N_x+N_y+2}u(x,y)}{\partial x^{N_x+1}\partial y^{N_y+1}}\right| \leq C_3. \tag{53}$$

4.1. Error bound theorem on a single domain

The error bound for bivariate polynomial interpolation using Chebyshev-Gauss-Lobatto nodes on a single domain is governed by the theorem below

Theorem 2 (**The error bound in a single domain**). *The resulting error bound when CGL grid points, $\{x_i\}_{i=0}^{N_x} \in [a,b]$, in x-variable and, $\{y_j\}_{j=0}^{N_y} \in [c,d]$, in y-variable are used in bivariate polynomial interpolation is given by*

$$E(x,y) \leq C_1\frac{8\left(\frac{b-a}{4}\right)^{N_x+1}}{(N_x+1)!} + C_2\frac{8\left(\frac{d-c}{4}\right)^{N_y+1}}{(N_y+1)!} + C_3\frac{8^2\left(\frac{b-a}{4}\right)^{N_x+1}\left(\frac{d-c}{4}\right)^{N_y+1}}{(N_x+1)!(N_y+1)!}. \tag{54}$$

Proof. First, using the relation stated in [32] we express Eq.(50) as

$$L_{N_x+1}(\hat{x}) = (1-\hat{x}^2)T'_{N_x}(\hat{x}) = -N_x\hat{x}T_{N_x}(\hat{x}) + N_xT_{N_x-1}(\hat{x}). \tag{55}$$

Using the triangle inequality and noting that $|T_{N_x}(\hat{x})| \leq 1,\ \forall\ \hat{x} \in [-1,1]$, we have

$$|L_{N_x+1}(\hat{x})| = |-N_x\hat{x}T_{N_x}(\hat{x}) + N_xT_{N_x-1}(\hat{x})| \leq |-N_x\hat{x}T_{N_x}(\hat{x})| + |N_xT_{N_x-1}(\hat{x})| \leq 2N_x. \tag{56}$$

The leading coefficient of $L_{N_x+1}(\hat{x})$ is $2^{N_x-1}N_x$, where the components 2^{N_x-1} and N_x come, respectively, from the leading coefficient of $T_{N_x}(\hat{x})$ and the application of N_x-th rule of differentiation on $T_{N_x}(\hat{x})$. The product factor in the first term of the error bound expression given in Eq.(52) can therefore be taken as the factorized form of a monic polynomial $\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}$. We write

$$\prod_{i=0}^{N_x}(\hat{x}-\hat{x}_i) = \frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x},\ \ \hat{x} \in [-1,1]. \tag{57}$$

Using Eq.(56), it is easy to establish that the monic polynomial Eq.(57) is bounded by

$$\left|\prod_{j=0}^{N_x}(x-\hat{x}_i)\right| = \left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right| \leq \frac{2N_x}{2^{N_x-1}N_x} = \frac{4}{2^{N_x}}. \tag{58}$$

Considering a general interval $x \in [a,b]$, we can show that the first product factor in Eq.(52) is bounded by

$$\begin{aligned}\max_{a\leq x\leq b}\left|\prod_{i=0}^{N_x}(x-x_i)\right| &= \max_{-1\leq\hat{x}\leq 1}\left|\prod_{i=0}^{N_x}\frac{(b-a)}{2}(\hat{x}-\hat{x}_i)\right| = \left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\leq\hat{x}\leq 1}\left|\prod_{i=0}^{N_x}(\hat{x}-\hat{x}_i)\right| \\ &= \left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\leq\hat{x}\leq 1}\left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right| \leq \frac{4\left(\frac{b-a}{2}\right)^{N_x+1}}{2^{N_x}} = 8\left(\frac{b-a}{4}\right)^{N_x+1}.\end{aligned} \tag{59}$$

Similarly, we conclude that the second product factor is bounded by

$$\max_{c\leq y\leq d}\left|\prod_{j=0}^{N_y}(y-y_j)\right| = \left(\frac{d-c}{2}\right)^{N_y+1}\max_{-1\leq\hat{y}\leq 1}\left|\frac{L_{N_y+1}(\hat{y})}{2^{N_y-1}N_y}\right| \leq \frac{4\left(\frac{d-c}{2}\right)^{N_y+1}}{2^{N_y}} = 8\left(\frac{d-c}{4}\right)^{N_y+1}. \tag{60}$$

Using Eq.(53), Eq.(59), and Eq.(60) in Eq.(52) the proof is completed. □

4.2. Error bound theorem on decomposed domain

Here, we extend the bivariate polynomial interpolation error bound Theorem 2 to obtain its variant on a decomposed domain. We assume that the number of grid points is the same for all subintervals in either direction of the space variable, x or y.

Theorem 3 (**The error bound in the decomposed domain**). *The error bound, when Chebyshev-Gauss-Lobatto grid points* $\{x_i\}_{i=0}^{N_x} \in [x_{l-1}, \bar{x}_l],\ l = 1, 2, \ldots, q$, *for the decomposed domain in x-variable and* $\{y_j\}_{j=0}^{N_y} \in [y_{k-1}, \bar{y}_k],\ k = 1, 2, \ldots, p$, *for the decomposed domain in y-variable are used in bivariate polynomial interpolation, is given by*

$$E(x,t) \leq C_1 \frac{8\left(\frac{L}{4}\right)^{N_x+1}}{(N_x+1)!} + C_2 \frac{8\left(\frac{K}{4}\right)^{N_y+1}}{(N_y+1)!} + C_3 \frac{8^2\left(\frac{L}{4}\right)^{N_x+1}\left(\frac{K}{4}\right)^{N_y+1}}{(N_x+1)!(N_y+1)!}, \tag{61}$$

where L *and* K *denote the length of each subinterval in* x *and* y, *respectively as defined in Eq.*(13).

Proof. First, we consider the x variable. In the entire domain $[a, b]$, we have that

$$\left|\prod_{i=0}^{N_x}(x - x_i)\right| \leq 8\left(\frac{b-a}{4}\right)^{N_x+1}, \quad x \in [a, b]. \tag{62}$$

Since interpolation is performed piece-wise and noting from the previous theorem that the error bound depends on the length of the interval, there is a direct implication that in the decomposed domain and at each subinterval in x, we must have

$$\left|\prod_{i=0}^{N_x}(x - x_i)\right| \leq 8\left(\frac{L}{4}\right)^{N_x+1}, \quad \{x_i\}_{i=0}^{N_x} \in [x_{l-1}, \bar{x}_l],\ l = 1, 2, \ldots, q. \tag{63}$$

Under the assumption that the unknown function $u(x, y)$ is smooth, it is logical to assert that $\exists\, \xi_\mu \in (x_{\mu-1}, \bar{x}_\mu)$, $\mu = 1, 2, \ldots, q$, for which the values of the $(N_x+1)^{th}$ partial derivatives of u with respect to x in each subinterval, is the absolute extrema. This enables us to break the first term $C_1 \frac{8\left(\frac{b-a}{4}\right)^{N_x+1}}{(N_x+1)!}$, which appears in the error bound expression at Eq.(54), into different components that are necessarily not equal in the decomposed x domain, as

$$\left\{ \overset{(l)}{C}_1 \frac{8\left(\frac{L}{4}\right)^{N_x+1}}{(N_x+1)!} \right\}_{l=1}^{q}, \tag{64}$$

where

$$\max_{(x,y)\in\Omega}\left|\frac{\partial^{N_x+1}u(x,y)}{\partial x^{N_x+1}}\right| = \left|\frac{\partial^{N_x+1}u(\xi_l,y)}{\partial x^{N_x+1}}\right| \leq \overset{(k)}{C}_1, \quad x \in [x_{l-1}, \bar{x}_l].$$

We define

$$\| \hat{C}_1 \|_\infty \equiv \max\{\overset{(1)}{C}_1, \overset{(2)}{C}_1, \ldots, \overset{(q)}{C}_1\}, \tag{65}$$

to denote the maximum absolute value of the $(N_x+1)^{th}$ partial derivatives of u with respect to x in $[a, b]$. Clearly, $\| \hat{C}_1 \|_\infty = C_1$, where C_1 is identical to that given in Eq.(54). To expand the error bound over the entire x domain, we shall take the largest possible error across all overlapping subintervals in x which is

$$C_1 \frac{8\left(\frac{L}{4}\right)^{N_x+1}}{(N_x+1)!}. \tag{66}$$

Similar reasoning can be applied to show that the second component in error bound Eq.(54) in the decomposed y domain translates to

$$C_2 \frac{8\left(\frac{K}{4}\right)^{N_y+1}}{(N_y+1)!}. \tag{67}$$

Consequently, the third component in Eq.(54) becomes,

$$C_3 \frac{8^2\left(\frac{L}{4}\right)^{N_x+1}\left(\frac{K}{4}\right)^{N_y+1}}{(N_x+1)!(N_y+1)!}, \tag{68}$$

in the decomposed domain and $\left(\frac{L}{4}\right)^{N_x+1} << \left(\frac{b-a}{4}\right)^{N_x+1}$ and $\left(\frac{K}{4}\right)^{N_y+1} << \left(\frac{d-c}{4}\right)^{N_y+1}$ for a large number of subintervals q and p in x and y, respectively. Using Eqs.(66) to (68) in Eq.(54) completes the proof. □

Comparing Eq.(54) and Eq.(61), we note that the error in bivariate polynomial interpolation is smaller when interpolation is conducted on multiple overlapping domains than on a single domain. Further, we remark that CGL nodes are the preferred candidates for interpolation when using spectral collocation methods to solve boundary value problems because they are convenient for constructing differentiation matrices as they contain the boundary nodes, which is advantageous when treating the boundary conditions of the problem. To conduct a numerical validation of error bound theorems, we consider the interpolation error in approximating the function of two variables, as given below.

Example 4. Consider constructing an interpolating polynomial to approximate the function

$$u(x,y) = e^{x+y},\ x \in [0,4],\ y \in [-2,2]. \tag{69}$$

5. Results and discussion

In this section, numerical results obtained after solving the test Examples 1 to 3 using the overlapping grid based spectral collocation method are presented in tabular and graphical forms and discussed. The numerical schemes were implemented on the MatLab 2017b platform. We demonstrate the superiority of the proposed method in terms of accuracy by comparing absolute errors obtained using the overlapping grids approach against the equivalent results using a single domain. The absolute error values computed at selected grid points in x and y for Examples $1-2$ are defined by

$$E_{i,j} = \left| \overset{a}{U}(x_i, y_j) - U(x_i, y_j) \right|. \tag{70}$$

Here $\overset{a}{U}$ and U denote the approximate and the analytic solution, respectively. To highlight the advantages of adopting the overlapping grids approach, we examine properties of the coefficient matrices for the resulting linear system of algebraic equations. To set a base for comparison, the number of grid points across the entire domain is maintained constant. We investigate the effect of increasing the number of subintervals in both x and y directions, as opposed to incrementing the number of grid points on a single large domain. The aim is to quantify the size of the resulting linear system of equations, find condition numbers of the associated coefficient matrices and keep track of the CPU time that the proposed method of solution takes to realize results. The condition number of the matrix $\mathbf{A}$ is found using the MatLab build in command $cond\,(\mathbf{A})$. For the nonlinear system of equations in Example 3, for which no exact solutions exist, the accuracy of the numerical scheme is evaluated by presenting residual error values at different iteration levels. To define the residual error, it is convenient to first rewrite Eqs.(37)-(39), respectively, in the form;

$$F_{\psi}\left(\psi,\theta,\phi\right)=0,\quad F_{\theta}\left(\psi,\theta,\phi\right)=0,\ \text{ and }\ F_{\phi}\left(\psi,\theta,\phi\right)=0, \tag{71}$$

where F_{ψ}, F_{θ}, and F_{ϕ} are nonlinear differential operators. The residual error is thus computed as

$$R_{\psi}=||\mathbf{F}_{\psi,s+1}||_{\infty},\quad R_{\theta}=||\mathbf{F}_{\theta,s+1}||_{\infty},\quad R_{\phi}=||\mathbf{F}_{\phi,s+1}||_{\infty}. \tag{72}$$

To assess convergence of the iterative scheme for the system of equations, the graph of solution error values versus iteration number is presented. Solution error is measured using the infinity norm of the vector obtained by taking the difference between solutions at successive iterations as

$$S_{\psi}=||\mathbf{\Psi}_{s+1}-\mathbf{\Psi}_{s}||_{\infty},\quad S_{\theta}=||\mathbf{\Theta}_{s+1}-\mathbf{\Theta}_{s}||_{\infty},\quad S_{\phi}=||\mathbf{\Phi}_{s+1}-\mathbf{\Phi}_{s}||_{\infty}. \tag{73}$$

The effect of several flow parameters on the temperature and concentration profiles is also investigated and compared with results already published in the literature.

The numerical results obtained from approximating the solution of Eq.(31) using the spectral collocation method on overlapping grids are given in Table 1. Results indicate that the method is very accurate, as absolute errors of order 10^{-13} were registered. The algorithm is also executed over a short CPU time as shown in Table 1 by the small CPU time of less than 0.14 seconds. Equivalent results obtained using the single domain approach are displayed in Table 2. It can be seen that absolute errors of order 10^{-11} are reported, which is 100 times larger than those in Table 1, and the CPU time is more than 5 times greater than for the overlapping grids approach. The number of grids points across the entire interval is maintained constant in both approaches. The numerical

results from Table 1 and 2 suggest that the overlapping grids based spectral collocation method is more accurate and computationally more efficient than the single domain approach when applied to solve nonlinear elliptic PDEs defined on large rectangular domains. The improved accuracy for the overlapping grids approach can be attributed to smaller condition numbers of the matrices. The shorter CPU time required to realize results in the multiple domains approached can be explained by the sparse nature of the coefficient matrices, which are then easy to invert. We remark that the unanticipated difference in matrix dimension between both approaches relates to the number of common nodes that are discarded when assembling the spatial differentiation matrix in the multidomain overlapping grids based approach.

The results obtained from approximating the solution of Eq.(34) are given in Table 3 and Table 4. Table 3 shows the results generated with the multidomain overlapping grids based spectral collocation method whereas Table 4 presents the results obtained using the single domain based approach namely, the bivariate spectral collocation method is used. As in the previous example, the number of grid points over the entire interval is kept constant. It is evident that the results obtained when using the multidomain approach are more accurate than those realized from the single domain approach, with absolute errors of order 10^{-10} and 10^{-09}, respectively, being recorded. Additionally, we notice that the CPU time required for the multidomain approach is about 5 times shorter than for the single domain approach. For the new approach, the need for only a few numbers of grid points at each subinterval also reduces the effects of roundoff errors that are associated with approximating functions with interpolating polynomials of higher degree. Furthermore, sparse matrices are also advantageous as they are well conditioned and so render a well-posed problem, ensuring that stable results are obtained [33]. Noticeable difference in magnitudes of absolute errors obtained when Example 1 and Example 2 are solved is caused by the smoothness of solution for Eq.(31), which contrasts with the discontinuity of the solution of Eq.(34) at $x = -1$. Finally, we observe that the relative difference between the absolute errors in Tables 3 and 4 are smaller than those in Table 1 and 2. This is because in Example 2, we considered a relatively smaller x domain $b - a = 5$, compared to that in Example 1, $b - a = 10$. This comparison is significant as it indicates that the superiority of overlapping grids approach is most noticeable in problems that are defined over large domains.

Numerical results for the system of nonlinear equations given in Example 3 are presented in terms of graphs of the residual error and solution error values at different iteration levels, in line with findings

of investigating the effect of several parameters on the behaviour of the flow. The following default values of flow parameters are adopted during experimentation; $Gr = 2.0$, $Gc = 2.0$, $M = 1.0$, $K = 1.0$, $Ec = 0.01$, $Pr = 0.71$, $Sc = 0.6$, and $Kr = 0.5$. The angle of inclination of the porous plate was taken to be $\alpha = \frac{\pi}{6}$. All graphs, therefore, correspond to these values unless indicated otherwise on the appropriate graphs. The domain of approximation was truncated as $L_x = 30$ and $L_y = 5$, in x and y directions, respectively. In the case of single domain approach, 40 grid points were used in x domain whereas 20 grid points were used in y. Spectral collocation on overlapping grids is invoked using $q = 2$ overlapping subintervals in x with 20 grid points in each subinterval, and $p = 2$ overlapping subintervals in y with 10 grid points in each. The values of residual error against iterations are depicted in Figure 2. The graphs in Figure 2 suggests that the present numerical algorithm works perfectly well on systems of nonlinear elliptic PDEs as small residual error values of order 10^{-9} are recorded. The dotted lines represent residual error profiles obtained from the overlapping grid approach whereas continuous line denotes the residual error profiles for the case of single domain approach. The superiority of the overlapping grids based approach is manifested in the graphs. As confirmed by the residual error profiles displayed in Figure 2, the best approximate solutions are obtained after 4 iterations. In addition, Figure 3 shows that the numerical scheme converges quadratically and that convergence is indeed achieved after the 4^{th} iteration.

The overlapping grid approach was employed to generate graphs shown in Figures 4-7, which demonstrate the effect of different parameters on flow fields. Figures 4-9. Figures 4-7 show the temperature profiles for different parameters. The Prandtl number, Pr, defines the ratio of momentum diffusivity to thermal diffusivity. From Figure 4, it is observed that an increase in the Prandtl number decreases the thermal boundary layer thickness and in general lowers the average temperature within the boundary layer. This phenomenon can be explained by the decreased values of Prandtl number being equivalent to increased thermal conductivity, consequently, heat diffuses away from the heated plate more rapidly than it would for higher values of Prandtl number. The effect of the viscous dissipation parameter, Eckert number Ec, on temperature is shown in Figure 5. The Eckert number expresses the relationship between the kinetic energy and enthalpy in the flow and it embodies the conversion of kinetic energy into internal energy by work done against the viscous fluid stress. Greater viscous dissipative heat causes an increase in temperature, which is evident from Figure 5. The effect of different values of the magnetic field parameter M on temperature profiles is plotted in Figure 6. It is observed that as the magnetic parameter increases,

the temperature also increase. Figure 7 represents the effect of the porosity parameter K on the temperature profiles. We notice that very small changes occur in the thermal boundary layer when changes are made to the porosity parameter. The temperature increases with an increase in the porosity parameter. The influence of the Schmidt number, Sc, on the concentration profiles, is plotted in Figure 8. The Schmidt number embodies the ratio of the momentum to mass diffusivity and therefore it quantifies the relative effectiveness of momentum and mass transport by diffusion in the concentration (species) boundary layer. As can be seen from Figure 8, increasing the Schmidt number decreases the species concentration. The influence of the chemical reaction parameter Kr on the concentration profiles across the boundary layer are presented in Figure 9. We see that the concentration distribution across the boundary layer decreases with an increase in chemical reaction parameter. Overall, the observations made on the effects of different parameters on flow as shown in our results are in agreement with those made by Reddy et al. [26].

Finally, Table 5 presents interpolation errors obtained from approximating the function given in Eq.69 using a bivariate Lagrange interpolating polynomial at CGL points. The results for overlapping grids based approximation are shown in bold to distinguish them from those obtained by interpolation on a single domain. The column label, Bound, denotes the maximum possible theoretical value of interpolation error evaluated using formulas given in the error bound theorems Eq.(54) and Eq.(61). The domain of approximation was subdivided into 2 overlapping subintervals of equal length in both x and y directions. The number of grid points over the entire domain was maintained constant to achieve a fair comparison. Table 5 shows that the interpolation errors decrease as the number of grid points increase. It also demonstrates that numerical values of interpolation errors are always smaller than the theoretical error bound values, thereby verifying that the function considered here obeys the bivariate interpolating polynomial error bound theorems that were presented in Section 4. We observe that performing interpolation over many smaller overlapping domains records a significantly shorter CPU time than the equivalent process over a single domain. Further, improved accuracy is registered when 40 grid points are used in multiple domains, whereas for a single domain 40 grid points results in a drop in accuracy.

Table 1: Absolute error values obtained when Example 1 is solved on overlapping domains: $N_x = 20$, $q = 2$, $N_y = 20$, $p = 2$.

	y		
x	9.9384	5.7822	0.5450
9.9846	1.08637e-15	3.16414e-15	3.00255e-14
8.8020	2.68188e-15	5.75581e-15	8.24072e-14
7.6125	5.65693e-15	3.49876e-14	2.51086e-13
6.1672	6.96491e-15	1.47209e-14	2.09039e-13
3.0866	5.94733e-14	5.78860e-13	2.09039e-13
CPU time (sec)		0.135284	
Matrix Size		1600×1600	
Cond Number		4.87352e+003	

Table 2: Absolute error values obtained when Example 1 is solved on a single domain: $N_x = 40$, $N_y = 40$.

	y		
x	9.9384	5.7822	0.5450
9.9846	9.42779e-13	6.38465e-13	5.78079e-13
8.8020	5.26489e-14	1.91149e-13	3.81223e-13
7.6125	3.51628e-13	5.07198e-13	2.98928e-13
6.1672	1.50756e-12	1.39328e-12	4.09846e-13
3.0866	3.17623e-11	8.84608e-12	3.97314e-11
CPU time (sec)		0.743578	
Matrix Size		1681×1681	
Cond Number		7.6543e+004	

Table 3: Absolute error values obtained when Example 2 is solved on overlapping domains: $N_x = 20$, $q = 2$, $N_y = 10$, $p = 5$.

	y		
x	9.9691	7.1000	1.7328
4.9923	1.07443e-11	1.38911e-11	8.56769e-11
4.2739	2.23288e-11	1.74158e10	1.12733e-10
2.9300	3.18758e-11	1.06726e-10	3.52343e-10
1.5909	5.98632e-12	6.45883e-11	1.78728e-11
0.2588	5.16565e-11	2.39211e-10	2.02153e-10
CPU time (sec)		0.243617	
Matrix Size		1940×1940	
Cond Number		4.2389e+007	

Table 4: Absolute error values obtained when Example 2 is solved on a single domain: $N_x = 40$, $N_y = 50$.

	y		
x	9.9691	7.1000	1.7328
4.9923	6.49512e-11	3.81219e-10	1.27900e-10
4.2739	1.20477e-10	1.94005e-09	9.43050e-10
2.9300	3.15801e-11	1.94005e-09	8.00239e-10
1.5909	4.48859e-10	9.23929e-10	1.27900e-10
0.2588	9.82577e-10	1.34028e-09	4.07725e-09
CPU time (sec)		1.383625	
Matrix Size		2091×2091	
Cond Number		2.7812e+009	

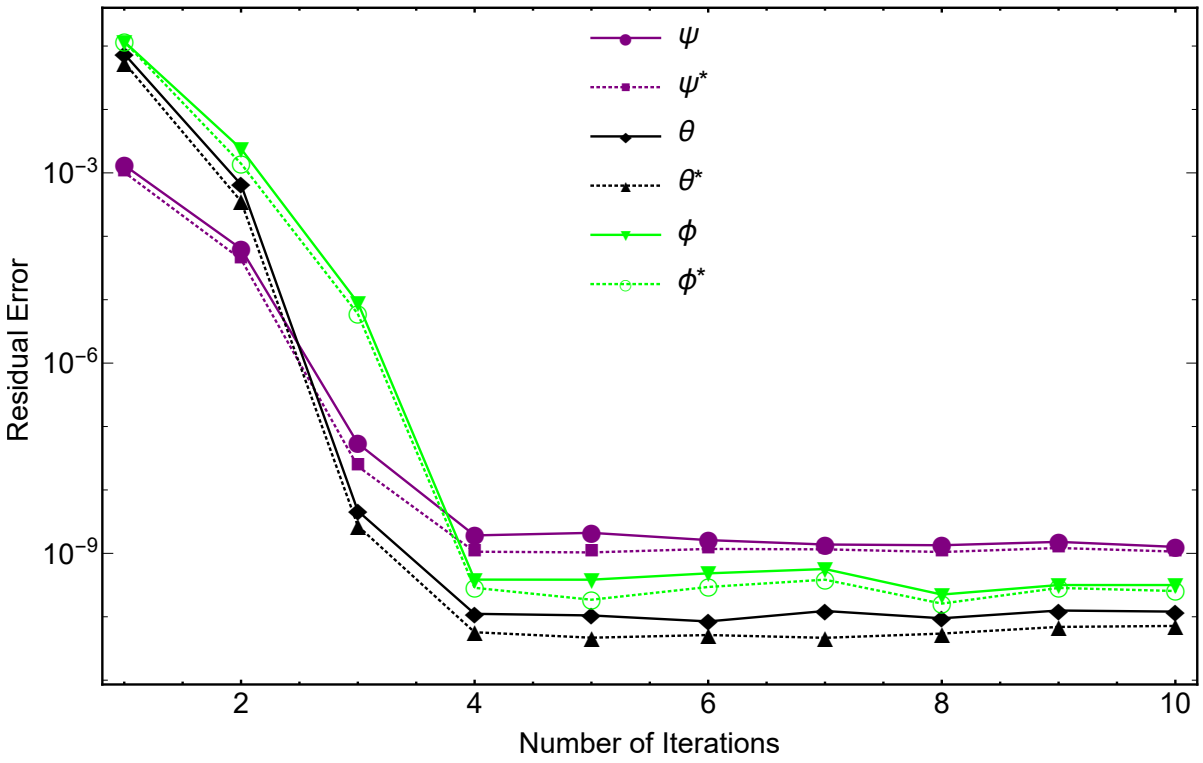


Figure 2: Residual error profile against the number of iterations

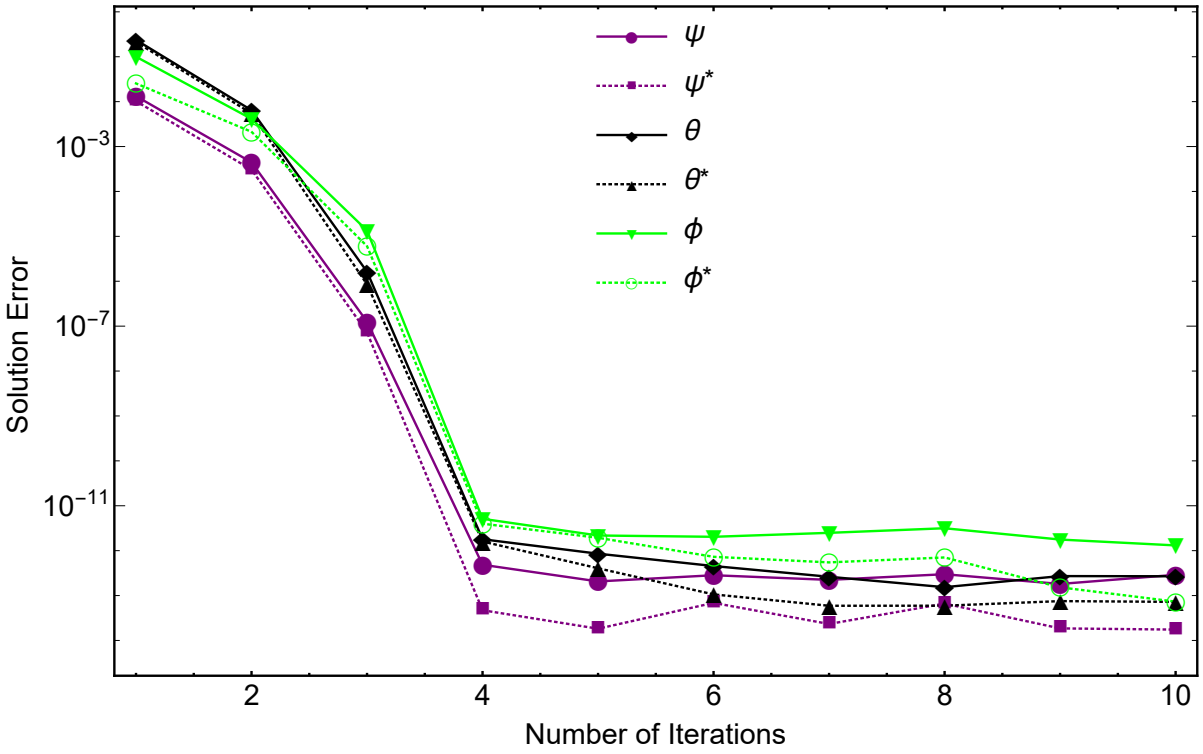


Figure 3: Solution error profile against the number of iterations

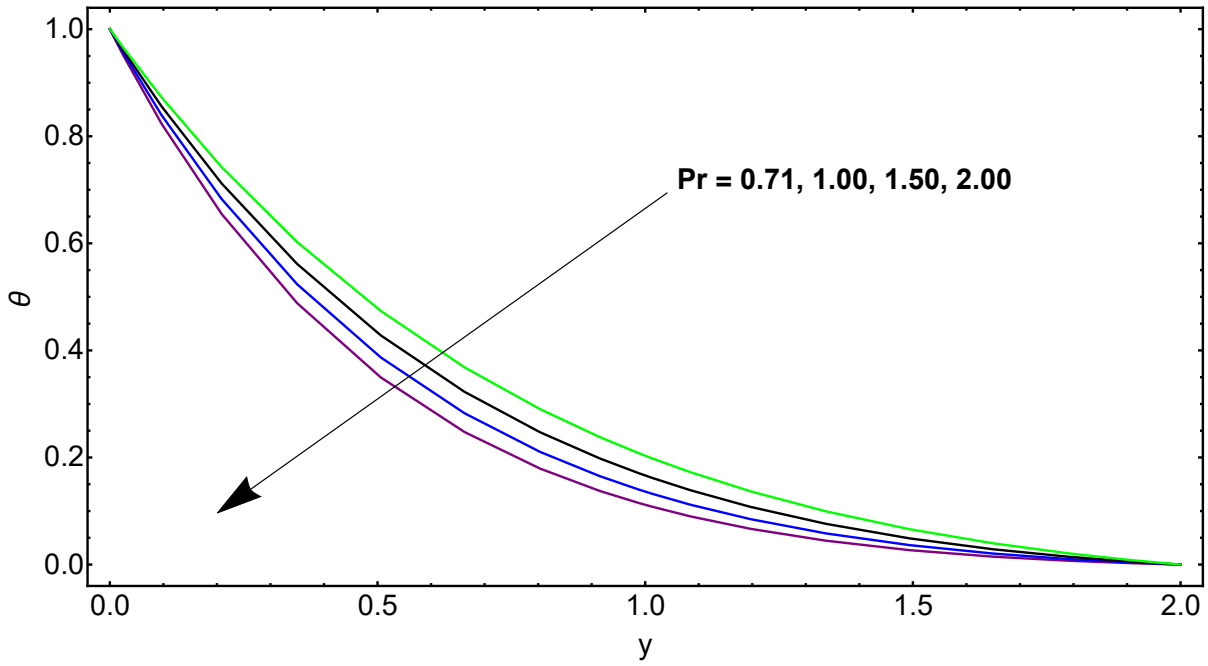


Figure 4: Temperature profiles for different values of Pr

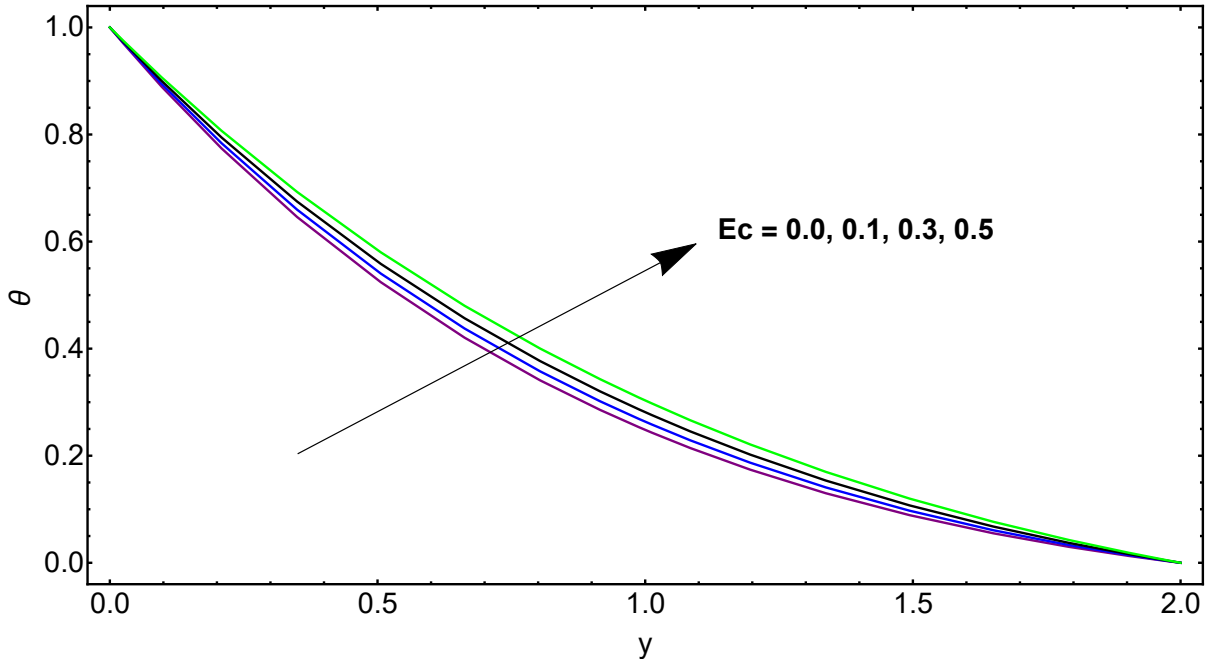


Figure 5: Temperature profiles for different values of Ec

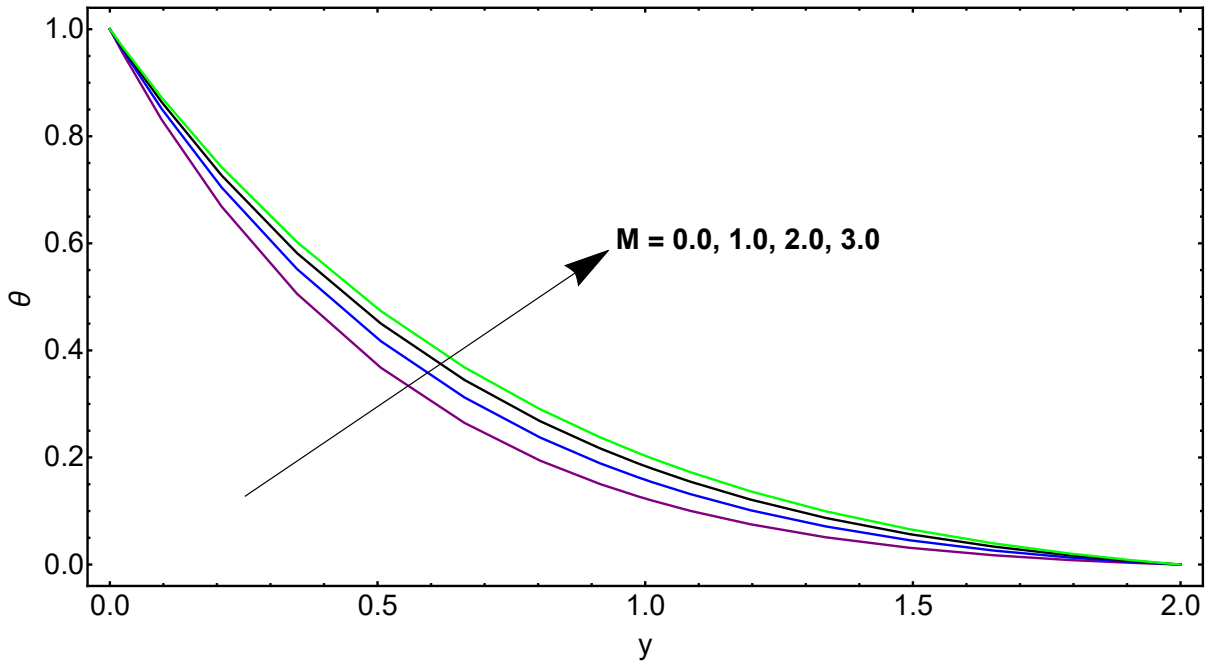


Figure 6: Temperature profiles for different values of M

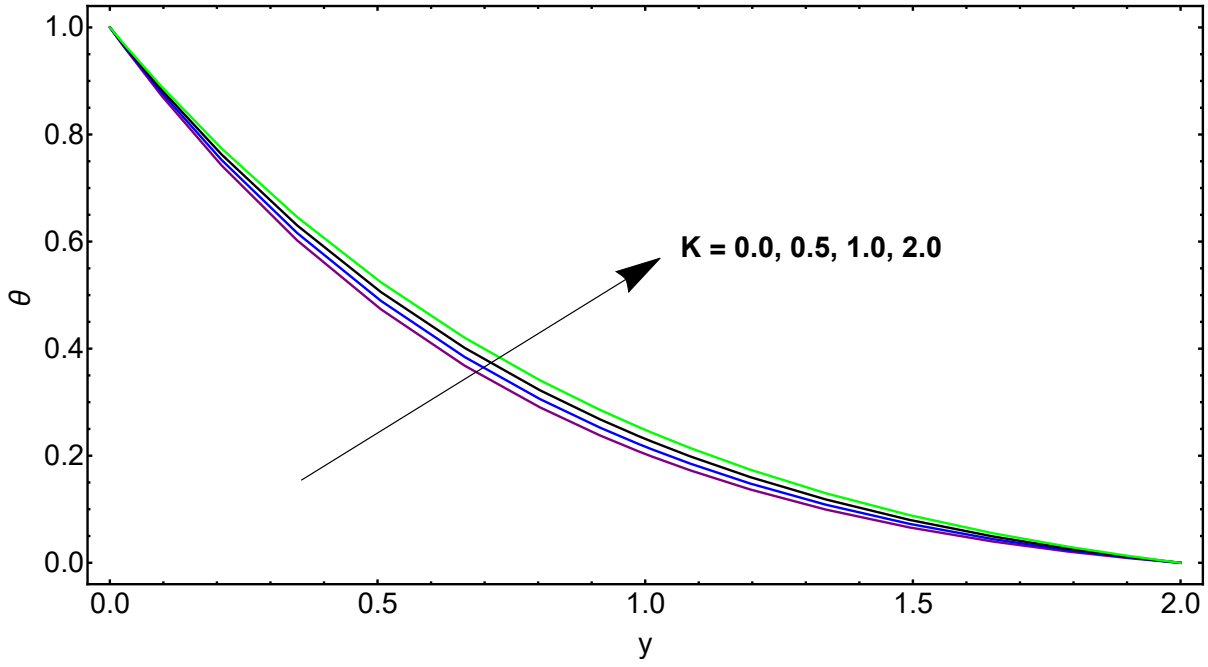


Figure 7: Temperature profiles for different values of K

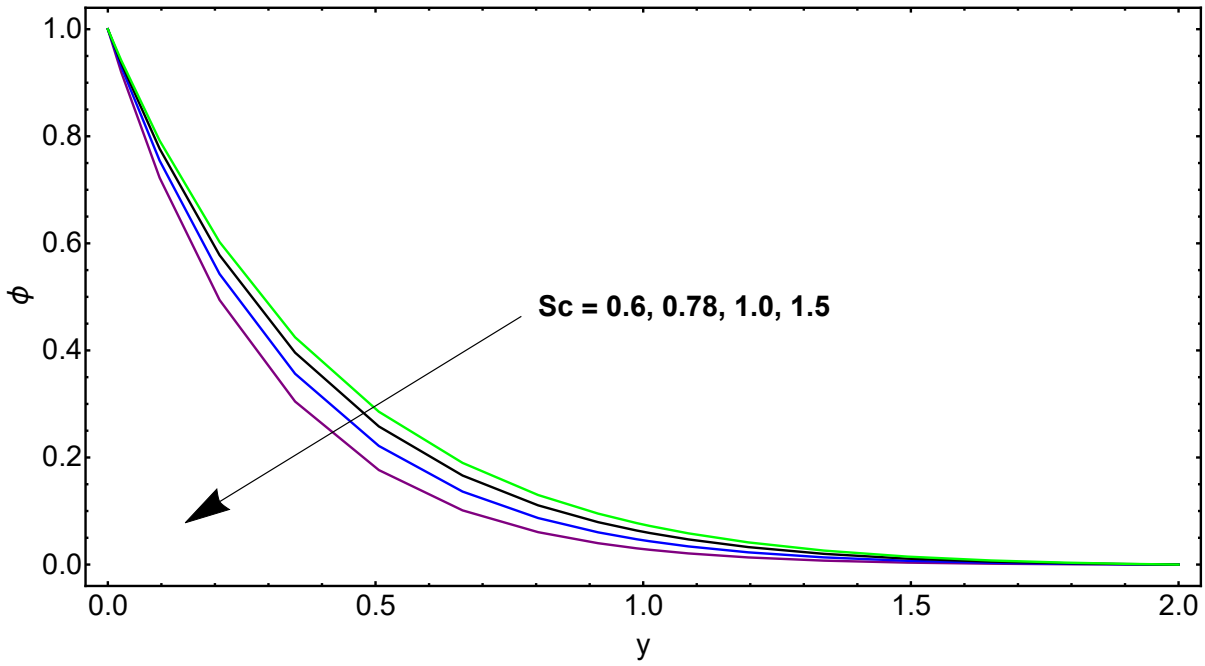


Figure 8: Concentration profiles for different values of Sc

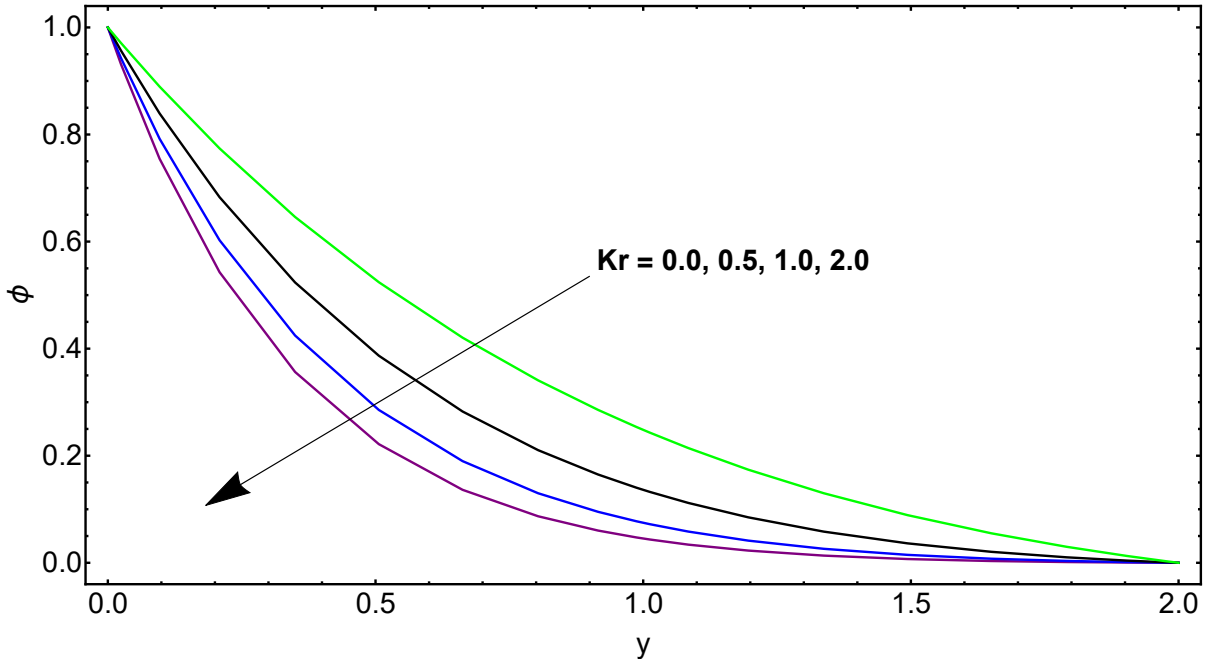


Figure 9: Concentration profiles for different values of Kr

Table 5: Numerical values of interpolation errors obtained when interpolating Eq.(69) in a single domain and overlapping domains.

N_x	N_y	q	p	CPU Time(Sec)	Bound	Interpolation Error
10	10	1	1	1.5760	1.4874×10^{-6}	2.8320×10^{-7}
10	**5**	**2**	**2**	**1.1437**	$\mathbf{8.2877 \times 10^{-5}}$	$\mathbf{3.8721 \times 10^{-5}}$
20	20	1	1	6.7860	2.0460×10^{-8}	6.4615×10^{-14}
20	**10**	**2**	**2**	**4.7495**	$\mathbf{1.7551 \times 10^{-8}}$	$\mathbf{1.8046 \times 10^{-13}}$
40	40	1	1	38.0700	6.3882×10^{-9}	1.7955×10^{-13}
40	**20**	**2**	**2**	**15.3842**	$\mathbf{5.3520 \times 10^{-10}}$	$\mathbf{2.8411 \times 10^{-16}}$

6. Conclusion

We have described and successfully applied an overlapping grid based spectral collocation method to solve typical examples of two-dimensional nonlinear elliptic partial differential equations defined on large rectangular domains. We have demonstrated that in comparison to the single domain approach applying the spectral collocation method on overlapping grids gives superior accuracy and computational efficiency when applied to solve nonlinear elliptic PDEs defined on large domains. Improved accuracy is attributed to the small condition numbers of coefficient matrices in the overlapping grids approach. The computational efficiency of the proposed numerical algorithm rests on the ease of inverting the sparse coefficient matrices of the linear algebraic systems of equations that are to be solved. The evaluations of the method have confirmed that the benefits of overlapping grid spectral collocation approach only manifest when the domain of approximation is large. Current numerical results show a good agreement with published results in the literature. Although the numerical scheme developed has been tested on nonlinear elliptic PDEs, the same concept could be adapted and applied to solve other class of nonlinear two-dimensional problems representing certain real-life phenomena. New error bound theorems and proofs on bivariate polynomial interpolation have been proven and tested. The error bound theorems unfold the theory behind the benefits of the proposed numerical method of solution and indeed, it is what motivated adoption of current numerical technique to nonlinear elliptic problems.

7. Acknowledgement

The authors are thankful to the University of KwaZulu-Natal for providing essential research resources.

Declaration of interest:

There is no conflict of interest regarding publication of this paper.

References

[1] A.I. Fedoseyev, M.J. Friedman, and E.J. Kansa, *Improved Multiquadric Method for Elliptic Partial Differential Equations Via PDE Collocation on the Boundary*, Computers & Mathematics with Applications, 43(2002), pp. 439455.

[2] A.D. Polyanin, and V.F. Zaitsev, *Handbook of Nonlinear Partial Differential Equations*, Chapman & Hall/CRC, London, 2004, ISBN 1-58488-355-3.

[3] A. Tadmor, *A Review of Numerical Methods for Nonlinear Partial Differential Equations*, Bulletin of the American Mathematical Society, 49(2012), pp. 507-554.

[4] B. Carnahan, H.A. Luther, and J.O. Wilkes, *Applied Numerical Methods*, John Wiley and Sons, New York, 1969.

[5] S. Klaus, *Numerical Methods for Nonlinear Elliptic Partial Differential Equation*, Oxford University Press, New York, 2010.

[6] T. Liszka, and J. Orkisz, *The Finite Difference Method at Arbitrary Irregular Grids and its Application in Applied Mechanics*, Computer & Structures, 11(1980), pp. 83-95.

[7] M. Avellaneda, T. Hou, and G. Papanicolaou, *Finite Difference Approximations for Partial Differential Equations With Rapidly Oscillating Coefficients*, Mathematical Modelling and Numerical Analysis, 25(1991), pp. 693710.

[8] C. Conca, and S. Natesan, *Numerical Methods for Elliptic Partial Differential Equations with Rapidly Oscillating Coefficients*, Computer Methods in Applied Mechanics and Engineering, 192(2003). pp. 4776.

[9] H.F. Chan, C.M. Fan, and C.W. Kuo, *Generalized Finite Difference Method for Solving Two-Dimensional Nonlinear Obstacle Problems*, Engineering Analysis with Boundary Elements, 37(2013), pp. 1189-1196.

[10] P.S. Jensen, *Finite Difference Technique for Variable Grids*, Computer & Structures, 2(1972), pp. 17-29.

[11] O. Lakkis, and T. Pryer, *A Finite Element Method for Nonlinear Elliptic Problems*, SIAM Journal on Scientific Computing, 35(2013), pp.2025-2045.

[12] T. Hou, and X. Wu, *A Multiscale Finite Element Method for Elliptic Problems in Composite Materials and Porous Media*, Computational Physics, 134(1997), pp. 169189.

[13] A. Matache, I. Babuska, and C. Schwab, *Generalized p-FEM in Homogenization*, Numerical Mathematics, 86(2000), pp. 319375.

[14] B.K. Ghimire, H.Y. Tian, and A.R. Lamichhane, *Numerical Solutions of Elliptic Partial Differential Equations Using Chebyshev Polynomials*, Computers & Mathematics with Applications, 72(2016), pp. 1042-1054.

[15] L. Yi, and Z. Wang, *Legendre Spectral Collocation Method for Second-order Nonlinear Ordinary/Partial Differential Equations*, , Discrete & Continuous Dynamical Systems - Series B, 19(2014), pp. 299-322.

[16] M.G. Macaraeg, and C.L. Streett, *Improvements in Spectral Collocation Discretization Through a Multiple Domain Technique*, Applied Numerical Mathematics, 2(1986), pp. 95-108.

[17] W.S. Don, and A. Solomonoff, *Accuracy and Speed of Computing the Chebyshev Collocation Derivative*, SIAM, Journal of Scientific Computing, 16(1995), pp. 1253-1268.

[18] F.J. Narcowich, and J.D. Ward, *Norms of Inverses and Condition Numbers of Matrices Associated with Scattered Data*, Journal of Approximation Theory, 64(1991), pp. 69-94.

[19] D. Gottlieb, and S.A. Orszag, *Numerical Analysis of Spectral Methods*, Society for Industrial and Applied Mathematics, Philadelphia, 1977.

[20] P. Demaret, and M.O. Deville, *Chebyshev Collocation Solutions of the Navier-Stokes Equations using Multi-domain Decomposition and Finite Element Preconditioning*, Computational Physics, 95(1991), pp. 359-386.

[21] A. Pinelli, A. Vacca, and A. Quarteroni, *A Spectral Multi-domain Method for the Numerical Simulation of Turbulent Flows*, Computational Physics, 136(1997), pp. 546-558.

[22] M.H. Chou, *A Multigrid Pseudospectral Method for Steady Flow Computation*, Numerical Methods in Fluids, 43(2003), pp. 25-42.

[23] W. Zhang, C.H. Zhang, and G. Xi, *An Explicit Chebyshev Pseudospectral Multigrid Method for Incompressible NavierStokes Equations*, Computers & Fluids, 39(2010), pp. 178-188.

[24] J.L. Vazquez, *Domain of Existence and Blow-up for the Exponential Reaction-Diffusion Equation*, Indiana University Mathematics Journal, (48)1999, pp. 677-709.

[25] E.A. Zabolotskaya, and R.V. Khokhlov, *Quasi-planes Waves in the Nonlinear Acoustic of Confined Beams*, Soviet physics Acoustics, 15(1969), pp. 35-40.

[26] G.V.R. Reddy, S.M. Ibrahim, and V.S. Bhagavan, *Similarity Transformations of Heat and Mass Transfer Effects on Steady MHD Free Convection Dissipative Fluid Flow Past an Inclined Porous Surface With Chemical Reaction*, Naval Architecture & Marine Engineering, 11(2014), pp. 157-166.

[27] R.E. Bellman, and R.E. Kalaba, *Quasilinearization and Nonlinear Boundary-Value Problems*, Elsevier Publishing Company, New York, 1965.

[28] H.H. Yang, B.R. Seymour, and B.D. Shizgal, *A Chebyshev Pseudospectral multi-Domain Method for Flow Past a Cylinder, up to Re=150*, Computers & Fluids, 23(1994), pp. 829-851.

[29] L.N. Trefethen, *Spectral Methods in MATLAB*, SIAM, Philadelphia, 2000.

[30] [30] M. Gasca and T. Sauer, *On the History of Multivariate Polynomial Interpolation*, Journal of Computational and Applied Mathematics, 122(2000), pp. 23-35.

[31] A.H Bhrawy, *A Highly Accurate Collocation Algorithm for* 1+1 *and* 2+1 *Fractional Percolation Equations*, Vibration and Control, 5(2015), pp. 1-23.

[32] H.E. Salzer, *Converting Interpolation Series into Chebyshev Series by Recurrence Formulas*, Mathematics of Computation, 30(1976), pp. 295-302.

[33] R. Baltensperger, and J.P. Berrut, *The Errors in Calculating the Pseudospectral Differentiation Matrices for Chebyshev-Gauss-Lobatto Points*, Computers & Mathematics with Applications, 37(1999), pp. 41-48.

Chapter 6

A New Spectral Collocation Method of Solution for Two-Dimensional Initial-Boundary Value Problems

This chapter is the climax of the study. We propose herein an accurate and computationally efficient overlapping grid based multidomain trivariate spectral collocation method for solving two-dimensional nonlinear initial-boundary value problems defined over large spatial and time domains. The numerical scheme is tested on typical examples of two-dimensional nonlinear parabolic PDEs reported in the literature as a single equation or system of equations. Numerical results confirm that the method is highly accurate and computationally efficient when applied to solve two-dimensional initial-boundary value problems defined on large spatial domains and time intervals when compared to the standard method on a single domain. Adopting the structure of the previous chapters, this particular one ends with a discussion of error bound theorems and proofs on trivariate polynomial interpolation. Extension of the trivariate spectral collocation method described in this chapter to the solution of three-dimensional problems is achievable in a straight forward manner. A manuscript on the single domain version of this method has been published in the Journal of Applied Mathematics and Computation. This paper is attached, as an appendix, at the end of this thesis. A paper detailing incorporation of a domain decomposition technique on trivariate spectral collocation method, then, constitutes chapter 6, thereby, retaining the central focus of the thesis.

Improved trivariate spectral collocation method of solution for two-dimensional nonlinear initial-boundary value problems via domain decomposition

F.M. Samuel[1,a,c], S.S. Motsa[1,a,b]

[a] *University of KwaZulu-Natal, School of Mathematics, Statistics, and Computer Science, Private Bag X01, Scottsville, 3209, South Africa;*
[b] *University of Swaziland, Department of Mathematics, Private Bag 4, Kwaluseni, Swaziland;*
[c] *Taita Taveta University, Department of Mathematics, 635-80300, Voi, Kenya.*

Abstract

In this article, we propose an accurate and computationally efficient overlapping grid based multidomain trivariate spectral collocation method for solving two-dimensional nonlinear initial-boundary value problems over large time intervals. In the current solution approach, the quasi-linearization method is used to simplify the nonlinear PDEs. The spatial domain is decomposed into a sequence of overlapping subintervals of equal length whereas the time domain is broken into equal non-overlapping subintervals. Trivariate Lagrange interpolating polynomials constructed using Chebyshev-Gauss-Lobatto (CGL) points are used to approximate the solutions to the nonlinear PDEs. A purely spectral collocation-based discretization is employed on the two space variables and the time variable on each subinterval to yield a system of linear algebraic equations that is solved. The PDEs are solved simultaneously across all subintervals in space, but independently on each subinterval in time, with the continuity condition been applied to obtain initial conditions in subsequent time subintervals. The numerical scheme is tested on typical examples of two dimensional nonlinear parabolic PDEs reported in the literature as a single equation or system of equations. Numerical results confirm that the proposed solution approach is highly accurate and computationally efficient, when applied to solve two-dimensional initial-boundary value problems defined on large time intervals and large spatial domains when compared with the standard method on a single domain. In addition, it is demonstrated that the overlapping grids technique preserves the stability of the numerical scheme when solving fluid mechanics problems for large Reynolds numbers. The new error bound theorems and proofs on trivariate polynomial interpolation that we present support findings from the numerical simulations.

*Corresponding author: felexmutua@gmail.com

Keywords: Trivariate Lagrange Interpolating Polynomials, Spectral Collocation, Two-dimensional Parabolic PDEs, Overlapping Grids, Non-overlapping Grids, CGL Points.
2010 MSC: 65M15, 65M55, 65M70.

1. Introduction

Spectral collocation-based methods, have since their first appearance, gained popularity for the numerical approximation of the solution of partial differential equations, owing to their superior accuracy when applied to solve problems with smooth solutions [1]. They are particularly appropriate for approximating solutions of nonlinear PDEs defined on regular geometries, and they require few grid points to achieve highly accurate results [2]. Despite the benefits of these spectral collocation methods, previous applications of purely spectral collocation methods have focused only on the solutions of ordinary differential equations For instance, or partial differential equations involving two independent variables [3]. For instance, Zhao et al. [4] and Tadmor [5] indicate extensive examples in the literature where the spectral collocation methods have been applied successfully on such problems with highly accurate results being achieved in a computationally efficient manner. In nearly all cases these results have been achieved through the application of spectral collocation discretization on the space variables and finite difference discretization on the time variable. The few noticeable examples include [6] and references give therein, where spectral collocation methods have been applied to obtain numerical solutions of two-dimensional time-dependent PDEs. It is well known that finite difference methods require many grid points to yield accurate results which can hardly match those obtained when spectral collocation methods are applied on PDEs defined on simple geometries particularly if the underlying solutions are smooth. As observed in [7], the accuracy of a single domain based spectral collocation method deteriorates when the method is applied to solve PDEs defined over a large time interval even with a large number of grid points. As shown by Motsa et al. [8], the utility of the spectral collocation method can be improved by decomposing the large time domain into smaller non-overlapping subintervals and solving the differential equation independently for each of these subintervals in time. Further, as will be shown later, decomposing the spatial domains into a sequence of overlapping subintervals improves accuracy when solving differential equations defined over large spatial domains and presents a stable numerical scheme when solving a differential equation for very large or small values of the parameters. Motivated by these facts, we propose an overlapping grid based multidomain trivariate spectral collocation method for solving nonlinear two-dimensional time-dependent PDEs defined

on large rectangular domains over large time intervals.

Typical examples of two-dimensional nonlinear time-dependent PDEs considered in this study include the nonlinear PDEs given as single equations that describe the problem of unsteady two-dimensional heat and mass transfer, firstly, in quiescent media with a chemical reaction [9], and secondly, with power-law temperature-dependent thermal conductivity [10]. Heat and mass transfer phenomena are found throughout the physical world and the industrial domain. For nonlinear PDEs described as systems of equations we consider the case of the coupled two-dimensional Burgers system [11] and the two-dimensional reaction-diffusion Brusselator system [12]. The Burgers equation is a fundamental PDEs from fluid mechanics, that appears in various areas of applied mathematics, such as modeling of hydrodynamic turbulence, shock waves theory, and traffic flow problems. It also describes the sedimentation of particles in fluid suspensions under the effect of gravity, transport, and dispersion of pollutants in rivers [13]. The second system of PDEs considered is the Brusselator system arising in the mathematical modeling of chemical systems such as enzymatic reactions, and in plasma and laser physics in multiple coupling between certain modes [14]. The Brusselator model is also evident in the formation of the ozone layer through a triple collision of oxygen atoms. The problems considered here have exact solutions and have been reported in the literature to be very useful in testing newly developed numerical methods of solution for nonlinear partial differential equations arising in modeling of various aspects of the real world. We, therefore, consider them appropriate to demonstrate the effectiveness of the current method of solution.

Exact solutions of equations modelling two-dimensional heat and mass transfer problem in quiescent media with chemical reaction were discussed by Polyanin and Zaitsev [9]. The problem has been solved numerically using an implicit finite-difference method by Chamkha and Aly [15]. On the other hand, the problem of two-dimensional heat and mass transfer with power-law temperature-dependent thermal conductivity was examined by Pamuk and Pamuk [10] who obtained a particular exact solution using the Adomian decomposition method. Further numerical methods of solution for this problem can be found in the references they give. Burgers [11] pioneered the investigation of the mathematical properties of Burgers equation. The analytical solution of the unsteady two dimensional coupled Burgers equation was first given by Fletcher [16] using the Hopf-Cole transformation, and in, [17], the differential transformation method was applied to obtain the analytical solution of a coupled unsteady Burgers equation. Numerical solution of the Burgers equation is,

therefore, a natural first step towards developing methods for the computation of complex flows. Accordingly, the Burgers equation has been used intensively to test new approaches in computational fluid dynamics. For instance, a numerical method based on local discontinuous Galerkin finite element was analyzed in [18] to solve the two-dimensional Burgers equation. The local radial basis functions collocation method to approximate the numerical solution of the transient coupled Burgers equation was examined in [19]. In another development, the Brusellor model has been extensively studied, both numerically and analytically. For example, Twizell et al. [20] developed a second-order finite difference method for the numerical solution of the initial-boundary value problems of the Brusselator model. Khani et al. [21] found exact solutions of the Brusselator reaction-diffusion model using the exp-function method and Biazar and Ayati [22] obtained an approximate solution to the Brusselator system by applying the Adomian decomposition method.

In this paper, a purely spectral collocation method, namely, the overlapping grids multidomain trivariate spectral collocation method is introduced and applied to solve two-dimensional initial-boundary value problems defined on over large space and time intervals. The solution process involves, first, the simplification of the PDE using the quasi-linearization method. Next, the spatial domain is decomposed into a sequence of equal overlapping subintervals and the large time domain is decomposed into equal non-overlapping subintervals. The solution to the linearized PDE is assumed to be a trivariate Lagrange interpolating polynomials constructed on Chebyshev-Gauss-Lobatto points defined on each subinterval. The PDE is discretized in all space variables and time variable using the spectral collocation method to yield a system of linear algebraic equations that are then solved iteratively. The solution of the PDE is computed simultaneously across all subintervals in space and independently at each subinterval in time. Applying the continuity condition allows us to obtain the initial condition for the second to last subintervals. The current numerical method is tested using typical examples of initial-boundary value problems reported in the literature. The accuracy of the numerical scheme is assessed by computing the absolute difference between the numerical results with the exact solutions, these are presented in tabular and graphical form and discussed. Numerical results obtained using the current solution approach are also compared with those obtained using the standard single domain based trivariate spectral collocation method. Findings from numerical simulations show that the current method yields highly accurate results in a computationally efficient manner when applied to problems defined on a large time interval, large spatial domains, and obtaining numerical approximations of solutions to the differential equation

for large parameter values. To the best of our knowledge, the spectral collocation-based method with Chebyshev-Gauss-Lobatto points has not been applied simultaneously on overlapping grids in two-dimensional space and non-overlapping grids in time. The new theoretical results of error bound theorems support the finding of the numerical simulations.

The rest of this paper is organized as follows. In Section 2, we describe the overlapping grids based multidomain trivariate spectral collocation method for approximating the solution of two-dimensional nonlinear initial-boundary value problems described by single equations or systems of equations. In Section 3, the error bound theorems and proofs emanating from trivariate Lagrange interpolating polynomials constructed on Chebyshev-Gauss-Lobatto grid points emerge. In Section 4, we give four test examples where the numerical method is applied to demonstrate its applicability. Section 5 is devoted to results and discussion. In Section 6, we summarize the findings and point out the direction of future work.

2. The method of solution

In this section, we describe the algorithm for solving two-dimensional partial differential equations of initial-boundary value problems type. The present investigation focuses on the partial differential equations of the second order. For purposes of simplicity, this section is divided into two subsections. In the first subsection, we construct numerical algorithms for solving nonlinear PDEs that are expressible as a single equation and in the second subsection, the idea is extended to systems of nonlinear PDEs.

2.1. ***Method of solution for a single nonlinear PDE***

In this subsection, the overlapping grids based multidomain spectral collocation algorithm for solving two-dimensional partial differential equations of initial-boundary value problems type given as a single nonlinear equation is described. To illustrate the solution process, we consider a general second-order nonlinear PDE that takes the form

$$\frac{\partial u}{\partial t} = F\left(\frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, u\right), \quad (x,y) \in (a,b)\times(c,d),\ t \in (0,T), \tag{1}$$

where F is a nonlinear operator operating on the unknown function u and its first and second order spatial derivatives. Eq.(1) is solved subject to the boundary conditions

$$\begin{aligned} &\alpha_1^a \frac{\partial u}{\partial x}(a,y,t) + \alpha_0^a u(a,y,t) = f_a(y,t), \quad \alpha_1^b \frac{\partial u}{\partial x}(b,y,t) + \alpha_0^b u(b,y,t) = f_b(y,t), \\ &\beta_1^c \frac{\partial u}{\partial y}(x,c,t) + \beta_0^c u(x,c,t) = g_c(x,t), \quad \beta_1^d \frac{\partial u}{\partial y}(x,d,t) + \beta_0^d u(x,d,t) = g_d(x,t), \end{aligned} \tag{2}$$

where $\alpha_1^a,\ \alpha_0^a,\ \alpha_1^b,\ \alpha_0^b,\ \beta_1^c,\ \beta_0^c,\ \beta_1^d,\ \beta_0^d$ are known constants and $f_a(y),\ f_b(y),\ g_c(x),\ g_d(x)$, are known functions. The initial condition for this problem is given as

$$u(x, y, 0) = h(x, y). \tag{3}$$

The solution process involves the stages given in the subsections below.

2.1.1. The quasi-linearization method

The PDE Eq.(1) is first simplified using the quasi-linearization method (QLM) of Bellman and Kalaba [24]. The QLM is based on the Newton-Raphson method and is constructed from the linear terms of a Taylor series expansion about an initial approximation to the solution. The QLM assumes that the difference between solutions at two successive iterations, denoted by $u_{s+1} - u_s$, is very small. In particular, the QLM is comparable to giving the linear approximation of a function of several variables, where the derivatives of a different order, and the previous approximation to the solution, assume the respective roles of independent variables and the functional value at the reference point. Finer details about the linear approximation of functions can be found in any elementary book on differential calculus. Applying the QLM on Eq.(1) we obtain

$$\delta_{4,s}\frac{\partial^2 u_{s+1}}{\partial x^2} + \delta_{3,s}\frac{\partial^2 u_{s+1}}{\partial y^2} + \delta_{2,s}\frac{\partial u_{s+1}}{\partial x} + \delta_{1,s}\frac{\partial u_{s+1}}{\partial y} + \delta_{0,s}u_{s+1} - \dot{u}_{s+1} = R_s, \tag{4}$$

where

$$\begin{aligned} &\delta_{4,s} = \frac{\partial F}{\partial\left(u_{xx}\right)_s},\ \delta_{3,s} = \frac{\partial F}{\partial\left(u_{yy}\right)_s},\ \delta_{2,s} = \frac{\partial F}{\partial\left(u_{x}\right)_s},\ \delta_{1,s} = \frac{\partial F}{\partial\left(u_{y}\right)_s},\ \delta_{0,s} = \frac{\partial F}{\partial\left(u\right)_s}, \\ &R_s = \delta_{4,s}(u_{xx})_s + \delta_{3,s}(u_{yy})_s + \delta_{2,s}(u_x)_s + \delta_{1,s}(u_y)_s + \delta_{0,s}u_s - F_s. \end{aligned} \tag{5}$$

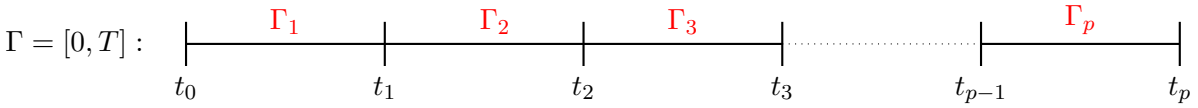
The subscripts in u_x and u_{xx}, denote, respectively, the first and second partial derivatives of u with respect to x. Similarly, the first and the second derivatives with respect to y are denoted by u_y and u_{yy}, respectively. The dot in $\dot{u}$ denotes the derivative with respect to t and s signifies the previous iteration.

2.1.2. Domain decomposition and discretization

If we let $t \in \Gamma$ where $\Gamma = [0, T]$. The domain of approximation in the time direction Γ is decomposed into p equal non-overlapping subintervals as

$$\Gamma_\tau = [t_{\tau-1}, t_\tau], \quad t_{\tau-1} < t_\tau, \quad t_0 = 0, \quad t_p = T, \quad \tau = 1, 2, \ldots, p. \tag{6}$$

The domain decomposition in the t variable is illustrated below.



The subdomain $t \in [t_{\tau-1}, t_\tau]$ in each of the τ^{th} subintervals is transformed into $\hat{t} \in [-1, 1]$ using the linear map

$$\hat{t}(t) = \frac{2}{t_\tau - t_{\tau-1}} \left[t - \frac{1}{2}(t_\tau + t_{\tau-1}) \right], \quad t \in [t_{\tau-1}, t_\tau], \quad \hat{t} \in [-1, 1], \quad \tau = 1, 2, \ldots, p, \tag{7}$$

before the spectral collocation is applied. The domain of approximation in each time subinterval is further discretized into $N_t + 1$ Chebyshev Gauss-Lobatto nodes defined in [25] as

$$\{\hat{t}_k\}_{k=0}^{N_t} = \cos\left(\frac{k\pi}{N_t}\right). \tag{8}$$

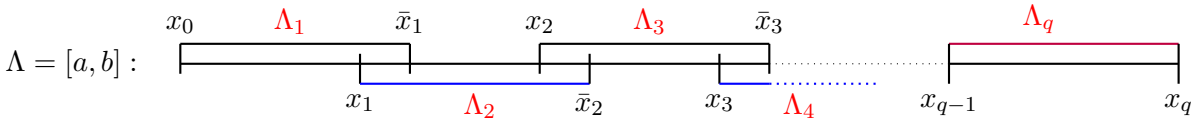
The entire grid in the t variable can be represented as

$$\{0 = t_{N_t}^{(1)}, \ldots, t_0^{(1)} = t_{N_t}^{(2)}, \ldots, t_0^{(\tau-1)} = t_{N_t}^{(\tau)}, \ldots, t_0^{(p)} = T, \ldots, \ 2 \leq \tau \leq p\}. \tag{9}$$

The superscripts and subscripts are the subinterval and grid points indices, respectively. The spatial domain $x \in [a, b]$ is decomposed into q overlapping subintervals of equal length as

$$\Lambda_l = [x_{l-1}, \bar{x}_l], \quad x_{l-1} < x_l < \bar{x}_l, \quad x_0 = a, \ \bar{x}_q = b, \quad l = 1, 2, \ldots, q, \tag{10}$$

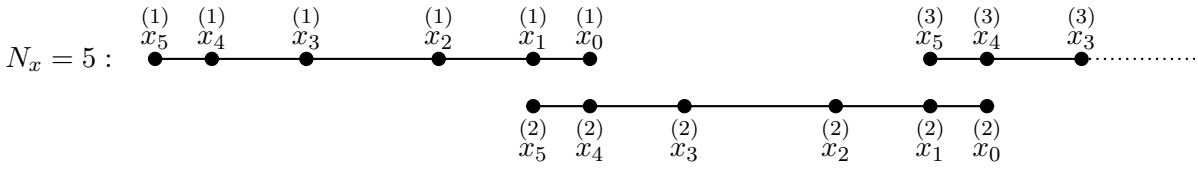
where $x_l < \bar{x}_l$, depicts the overlapping nature. Pictorially, this overlapping domain decomposition in x can be represented as



The computational domain $[x_{l-1}, \bar{x}_l]$ in the l^{th} subinterval is then transformed into $\hat{x} \in [-1, 1]$ where the standard Chebyshev differentiation matrix is defined by applying the linear map

$$\hat{x}(x) = \frac{2}{\bar{x}_l - x_{l-1}} \left[x - \frac{1}{2}(\bar{x}_l + x_{l-1}) \right], \quad x \in [x_{l-1}, \bar{x}_l], \quad \hat{x} \in [-1, 1]. \tag{11}$$

Further, each subinterval is discretized into $N_x + 1$ Chebyshev Gauss Lobatto points. The subintervals in the decomposed domain overlap in such a manner that the last two points in the Λ_l subinterval overlap with the first two points in the Λ_{l+1} subinterval and they remain common. For illustrative purpose, the case $N_x = 5$ is considered.



In general, the set of grid points over the entire x domain can be represented as

$$\{a = x_{N_x}^{(1)}, \ldots, x_1^{(1)} = x_{N_x}^{(2)}, x_0^{(1)} = x_{N_x-1}^{(2)}, \ldots, x_1^{(l-1)} = x_{N_x}^{(l)}, x_0^{(l-1)} = x_{N_x-1}^{(l)}, \ldots, x_0^{(q)} = b,\ 2 \leq l \leq q\}. \tag{12}$$

Similarly, the spatial domain in y variable $[c, d]$ is decomposed into m equal overlapping subintervals which are further discretized into $N_y + 1$ collocation points. The subdivision of the computational domain in y is illustrated by

$$\Omega_\zeta = [y_{\zeta-1}, \bar{y}_\zeta]\,,\ \ y_{\zeta-1} < y_\zeta < \bar{y}_\zeta,\ \ y_0 = c,\ \bar{y}_m = d,\ \ \zeta = 1, 2, \ldots, m, \tag{13}$$

and the overlapping nature is analogous to that in the x variable. The computational domain $[y_{\zeta-1}, \bar{y}_\zeta]$ in the ζ^{th} subinterval is transformed into $\hat{y} \in [-1, 1]$ by applying the linear transformation

$$\hat{y}(y) = \frac{2}{\bar{y}_\zeta - y_{\zeta-1}} \left[y - \frac{1}{2}\left(\bar{y}_\zeta + y_{\zeta-1}\right) \right],\ \ y \in [y_{\zeta-1}, \bar{y}_\zeta],\ \ \hat{y} \in [-1, 1], \tag{14}$$

before spectral collocation is applied. The grid points in the entire y domain can be represented as

$$\{c = y_{N_y}^{(1)}, \ldots, y_1^{(1)} = y_{N_y}^{(2)}, y_0^{(1)} = y_{N_y-1}^{(2)}, \ldots, y_1^{(\zeta-1)} = y_{N_y}^{(\zeta)}, y_0^{(\zeta-1)} = y_{N_y-1}^{(\zeta)}, \ldots, y_0^{(m)} = d,\ 2 \leq \zeta \leq m\}. \tag{15}$$

We remark that the number of subintervals q in x need not to be equal to the number of subintervals m in y. The ordering of grid points as illustrated in Eq.(12) and Eq.(15) signifies that the spectral collocation is done from right to left of the subinterval. The grid points in the l^{th} subinterval in x and the ζ^{th} subinterval in y variable are defined in [25] by

$$\{\hat{x}_i\}_{i=0}^{N_x} = \cos\left(\frac{i\pi}{N_x}\right),\ \text{ and }\ \{\hat{y}_j\}_{j=0}^{N_y} = \cos\left(\frac{j\pi}{N_y}\right). \tag{16}$$

To obtain explicit expression of the length of each subinterval $L = \bar{x}_l - x_{l-1}$ in x and $Z = \bar{y}_\zeta - y_{\zeta-1}$ in y in terms of the number of subintervals q and m, respectively, we solve

$$qL - L\,(q-1)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right) = b - a,\ \ mZ - Z\,(m-1)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_y}\right\}\right) = d - c, \tag{17}$$

to obtain

$$L = \frac{b-a}{q + (1-q)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right)},\ \text{and}\ Z = \frac{d-c}{m + (1-m)\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_y}\right\}\right)}. \tag{18}$$

Consequently, we obtain the following relations:

$$\bar{x}_l = x_l + L\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_x}\right\}\right),\ \ \bar{y}_\zeta = y_\zeta + Z\left(\frac{1}{2} - \frac{1}{2}\cos\left\{\frac{\pi}{N_y}\right\}\right). \tag{19}$$

Eq.(19) is used in defining the boundaries of the overlapping subintervals when performing discretization.

2.1.3. Spectral collocation

The overlapping grids multidomain trivariate spectral collocation method is implemented on the linearised QLM scheme Eq.(4) as detailed below. For simplicity, the label $\overset{(\tau)}{u}$, $\tau = 1, 2, \ldots, p$, will been used to distinguish solutions at different subintervals in time. We note that the application of the non-overlapping technique is limited to the time interval only. The PDE is solved independently at each subinterval in t, $\tau = 1, 2, \ldots, p$, and therefore we must solve

$$\begin{aligned} &\delta_{4,s}\frac{\partial^2 \overset{(\tau)}{u}_{s+1}}{\partial x^2} + \delta_{3,s}\frac{\partial^2 \overset{(\tau)}{u}_{s+1}}{\partial y^2} + \delta_{2,s}\frac{\partial \overset{(\tau)}{u}_{s+1}}{\partial x} + \delta_{1,s}\frac{\partial \overset{(\tau)}{u}_{s+1}}{\partial y} + \delta_{0,s}\overset{(\tau)}{u}_{s+1} - \frac{\partial \overset{(\tau)}{u}_{s+1}}{\partial t} = R_s, \\ &(x, y) \in [a, b] \times [c, d],\ t \in [t_{\tau-1}, t_\tau], \end{aligned} \tag{20}$$

subject to the boundary conditions

$$\begin{aligned} &\alpha_1^a \frac{\partial \overset{(\tau)}{u}}{\partial x}(a, y, t) + \alpha_0^a \overset{(\tau)}{u}(a, y, t) = f_a(y), \quad \alpha_1^b \frac{\partial \overset{(\tau)}{u}}{\partial x}(b, y, t) + \alpha_0^b \overset{(\tau)}{u}(b, y, t) = f_b(y), \\ &\beta_1^c \frac{\partial \overset{(\tau)}{u}}{\partial y}(x, c, t) + \beta_0^c \overset{(\tau)}{u}(x, c, t) = g_c(x), \quad \beta_1^d \frac{\partial \overset{(\tau)}{u}}{\partial y}(x, d, t) + \beta_0^d \overset{(\tau)}{u}(x, d, t) = g_d(x), \end{aligned} \tag{21}$$

and the initial condition

$$\overset{(1)}{u}(x, y, 0) = h(x, y), \quad \overset{(\tau)}{u}(x, y, t_{\tau-1}) = \overset{(\tau-1)}{u}(x, y, t_{\tau-1}),\ \tau = 2, \ldots, p, \quad (x, y) \in [a, b] \times [c, d]. \tag{22}$$

In the solution process, the approximate solution of the linearized QLM scheme Eq.(4) at each subinterval in time is assumed to be the trivariate Lagrange interpolating polynomial that takes the form

$$\overset{(\tau)}{u}(x, y, t) \approx \overset{(\tau)}{U}(x, y, t) = \sum_{p=0}^{N_x}\sum_{q=0}^{N_y}\sum_{r=0}^{N_t} \overset{(\tau)}{U}(x_p, y_q, t_r) L_p(x) L_q(y) L_r(t). \tag{23}$$

The spatial differentiation matrix in x is approximated at the collocation nodes, $(\hat{x}_i, \hat{y}_j, t_k)$ for $j = 0, 1, 2, \ldots, N_y$, and $k = 0, 1, 2, \ldots, N_t$, in the l-th subinterval as follows

$$\begin{aligned} \frac{\partial \overset{(\tau)}{u}}{\partial x}(\hat{x}_i, \hat{y}_j, \hat{t}_k) \approx \sum_{p=0}^{N_x}\sum_{q=0}^{N_y}\sum_{r=0}^{N_t} \overset{(\tau)}{U}(x_p, y_q, t_r) L_p'(\hat{x}_i) L_q(\hat{y}_j) L_r(\hat{t}_k) &= \sum_{p=0}^{N_x} \overset{(\tau)}{U}(x_p, y_j, t_k) L_p'(\hat{x}_i) = \overset{l}{\mathbf{D}}\overset{j}{\mathbf{U}}_k \\ &= \left(\frac{2}{L}\right) \overset{l}{\hat{\mathbf{D}}}\overset{j}{\mathbf{U}}_k, \end{aligned} \tag{24}$$

where $\overset{l}{\hat{\mathbf{D}}} = \left(\frac{L}{2}\right)\overset{l}{\mathbf{D}}$ is the standard first order Chebyshev differentiation matrix of size $(N_x+1)\times(N_x+1)$ as defined in [25]. The higher order differentiation matrices are obtained using matrix multiplication. The vector $\overset{j}{\mathbf{U}}_k$ is defined as

$$\overset{j}{\mathbf{U}}_k = [u(x_0,y_j,t_k), u(x_1,y_j,t_k),\ldots,u(x_{N_x},y_j,t_k)]^T,\ j=0,1,\ldots,N_y,\ k=0,1,\ldots,N_t, \tag{25}$$

where T denotes the matrix transpose. Similarly, the spatial differentiation matrix in y is approximated at the collocation points, $(\hat{x}_i,\hat{y}_j,\hat{t}_k)$ for $i=0,1,2,\ldots,N_x$, and $k=0,1,2,\ldots,N_t$, in the ζ-th subinterval as

$$\frac{\partial^{(\tau)}u}{\partial y}(\hat{x}_i,\hat{y}_j,\hat{t}_k) \approx \sum_{q=0}^{N_y}\overset{(\tau)}{U}(x_i,y_q,t_k)L'_q(\hat{y}_j) = \sum_{q=0}^{N_y}\overset{\zeta}{\bar{D}}_{j,q}\overset{q}{\mathbf{U}}_k = \sum_{q=0}^{N_y}\left(\frac{2}{Z}\right)\overset{\zeta}{\hat{\bar{D}}}_{j,q}\overset{q}{\mathbf{U}}_k, \tag{26}$$

where $\overset{\zeta}{\hat{\bar{D}}}_{j,q} = \left(\frac{Z}{2}\right)\overset{\zeta}{\bar{D}}_{j,q}$, $j,q=0,1,2,\ldots,N_y$, are entries of a standard first order Chebyshev differentiation matrix of size $(N_y+1)\times(N_y+1)$. The higher order differentiation matrix with respect to y can be obtained using matrix multiplication. Finally, the differentiation matrix in t at the collocation points $(\hat{x}_i,\hat{y}_j,\hat{t}_k)$, for $i=0,1,2,\ldots,N_x$, and $j=0,1,2,\ldots,N_y$, in the τ-th subinterval is approximated as

$$\begin{aligned}\frac{\partial^{(\tau)}u}{\partial t}(\hat{x}_i,\hat{y}_j,\hat{t}_k) &\approx \sum_{r=0}^{N_t}\sum_{q=0}^{N_y}\overset{(\tau)}{U}(x_i,y_j,t_r)L'_r(\hat{t}_k) = \sum_{r=0}^{N_t}\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,r}\overset{q}{\mathbf{U}}_r = \sum_{r=0}^{N_t}\sum_{q=0}^{N_y}\left(\frac{2}{t_\tau - t_{\tau-1}}\right)\hat{\hat{\bar{D}}}_{k,r}\overset{q}{\mathbf{U}}_r,\\ &\tau=1,2,\ldots,p,\end{aligned} \tag{27}$$

where $\hat{\hat{\bar{D}}}_{k,r} = \left(\frac{t_\tau - t_{\tau-1}}{2}\right)\bar{\bar{D}}_{k,r}$, $k,r=0,1,2,\ldots,N_t$, are entries of a standard first order Chebyshev differentiation matrix of size $(N_t+1)\times(N_t+1)$. We remark that the bar in $\bar{D}$ in Eq.(26) and double bar in $\bar{\bar{D}}$ in Eq.(27) distinguishes the differentiation matrices in y and t, respectively, from that in x. We note that in generating the sequence of vectors $\overset{j}{\mathbf{U}}_k$, $j=0,1,2,\ldots,N_y$, $k=0,1,2,\ldots,N_t$, the superscript j is varied of each subscript k. Such a pattern will be useful when arranging the system of linear algebraic equations to assemble coefficient matrices.

The solution in the spatial directions is computed simultaneously across all subintervals and the multidomain approach only becomes noticeable when assembling differentiation matrices. A key step towards the solution involve assembling the differentiation matrices in both space variables. To achieve this, we take x variable as an example, and since the last two points in the l^{th} subinterval

and the first two points in the $(l+1)^{th}$ subinterval overlap and remain common, we discard the rows corresponding to the recurrent grid points and assemble the Chebyshev differentiation matrix $\mathbf{D}$ (without the superscript l) for the overlapping grids multidomain approach in x as

$$\mathbf{D} = \begin{bmatrix}
\overset{q}{D}_{0,0} & \dots & \overset{q}{D}_{0,N_x-1} & \overset{q}{D}_{0,N_x} & & & & & & \\
\overset{q}{D}_{1,0} & \dots & \overset{q}{D}_{1,N_x-1} & \overset{q}{D}_{1,N_x} & & & & & & \\
\ddots & \ddots & \ddots & \ddots & & & & & & \\
\overset{q}{D}_{N_x-1,0} & \dots & \overset{q}{D}_{N_x-1,N_x-1} & \overset{q}{D}_{N_x-1,N_x} & & & & & & \\
 & & \ddots & \ddots & & & & & & \\
 & & \overset{2}{D}_{1,0} & \overset{2}{D}_{1,1} & \dots & \overset{2}{D}_{1,N_x-1} & \overset{2}{D}_{1,N_x} & & & \\
 & & \ddots & \ddots & \ddots & \ddots & \ddots & & & \\
 & & \overset{2}{D}_{N_x-1,0} & \overset{2}{D}_{N_x-1,1} & \dots & \overset{2}{D}_{N_x-1,N_x-1} & \overset{2}{D}_{N_x-1,N_x} & & & \\
 & & & & & \overset{1}{D}_{1,0} & \overset{1}{D}_{1,1} & \dots & \overset{1}{D}_{1,N_x} \\
 & & & & & \ddots & \ddots & \ddots & \ddots \\
 & & & & & \overset{1}{D}_{N_x-1,0} & \overset{1}{D}_{N_x-1,1} & \dots & \overset{1}{D}_{N_x-1,N_x} \\
 & & & & & \overset{1}{D}_{N_x,0} & \overset{1}{D}_{N_x,1} & \dots & \overset{1}{D}_{N_x,N_x}
\end{bmatrix}. \tag{28}$$

Here the empty entries of matrix $\mathbf{D}$ are zeros and $\overset{l}{D}$ represents the Chebyshev differentiation matrix in the l^{th} subinterval in space variable x. The size of matrix $\mathbf{D}$ is $(\delta+1)\times(\delta+1)$, where $\delta = N_x + (N_x-1)\times(q-1)$. The differentiation matrix in y is assembled using $\overset{\zeta}{\bar{\mathbf{D}}}$ in a similar manner to obtain a matrix $\bar{\mathbf{D}}$ of size $(\sigma+1)\times(\sigma+1)$, where $\sigma = N_y + (N_y-1)\times(m-1)$. The second order differentiation matrices with respect to x and y, $\mathbf{D}^2$ and $\bar{\mathbf{D}}^2$, respectively, can be obtained using matrix multiplication. Notation wise, the bar in $\bar{\mathbf{D}}$ has been used to distinguish the assembled differentiation matrices in y from those in x.

2.1.4. Assembling linear system of algebraic equations

Using Eq.(24), Eq.(26) and Eq.(27) in the QLM scheme Eq.(20), we obtain a $(N_t+1)(\sigma+1)(\delta+1)$ system of linear algebraic equations given by

$$\left[\boldsymbol{\delta_{4,s}}\mathbf{D}^2 + \boldsymbol{\delta_{2,s}}\mathbf{D} + \boldsymbol{\delta_{0,s}}\mathbf{I}\right]\overset{j}{\bar{\mathbf{U}}}_k + \sum_{q=0}^{\sigma}\left[\boldsymbol{\delta_{3,s}}\bar{D}^2_{j,q} + \boldsymbol{\delta_{1,s}}\bar{D}_{j,q}\right]\overset{q}{\bar{\mathbf{U}}}_k - \sum_{r=0}^{N_t}\sum_{q=0}^{\sigma}\bar{\bar{D}}_{k,r}\overset{q}{\bar{\mathbf{U}}}_r = \overset{j}{\bar{\mathbf{R}}}_k, \quad \tau = 1,\dots,p, \tag{29}$$

for $j = 0,1,2,\dots,\sigma,\ k = 0,1,2,\dots,N_t$ and where $\mathbf{I}$ is an identity matrix of size $(\delta+1)\times(\delta+1)$. The vector $\overset{j}{\bar{\mathbf{U}}}_k$ of size $(\delta+1)$ represents the values of the unknown function $u(x,y,t)$ evaluated at

the $(\delta+1)$ grid points in entire the x domain of Eq.(12) and the j^{th} and the k^{th} grid points in y and t, respectively. The bar in $\bar{\overset{j}{\mathbf{U}}}_k$ distinguishes it from vector $\overset{j}{\mathbf{U}}_k$ of size (N_x+1) defined at Eq.(25). The initial condition when evaluated at the collocation points yields

$$\begin{aligned}
&\overset{(1)}{u}(x_i,y_j,0)=\overset{(1)}{u}(x_i,y_j,t_{N_t})=h(x_i,y_j)=\bar{\overset{j}{\mathbf{U}}}_{N_t},\ \ i=0,1,2,\ldots,\delta,\ \ j=0,1,2,\ldots,\sigma,\\
&\overset{(\tau)}{u}(x_i,y_j,t_{\tau-1})=\overset{(\tau-1)}{u}(x_i,y_j,t_{\tau-1}),\ \ \tau=2,3,\ldots,p.
\end{aligned}\tag{30}$$

Using the initial condition Eq.(30), we can reduce Eq.(29) to

$$\begin{aligned}
&\left[\boldsymbol{\delta_{4,s}}\mathbf{D}^2+\boldsymbol{\delta_{2,s}}\mathbf{D}+\boldsymbol{\delta_{0,s}}\mathbf{I}\right]\bar{\overset{j}{\mathbf{U}}}_k+\sum_{q=0}^{\sigma}\left[\boldsymbol{\delta_{3,s}}\bar{D}^2_{j,q}+\boldsymbol{\delta_{1,s}}\bar{D}_{j,q}\right]\bar{\overset{q}{\mathbf{U}}}_k-\sum_{r=0}^{N_t-1}\sum_{q=0}^{\sigma}\bar{\bar{D}}_{k,r}\bar{\overset{q}{\mathbf{U}}}_r=\bar{\overset{j}{\mathbf{R}}}_k,\\
&\tau=1,2,3,\ldots,p.
\end{aligned}\tag{31}$$

where $j=0,1,2,\ldots,\sigma,\ k=0,1,2,\ldots,N_t-1$ and

$$\bar{\overset{j}{\mathbf{R}}}_k=\overset{j}{\mathbf{R}}_k+\sum_{q=0}^{\sigma}\bar{\bar{D}}_{k,N_t}\bar{\overset{q}{\mathbf{U}}}_{N_t},\ \ k=0,1,2,\ldots,N_t-1.\tag{32}$$

The linear system of equations in Eq.(31) can be expanded into an $N_t(\sigma+1)(\delta+1)\times N_t(\sigma+1)(\delta+1)$ matrix system given by

$$\begin{bmatrix}
\mathbf{A}_{0,0} & \mathbf{A}_{0,1} & \mathbf{A}_{0,2} & \ldots & \mathbf{A}_{0,N_t-1}\\
\mathbf{A}_{1,0} & \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \ldots & \mathbf{A}_{1,N_t-1}\\
\vdots & \vdots & \vdots & \ldots & \vdots\\
\mathbf{A}_{N_t-1,0} & \mathbf{A}_{N_t-1,1} & \mathbf{A}_{N_t-1,2} & \ldots & \mathbf{A}_{N_t-1,N_t-1}
\end{bmatrix}
\begin{bmatrix}
\bar{\overset{j}{\mathbf{U}}}_0\\ \bar{\overset{j}{\mathbf{U}}}_1\\ \vdots\\ \bar{\overset{j}{\mathbf{U}}}_{N_t-1}
\end{bmatrix}=
\begin{bmatrix}
\overset{j}{\mathbf{R}}_0\\ \overset{j}{\mathbf{R}}_1\\ \vdots\\ \overset{j}{\mathbf{R}}_{N_t-1}
\end{bmatrix},\ \ \tau=1,2,3,\ldots,p.\tag{33}$$

The matrix system Eq.(33) can be written compactly as

$$\left[\mathbf{A}\right]\left[\overset{(\tau)}{\mathbf{U}}\right]=\left[\overset{(\tau)}{\mathbf{R}}\right],,\ \ \tau=1,2,3,\ldots,p,\tag{34}$$

where

$$\begin{aligned}
&\mathbf{A}_{k,k}=\mathbf{B}-\bar{\bar{D}}_{k,k}\mathbf{I}_{xy},\quad \mathbf{A}_{k,r}=-\bar{\bar{D}}_{k,r}\mathbf{I}_{xy},\ \ k\neq r,\ \ k,r=0,1,2,\ldots,N_t-1,\\
&\mathbf{B}_{i,i}=\boldsymbol{\delta_{4,s}}\mathbf{D}^2+\boldsymbol{\delta_{2,s}}\mathbf{D}+\boldsymbol{\delta_{0,s}}\mathbf{I}_x+\boldsymbol{\delta_{3,s}}\bar{D}^2_{i,i}\mathbf{I}_x+\boldsymbol{\delta_{1,s}}\bar{D}_{i,i}\mathbf{I}_x,\\
&\mathbf{B}_{i,j}=\boldsymbol{\delta_{3,s}}\bar{D}^2_{i,j}\mathbf{I}_x+\boldsymbol{\delta_{1,s}}\bar{D}_{i,j}\mathbf{I}_x,\quad i\neq j,\ \ i,j=0,1,2,\ldots,\sigma,\\
&\overset{(\tau)}{\mathbf{U}}=\left[\overset{(\tau)}{u}(x_0,y_0,t_0),\ldots,\overset{(\tau)}{u}(x_\delta,y_0,t_0),\overset{(\tau)}{u}(x_0,y_1,t_0),\ldots,\overset{(\tau)}{u}(x_{N_x},y_\sigma,t_0),\ldots,\overset{(\tau)}{u}(x_\delta,y_\sigma,t_{N_t})\right]^T,
\end{aligned}\tag{35}$$

and $\overset{(\tau)}{\mathbf{R}}$ is the right hand side corresponding to the unknown vector $\overset{(\tau)}{\mathbf{U}}$. The quantities $\mathbf{I}_{xy}$ and $\mathbf{I}_x$ are identity matrices of size $(\sigma+1)(\delta+1)\times(\sigma+1)(\delta+1)$ and $(\delta+1)\times(\delta+1)$, respectively. The boundary conditions Eq.(21) are evaluated at the collocation nodes as

$$\begin{aligned}
&\alpha_1^a \sum_{p=0}^{\delta} D_{\delta,p}\, \overset{(\tau)}{u}(x_p, y_j, t_k) + \alpha_1^a \overset{(\tau)}{u}(x_\delta, y_j, t_k) = f_a(y_j, t_k),\\
&\alpha_1^b \sum_{p=0}^{\delta} D_{0,p}\, \overset{(\tau)}{u}(x_p, y_j, t_k) + \alpha_0^b \overset{(\tau)}{u}(x_0, y_j, t_k) = f_b(y_j, t_k),\ j=0,1,2,\ldots,\sigma,\ k=0,1,2,\ldots,N_t,
\end{aligned} \tag{36}$$

and

$$\begin{aligned}
&\beta_1^c \sum_{q=0}^{\sigma} \bar{D}_{\sigma,q}\, \overset{(\tau)}{u}(x_i, y_q, t_k) + \beta_0^c \overset{(\tau)}{u}(x_i, y_\sigma, t_k) = g_c(x_i, t_k),\\
&\beta_1^d \sum_{q=0}^{\sigma} \bar{D}_{0,q}\, \overset{(\tau)}{u}(x_i, y_q, t_k) + \beta_0^d \overset{(\tau)}{u}(x_i, y_0, t_k) = g_d(x_i, t_k),\ \ i=0,1,2,\ldots,\delta,\ k=0,1,2,\ldots,N_t.
\end{aligned} \tag{37}$$

The boundary conditions are imposed on the main diagonal sub-blocks of matrices in Eq.(33) to yield a new consistent system of linear algebraic equations. The required solution of this system is obtained through an iterative process starting with an appropriate initial approximation to the solution.

2.2. ***Method of solution for systems of nonlinear PDEs***

In this subsection, we extend the algorithm described in the previous subsection to the solution of systems of nonlinear PDEs. To demonstrate the construction of the algorithm, we consider a system of two nonlinear PDEs that can be expressed in the form

$$\begin{aligned}
\frac{\partial u}{\partial t} &= F_1\left(\frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 v}{\partial x^2}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial^2 v}{\partial y^2}, \frac{\partial u}{\partial x}, \frac{\partial v}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial y}, u, v\right),\\
\frac{\partial v}{\partial t} &= F_2\left(\frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 v}{\partial x^2}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial^2 v}{\partial y^2}, \frac{\partial u}{\partial x}, \frac{\partial v}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial y}, u, v\right),
\end{aligned} \tag{38}$$

where F_1 and F_2 are nonlinear operators acting on the unknown functions u and v and their derivatives with respect to x and y as illustrated. The system Eq.(38) is solved subject to the boundary conditions

$$\begin{aligned}
&\alpha_1^a \frac{\partial u}{\partial x}(a, y, t) + \alpha_0^a u(a, y, t) = f_a(y, t),\ \ \alpha_1^b \frac{\partial u}{\partial x}(b, y, t) + \alpha_0^b u(b, y, t) = f_b(y, t),\\
&\beta_1^c \frac{\partial u}{\partial y}(x, c, t) + \beta_0^c u(x, c, t) = g_c(x, t),\ \ \beta_1^d \frac{\partial u}{\partial y}(x, d, t) + \beta_0^d u(x, d, t) = g_d(x, t),
\end{aligned} \tag{39}$$

and

$$\begin{aligned}
&\alpha_1^{a*}\frac{\partial v}{\partial x}(a,y,t)+\alpha_0^{a*}v(a,y,t)=f_{a*}(y,t),\quad \alpha_1^{b*}\frac{\partial v}{\partial x}(b,y,t)+\alpha_0^{b*}v(b,y,t)=f_{b*}(y,t),\\
&\beta_1^{c*}\frac{\partial v}{\partial y}(x,c,t)+\beta_0^{c*}v(x,c,t)=g_{c*}(x,t),\quad \beta_1^{d*}\frac{\partial v}{\partial y}(x,d,t)+\beta_0^{d*}v(x,d,t)=g_{d*}(x,t).
\end{aligned}\tag{40}$$

The star * in Eq.(40) distinguishes the boundary conditions in v from those of u. The initial conditions for this problem are

$$u(x,y,0)=h_1(x,y),\quad v(x,y,0)=h_2(x,y).\tag{41}$$

Applying the QLM, Eq.(38) can be expressed as the linearized coupled system

$$\begin{aligned}
&\delta_{4,s}^1\frac{\partial^2 u_{s+1}}{\partial x^2}+\delta_{3,s}^1\frac{\partial^2 u_{s+1}}{\partial y^2}+\delta_{2,s}^1\frac{\partial u_{s+1}}{\partial x}+\delta_{1,s}^1\frac{\partial u_{s+1}}{\partial y}+\delta_{0,s}^1u_{s+1}\\
&-\dot{u}_{s+1}+\sigma_{4,s}^1\frac{\partial^2 v_{s+1}}{\partial x^2}+\sigma_{3,s}^1\frac{\partial^2 v_{s+1}}{\partial y^2}+\sigma_{2,s}^1\frac{\partial v_{s+1}}{\partial x}+\sigma_{1,s}^1\frac{\partial v_{s+1}}{\partial y}+\sigma_{0,s}^1v_{s+1}\quad=R_{1,s},
\end{aligned}\tag{42}$$

$$\begin{aligned}
&\delta_{4,s}^2\frac{\partial^2 u_{s+1}}{\partial x^2}+\delta_{3,s}^2\frac{\partial^2 u_{s+1}}{\partial y^2}+\delta_{2,s}^2\frac{\partial u_{s+1}}{\partial x}+\delta_{1,s}^2\frac{\partial u_{s+1}}{\partial y}+\delta_{0,s}^2u_{s+1}\\
&+\sigma_{4,s}^2\frac{\partial^2 v_{s+1}}{\partial x^2}+\sigma_{3,s}^2\frac{\partial^2 v_{s+1}}{\partial y^2}+\sigma_{2,s}^2\frac{\partial v_{s+1}}{\partial x}+\sigma_{1,s}^2\frac{\partial v_{s+1}}{\partial y}+\sigma_{0,s}^2v_{s+1}-\dot{v}_{s+1}\quad=R_{2,s},
\end{aligned}\tag{43}$$

where

$$\begin{aligned}
&\delta_{4,s}^\nu=\frac{\partial F_\nu}{\partial\left(u_{xx}\right)_s},\ \delta_{3,s}^\nu=\frac{\partial F_\nu}{\partial\left(u_{yy}\right)_s},\ \delta_{2,s}^\nu=\frac{\partial F_\nu}{\partial\left(u_x\right)_s},\ \delta_{1,s}^\nu=\frac{\partial F_\nu}{\partial\left(u_y\right)_s},\ \delta_{0,s}^\nu=\frac{\partial F_\nu}{\partial\left(u\right)_s},\\
&\sigma_{4,s}^\nu=\frac{\partial F_\nu}{\partial\left(u_{xx}\right)_s},\ \sigma_{3,s}^\nu=\frac{\partial F_\nu}{\partial\left(u_{yy}\right)_s},\ \sigma_{2,s}^\nu=\frac{\partial F_\nu}{\partial\left(u_x\right)_s},\ \sigma_{1,s}^\nu=\frac{\partial F_\nu}{\partial\left(u_y\right)_s},\ \sigma_{0,s}^\nu=\frac{\partial F_\nu}{\partial\left(u\right)_s},\quad \nu=1,2,
\end{aligned}\tag{44}$$

$$\begin{aligned}
R_{\nu,s}=&\delta_{4,s}^\nu(u_{xx})_s+\delta_{3,s}^\nu(u_{yy})_s+\delta_{2,s}^\nu(u_x)_s+\delta_{1,s}^\nu(u_y)_s+\delta_{0,s}^\nu u_s+\\
&\sigma_{4,s}^\nu(v_{xx})_s+\sigma_{3,s}^\nu(v_{yy})_s+\sigma_{2,s}^\nu(v_x)_s+\sigma_{1,s}^\nu(v_y)_s+\sigma_{0,s}^\nu v_s-F_{\nu,s}\ ,\quad \nu=1,2.
\end{aligned}\tag{45}$$

For each subinterval in time, we must solve

$$\begin{aligned}
&\delta_{4,s}^1\frac{\partial^2\overset{(\tau)}{u}_{s+1}}{\partial x^2}+\delta_{3,s}^1\frac{\partial^2\overset{(\tau)}{u}_{s+1}}{\partial y^2}+\delta_{2,s}^1\frac{\partial\overset{(\tau)}{u}_{s+1}}{\partial x}+\delta_{1,s}^1\frac{\partial\overset{(\tau)}{u}_{s+1}}{\partial y}+\delta_{0,s}^1\overset{(\tau)}{u}_{s+1}\\
&-\frac{\partial\overset{(\tau)}{u}_{s+1}}{\partial t}+\sigma_{4,s}^1\frac{\partial^2\overset{(\tau)}{v}_{s+1}}{\partial x^2}+\sigma_{3,s}^1\frac{\partial^2\overset{(\tau)}{v}_{s+1}}{\partial y^2}+\sigma_{2,s}^1\frac{\partial\overset{(\tau)}{v}_{s+1}}{\partial x}+\sigma_{1,s}^1\frac{\partial\overset{(\tau)}{v}_{s+1}}{\partial y}+\sigma_{0,s}^1\overset{(\tau)}{v}_{s+1}\quad=R_{1,s},
\end{aligned}\tag{46}$$

$$\begin{aligned}
&\delta_{4,s}^2\frac{\partial^2\overset{(\tau)}{u}_{s+1}}{\partial x^2}+\delta_{3,s}^2\frac{\partial^2\overset{(\tau)}{u}_{s+1}}{\partial y^2}+\delta_{2,s}^2\frac{\partial\overset{(\tau)}{u}_{s+1}}{\partial x}+\delta_{1,s}^2\frac{\partial\overset{(\tau)}{u}_{s+1}}{\partial y}+\delta_{0,s}^2\overset{(\tau)}{u}_{s+1}\\
&+\sigma_{4,s}^2\frac{\partial^2\overset{(\tau)}{v}_{s+1}}{\partial x^2}+\sigma_{3,s}^2\frac{\partial^2\overset{(\tau)}{v}_{s+1}}{\partial y^2}+\sigma_{2,s}^2\frac{\partial\overset{(\tau)}{v}_{s+1}}{\partial x}+\sigma_{1,s}^2\frac{\partial\overset{(\tau)}{v}_{s+1}}{\partial y}+\sigma_{0,s}^2\overset{(\tau)}{v}_{s+1}-\frac{\partial\overset{(\tau)}{v}_{s+1}}{\partial t}\quad=R_{2,s},
\end{aligned}\tag{47}$$

subject to the corresponding boundary conditions. Here the initial conditions are evaluated at collocation points as

$$\begin{aligned}
&\overset{(1)}{u}(x_i,y_j,0)=\overset{(1)}{u}(x_i,y_j,t_{N_t})=h_1(x_i,y_j)=\overset{j}{\mathbf{U}}_{N_t},\quad \overset{(1)}{v}(x_i,y_j,0)=\overset{(1)}{v}(x_i,y_j,t_{N_t})=h_2(x_i,y_j)=\overset{j}{\mathbf{V}}_{N_t},\\
&\overset{(\tau)}{u}(x_i,y_j,t_{\tau-1})=\overset{(\tau-1)}{u}(x_i,y_j,t_{\tau-1}),\quad \overset{(\tau)}{v}(x_i,y_j,t_{\tau-1})=\overset{(\tau-1)}{v}(x_i,y_j,t_{\tau-1}),\ \ \tau=2,3,\ldots,p.
\end{aligned} \tag{48}$$

Applying the spectral collocation method to the linearized QLM scheme Eqs.(46)-(47) and accounting for the effect of the initial condition we obtain a system of $2N_t(\sigma+1)(\delta+1)$ linear algebraic equations for each subinterval $\tau=1,2,\ldots,p$, in t given by

$$\begin{aligned}
&\left[\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{4},s}\mathbf{D}^2+\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{2},s}\mathbf{D}+\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{0},s}\mathbf{I}\right]\overset{j}{\bar{\mathbf{U}}}_k+\sum_{q=0}^{\sigma}\left[\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{3},s}\bar{D}^2_{j,q}+\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{1},s}\bar{D}_{j,q}\right]\overset{q}{\bar{\mathbf{U}}}_k-\sum_{r=0}^{N_t-1}\sum_{q=0}^{\sigma}\bar{\bar{D}}_{k,r}\overset{q}{\bar{\mathbf{U}}}_r\\
&+\left[\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{4},s}\mathbf{D}^2+\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{2},s}\mathbf{D}+\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{0},s}\mathbf{I}\right]\overset{j}{\bar{\mathbf{V}}}_k+\sum_{q=0}^{\sigma}\left[\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{3},s}\bar{D}^2_{j,q}+\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{1},s}\bar{D}_{j,q}\right]\overset{q}{\bar{\mathbf{V}}}_k=\overset{j}{\bar{\mathbf{R}}}_{1,k},
\end{aligned} \tag{49}$$

$$\begin{aligned}
&\left[\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{4},s}\mathbf{D}^2+\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{2},s}\mathbf{D}+\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{0},s}\mathbf{I}\right]\overset{j}{\bar{\mathbf{U}}}_k+\sum_{q=0}^{\sigma}\left[\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{3},s}\bar{D}^2_{j,q}+\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{1},s}\bar{D}_{j,q}\right]\overset{q}{\bar{\mathbf{U}}}_k+\left[\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{4},s}\mathbf{D}^2+\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{2},s}\mathbf{D}+\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{0},s}\mathbf{I}\right]\overset{j}{\bar{\mathbf{V}}}_k\\
&\qquad+\sum_{q=0}^{\sigma}\left[\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{3},s}\bar{D}^2_{j,q}+\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{1},s}\bar{D}_{j,q}\right]\overset{q}{\bar{\mathbf{V}}}_k-\sum_{r=0}^{N_t-1}\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,r}\overset{q}{\bar{\mathbf{V}}}_r=\overset{j}{\bar{\mathbf{R}}}_{2,k}.
\end{aligned} \tag{50}$$

Eqs.(38)-(39) can be expressed as a $2N_t(\sigma+1)(\delta+1)\times 2N_t(\sigma+1)(\delta+1)$ matrix system given by

$$\begin{bmatrix}\mathbf{A}_{1,1} & \mathbf{A}_{1,2}\\ \mathbf{A}_{2,1} & \mathbf{A}_{2,2}\end{bmatrix}\begin{bmatrix}\overset{(\tau)}{\mathbf{U}}\\ \overset{(\tau)}{\mathbf{V}}\end{bmatrix}=\begin{bmatrix}\overset{(\tau)}{\mathbf{R}}_1\\ \overset{(\tau)}{\mathbf{R}}_2\end{bmatrix}, \tag{51}$$

where

$$\overset{(\tau)}{\mathbf{R}}_1=\mathbf{R}_{1,s}+\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,N_t}\overset{q}{\mathbf{U}}_{N_t},\quad \overset{(\tau)}{\mathbf{R}}_2=\mathbf{R}_{2,s}+\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,N_t}\overset{q}{\mathbf{V}}_{N_t},\ \ k=0,1,2,\ldots,N_t-1. \tag{52}$$

The vector $\overset{(\tau)}{\bar{\mathbf{V}}}$ is defined in similar manner to vector $\overset{(\tau)}{\bar{\mathbf{U}}}$ given in Eq.(35). The right hand side vectors $\overset{(\tau)}{\mathbf{R}}_1$, and $\overset{(\tau)}{\mathbf{R}}_2$ correspond to vectors $\overset{(\tau)}{\bar{\mathbf{U}}}$ and $\overset{(\tau)}{\bar{\mathbf{V}}}$, respectively. The boundary conditions Eq.(39) and Eq.(40) are evaluated at the collocation points in a manner similar to that illustrated at Eq.(36) and Eq.(37). We remark that if the initial conditions $u(x,y,0)$ and $v(x,y,0)$ are not available,

the system of differential equations is first solved for $t = 0$ to approximate the initial conditions. In summary, we assert that the algorithms described so far can be adjusted easily and applied to PDEs of higher orders (> 2) and on problems modeled by larger systems of nonlinear PDEs. An extension to numerical schemes of solutions of two-dimensional time-dependent PDEs exhibiting slightly different forms of nonlinearity can be achieved in a straightforward manner.

3. Error bound theorems in a trivariate polynomial interpolation

In this section, we present new error bound theorems that govern polynomial interpolation errors in a trivariate Lagrange interpolating polynomial constructed using Chebyshev-Gauss-Lobatto nodes. Fundamental elements related to the construction of proofs of the theorems include; first, understanding that the CGL nodes are the relative extremes of the N_x-th degree Chebyshev polynomial of the first kind $T_{N_x}(\hat{x}) = \cos\left[N_x \arccos(\hat{x})\right], \ \hat{x} \in [-1, 1]$, secondly, understanding the general properties of Chebyshev polynomials, and thirdly, understanding the mean value theorem in calculus. Although to the best of our knowledge, there does not exist a well-known family of polynomials whose roots are the CGL nodes, it is easy to discern that the interior CGL nodes are roots of $T'_{N_x}(\hat{x}) = 0$. This fact leads to the discovery of a complete set of CGL nodes as the roots of the $N_x + 1$-th degree polynomial given by

$$L_{N_x+1}(\hat{x}) = (1 - \hat{x}^2)T'_{N_x}(\hat{x}). \tag{53}$$

The theorem given below benchmarks the formulation of the error bound theorems on trivariate polynomial interpolation.

Theorem 1. *[27] Let $u(x, y, t) \in C^{N_x+N_y+N_t+3}([a, b] \times [c, d] \times [0, T])$ be sufficiently smooth such that at least the $(N_x+1)^{th}$ partial derivative with respect to x, the $(N_y+1)^{th}$ partial derivative with respect to y, the $(N_t+1)^{th}$ partial derivative with respect to t, and the $(N_x + N_y + N_t + 3)^{th}$ mixed partial derivative with respect to x, y, and t exist and are all continuous, then there exist values $\xi_x, \xi'_x \in (a, b)$, $\xi_y, \xi'_y, \in (c, d)$, and $\xi_t, \xi'_t, \in (0, T)$, such that*

$$\begin{aligned} u(x, y, t) - U(x, y, t) =& \frac{\partial^{N_x+1} u(\xi_x, y, t)}{\partial x^{N_x+1}(N_x + 1)!} \prod_{i=0}^{N_x}(x - x_i) + \frac{\partial^{N_y+1} u(x, \xi_y, t)}{\partial y^{N_y+1}(N_y + 1)!} \prod_{j=0}^{N_y}(y - y_j) + \frac{\partial^{N_t+1} u(x, y, \xi_t)}{\partial t^{N_t+1}(N_t + 1)!} \prod_{k=0}^{N_t}(t - t_k) \\ &- \frac{\partial^{N_x+N_y+N_t+3} u(\xi'_x, \xi'_y, \xi'_t)}{\partial x^{N_x+1} \partial y^{N_y+1} \partial t^{N_t+1}(N_x + 1)!(N_y + 1)!(N_t + 1)!} \prod_{i=0}^{N_x}(x - x_i) \prod_{j=0}^{N_y}(y - y_j) \prod_{k=0}^{N_t}(t - t_k), \end{aligned} \tag{54}$$

where $U(x,y,t)$ is a trivariate interpolating polynomial of $u(x,y,t)$ at $\{x_i\}_{i=0}^{N_x}$ grid points in x-variable, $\{y_j\}_{j=0}^{N_y}$ grid points in y-variable, and $\{t_k\}_{k=0}^{N_t}$ grid points in t-variable.

The remainder formula Eq.(54) is based on the mean value theorem and is derived recursively from the corresponding univariate error formula given in [28] for a sufficiently smooth function $u(x,y,t)$. Taking the absolute value of Eq.(54) we obtain

$$\begin{aligned}
|u(x,y,t)-U(x,y,t)| \leq & \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_x+1}u(\xi_x,y,t)}{\partial x^{N_x+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x-x_i)\right|}{(N_x+1)!} \\
& + \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_y+1}u(x,\xi_y,t)}{\partial y^{N_y+1}}\right|\frac{\left|\prod_{j=0}^{N_y}(y-y_j)\right|}{(N_y+1)!} \\
& + \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x,y,\xi_t)}{\partial t^{N_t+1}}\right|\frac{\left|\prod_{k=0}^{N_t}(t-t_k)\right|}{(N_t+1)!} \\
& + \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_x+N_y+N_t+3}u(\xi_x',\xi_y',\xi_t')}{\partial x^{N_x+1}\partial y^{N_y+1}\partial t^{N_t+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x-x_i)\right|\left|\prod_{j=0}^{N_y}(y-y_j)\right|\left|\prod_{k=0}^{N_t}(t-t_k)\right|}{(N_x+1)!(N_y+1)!(N_t+1)!},
\end{aligned} \tag{55}$$

where $\Omega=[a,b]\times[c,d]\times[0,T]$. Since the function $u(x,y,t)$ is assumed to be smooth on the interval of approximation, it follows that its derivatives are bounded and thus $\exists$ constants C_1, C_2, C_3 and C_4, such that

$$\begin{aligned}
&\max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_x+1}u(x,y,t)}{\partial x^{N_x+1}}\right|\leq C_1, \quad \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_y+1}u(x,y,t)}{\partial y^{N_y+1}}\right|\leq C_2, \\
&\max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x,y,t)}{\partial t^{N_t+1}}\right|\leq C_3, \quad \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_x+N_y+N_t+3}u(x,y,t)}{\partial x^{N_x+1}\partial y^{N_y+1}\partial t^{N_t+1}}\right|\leq C_4.
\end{aligned} \tag{56}$$

The error bound for trivariate polynomial interpolation using Chebyshev-Gauss-Lobatto nodes on a single domain is governed by the theorem below

Theorem 2 (**The error bound in a single domain**). *The resulting error bound when CGL grid points $\{x_i\}_{i=0}^{N_x}\in[a,b]$, in x-variable, $\{y_j\}_{j=0}^{N_y}\in[c,d]$ in y-variable, and $\{t_k\}_{k=0}^{N_t}\in[0,T]$, in t-variable are used in trivariate polynomial interpolation is given by*

$$E(x,y,t)\leq C_1\frac{8\left(\frac{b-a}{4}\right)^{N_x+1}}{(N_x+1)!}+C_2\frac{8\left(\frac{d-c}{4}\right)^{N_y+1}}{(N_y+1)!}+C_3\frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}+C_4\frac{8^3\left(\frac{b-a}{4}\right)^{N_x+1}\left(\frac{d-c}{4}\right)^{N_y+1}\left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_y+1)!(N_t+1)!}. \tag{57}$$

Proof. First, using the relation stated in [29] we express Eq.(53) as

$$L_{N_x+1}(\hat{x}) = (1-\hat{x}^2)T'_{N_x}(\hat{x}) = -N_x\hat{x}T_{N_x}(\hat{x}) + N_xT_{N_x-1}(\hat{x}). \tag{58}$$

Using the triangle inequality and noting that $|T_{N_x}(\hat{x})| \leq 1,\ \forall\ \hat{x} \in [-1,1]$, we have

$$|L_{N_x+1}(\hat{x})| = |-N_x\hat{x}T_{N_x}(\hat{x}) + N_xT_{N_x-1}(\hat{x})| \leq |-N_x\hat{x}T_{N_x}(\hat{x})| + |N_xT_{N_x-1}(\hat{x})| \leq 2N_x. \tag{59}$$

The leading coefficient of $L_{N_x+1}(\hat{x})$ is $2^{N_x-1}N_x$, where the components 2^{N_x-1} and N_x come from the leading coefficient of $T_{N_x}(\hat{x})$ and the application of N_x-th rule of differentiation on $T_{N_x}(\hat{x})$, respectively. The product factor in the first term of the error bound expression given at Eq.(55) can therefore be taken as the factorized form of the monic polynomial $\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}$. We write

$$\prod_{i=0}^{N_x}(\hat{x}-\hat{x}_i) = \frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x},\ \ \hat{x} \in [-1,1]. \tag{60}$$

Using Eq.(59), it is easy to establish that the monic polynomial Eq.(60) is bounded by

$$\left|\prod_{j=0}^{N_x}(x-\hat{x}_i)\right| = \left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right| \leq \frac{2N_x}{2^{N_x-1}N_x} = \frac{4}{2^{N_x}}. \tag{61}$$

Considering a general interval $x \in [a,b]$, we can show that the first product factor in Eq.(55) is bounded by

$$\begin{aligned}\max_{a\leq x\leq b}\left|\prod_{i=0}^{N_x}(x-x_i)\right| &= \max_{-1\leq\hat{x}\leq 1}\left|\prod_{i=0}^{N_x}\frac{(b-a)}{2}(\hat{x}-\hat{x}_i)\right| = \left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\leq\hat{x}\leq 1}\left|\prod_{i=0}^{N_x}(\hat{x}-\hat{x}_i)\right| \\ &= \left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\leq\hat{x}\leq 1}\left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right| \leq \frac{4\left(\frac{b-a}{2}\right)^{N_x+1}}{2^{N_x}} = 8\left(\frac{b-a}{4}\right)^{N_x+1}.\end{aligned} \tag{62}$$

Similarly, we conclude that the second and the third product factors are bounded, respectively, by

$$\max_{c\leq y\leq d}\left|\prod_{j=0}^{N_y}(y-y_j)\right| = \left(\frac{d-c}{2}\right)^{N_y+1}\max_{-1\leq\hat{y}\leq 1}\left|\frac{L_{N_y+1}(\hat{y})}{2^{N_y-1}N_y}\right| \leq \frac{4\left(\frac{d-c}{2}\right)^{N_y+1}}{2^{N_y}} = 8\left(\frac{d-c}{4}\right)^{N_y+1}, \tag{63}$$

and

$$\max_{0\leq t\leq T}\left|\prod_{k=0}^{N_t}(t-t_k)\right| = \left(\frac{T}{2}\right)^{N_t+1}\max_{-1\leq\hat{t}\leq 1}\left|\frac{L_{N_t+1}(\hat{t})}{2^{N_t-1}N_t}\right| \leq \frac{4\left(\frac{T}{2}\right)^{N_t+1}}{2^{N_t}} = 8\left(\frac{T}{4}\right)^{N_t+1}. \tag{64}$$

Using Eqs.(62) – (64) and Eq.(56) in Eq.(55) the proof is completed. □

Theorem 3 (**The error bound in a decomposed domain**). *The resulting error bound when CGL grid points* $\{x_i\}_{i=0}^{N_x} \in [x_{l-1}, \bar{x}_l],\ l = 1,2,\ldots,q$, *for the decomposed domain in x-variable and*

$\{y_j\}_{j=0}^{N_y} \in [y_{\zeta-1}, \bar{y}_\zeta],\ \zeta = 1, 2, \ldots, m$, *for the decomposed domain in y-variable, and* $\{t_k\}_{k=0}^{N_t} \in [t_{\tau-1}, t_\tau],\ \tau = 1, 2, \ldots, p$, *for the decomposed domain in t-variable are used in trivariate polynomial interpolation is given by*

$$\begin{aligned} E(x,y,t) \leq & C_1 \frac{8\left(\frac{L}{4}\right)^{N_x+1}}{(N_x+1)!} + C_2 \frac{8\left(\frac{Z}{4}\right)^{N_y+1}}{(N_y+1)!} + C_3 \frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1} \\ & + C_4 \frac{8^3\left(\frac{L}{4}\right)^{N_x+1}\left(\frac{Z}{4}\right)^{N_y+1}\left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_y+1)!(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1}. \end{aligned} \tag{65}$$

Proof. First, we consider the t variable. In the entire domain $[0, T]$, we have that

$$\left|\prod_{k=0}^{N_t}(t - t_k)\right| \leq 8\left(\frac{T}{4}\right)^{N_t+1},\ \ t \in [0, T]. \tag{66}$$

The implication is that in the decomposed domain and at each subinterval, we must have

$$\left|\prod_{k=0}^{N_t}(t - t_k)\right| \leq 8\left(\frac{T}{4p}\right)^{N_t+1} = 8\left(\frac{T}{4}\right)^{N_t+1}\left(\frac{1}{p}\right)^{N_t+1},\ \ t \in [t_{\tau-1}, t_\tau],\ \tau = 1, 2, \ldots, p, . \tag{67}$$

For smooth u, there exists $\xi_\mu \in (t_{\mu-1}, t_\mu),\ \mu = 1, 2, \ldots, p$, for which the values of the $(N_t+1)^{th}$ partial derivatives of u with respect to t in each subinterval is the absolute extrema. This enables us to break the third term $C_3\frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}$ that appears in the error bound expression at Eq.(57) into different components, which are necessarily not equal in the decomposed domain, as

$$\left\{ \overset{(\tau)}{C}_3 \frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1} \right\}_{\tau=1}^{p}, \tag{68}$$

where

$$\max_{(x,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x,t)}{\partial t^{N_t+1}}\right| = \left|\frac{\partial^{N_t+1}u(x,\xi_\tau)}{\partial t^{N_t+1}}\right| \leq \overset{(\tau)}{C}_3,\ \ t \in [t_{\tau-1}, t_\tau].$$

We define

$$\| \hat{C}_3 \|_\infty \equiv \max\{\overset{(1)}{C}_3, \overset{(2)}{C}_3, \ldots, \overset{(p)}{C}_3\}, \tag{69}$$

to denote the maximum absolute value of the $(N_t+1)^{th}$ partial derivatives of u with respect to t in $[0, T]$. Clearly, $\| \hat{C}_3 \|_\infty = C_3$, where C_3 is identical to the one given at Eq.(57). To expand the error bound over the entire t domain, we shall take the largest possible error across all subintervals in t, which is

$$C_3 \frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}\left(\frac{1}{p}\right)^{N_t+1}. \tag{70}$$

Similar reasoning can be applied to show that the first and the second component in error bound Eq.(57) in the decomposed x and y domains translate to

$$C_2 \frac{8\left(\frac{L}{4}\right)^{N_y+1}}{(N_y+1)!},\ \text{and}\ C_2 \frac{8\left(\frac{Z}{4}\right)^{N_y+1}}{(N_y+1)!},\ \text{respectively.} \tag{71}$$

Consequently, the forth component in Eq.(57) becomes

$$C_4 \frac{8^3 \left(\frac{L}{4}\right)^{N_x+1} \left(\frac{Z}{4}\right)^{N_y+1} \left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_y+1)!(N_t+1)!} \left(\frac{1}{p}\right)^{N_t+1}, \tag{72}$$

in the decomposed domain. Using Eqs.(70) − (72) in Eq.(57) completes the proof. □

Comparing Eq.(57) and Eq.(64) we note that the error in trivariate polynomial interpolation is smaller when interpolation is conducted on multiple domains than it would be on a single domain. Further, we remark that CGL nodes are preferable candidates of interpolation when using spectral collocation methods to solve boundary value problems because they are convenient in constructing differentiation matrices. For the reason that they contain the boundary nodes they are advantageous when treating the boundary conditions of the problem.

4. Numerical experiment

In this section, we apply the method described in the previous section to a selected class of two-dimensional nonlinear initial-boundary value problems. The accuracy and efficiency of the proposed method are demonstrated by comparing the numerical results with the exact solution.

Example 1. Consider the problem of two-dimensional heat and mass transfer in quiescent media with chemical reaction given in its general form by

$$\frac{\partial u}{\partial t} = a\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + f(u), \tag{73}$$

where a is a real constant. As a special case, we analyze this problem when $a = 1$ and $f(u) = -4u^3 - 2u^2$ and solve the problem subject to the boundary conditions

$$\begin{aligned} &u(2,y,t) = (2+y+2t)^{-1}, \quad u(5,y,t) = (5+y+2t)^{-1}, \quad 2 \le y \le 5, \quad t > 0, \\ &u(x,2,t) = (2+x+2t)^{-1}, \quad u(x,5,t) = (5+x+2t)^{-1}, \quad 2 \le x \le 5, \quad t > 0. \end{aligned} \tag{74}$$

The initial condition for this problem is given by

$$u(x,y,0) = (x+y)^{-1}, \;\; 2 \le x \le 5, \;\; 2 \le y \le 5. \tag{75}$$

The exact solution is given in [9] as

$$u(x,y,t) = (x+y+2t)^{-1}. \tag{76}$$

Example 2. We consider the two-dimensional heat and mass transfer equation with power-law temperature-dependent thermal conductivity [10]

$$\frac{\partial u}{\partial t} = a\left[\frac{\partial}{\partial x}\left(u^n\frac{\partial u}{\partial x}\right)\right] + b\left[\frac{\partial}{\partial y}\left(u^m\frac{\partial u}{\partial y}\right)\right] + \beta u, \tag{77}$$

where $n,\ m$ can be integers or fractions and a, b, and β are some parameters. For simplicity, we consider the special case of Eq.(77) when $a = 1$, $b = 1$, $m = n = 1$, and $\beta = 0$ in which Eq.(77) reduces to the Boussinesq equation

$$\frac{\partial u}{\partial t} = \left[\frac{\partial}{\partial x}\left(u\frac{\partial u}{\partial x}\right)\right] + \left[\frac{\partial}{\partial y}\left(u\frac{\partial u}{\partial y}\right)\right]. \tag{78}$$

The Boussinesq equation arises in nonlinear heat conduction and the theory of unsteady flow through porous media with a free surface. Eq.(78) is solved subject to boundary conditions

$$\begin{aligned} &u(0,y,t) = y + 2t, \quad u(5,y,t) = 5 + y + 2t, \quad 0 \le y \le 4, \quad t > 0, \\ &u(x,0,t) = x + 2t, \quad u(x,4,t) = 4 + x + 2t, \quad 0 \le x \le 5, \quad t > 0. \end{aligned} \tag{79}$$

The initial condition for this problem is given by

$$u(x,y,0) = x + y. \tag{80}$$

The exact solution is given by authors in [9] as

$$u(x,y,t) = x + y + 2t. \tag{81}$$

Example 3. We consider the system of two-dimensional Burgers equations given by

$$\begin{aligned} \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} &= \frac{1}{Re}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right), \\ \frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} &= \frac{1}{Re}\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right), \end{aligned} \tag{82}$$

where $u(x,y,t)$ and $v(x,y,t)$ are velocity components to be determined and Re is the Reynolds number. Eq.(82) is subject to boundary conditions

$$\begin{aligned} u(0,y,t) &= \frac{3}{4} - \frac{1}{4\left[1 + e^{(Re(4y-t)/32)}\right]}, \quad u(5,y,t) = \frac{3}{4} - \frac{1}{4\left[1 + e^{(Re(4y-20-t)/32)}\right]}, \\ u(x,0,t) &= \frac{3}{4} - \frac{1}{4\left[1 + e^{(Re(-4x-t)/32)}\right]}, \quad u(x,5,t) = \frac{3}{4} - \frac{1}{4\left[1 + e^{(Re(20-4x-t)/32)}\right]}, \end{aligned} \tag{83}$$

$$\begin{aligned} v(0,y,t) &= \frac{3}{4} + \frac{1}{4\left[1 + e^{(Re(4y-t)/32)}\right]}, \quad v(5,y,t) = \frac{3}{4} + \frac{1}{4\left[1 + e^{(Re(4y-20-t)/32)}\right]}, \\ v(x,0,t) &= \frac{3}{4} + \frac{1}{4\left[1 + e^{(Re(-4x-t)/32)}\right]}, \quad v(x,5,t) = \frac{3}{4} + \frac{1}{4\left[1 + e^{(Re(20-4x-t)/32)}\right]}. \end{aligned} \tag{84}$$

The initial condition are

$$u(x,y,0) = \frac{3}{4} - \frac{1}{4\left[1+e^{(Re(y-x)/8)}\right]}, \qquad v(x,y,0) = \frac{3}{4} + \frac{1}{4\left[1+e^{(Re(y-x)/8)}\right]}. \tag{85}$$

The exact solutions for this problem are given in [30] as

$$u(x,y,t) = \frac{3}{4} - \frac{1}{4\left[1+e^{(Re(4y-4x-t)/32)}\right]}, \qquad v(x,y,t) = \frac{3}{4} + \frac{1}{4\left[1+e^{(Re(4y-4x-t)/32)}\right]}. \tag{86}$$

Example 4. We consider the two-dimensional Brusselator system given by

$$\begin{aligned} \frac{\partial u}{\partial t} &= B + u^2 v - (A+1)u + \mu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right), \\ \frac{\partial v}{\partial t} &= Au - u^2 v + \mu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right), \end{aligned} \tag{87}$$

where $u(x,y,t)$ and $v(x,y,t)$, representing dimensionless concentration of two reactants A and B are constant concentrations of the two reactants, and μ is the diffusion coefficient. The exact solutions for this problem are given in [31] for $A = 1,\ B = 0,\ \mu = \frac{1}{4}$ as

$$u(x,y,t) = e^{-x-y-t/2}, \qquad v(x,y,t) = e^{x+y+t/2}. \tag{88}$$

Eq.(87) is solved subject to boundary conditions

$$u(0,y,t) = e^{-y-t/2}, \quad u(2,y,t) = e^{-2-y-t/2}, \quad u(x,0,t) = e^{-x-t/2}, \quad u(x,2,t) = e^{-2-x-t/2}, \tag{89}$$

$$v(0,y,t) = e^{y+t/2}, \quad v(2,y,t) = e^{2+y+t/2}, \quad v(x,0,t) = e^{x+t/2}, \quad v(x,2,t) = e^{2+x+t/2}. \tag{90}$$

The initial condition for this problem are

$$u(x,y,0) = e^{-x-y}, \qquad v(x,y,0) = e^{x+y}. \tag{91}$$

5. Results and discussion

In this section, results obtained after solving selected initial-boundary value problems using the overlapping grids based multidomain spectral collocation method are presented in tabular and graphical forms and then discussed. The results demonstrate various aspects of the proposed numerical method of solution. For comparison purposes, results from the new modified solution approach are compared with those from the standard spectral collocation method on a single

domain to reveal the benefits of incorporating techniques of both overlapping and non-overlapping grids in the solution algorithm. The selected test examples possess exact solutions and, therefore, we evaluate the accuracy of the method by displaying the absolute error norms obtained after 5 iterations, upon which the iterative numerical scheme is discerned to have converged. The error norms are computed as

$$E_5 = |u - U|_\infty \tag{92}$$

where u and U are vectors representing, respectively, the exact and approximate solutions evaluated at selected collocation points. We examine the dependence of accuracy on the length of time interval, length of the spatial domain and the size of the parameters present in the given differential equation.

Results obtained when Eq.(73) is solved are displayed in Table 1. The single domain approach is invoked using $N_x = 20$ and $N_y = 20$ grid points in the spatial domains and $N_t = 10$ grid points per unit interval in the time domain, in such a manner that when $T = 1$, $N_t = 10$, when $T = 2$, $N_t = 20$ and so on until $N_t = 50$ is used for $T = 5$. In the multidomain domain approach, each spatial domain is broken into two overlapping subintervals, with 10 grid points in each subinterval; thus $N_x = 10,\ N_y = 10,\ q = 2,\ m = 2$ in all five cases of time interval considered. The domain decomposition on the time interval is achieved as follows: $N_t = 10$ is used per unit time, which implies that when $T = 1$ only one interval in time is considered with $N_t = 10$, but when $T = 2$, two non-overlapping subintervals are used, each with $N_t = 10$ grid points and so on such when $T = 5$, five non-overlapping subintervals are used, each with $N_t = 10$ grid points. The CPU time displayed in this table is the cumulative computation time required to run the algorithm over the different time levels, $T = 1$ to $T = 5$. Comparison of results in the second and the third columns suggests that incorporating the domain decomposition technique for large time intervals yields more accurate results than those given by increasing the number of grid points on a single interval. The entries in the first row show that this is so, even for smaller time intervals. Decomposing the large spatial domain into overlapping subintervals improves the accuracy of the numerical approximation. It can be seen in Table 1 that shorter computation time is taken to yield results with the multidomain approach than with the single domain approach. The condition number displayed is that of the coefficient matrix of the system of linear algebraic equations when $T = 5$. With the single domain approach, deterioration of accuracy with large time intervals can be explained by the large condition number of the coefficient matrix resulting from many grid points. In the multidomain approach,

the shorter computational time is attributed to the resulting small-sized matrices, which are easier to invert.

Results obtained when Eq.(78) is solved are displayed in Table 2. The grid points distribution used to generate these results is similar to that for the previous table. From Table 2, we observe that, although the error norm is slightly larger in the single domain approach than in the multidomain case, it remains relatively steady as the length of the computational domain in time increase. Close scrutiny of the exact solution of Eq.(78) reveals that its higher ordered derivatives are bounded within the interval of approximation. For problems whose higher ordered derivatives are bounded, the accuracy ought not to deteriorate with an increase in the number of grid points. This observation agrees with the error bound theorems given in the previous section. However, since, in many practical applications, there is no prior knowledge of the exact solution of the differential equation and in view of the computational time constraint, the multidomain approach is superior to the single domain approach where the domain of approximation is large. The trend in CPU time and condition numbers is similar to that of the previous table.

Results obtained when Eq.(82) (the 2-D coupled Burgers system) is solved are displayed in Table 3, and Figures 1 and 2. In Table 3, the single domain approach is invoked using $N_x = 40,\ N_y = 40$ grid points in the spatial domains and $N_t = 50$ grid points over the entire time domain of length $T = 10$. In the multidomain approach, $q = m = 4$ overlapping subintervals are used in space directions with $N_x = N_y = 10$ grid points, and in time direction $p = 5$ non-overlapping subintervals with $N_t = 10$ are used. In Table 3, we observe that as the values of the Reynolds (Re) number increase, the single domain approach shows a greater drop in accuracy than the multidomain approach. It is worth mentioning that the factor $\frac{1}{Re}$ is the coefficient of higher ordered space derivatives in Eq.(82) and so for large values of Re, this coefficient is small. Multiplying the differentiation matrices by this small value leads to reduced in the precision of the entries of the coefficient matrix derived from the resulting system of linear algebraic equations. Numerical computations with small valued matrix entries have pronounced round-off errors, which reduces the accuracy of numerical approximations. Incorporating the overlapping grids technique in the spatial domain introduces scaling in the coefficient matrix, which in turn counters the effect of large values of Re. Figure 1 shows the graph of error norms plotted against the length of the time interval. We examine the variation of error norms by considering results from both a single domain approach ($p = 1$) and a multidomain approach, in which the large time domain has been broken into 5 and 10 subintervals

($p = 5$ and $p = 10$). The total number of grid points over the entire domain is kept constant with Nt = 10 per unit time. The algorithm is run using $T = 4,\ T = 8,\ T = 12,\ T = 16,\ T = 20,$ time intervals. Here the spatial domain is not decomposed and values of Reynolds number are kept small at $Re = 5$. The results in Figure 1 make clear the need for the non-overlapping grids technique when solving problems over large time intervals. It can be observed that for small time intervals, single domain approach gives the best accuracy for a fixed number of grid points across all the three cases considered. Breaking a small time interval into further intervals comes at the expense of increasing the number of grid points, which is equivalent to reducing the degree of the interpolating polynomial approximating the solution of the differential equation, thereby, leading to poor approximations. Having examined the usefulness of the non-overlapping grids technique in the case of large time intervals, in Figure 2 we maintain a large time interval at $T = 20$ and subdivide it into 5 subintervals, then compare this to the single domain approach while investigating the effect of varying the Reynolds number values on the accuracy of numerical approximations. We observe that for the large Reynolds (Re) numbers and large time intervals, using both overlapping subintervals in space and non-overlapping subintervals in time give the best accuracy.

Results obtained when Eq.(87) (the Brusselator system) is solved are displayed in Table 4, and Figures 3 and 4. Table 4 shows results obtained using the single and multiple domains approach for both small and large time intervals. Results given in Table 4 are closely explained by theoretical results of the error bound theorems. We notice that unlike in the previous three examples, in which higher ordered derivatives of the solution are bounded, the solution of Eq.(87) is unbounded within the domain of approximation. Consequently, increasing the number of grid points leads to large interpolation errors, which are propagated into the numerical solution of the differential equation. For a large time domain, the single approach gives less accurate results; there is a noticeable improvement in accuracy when the multiple domain approach is adopted for large time intervals. In Figure 3, error norms resulting from the use of $N_t = 20$ and $N_t = 10$ grid points over a single time interval of length $T = 5$ units are compared. We notice that when the interval is large, using many grid points has little effect on accuracy. This phenomenon is caused by the unbounded nature of higher ordered derivatives of the solution of Eq.(87). In Figure 4, we re-examine the decomposition of a large time domain and compare the solution by a single domain approach with that for the multiple domain approach, while maintaining a constant number of grid points over the entire time interval. Consequently, as supported by error bound theorems, the use of a large number of

grid points on a single domain records less accurate results for problems with unbounded higher ordered derivatives. In fact, beyond a certain threshold length of the time interval, the numerical scheme becomes unstable. A domain decomposition approach improves the accuracy of numerical approximations and constitutes a stable numerical algorithm.

Table 1: Absolute error norms values for spectral approximation of Example 1 using different lengths of time intervals $[0, T]$: $N_x = 10,\ N_y = 10, q = 2, m = 2,\ N_t = 10/$ per unit length in time.

T	Single Domain	Multiple Domains
1.0	2.26291e-12	2.97384e-13
2.0	2.95380e-12	4.15001e-13
3.0	7.35126e-11	2.37643e-13
4.0	6.01130e-10	1.85810e-13
5.0	2.75105e-09	1.52454e-13
CPU time (sec)	15.259924	1.717759
Cond Number	4.7356e+05	7.9950e+03

Table 2: Absolute error norms values for spectral approximation of Example 2 using different lengths of time intervals $[0, T]$: $N_x = 10,\ N_y = 10, q = 2, m = 2,\ N_t = 10$ per unit length in time.

T	Single Domain	Multiple Domains
1.0	2.29941e-11	1.71063e-12
2.0	2.19984e-11	2.13518e-12
3.0	2.30047e-11	3.24896e-12
4.0	5.26619e-11	6.46949e-12
5.0	3.48477e-11	5.11946e-12
CPU time (sec)	6.507918	0.636237
Cond Number	7.3179e+04	1.8094e+03

Table 3: Error norm values obtained when Example 3 is solved for different values of Reynolds number, Re : $N_x = 10$, $N_y = 10$, $q = 4$, $m = 4$, $[0, T] = [0, 10]$, $N_t = 10$, $p = 5$.

	Single Domain		Multiple Domains	
Re	Error Norm in u	Error Norm in v	Error Norm in u	Error Norm in v
1.0	4.09006e-13	5.62328e-13	2.37022e-12	4.03122e-12
5.0	4.26503e-12	4.26625e-12	2.46692e-12	3.52773e-12
10.0	9.17247e-09	9.17326e-09	3.56237e-12	6.11633e-12
15.0	4.22525e-08	4.22845e-08	3.00593e-11	4.43479e-11
20.0	4.23489e-07	4.32432e-07	2.69562e-10	3.20433e-10

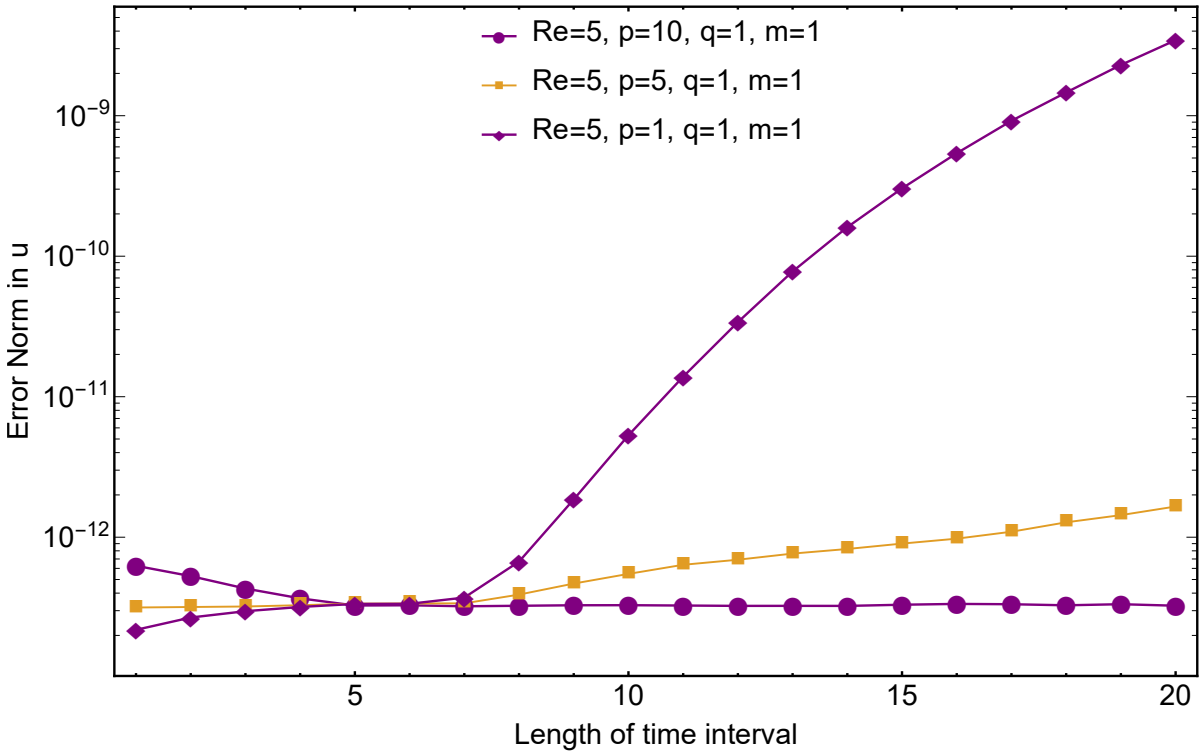


Figure 1: Error norms in u for different time intervals $[0, T]$ for 2D coupled Burgers system

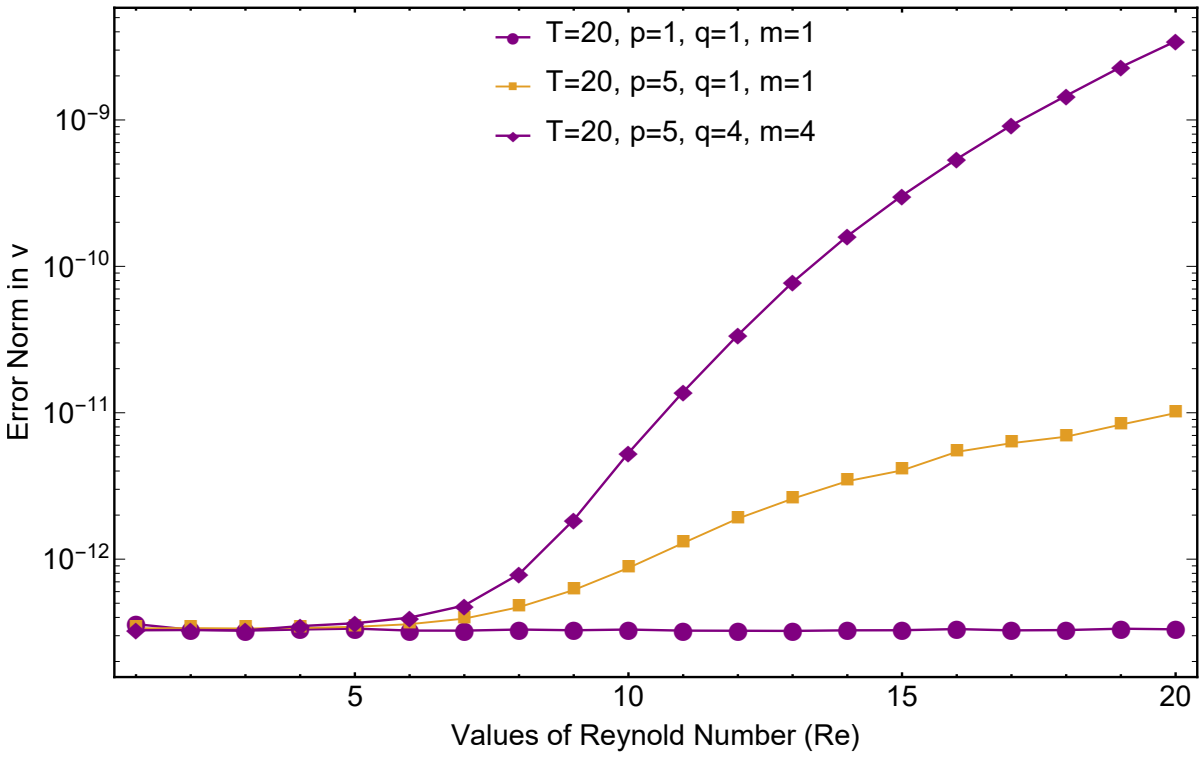


Figure 2: Error norms in v for different values of Reynolds number for 2D coupled Burgers system

Table 4: Error norm values for different time intervals obtained when Example 4 is solved: $N_x = 10$, $N_y = 10$, $q = 2$, $m = 2$, $N_t = 10/$ per unit length in time.

	Single Domain		Multiple Domain	
T	Absolute Error u	Absolute Error v	Absolute Error u	Absolute Error v
1.0	1.89722e-13	2.13396e-13	9.53246e-14	1.10242e-13
2.0	1.10363e-12	1.19616e-12	7.75372e-13	8.65135e-13
3.0	2.23119e-09	2.33731e-09	2.80567e-12	3.01118e-12
4.0	4.35874e-01	3.38041e-01	2.22283e-11	2.31627e-11
5.0	4.71456e+02	5.69322e+02	4.13003e-09	7.22800e-09
CPU time (sec)	17.345153		1.826348	
Cond Number	3.6427e+07		2.9215e+03	

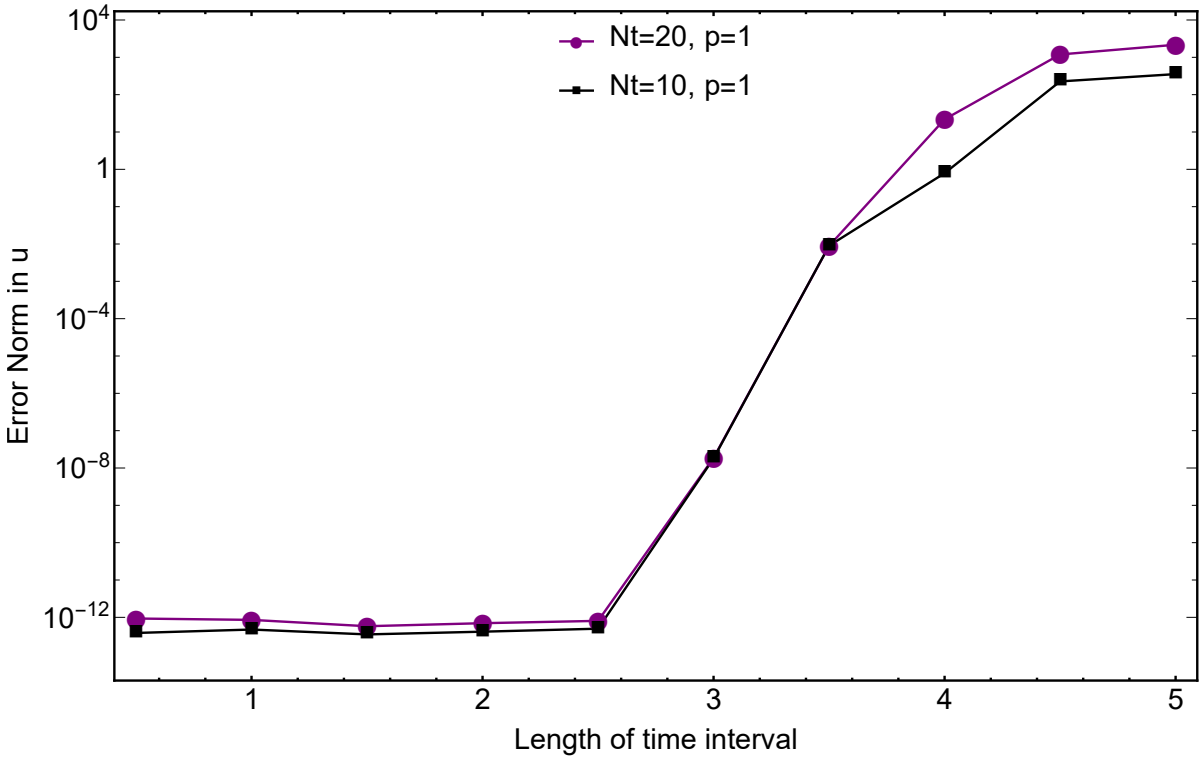


Figure 3: Error norms in u for different time intervals $[0, T]$ for 2D coupled Brusselator system (Varying grid points)

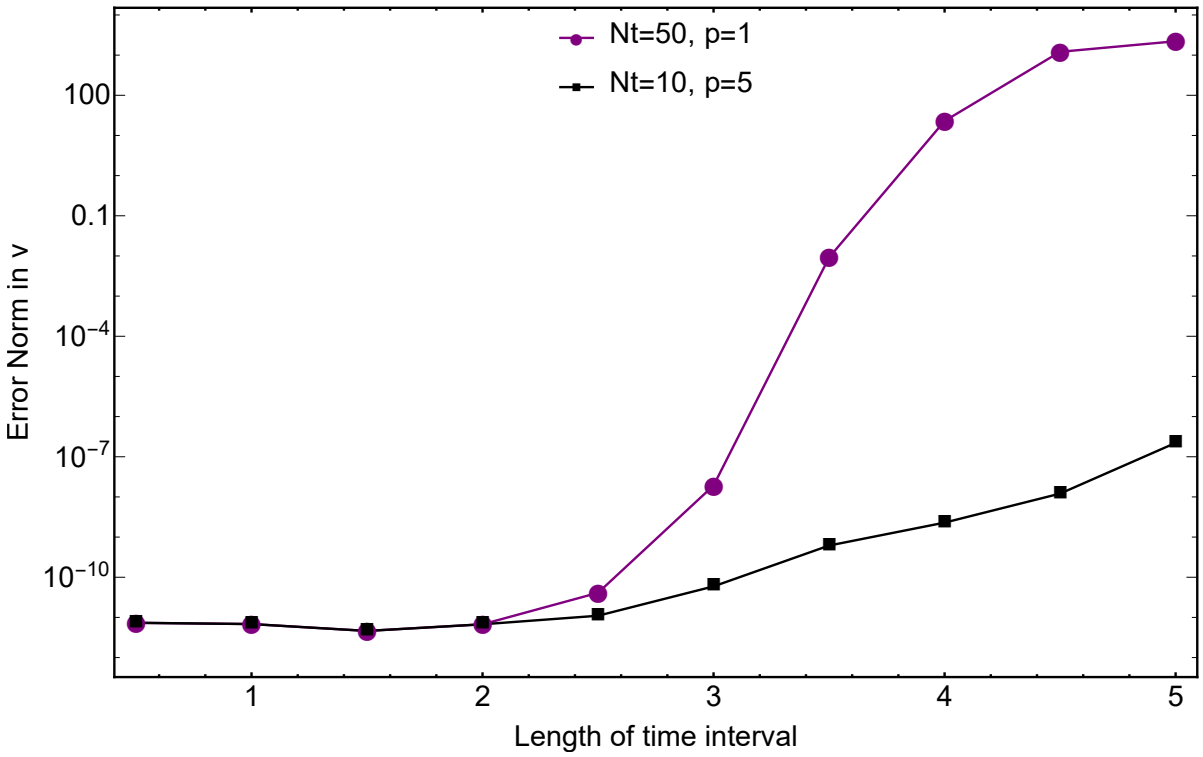


Figure 4: Error norms in v for different time intervals $[0, T]$ for 2D coupled Brusselator system (fixed grid points)

6. Conclusion

In this work, a modified numerical method, namely the overlapping grids based multidomain trivariate spectral collocation method, for solving nonlinear two-dimensional initial-boundary value problems is proposed. The method has been described and successively applied on typical examples of two-dimensional initial-boundary value problems reported in the literature as a single nonlinear equation or systems of nonlinear equations. From the numerical simulations, we arrive at the following conclusions. The standard single domain based trivariate spectral collocation method, when applied to solve nonlinear two-dimensional initial-boundary value problems, is accurate only for those defined on a small time interval. When the time interval is large, many grid points are required to achieved results with sufficient accuracy and this is only applicable if the solutions' higher ordered derivatives are bounded within the computational domain. For solutions with unbounded higher ordered derivatives, the application of the standard single domain based trivariate spectral collocation method is limited to problems defined on a small time interval because a large number of grid points would be computationally expensive and would not guarantee the accuracy of the numerical approximations. For fluid flow problems, demonstrated in this case by the two-dimensional Burgers system, the single domain approach gives accurate results only for small values of the parameters governing the flow. The main challenge in deciding whether to use the standard trivariate spectral collocation method is that in many practical problems the exact solutions are not known at the outset. Therefore, the standard single domain trivariate spectral collocation method has many shortcomings. By contrast, the modified overlapping grids based multidomain spectral collocation method is reliable, as it gives highly accurate results in short CPU time. In terms of accuracy, the superiority of the current method of solution can be attributed to the purely spectral collocation discretization performed in all variables coupled with the small condition numbers of the coefficient matrices in the multidomain approach. The short CPU time required to generate results is linked to the small-sized coefficient matrix in the multidomain approach, which is easy to invert. Owing to its remarkable benefits, the current method of solution is a suitable alternative method for solving two-dimensional nonlinear initial-boundary value problems defined on large regular spatial domains over a large time interval. In addition, the overlapping grids technique is a useful tool when solving differential equations with large parameter values and those defined over large spatial domains. Future work will entail an extension of the current numerical method to problems defined on circular domains.

7. Acknowledgement

The authors are thankful to the University of KwaZulu-Natal for providing essential research resources.

Declaration of interest:

There is no conflict of interest regarding publication of this paper.

References

[1] B. Costa, *Spectral Methods for Partial Differential Equations*, CUBO: A Mathematical Journal, 6(2004), pp. 1-32.

[2] M.G. Macaraeg, and C.L. Streett, *Improvements in Spectral Collocation Discretization Through a Multiple Domain Technique*, Applied Numerical Mathematics, 2(1986), pp. 95-108.

[3] L. Yi, and Z. Wang, *Legendre Spectral Collocation Method for Second-order Nonlinear Ordinary/Partial Differential Equations*, Discrete and Continuous Dynamical Systems-B, 19(2014), pp. 299-322.

[4] T. Zhao, C. Li, Z. Zang, and Y. Wua, *Chebyshev Legendre Pseudospectral Method for the Generalised Burgers-Fisher Equation*, Applied Mathematical Modelling, 36(2012), pp. 1046-1056.

[5] A. Tadmor, *A Review of Numerical Methods for Nonlinear Partial Differential Equations*, Bulletin of the American Mathematical Society, 49(2012), pp. 507-554.

[6] M.R. Malik, T.A. Zang, and M.Y. Hussaini, *A Spectral Collocation Method for the Navier-Stokes Equations*, Journal of Computational Physics, 61(1985), pp. 64-88.

[7] R. Baltensperger, and J.P. Berrut, *The Errors in Calculating the Pseudospectral Differentiation Matrices for Chebyshev Gauss-Lobatto points*, Computers & Mathematics with Applications, 37(1999), pp. 41-48.

[8] S.S. Motsa, F.M. Samuel, and S. Shateyi (2016), *Solving Nonlinear Parabolic Partial Differential Equations Using Multidomain Bivariate Spectral Collocation Method*, Nonlinear Systems - Design, Analysis, Estimation and Control, In D. Lee (Ed.), InTech, DOI: 10.5772/64600.

[9] A.D. Polyanin, and V.F. Zaitsev, *Handbook of Nonlinear Partial Differential Equations*, Chapman and Hall/CRC Press, Boca Raton, 2004.

[10] S. Pamuk, and N. Pamuk, *Solution of Two-Dimensional Heat and Mass Transfer Equation With Power-Law Temperature-Dependent Thermal Conductivity*, TWMS Journal of Applied and Engineering Mathematics, 4(2014), pp. 199-208.

[11] J.M. Burgers, *Application of a Model System to Illustrate Some Points to the Statistical Theory of Free Turbulance*, The Royal Netherlands Academy of Art and Sciences, 43(1940), pp. 2-12.

[12] I. Prigogine, and R. Lefever, *Symmetry Breaking Instabilities in Dissipative Systems II*, Journal of Chemical Physics, 48(1968), pp. 1695-1700.

[13] A.R. Bahadir, *A Fully Implicit Finite-Difference Scheme for Two-Dimensional Burgers' Equations*, Applied Mathematics and Computation, 137(2003), pp. 131-137.

[14] W.T. Ang, *The Two Dimensional Reaction-diffusion Brusselator System: a Dual-Reciprocity Boundary Element Solution*, Engineering Analysis with Boundary Elements, 27(2003), pp. 897-903.

[15] A.J. Chamkha, and A.M. Aly, *Heat and Mass Transfer in Stagnation-Point Flow of a Polar Fluid Towards a Stretching Surface in Porous Media in the Presence of Soret, Dufour and Chemical Reaction Effects*, Chemical Engineering Communications, 198(2010), pp. 214-234.

[16] C.A.J. Fletcher, *Generating exact Solution of the Two-Dimensional Burgers' Equation*, Numerical Methods in Fluids, 3(1983), pp. 213-216.

[17] R. Abazari, and A. Borhanifar, *Numerical Study of the Solution of the Burgers and Coupled Burgers Equations by a Differential Transformation Method*, Computers & Mathematics with Applications, 59(2010), pp. 27112722.

[18] G. Zhao, X. Yu, and R. Zhang, *The New Numerical Method for Solving the System of Two-Dimensional Burgers Equations*, Computers & Mathematics with Applications, 62(2011), pp.32793291 .

[19] S.U. Islam, B. Sarler, R. Vertnik, and G. Kosec, *Radial Basis Function Collocation Method for the Numerical Solution of the Two-Dimensional Transient Nonlinear Coupled Burgers Equations*, Applied Mathematical Modelling, 36(2012), pp. 11481160.

[20] E.H. Twizell, A.B. Gumel, and Q. Cao, *A Second-Order Scheme for the Brusselator Reaction-Diffusion System*, Journal of Mathematical Chemistry, 26(1999), pp. 297316.

[21] F. Khani, F. Samadi, and S.H. Nezhad, *New Exact Solutions of the Brusselator Reaction-Diffusion Model Using the Exp-Function Method*, Mathematical Problems in Engineering, vol. 2009, Article ID 346461, 9 pages, 2009. https://doi.org/10.1155/2009/346461.

[22] J. Biazar, and Z. Ayati *An Approximation to the Solution of the Brusselator System by Adomian decomposition Method and Comparing the Results with Runge-Kutta Method*, International Journal of Contemporary Mathematical Sciences, 2(2007), pp. 983-989.

[23] G.B. Folland, *Introduction to Partial Differential Equations*, Princeton University Press, Washington, 1995.

[24] R.E. Bellman, and R.E. Kalaba, *Quasilinearization and Nonlinear Boundary-Value Problems*, Elsevier Publishing Company, New York, 1965.

[25] L.N. Trefethen, *Spectral Methods in MATLAB*, SIAM, Philadelphia, 2000.

[26] [26] H.H. Yang, B.R. Seymour, and B.D. Shizgal, *A Chebyshev Pseudospectral multi-Domain Method for Flow Past a Cylinder, up to Re=150*, Computers Fluids, 23(1994), pp. 829-851.

[27] [27] M. Gasca and T. Sauer, *On the History of Multivariate Polynomial Interpolation*, Computational and Applied Mathematics, 122(2000), pp. 23-35.

[28] [28] A.H Bhrawy, *A Highly Accurate Collocation Algorithm for* $1+1$ *and* $2+1$ *Fractional Percolation Equations*, Journal of Vibration and Control, 5(2015), pp. 1-23.

[29] H.E. Salzer, *Converting Interpolation Series into Chebyshev Series by Recurrence Formulas*, Mathematics of Computation, 30(1976), pp. 295-302.

[30] G. Zhao, X. Yu, and R. Zhang, *The new Numerical Method for Solving the system of Two-Dimensional Burgers Equation*, Computers and Mathematics with Applications, 62(2011), pp. 3279-3291.

[31] K. Yildirim, *A Solution Method for Solving Systems of Nonlinear PDEs*, World Applied Sciences Journal, 18(2012), pp. 1527-1532.

[32] S.U. Islam, and A. Ali, S. Haq, *A Computational Modeling of the Behavior of the Two-Dimensional ReactionDiffusion Brusselator System*, Applied Mathematical Modelling, 34(2010), pp. 3896-3909.

Chapter 7

Conclusion

In this thesis, we have successfully developed and applied new modified spectral collocation-based methods to find solutions of nonlinear ordinary differential equations, one-dimensional partial differential equations of hyperbolic and parabolic type, and two-dimensional nonlinear elliptic PDEs. The final chapter showed the development and application of the new numerical method, namely the trivariate spectral collocation method, for the solution of two-dimensional initial-boundary value problems. The numerical methods used relied on the quasi-linearization method to simplify the nonlinear differential equations and Lagrange interpolating polynomial as the approximating functions for the solution of the differential equations. A purely spectral collocation discretization at Chebyshev-Gauss-Lobatto points was applied in both space and time variables. The improvement in the current solution techniques from the traditional, which were based on a single domain, was achieved through the introduction of an overlapping grids approach on large spatial domains and using the non-overlapping subdomains approach on large time intervals. New error bound theorems, and proofs, on polynomial interpolation for univariate, bivariate, and trivariate cases were presented. Most importantly, new versions of the error bound theorems, and proofs, that are as a consequence of incorporating the overlapping grid and non-overlapping subdomains were added to demonstrate the theory behind the remarkable benefits of both approaches. From the general numerical results and the new findings of this study, we arrive at the following conclusions.

i. Incorporation of an overlapping grid approach on large spatial domains and the use of non-overlapping subintervals when solving differential equation over large time intervals using spectral collocation methods give accurate numerical results over relatively short computation time.

ii. the overlapping grid approach introduces sparsity in the spatial differentiation matrices, thereby rendering a stable numerical scheme.

iii. Improved accuracy is attributed to the small condition numbers of the coefficient matrices

obtained when using the multidomain approach.

iv. The small-sized matrices in the multidomain approach are easy into invert, and this translates to short computation time for generating numerical results.

v. The performance of the numerical methods described in this thesis is reliable. These numerical methods give general schemes, that are easily adaptable to solve a wide range of differential equations encountered in real life.

vi. Methods of solution used to solve ordinary differential equations of boundary layer flow types represent a modified solution approach to the existing method. In addition, the methods developed and applied on one-dimensional hyperbolic, one-dimensional parabolic, and two-dimensional elliptic partial differential equations represent an extension of the spectral collocation methods that already exist. Furthermore, incorporating overlapping grid concepts on a purely spectral collocation numerical method is new.

vii. Error bound theorems, and proofs, on Lagrange polynomial interpolation using Chebyshev-Gauss-Lobatto nodes, are new. The error bounds are useful as they give an idea of the sizes of errors that are propagated in the solution of the differential equations when using interpolating polynomials as the approximating functions. Indeed, the different version of error bounds in the cases of a single domain and multiple domains gives an interesting theoretical illustration of the benefits of adopting domain decomposition technique when solving differential equations over large domains.

viii. The trivariate spectral collocation method developed and applied to solve two-dimensional initial-boundary value problems is new. We have demonstrated that incorporation of the domain decomposition technique improves its accuracy and computational efficiency over that of the single domain approach. It is worth noting that this method can be adapted to solve three-dimensional problems.

ix. As demonstrated in the numerical results for two-dimensional Burgers equations, benefits of the overlapping grid approach are also evident in solving reaction-diffusion problems for very small or large values of the diffusion coefficient.

x. The benefits of incorporating the domain decomposition technique in a spectral collocation method over the single domain approach are not manifested when differential equations are

solved over small domains.

Future research would include extensions of similar solution techniques to solve differential equations defined over irregular domains. This can be achieved through development of new numerical methods of solution of differential equations, that, combines the benefits of the spectral collocation-based methods and the finite element-based methods.

Bibliography

[1] F. John (1982), *Partial Differential Equations*, 4th edition, New York: Springer-Verlag.

[2] P. Aviles, and J. Sandefur, Nonlinear Second Order Equations with Applications to Partial Differential Equations, *Journal of Differential Equations*, 58(1985), pp. 404-427.

[3] A. Ralston, and P. Rabinowitz (1988), *A First Course in Numerical Analysis*, New York: McGraw Hill International Editions.

[4] S.D. Conte, and C.W. de Boor (1981), *Elementary Numerical Analysis*, New York: McGraw Hill International Editions.

[5] D. Zill, and W. Wright (2013), *Differential Equations and Boundary Value Problems*, 8th edition, Brooks/Cole: Boston.

[6] J.H. He, Linearized Perturbation Technique and its Applications to Strongly Nonlinear Oscillators, *Computers and Mathematics with Applications*, 45(2003), pp. 1-8.

[7] A.H. Nayfeh, C.M. Chin, and J. Pratt, Perturbation Methods in Nonlinear Dynamics and Applications to Machining Dynamics, *Manufacturing Science and Engineering*, 119(997), pp. 485-493.

[8] S. Somali, and G. Gokmen, Adomian Decomposition Method for Nonlinear Sturm-Liouville Problems, *Surveys in Mathematics and its Applications*, 2(2007), pp. 11-20.

[9] S. Liao, On the Homotopy Analysis Method for Nonlinear Problems, *Applied Mathematics and Computation*, 147(2004), pp. 499-513.

[10] J.H. He, Variational Iteration Method-Some Recent Results and New Interpretations, *Computational and Applied Mathematics*, 207(2007), pp. 3-17.

[11] M.A. Abdou, and A.A. Soliman, New Applications of Variational Iteration Method, *Physica D: Nonlinear Phenomena*, 211(2005), pp. 1-8.

[12] M.A. Abdou, and A.A. Soliman, Variational Iteration Method for Solving Burger's and Coupled Burger's Equations, *Computational and Applied Mathematics*, 181(2005), pp. 245-251.

[13] F.B. Hildebrand (1956), *Introduction to Numerical Analysis*, New York: McGraw-Hill.

[14] F.B. Hildebrand (1968), *Finite-Difference Equations and Simulations*, Englewood Cliffs, NJ: Prentice-Hall.

[15] X. Dexuan, An Improved Approximate Newton Method for Implicit Runge-Kutta Formulas, *Computational and Applied Mathematics*, 33(2010), pp. 78-89.

[16] W. Cheney, and D. Kincaid (2004), *Numerical Mathematics and Computing*, 6th edition, Belmont, CA: Thomson Higher Education.

[17] M.Y. Hussaini, D.A. Kopriva, and A.T. Patera, Spectral Collocation Methods, *Applied Numerical Mathematics*, 5(1989), pp. 177-208.

[18] P.J. Davis (1965), *Interpolation and Approximation*, 2nd edition; New York: Blais-dell.

[19] M. Dehghan, and F. Izadi, The Spectral Collocation Method With Three Different Bases for Solving a Nonlinear Partial Differential Equation Arising in Modeling of Nonlinear Waves, *Mathematical and Computer Modelling*, 53(2011), pp. 1865-1877.

[20] S.S. Motsa, P. Sibanda, and S. Shateyi, A New Spectral-Homotopy Analysis Method for Solving a Nonlinear Second Order BVP, *Communications in Nonlinear Science and Numerical Simulation*, 15(2010), pp. 2293-2302.

[21] S.J. Liao, *The Proposed Homotopy Analysis Technique For the Solution of Nonlinear Problems, PhD thesis*, Shanghai, Jiao Tong University, 1992.

[22] S.S. Motsa, and S. Shateyi, A Successive Linearization Method Approach to Solve Lane-Emden Type of Equations, *Mathematical Problems in Engineering*, 14(2012), pp. 19-32.

[23] S.S. Motsa, A New Spectral Local Linearization Method for Nonlinear Boundary Layer Flow Problems, *Journal of Applied Mathematics*, 15(2013), pp. 67-78.

[24] S.S. Motsa, and P. Sibanda, On Extending the Quasilinearization Method to Higher Order Convergent Hybrid Schemes Using the Spectral Homotopy Analysis Method, *Journal of Applied Mathematics*, 13(2013), pp. 107-115.

[25] S.S. Motsa, P. Dlamini, and M. Khumalo, A New Multi-stage Spectral Relaxation Method for Solving Chaotic Initial Value Systems, *Nonlinear Dynamics*, 72(2013), pp. 265-283.

[26] S.S. Motsa, V.M. Magagula, and P. Sibanda, A Bivariate Chebyshev Spectral Collocation Quasilinaerization Methods for Nonlinear Evolution Parabolic Equations, *The Scientific World Journal*, 20(2014), pp. 251-260.

[27] A. Rezazadeh, M. Mahmoudi, and M. Darehmiraki, Space time Spectral Collocation Method for One-dimensional PDE Constrained Optimisation, *International Journal of Control*, 1(2018), pp. 34-42.

[28] A. Shadrin, Error Bounds for Lagrange Interpolation, *Journal of Approximation Theory*, 80(1995), pp. 25-49.

[29] S.A.M. Ismail, On Bivariate Polynomial Interpolation, *East Journal on Approximations*, 8(2002), pp. 209-228.

[30] G. W. Howell, Derivative Error Bounds for Lagrange Interpolation: An extension of Cauchy's Bound for the Error of Lagrange Interpolation, *Journal of Approximation Theory*, 67(1991), pp. 164-173.

[31] F.J. Narcowich, and J.D. Ward, Norms of Inverses and Condition Numbers of Matrices Associated With Scattered Data, *Journal of Approximation Theory*, 64(1991), pp. 69-94.

[32] C. Canuto, M.Y. Husseini, A. Quarteroni, and T.A. Zang (2006), *Spectral Methods: Fundamentals in Single Domains*, Springer-Verlag: New York.

[33] S. Ghili, and G. Iaccarino, Reusing Chebyshev Points for Polynomial Interpolation, *Numerical Algorithms*, 70(2015), pp. 249-267.

[34] B. Fischer, and R. Freund, Chebyshev Polynomials Are Not Always Optimal, *Journal of Approximation Theory*, 65(1991), pp. 261-272.

[35] A.H. Turetskii, The Bounding of Polynomials Prescribed at Equally Distributed Points, *Proc. Pedag. Inst. Vitebsk*, 3(1940), pp. 117-127.

[36] L. Bos, Bounding the Lebesgue Function for Lagrange Interpolation in a Simplex, *Jaorunal of Approximation Theory*, 38(1983), pp. 43-59.

[37] T. Bloom, The Lebesgue Constant for Lagrange Interpolation in the Simplex, *Journal of Approximation Theory*, 54(1988), pp. 338-353.

[38] S.J. Smith, Lebesgue Constant in Polynomial Interpolation, *Annales Mathematics et Informaticae*, 33(2006), pp. 109-123.

[39] A.H. Bhrawy, A Highly Accurate Collocation Algorithm for $1+1$ and $2+1$ Fractional Percolation Equations, *Journal of Vibration and Control*, 5(2015), pp. 1-23.

[40] A.H. Bhrawy, and M.A. Zaky, A Method Based on the Jacobi Tau Approximation for Solving Multi-term Time-space Fractional Partial Differential Equations, *Journal of Computational Physics*, 60(2014), pp. 10-16.

[41] L.N. Trefethen, and J.A.C. Weideman, Two Results on Polynomial Interpolation in Equally Spaced Points, *Journal of Approximation Theory*, 65(1991), pp. 247-260.

[42] A. Chakrabarti, J. Hamsapriye, Derivation of the Errors Involved in Interpolation and Their Application to Numerical Quadrature Formulae, *Computational and Applied Mathematics*, 92(1998), pp. 59-68.

[43] J.F. Epperson, On the Runge example, *American Mathematical Monthly*, 94(1987), pp. 329-341.

[44] S. Shateyi, S.S. Motsa, and Y. Khan, A New Piecewise Spectral Homotopy Analysis of the Michaelis-Menten Enzymatic Reactions Model, *Numerical Algorithms*, 66(2014), pp. 495-510.

[45] S.S. Motsa, A New Piecewise-Quasilinearization Method for Solving Chaotic Systems of Initial Value Problems, *Central European Journal of Physics*, 10(2012), pp. 936-946.

[46] T.F. Chan, and T.P. Mathew, Domain Decomposition Algorithms, *Acta Numerica*, 3(1994), pp. 61-143.

[47] R.E. Bellman, and R.E. Kalaba (1965), *Quasilinearization and Nonlinear Boundary-Value Problems*, Elsevier: New York.

[48] X. Su, L. Zheng, and X. Zhang, DTM-BF Method and Dual Solutions for an Unsteady MHD Flow Over a Permeable Shrinking Sheet With Velocity Slip, *Applied Mathematics & Mechanics*, 33(2012), pp. 1555-1568.

[49] K. Parand, N. Pakniat, and Z. Delafkar, Numerical Solution of the Falkner-Skan Equation with Stretching Boundary by Collocation Method, *International Journal of Nonlinear Science*, 11(2011), pp. 275-283.

[50] M.A. El-Aziz, Flow and heat transfer over an unsteady stretching surface with Hall effect, *Meccanica*, 45(2010), pp. 97-109.

[51] E. Magyari, and B. Keller, Heat and Mass Transfer in the Boundary Layers on an Exponentially Stretching Continuous Surface, *Physics D Applied Physics*, 32(1999), pp. 577-585.

[52] B.C. Sakiadis, Boundary-Layer Behavior on Continuous Solid Surfaces: II. The Boundary-Layer on Continuous Flat Surface, *AIChE Journal*, 7(1961), pp. 221- 225.

[53] L. Crane, Flow Past a Stretching Plate, *Zeitschrift für. Angewandte Mathematik und Physik (ZAMP)*, 21(1970), pp. 645-647.

[54] A. Chakrabarthi, and A.S. Gupta, Hydromagnetic Flow and Heat Transfer Over a Stretching Sheet, *Quarterly of Applied Mathematics*, 37(1979), pp. 73-78.

[55] T. Chiam, Magneto Hydrodynamic Boundary Layer Flow Due to a Continuous Moving Flat Plate, *Computers & Mathematics with Applications*, 26(1993), pp. 1-8.

[56] A.K. Jhankal, and M. Kumar, MHD Boundary Layer Flow Past a Stretching Plate with Heat Transfer, *International Journal of Engineering and Science*, 2(2013), pp. 9-13.

[57] N.F.M, Noor, O. Abdulaziz, and I. Hashim, MHD Flow and Heat Transfer in a Thin Liquid Film on an Unsteady Stretching Sheet by the Homotopy Analysis Method, *International Journal for Numerical Methods in Fluids*, 63(2010), pp. 357-373.

[58] C.Y. Wang, Liquid Film on an Unsteady Stretching Sheet, *Quarterly of Applied Mathematics*, 48(1990), pp. 601- 610.

[59] M. Miklavcic, and C. Y. Wang, Viscous Flow Due to a Shrinking Sheet, *Quarterly of Applied Mathematics*, 64(2006), pp. 283-290.

[60] M. Katagiri, The Effect Hall Currents on the Viscous Flow Magnetohydrodynamic Boundary Layer Flow past a semi-infinite flat plate, *Physical Society of Japan*, 27(2015), pp. 40-51.

[61] H. Sato, The Hall Effect in the Viscous Flow of Ionized Gas Between Two Parallel Plates Under Transverse Magnetic Field, *Physical Society of Japan*, 16(1961), pp. 14-27.

[62] I. Pop, and V.M. Soundalgekar, Effects of Hall-Current on Hydromagnetic Flow Near a Porous Plate, *Acta Mechanica*, 20(1974), pp. 315-318.

[63] V.M. Falkner, and S.W. Skan, Some Approximate Solutions of the Boundary Layer Equations, *Philosophical Magazine*, 12(1931), pp. 865-896.

[64] M.B. Zaturska, and W.H.H. Banks, A New Solution Branch of the Falkner-Skan Equation, *Acta Mechanica*, 152(2001), pp. 197-201.

[65] A. Asaithambi, Solution of the Falkner-Skan Equation by Recursive Evaluation of Taylor Coefficients, *Computational and Applied Mathematics*, 176(2005), pp. 203-214.

[66] N.S. Elgazery, Numerical Solution for the Falkner-Skan Equation, *Chaos, Solitons & Fractals*, 35(2008), pp. 738-746.

[67] E.M.A. Elbashbeshy, and M.A.A. Bazid, Heat Transfer Over an Unsteady Stretching Surface, *Heat and Mass Transfer*, 41(2004), pp. 1-4.

[68] S.J. Liao, An Analytic Solution of Unsteady Boundary Layer Flows Caused by an Impulsively Stretching Plate, *Communications in Nonlinear Science and Numerical Simulation*, 11(2006), pp. 326-339.

[69] Y. Tan, and S.J. Liao, Series Solution of Three-Dimensional Unsteady Laminar Viscous Flow Due to a Stretching Surface in a Rotating Fluid, *Journal of Applied Mechanics*, 74(2007), pp. 1011-1018.

[70] T. Hayat, Z. Abbas, and M. Sajid, On The Analytic Solution of Magnetohydrodynamic Flow of a Second Grade Fluid Over a Shrinking Sheet, *Journal of Applied Mechanics*, 74(2007), pp. 1165-1171.

[71] N.F.M. Noor, S.A. Kechilb, and I. Hashim, Simple Nonperturbative Solution For MHD Viscous Flow Due to a Shrinking Sheet, *Communications in Nonlinear Science and Numerical Simulation*, 15(2010), pp. 144-148.

[72] T. Fang, and J. Zhang, Closed Form Exact Solutions of MHD Viscous Flow Over a Shrinking Sheet, *Communications in Nonlinear Science and Numerical Simulation*, 14(2009), pp. 2853-2857.

[73] C. Midya, Exact Solutions of Chemically Reactive Solute Distribution in MHD Boundary Layer Flow over a Shrinking Surface, *Chinese Physics Letter*, 29(2012), pp. 14-27.

[74] M. Sajid, and T. Hayat, The Application of Homotopy Analysis Method for MHD Viscous Flow Due to a Shrinking Sheet,*Chaos, Solitons & Fractals*, 39(2009), pp. 1317-1323.

[75] D.R. Hartree, On an Equation Occurring in Falkner and Skan's Approximate Treatment of the Equations of the Boundary Layer, *Proceedings of the Cambridge Philosophical Society*, 33(1937), pp. 223-239.

[76] H. Weyl, On the Differential Equations of the Simplest Boundary-Layer Problem, *Annals of Mathematics*, 43(1942), pp. 381-407.

[77] B.L. Kuo, Application of the Differential Transformation Method to the Solutions of Falkner-Skan Wedge Flow, *Acta Mechanica*, 164(2003), pp. 161-174.

[78] S.J. Liao , and A. Campo, Analytic Solutions of the Temperature Distribution in Blasius Viscous Flow Problems, *Journal of Fluid Mechanics*, 453(2002), pp. 411-425.

[79] S. Abbasbandy, and T. Hayat, Solution of the MHD Falkner-Skan Flow by Homotopy Analysis Method, *Communications in Nonlinear Science and Numerical Simulation*, 14(2009), pp. 3591-3598.

[80] S.S. Motsa, and P. Sibanda, An Efficient Numerical Method For Solving Falkner-Skan Boundary Layer Flows, *International Journal for Numerical Methods in Fluids*, 69(2012), pp. 499-508.

[81] M. Lakestani, Numerical Solution For the Falkner-Skan Equation Using Chebyshev Cardinal Functions, *Acta Universitas Apulensis*, 27(2011), pp. 229-238.

[82] H.O. Kreiss, Initial Boundary Value Problems for Hyperbolic Systems, *Communications on Pure and Applied Mathematics*, 23(1970), pp. 277-298.

[83] M. Witten (1983), *Hyperbolic Partial Differential Equation*, Pergamon Press: New York.

[84] A.P.S. Selvadurai (2001), *Partial Differential Equations in Mechanics 1*, Springer: Stuttgart.

[85] M. Dehghan, and A. Taleei, A Chebyshev Pseudospectral Multidomain Method for the Soliton Solution of Coupled Nonlinear Schrodinger Equations, *Computer Physics Communications*, 182(2011), pp. 2519-2529.

[86] M. Dehghan, and A. Taleei, Numerical Solution of the Yukawa-Coupled Klein–Gordon–Schrödinger Equations Via a Chebyshev Pseudospectral Multidomain Method, *Applied Mathematical Modelling*, 36(2012), pp. 2340-2349.

[87] D.A. Kopriva, A Spectral Multidomain Method for the Solution of Hyperbolic Systems, *Applied Numerical Mathematics*, 2(1986), pp. 221-241.

[88] A. Taleei, and M. Dehghan, A Pseudo-Spectral Method That Uses an Overlapping Multidomain Technique for the Numerical Solution of Sine-Gordon Equation in One and Two Spatial Dimensions, *Mathematical Methods in the Applied Sciences*, 37(2014), pp. 1909–1923.

[89] M. Dehghan, and R. Salehi, A Method Based on Meshless Approach for the Numerical Solution of the Two-Space Dimensional Hyperbolic Telegraph Equation, *Mathematical Methods in the Applied Sciences*, 35(2012), pp. 1220–1233.

[90] M. Dehghan, and A. Mohebbi, High Order Implicit Collocation Method for the Solution of Two-Dimensional Linear Hyperbolic Equation, *Numerical Methods for Partial Differential Equations*, 25(2009), pp. 232-243.

[91] S. Fahy, J. O'Riordan, C. O'Sullivan, and P. Twomey, How Reflected Wave Fronts Dynamically Establish Hooke's Law in a Spring, *European Journal of Physics*, 33(2012), pp. 417-427.

[92] J.A. Elaf, Modified Treatment of Initial Boundary Value Problems for One-Dimensional Heat-like and Wave-like Equations using Adomian Decomposition Method, *Basrah Journal of Science*, 30(2012), pp. 86-105.

[93] T. Elzaki1, and E.M.A. Hilal, Analytical Solution for Telegraph Equation by Modified of Sumudu Transform "Elzaki Transform", *Mathematical Theory and Modeling*, 2(2012), pp. 104-112.

[94] M.A.E. AbdelrahmanA, E.H.M. Zahran, and M.M.A. Khater, Exact Traveling Wave Solutions for Modified Liouville Equation Arising in Mathematical Physics and Biology, *International Journal of Applied Mathematics*, 112(2015), pp. 75-87.

[95] M.A. Salam, Traveling-Wave Solution of Modified Liouville Equation by Means of Modified Simple Equation Method, *International Scholarly Research Network: Applied Mathematics*, 47(2012), pp. 52-56.

[96] E. Farshad, E. Farzad, H. Amin, and H. Nahid, Analytical Solution of Phi-Four Equation, *Technical Journal of Engineering and Applied Sciences*, 14(2013), pp. 1378-1388.

[97] G.B. Folland (1995), *Introduction to Partial Differential Equations*, Princeton: Princeton University Press.

[98] V. Chandraker, A. Awasthi, and S. Jayaraj, Numerical Treatment of Burgers-Fisher equation, *Procedia Technology*, 25(2016), pp. 1217-1225.

[99] T. Kawahara, and M. Tanaka, Interactions of Traveling Fronts: an Exact Solution of a Nonlinear Diffusion Equations, *Physics Letters*, 97(1983), pp. 3-11.

[100] W. Wang, and A.J. Roberts, Diffusion Approximation for Self-similarity of Stochastic Advection in Burger's Equation, *Communications in Mathematical Physics*, 5(2014), pp. 37-48.

[101] R. FitzHugh, Mathematical Models of Threshold Phenomena in the Nerve Membrane, *Bulletin of Mathematical Biophysics*, 17(1955), pp. 257-278.

[102] H. Li, and Y. Guo, New Exact Solutions to the FitzHugh-Nagumo Equation, *Applied Mathematics and Computation*, 18(2006), pp. 524-528.

[103] M.C. Nucci, and P.A. Clarkson, The Nonclassical Method is More General Than the Direct Method for Symmetry Reductions. An Example of the FitzHugh-Nagumo Equation, *Physics Letters A*, 164(1992), pp. 49-56.

[104] R. FitzHugh, Impulse and the Physiological States in Models of Nerve Membrane, *Biophysical Journal*, 1(1961), pp. 445-466.

[105] H.T. Lin, and W.S. Yu, Combined Heat and Mass Transfer by Laminar Natural Convection From a Vertical Plate With Uniform Heat Flux and Concentration, *Heat & Mass Transfer*, 32(1997), pp. 293-299.

[106] K.R. Khair, and A. Bejan, Mass Transfer to Natural Convection Boundary Layer Flow Driven by Heat Transfer, *Heat & Mass Transfer*, 30(1985), pp. 369-376.

[107] A. Mongruel, M. Cloitre, and C. Allain, Scaling of Boundary-Layer Flows Driven by Double Diffusive Convection, *International Journal of Heat and Mass Transfer*, 39(1996), pp. 3899-3910.

[108] A.M. Wazwaz, The Tanh Method for Generalized Forms of Nonlinear Heat Conduction and Burgers-Fisher Equations, *Applied Mathematics and Computation*, 169(2005), pp. 321-338.

[109] H.N.A. Ismail, K. Raslan, and A.A.A. Rabboh, Adomian Decomposition Method for Burger's-Huxley and Burger's-Fisher's Equations, *Applied Mathematics and Computation*, 159(2004), pp. 291-301.

[110] I. Hashim, M.S.M. Noorani, and M.R.S. Al-Hadidi, Solving the Generalized Burger's-Huxley Equation Using the Adomian Decomposition Method, *Mathematical and Computer Modelling*, 16(2006), pp. 11-19.

[111] J.H. He, Homotopy Perturbation Method for Bifurcation of Nonlinear Problems, *Nonlinear Sciences and Numerical Simulation*, 6(2005), pp. 27-33.

[112] A.A. Soliman, and M.A. Abdou, Numerical Solutions of Nonlinear Evolution Equations Using Variational Iteration Method, *Computational and Applied Mathematics*, 207(2007), pp. 111-120.

[113] S.A. Manaa, F.H. Easif, and A.S. Faris, The Finite Difference Methods for FitzHugh-Nagumo Equation, *IOSR Journal of Mathematics*, 11(2015), pp. 51-55.

[114] M.Y. Hussain, and T.A. Zang, Spectral Methods in Fluid Dynamics, *Annual Review of Fluid Mechanics*, 19(1987), pp. 339-367.

[115] M. Javidi, Spectral Collocation Method for the Solution of the Generalized Burgers-Fisher equation, *Applied Mathematics and Computation*, 174(2006), pp. 345-352.

[116] A. Golbabai, and M. Javidi, A Spectral Domain Decomposition Approach for the Generalized Burgers-Fisher equation, *Chaos, Solitons & Fractals*, 39(2009), pp. 385-392.

[117] E. Babolian, and J. Saeidian, Analytic Approximate Solutions to Burger's, Fisher's, Huxley Equations and Two Combined Forms of these Equations, *Communications in Nonlinear Science and Numerical Simulation*, 14(2009), pp. 1984-1992.

[118] D. Kaya, and S.M. El-Sayed, A Numerical Simulation and Explicit Solutions of the Generalized Burgers-Fisher Equation, *Applied Mathematics and Computation*, 152(2004), pp. 403-413.

[119] R.E. Mickens, and A.B. Gumel, Construction and Analysis of a Non-standard Finite Difference Scheme for the Burgers-Fisher Equation, *Journal of Sound and Vibration*, 257(2002), pp. 791-797.

[120] T. Zhao, C. Li, Z. Zang, and Y. Wua, Chebyshev Legendre Pseudospectral Method for the Generalised Burgers-Fisher Equation, *Applied Mathematical Modelling*, 36(2012), pp. 1046-1056.

[121] P. Gordon, Nonsymmetric Difference Equations, *Journal of the Society for Industrial and Applied Mathematics*, 13(1965), pp. 667-673.

[122] A.R. Gourlay, "Hopscotch": A Fast Second Order Partial Differential Equation Solver, *Journal of the Institute of Mathematics and its Applications*, 6(1970), pp. 375-390.

[123] D. Olmos, and B.D. Shizgal, Pseudospectral Method of Solution of the Fitzhugh-Nagumo Equation, *Mathematics and Computers in Simulation*, 77(2009), pp. 2258-2278.

[124] R. Jiwari, R.K. Gupta, and V. Kumar, Polynomial Differential Quadrature Method for Numerical Solutions of the Generalized Fitzhugh-Nagumo Equation with Time-Dependent Coefficients, *Ain Shams Engineering Journal*, 5(2014), pp. 1343-1350.

[125] S. Hussain, M.A. Hossain, and M. Wilson, Natural Convection Flow From a Vertical Permeable Flat Plate With Variable Surface Temperature and Species Concentration, *Engineering Computations*, 17(2000), pp. 789-812.

[126] H.T. Lin, and W.S. Yu, Combined Heat and Mass Transfer by Laminar Natural Convection Flow From a Vertical Plate, *Heat & Mass Transfer*, 30(1995), pp. 369-376.

[127] E. Tadmor, Convergence of Spectral Methods for Conservative Laws, *SIAM, Journal of Numerical Analysis*, 26(1989), pp. 30-44.

[128] B. Costa, Spectral Methods for Partial Differential Equations, *CUBO: A Mathematical Journal*, 6(2004), pp. 1-32.

[129] R. Baltensperger, and J.P. Berrut, The Errors in Calculating the Pseudospectral Differentiation Matrices for Chebyshev Gauss-Lobatto points, *Computers and Mathematics with Applications*, 37(1999), pp. 41-48.

[130] S.S. Motsa, F.M. Samuel, and S. Shateyi (2016), Solving Nonlinear Parabolic Partial Differential Equations Using Multidomain Bivariate Spectral Collocation Method, *In D. Lee (Ed.), Nonlinear Systems - Design, Analysis, Estimation and Control*, InTech, DOI: 10.5772/4600.

[131] A.I. Fedoseyev, M.J. Friedman, and E.J. Kansa, Improved Multiquadric Method for Elliptic Partial Differential Equations Via PDE Collocation on the Boundary, *Computers & Mathematics with Applications*, 43(2002), pp. 439–455.

[132] A.D. Polyanin, and V.F. Zaitsev (2004), *Handbook of Nonlinear Partial Differential Equations*, London: Chapman & Hall/CRC.

[133] A. Tadmor, A Review of Numerical Methods for Nonlinear Partial Differential Equations, *Bulletin of the American Mathematical Society*, 49(2012), pp. 507-554.

[134] B. Carnahan, H.A. Luther, and J.O. Wilkes (1969), *Applied Numerical Methods*, New York: Wiley.

[135] S. Klaus (2010), *Numerical Methods for Nonlinear Elliptic Partial Differential Equation*, New York: Oxford University Press.

[136] T. Liszka, and J. Orkisz, The Finite Difference Method at Arbitrary Irregular Grids and its Application in Applied Mechanics, *Computer & Structures*, 11(1980), pp. 83-95.

[137] M. Avellaneda, T. Hou, and G. Papanicolaou, Finite Difference Approximations for Partial Differential Equations With Rapidly Oscillating Coefficients, *Mathematical Modelling and Numerical Analysis*, 25(1991), pp. 693–710.

[138] C. Conca, and S. Natesan, Numerical Methods for Elliptic Partial Differential Equations with Rapidly Oscillating Coefficients, *Computer Methods in Applied Mechanics and Engineering*, 192(2003). pp. 47–76.

[139] H.F. Chan, C.M. Fan, and C.W. Kuo, Generalized Finite Difference Method for Solving Two-Dimensional Nonlinear Obstacle Problems, *Engineering Analysis with Boundary Elements*, 37(2013), pp. 1189-1196.

[140] P.S. Jensen, Finite Difference Technique for Variable Grids, *Computers & Structures*, 2(1972), pp. 17-29.

[141] O. Lakkis, and T. Pryer, A Finite Element Method for Nonlinear Elliptic Problems, *SIAM Journal on Scientific Computing*, 35(2013), pp. 2025-2045.

[142] T. Hou, and X. Wu, A Multiscale Finite Element Method for Elliptic Problems in Composite Materials and Porous Media, *Computational Physics*, 134(1997), pp. 169–189.

[143] A. Matache, I. Babuška, and C. Schwab, Generalized p-FEM in Homogenization, *Numerical Mathematics*, 86(2000), pp. 319–375.

[144] B.K. Ghimire, H.Y. Tian, and A.R. Lamichhane, Numerical Solutions of Elliptic Partial Differential Equations Using Chebyshev Polynomials, *Computers & Mathematics with Applications*, 72(2016), pp. 1042-1054.

[145] L. Yi, and Z. Wang, Legendre Spectral Collocation Method for Second-order Nonlinear Ordinary/Partial Differential Equations, *Discrete & Continuous Dynamical Systems - Series B* , 19(2014), pp. 299-322.

[146] D. Gottlieb, and S.A. Orszag (1977), *Numerical Analysis of Spectral Methods,*: Theory and Applications, Philadelphia: Society for Industrial and Applied Mathematics.

[147] P. Demaret, and M.O. Deville, Chebyshev Collocation Solutions of the Navier-Stokes Equations using Multi-domain Decomposition and Finite Element Preconditioning, *Computational Physics*, 95(1991), pp. 359-386.

[148] A. Pinelli, A. Vacca, and A. Quarteroni, A Spectral Multi-domain Method for the Numerical Simulation of Turbulent Flows, *Computational Physics*, 136(1997), pp. 546-558.

[149] M.H. Chou, A Multigrid Pseudospectral Method for Steady Flow Computation, *Numerical Methods in Fluids*, 43(2003), pp. 25-42.

[150] W. Zhang, C.H. Zhang, and G. Xi, An Explicit Chebyshev Pseudospectral Multigrid Method for Incompressible Navier–Stokes Equations, *Computers & Fluids*, 39(2010), pp. 178-188.

[151] J.L. Vazquez, Domain of Existence and Blow-up for the Exponential Reaction-Diffusion Equation, *Indiana University Mathematics Journal*, 48(1999), pp. 677-709.

[152] E.A. Zabolotskaya, R.V. Khokhlov, Quasi-planes Waves in the Nonlinear Acoustic of Confined Beams, *Soviet physics Acoustics*, 15(1969), pp. 35-40.

[153] G.V.R. Reddy, S.M. Ibrahim, and V.S. Bhagavan, Similarity Transformations of Heat and Mass Transfer Effects on Steady MHD Free Convection Dissipative Fluid Flow Past an Inclined Porous Surface With Chemical Reaction, *Naval Architecture & Marine Engineering*, 11(2014), pp. 157-166.

[154] G. K. Boateng, A Practical Chebyshev Collocation Method for Differential Equations, *International Journal of Computer Mathematics*, 5(2010), pp. 59-79.

[155] M.R. Malik, T.A. Zang, and M.Y. Hussaini, A Spectral Collocation Method for the Navier-Stokes Equations, *Computational Physics*, 61(1985), pp. 64-88.

[156] A.D. Polyanin, and V.F. Zaitsev (2004), *Handbook of Nonlinear Partial Differential Equations*, Boca Raton: Chapman and Hall/CRC Press.

[157] S. Pamuk, and N. Pamuk, Solution of Two-Dimensional Heat and Mass Transfer Equation With Power-Law Temperature-Dependent Thermal Conductivity, *TWMS Journal of Applied and Engineering Mathematics*, 4(2014), pp. 199-208.

[158] J.M. Burgers, Application of a Model System to Illustrate Some Points to the Statistical Theory of Free Turbulance, *The Royal Netherlands Academy of Art and Sciences*, 43(1940), pp. 2-12.

[159] I. Prigogine, R. Lefever, Symmetry Breaking Instabilities in Dissipative Systems II, *Chemical Physics*, 48(1968), pp. 1665-1700.

[160] A.R. Bahadir, A Fully Implicit Finite-Difference Scheme for Two-Dimensional Burgers' Equations, *Applied Mathematics and Computation*, 137(2003), pp. 131-137.

[161] A.W. Teong, The Two Dimensional Reaction-diffusion Brusselator System: a Dual-Reciprocity Boundary Element Solution, *Engineering Analysis with Boundary Elements*, 27(2003), pp. 897-903.

[162] A.J. Chamkha, and A.M. Aly, Heat and Mass Transfer in Stagnation-Point Flow of a Polar Fluid Towards a Stretching Surface in Porous Media in the Presence of Soret, Dufour and Chemical Reaction Effects, *Chemical Engineering Communications*, 198(2010), pp. 214-234.

[163] C.A.J. Fletcher, Generating exact Solution of the Two-Dimensional Burger's Equation, *Numerical Methods in Fluids*, 3(1983), pp. 213-216.

[164] R. Abazari, and A. Borhanifar, Numerical Study of the Solution of the Burgers and Coupled Burgers Equations by a Differential Transformation Method, *Computers & Mathematics with Applications*, 59(2010), pp. 2711–2722.

[165] G. Zhao, X. Yub and R. Zhang, The New Numerical Method for Solving the System of Two-Dimensional Burgers' Equations, *Computers & Mathematics with Applications*, 62(2011), pp. 3279–3291.

[166] B. Šarler, R. Vertnik, and G. Kosec, Radial Basis Function Collocation Method for the Numerical Solution of the Two-Dimensional Transient Nonlinear Coupled Burgers' Equations, *Applied Mathematical Modelling*, 36(2012), pp. 1148–1160.

[167] E.H. Twizell, A.B. Gumel, and Q. Cao, A Second-Order Scheme for the Brusselator Reaction–Diffusion System, *Mathematical Chemistry*, 26(1999), pp. 297–316.

[168] F. Khani, F. Samadi, and S.H. Nezhad, New Exact Solutions of the Brusselator Reaction-Diffusion Model Using the Exp-Function Method, *Mathematical Problems in Engineering*, 2009, 9 pages, DOI:org/10.1155/2009/346461

[169] J. Biazar, and Z. Ayati, An Approximation to the Solution of the Brusselator System by Adomian decomposition Method and Comparing the Results with Runge-Kutta Method, *Contemporary Mathematical Sciences*, 2(2007), pp. 983-989.

Appendix A

A Published Journal Article on Trivariate Spectral Collocation Method

Applied Mathematics and Computation 360 (2019) 221–235

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

A highly accurate trivariate spectral collocation method of solution for two-dimensional nonlinear initial-boundary value problems

F.M. Samuel [a,c,*], S.S. Motsa [a,b]

[a] School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal, Private Bag X01, Scottsville 3209, South Africa
[b] Department of Mathematics, University of Swaziland, Private Bag 4, Kwaluseni, Swaziland
[c] Department of Mathematics, Taita Taveta University, Voi 635-80300, Kenya

ARTICLE INFO

MSC:
35K61
65M15
65M70

Keywords:
Trivariate lagrange interpolating polynomials
Spectral collocation
Two-dimensional PDEs
Time dependent
CGL points

ABSTRACT

In this paper, we propose a new numerical method namely, the trivariate spectral collocation method for solving two-dimensional nonlinear partial differential equations (PDEs) arising from unsteady processes. The problems considered are nonlinear PDEs defined on regular geometries. In the current solution approach, the quasi-linearization method is used to simplify the nonlinear PDEs. The solutions of the linearized PDEs are assumed to be trivariate Lagrange interpolating polynomials constructed using Chebyshev Gauss-Lobatto (CGL) points. A purely spectral collocation-based discretization is employed on the two space variables and the time variable to yield a system of linear algebraic equations that are solved by iteration. The numerical scheme is tested on four typical examples of nonlinear PDEs reported in the literature as a single equation or system of equations. Numerical results confirm that the proposed solution approach is highly accurate and computationally efficient when applied to solve two-dimensional initial-boundary value problems defined on small time intervals and hence it is a reliable alternative numerical method for solving this class of problems. The new error bound theorems and proofs on trivariate polynomial interpolation that we present support findings from the numerical simulations.

Contents

* Corresponding author at: School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal, Private Bag X01, Scottsville 3209, South Africa.
E-mail address: felexmutua@gmail.com (F.M. Samuel).

https://doi.org/10.1016/j.amc.2019.04.082
0096-3003/

1. Introduction

Spectral collocation-based methods, since their existence, have gained popularity in the numerical approximation of the solution of partial differential equations owing to their superior accuracy when applied to solve problems with smooth solutions [1]. They are particularly desirable for approximating solutions of nonlinear PDEs defined on regular geometries and they require a few number of grid points to achieve results with stringent accuracy [2]. Despite the benefits of the spectral collocation methods, review of the literature indicates that previous application of purely spectral collocation methods has focused on the solutions of ordinary differential equations and partial differential equations involving two independent variables [3]. There exists extensive literature in the studies by Zhao et al. [4] and Tadmor [5] where the spectral collocation methods have been applied successfully on these type of problems and highly accurate results achieved in a computationally efficient manner have been reported. In a few noticeable exceptions, for instance, in [6] where spectral collocation methods have been applied to obtain numerical solutions of two-dimensional time-dependent PDEs, such has been achieved through the application of spectral collocation discretization on the space variables and finite difference discretization on the time variable. It is well known that finite difference methods require many grid points to yield accurate results which can hardly match those obtained when spectral collocation methods are applied on PDEs defined on simple geometries particularly if the underlying solutions are smooth. Motivated by this fact, we propose a purely spectral collocation-based method for solving nonlinear two-dimensional time-dependent PDEs defined on small rectangular domains.

Typical examples of two-dimensional nonlinear time-dependent PDEs considered in this study include the nonlinear PDEs given as a single equation that describes the problem of unsteady two-dimensional heat and mass transfer, firstly, in quiescent media with chemical reaction [7], and secondly, with power-law temperature-dependent thermal conductivity [8]. The problems of heat and mass transfer phenomena are found throughout virtually all of the physical world and the industrial domain. Further, we consider nonlinear PDEs described as systems of equations a case of the coupled two-dimensional Burger's system [9] and the two-dimensional reaction-diffusion Brusselator system [10]. The Burger's equation is a fundamental PDEs from fluid mechanics. Burger's equation occurs in various areas of applied mathematics, such as modeling of hydrodynamics turbulence, shock waves theory, and traffic flow problems. It also describes the sedimentation of particles in fluid suspensions under the effect of gravity, transport, and dispersion of pollutants in rivers [11]. On the other hand, the Brusselator system arises in the mathematical modeling of chemical systems such as enzymatic reactions, and in plasma and laser physics in multiple coupling between certain modes [12]. Brusselator model is also evident in the formation of the ozone layer through a triple collision of oxygen atoms. The problems considered here have exact solutions and have been reported in the literature to be very useful in testing newly developed numerical methods of solution for nonlinear partial differential equations arising in modeling of various aspects of the real world. We, therefore, consider them appropriate to demonstrate the effectiveness of the current method of solution.

The exact solution to the problem of two-dimensional heat and mass transfer in quiescent media with chemical reaction was discussed by Polyanin and Zaitsev [7]. The problem has been solved numerically using an implicit finite-difference method in [13]. The problem of two-dimensional heat and mass transfer with power-law temperature-dependent thermal conductivity was examined by authors in [8] where they obtained a particular exact solution using the Adomian decomposition method and numerical methods of solution for this problem can be found in references given therein. Mathematical properties of Burger's equation were first investigated by Burger [9]. Analytical solution of unsteady two dimensional coupled Burger's equation was first given by Fletcher[14] using the Hopf-Cole transformation. In [15], the differential transformation method was applied to obtain the analytical solution of a coupled unsteady Burger's equation. Numerical solution of Burger's equation is a natural and first step towards developing methods for the computation of complex flows. Burger's equation has been used intensively to test new approaches in computational fluid dynamics by first implementing novel and new approaches to it. A numerical method based on local discontinuous Galerkin finite element was analyzed in [16] to solve two-dimensional Burger's equation. The local radial basis functions collocation method to approximate the numerical solution of the transient coupled Burgers equation was examined in [17]. In recent the decades, Brusellor model has been extensively studied both numerically and analytically. Twizell et al. [18] developed a second-order finite difference method for the numerical solution of the initial-boundary value problems of the Brusselator model. Khan et al. [19] found exact solutions of the Brusselator reaction-diffusion model using the Exp-function method. Biazar and Ayati [20] obtained an approximate solution to the Brusselator system by applying the Adomian decomposition method.

Another set of studies that has considered the use of spectral-based collocation methods to solve differential equations in the same family as those expounded in this paper include; the work by Abbasbandy and Shivanian [21] where they applied pseudo-spectral collocation method in erudite way to analytically calculate dual solutions considering a model of mixed convection in a porous media with boundary conditions on semi-infinite interval. Ellahi et al. [22] used a hybrid technique based on pseudo-spectral collocation to obtain the solution of a nonlinear system resulting from a problem of generalized Couette flow Eyring-Powell fluid. Later, authors in [23] conducted a nonlinear analysis of generalized Couette flow problem using spectral homotopy analysis method (SHAM). On the other hand, Shivanian et al. [24] developed a spectral meshless radial point interpolation (SMRP) technique for solving the fractional reaction-subdiffusion equation in one

and two-dimensional cases. The spectral meshless radial point interpolation (SMRP) technique was later applied in [25] to solve inverse source problems of the time-fractional diffusion equation in two dimensions and in [26] to solve fractional evolution equations of arbitrary fractional order in two-dimensions.

In this paper, a purely spectral collocation-based method, namely, the trivariate spectral collocation method is introduced and applied to solve two-dimensional initial-boundary value problems. The solution process involves, first, the simplification of the PDE using the quasi-linearization method. The solution to the linearized problem is assumed to be a trivariate Lagrange interpolating polynomials constructed on Chebyshev-Gauss-Lobatto points. The PDE is discretized in all space variables and time variable using spectral collocation method to yield a system of linear algebraic equations that are solved iteratively. The current numerical method is tested using typical examples of initial-boundary value problems reported in the literature. Comparison of numerical results with the exact solutions is presented and discussed in tabular and graphical form. Findings from numerical simulations show that the current method yields highly accurate results in a computationally efficient manner when applied to problems defined on a small time interval.

The rest of this paper is organized as follows. In Section 2, we describe the trivariate spectral collocation method of approximating the solution of two-dimensional nonlinear initial-boundary value problems described as single equations or systems of equations. In Section 3, the error bound theorems and proofs emanating from trivariate Lagrange interpolating polynomials constructed on Chebyshev-Gauss-Lobatto grid points emerge. In Section 4, we give four test examples where the numerical method is applied to demonstrate its applicability. Section 5 is devoted to results and discussion. In Section 6, we summarize the findings and point out the direction of future work.

2. The method of solution

In this section, we describe the algorithm for solving two-dimensional partial differential equations of initial-boundary value problems type. The present investigation focuses on the partial differential equations of the second order. For purposes of simplicity, this section is divided into 2 sections. In the first section, we construct numerical algorithms for solving nonlinear PDEs that are expressible as a single equation. In the second section, the idea is extended to systems of nonlinear PDEs.

2.1. Method of solution for a single nonlinear PDE

In this section, we describe the algorithm for solving nonlinear PDEs given as a single nonlinear equation. To illustrate the solution process we consider a nonlinear PDE that takes the form;

$$\frac{\partial u}{\partial t} = F\left(\frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, u\right), \quad (x, y) \in (a, b) \times (c, d), \ t \in (0, T), \tag{1}$$

where F is a nonlinear operator operating on the unknown function u and its first and second order spatial derivatives. Eq. (1) is solved subject to the boundary conditions

$$\begin{aligned} &\alpha_1^a \frac{\partial u}{\partial x}(a, y, t) + \alpha_0^a u(a, y, t) = f_a(y, t), \quad \alpha_1^b \frac{\partial u}{\partial x}(b, y, t) + \alpha_0^b u(b, y, t) = f_b(y, t), \\ &\beta_1^c \frac{\partial u}{\partial y}(x, c, t) + \beta_0^c u(x, c, t) = g_c(x, t), \quad \beta_1^d \frac{\partial u}{\partial y}(x, d, t) + \beta_0^d u(x, d, t) = g_d(x, t), \end{aligned} \tag{2}$$

where α_1^a, α_0^a, α_1^b, α_0^b, β_1^c, β_0^c, β_1^d, β_0^d are known constants and $f_a(y, t)$,$f_b(y, t)$,$g_c(x, t)$,$g_d(x, t)$, are known functions. The initial condition for this problem is given as

$$u(x, y, 0) = h(x, y). \tag{3}$$

The PDE (1) is first simplified using the quasi-linearization method (QLM) of Bellman and Kalaba [27]. The QLM is based on the Newton-Raphson method and is constructed from the linear terms of Taylor series expansion about an initial approximation to solution. The QLM assumes that the difference between solutions at two successive iterations, denoted by $u_{s+1} - u_s$ is very small. In particular, the QLM is comparable to the linear approximation of a function of several variables where the derivatives of different order and the previous approximation to solution assumes the role of independent variables and the functional value at the reference point, respectively. Finer details about linear approximation of functions can be found in any elementary book on differential calculus. Applying the QLM on Eq. (1) we obtain

$$\delta_{4,s}\frac{\partial^2 u_{s+1}}{\partial x^2} + \delta_{3,s}\frac{\partial^2 u_{s+1}}{\partial y^2} + \delta_{2,s}\frac{\partial u_{s+1}}{\partial x} + \delta_{1,s}\frac{\partial u_{s+1}}{\partial y} + \delta_{0,s}u_{s+1} - \dot{u}_{s+1} = R_s, \tag{4}$$

where

$$\begin{aligned} &\delta_{4,s} = \frac{\partial F}{\partial (u_{xx})_s}, \ \delta_{3,s} = \frac{\partial F}{\partial (u_{yy})_s}, \ \delta_{2,s} = \frac{\partial F}{\partial (u_x)_s}, \ \delta_{1,s} = \frac{\partial F}{\partial (u_y)_s}, \ \delta_{0,s} = \frac{\partial F}{\partial (u)_s}, \\ &R_s = \delta_{4,s}(u_{xx})_s + \delta_{3,s}(u_{yy})_s + \delta_{2,s}(u_x)_s + \delta_{1,s}(u_y)_s + \delta_{0,s}u_s - F_s. \end{aligned} \tag{5}$$

The subscripts in u_{xx} and u_x denotes second and the first partial derivatives of u with respect to x. Similarly, the first and the second derivatives with respect to y are denoted u_y and u_{yy}, respectively. The dot in $\dot{u}$ denotes derivative with respect to t and s signifies the previous iteration. Using an initial approximation to solution u_0, the QLM scheme Eq. (4) is solved iteratively until the solution converges. As a rule of thumb, a simple choice of u_0 is a polynomial that satisfies the given boundary conditions. The domain of approximation is discretized into Chebyshev Gauss-Lobatto nodes defined in [28] as

$$\begin{aligned}
\left\{\hat{x}_i\right\}_{i=0}^{N_x} &= \cos\left(\frac{i\pi}{N_x}\right),\ \hat{x}(x) = \frac{2}{b-a}\left[x - \frac{1}{2}(b+a)\right],\ x \in [a,b]\\
\left\{\hat{y}_j\right\}_{j=0}^{N_y} &= \cos\left(\frac{j\pi}{N_y}\right),\ \hat{y}(y) = \frac{2}{d-c}\left[y - \frac{1}{2}(d+c)\right],\ y \in [c,d],\\
\left\{\hat{t}_k\right\}_{k=0}^{N_t} &= \cos\left(\frac{k\pi}{N_t}\right),\ \hat{t}(t) = \frac{2}{T}\left[t - \frac{T}{2}\right],\ t \in [0,T].
\end{aligned} \tag{6}$$

Here, N_x, N_y, and N_t denotes the number of grid points in x, y, and t, respectively. In the solution process, the approximate solution of the PDE Eq. (1) is assumed to be the trivariate Lagrange interpolating polynomial that takes the form;

$$u(x,y,t) \approx U(x,y,t) = \sum_{p=0}^{N_x}\sum_{q=0}^{N_y}\sum_{r=0}^{N_t} U(x_p,y_q,t_r)L_p(x)L_q(y)L_r(t), \tag{7}$$

where the functions $L_p(x)$ are the Lagrange cardinal polynomial

$$L_p(x) = \prod_{\substack{p=0\\ p\neq i}}^{N_x} \frac{x-x_p}{x_i-x_p},\ \text{with}\, l_p(x_i) = \delta_{pi} = \begin{cases} 1 & \text{if } p = i,\\ 0 & \text{if } p \neq i. \end{cases}$$

The functions $L_q(y)$, $L_{r(t)}$ are defined in a similar manner [29]. The spatial differentiation matrix in x is approximated at the collocation nodes $(\hat{x}_i, \hat{y}_j, t_k)$, for $j = 0, 1, 2, \ldots, N_y$, and $k = 0, 1, 2, \ldots, N_t$, as follows;

$$\frac{\partial u}{\partial x}(\hat{x}_i,\hat{y}_j,\hat{t}_k) \approx \sum_{p=0}^{N_x}\sum_{q=0}^{N_y}\sum_{r=0}^{N_t} U(x_p,y_q,t_r)L_p'(\hat{x}_i)L_q(\hat{y}_j)L_r(\hat{t}_k) = \sum_{p=0}^{N_x} U(x_p,y_j,t_k)L_p'(\hat{x}_i) = \mathbf{D}\overset{j}{\mathbf{U}}_k = \left(\frac{2}{b-a}\right)\hat{\mathbf{D}}\overset{j}{\mathbf{U}}_k, \tag{8}$$

where $\hat{\mathbf{D}} = \left(\frac{b-a}{2}\right)\mathbf{D}$ is the standard first order Chebyshev differentiation matrix of size $(N_x+1)\times(N_x+1)$ as defined in [28]. We remark that the labels, i, j, and k in Eq. (8) are used to index the grid points in x, y, and t variables, respectively. The labels p, q, and r are used as indices for Lagrange cardinal functions in x, y, and t variables, respectively. We note that the factor $\frac{2}{b-a}$ comes from the application of the chain rule of differentiation due to the transformation used in Eq. (6). Simplification of Eq. (8) is based on the properties of Lagrange cardinal functions. The higher order differentiation matrices are obtained using matrix multiplication. The vector $\mathbf{U}_k^j$ is defined as

$$\overset{j}{\mathbf{U}}_k = \left[u(x_0,y_j,t_k), u(x_1,y_j,t_k), \ldots, u(x_{N_x},y_j,t_k)\right]^T,\ j = 0,1,\ldots,N_y,\ k = 0,1,\ldots,N_t, \tag{9}$$

where T denotes matrix transpose. Similarly, the spatial differentiation matrix in y is approximated at the collocation points $(\hat{x}_i, \hat{y}_j, \hat{t}_k)$, for $i = 0, 1, 2, \ldots, N_x$, and $k = 0, 1, 2, \ldots, N_t$, as

$$\frac{\partial u}{\partial y}(\hat{x}_i,\hat{y}_j,\hat{t}_k) \approx \sum_{q=0}^{N_y} U(x_i,y_q,t_k)L_q'(\hat{y}_j) = \sum_{q=0}^{N_y} \bar{D}_{j,q}\overset{q}{\mathbf{U}}_k = \sum_{q=0}^{N_y}\left(\frac{2}{d-c}\right)\hat{\bar{D}}_{j,q}\overset{q}{\mathbf{U}}_k, \tag{10}$$

where $\hat{\bar{D}}_{j,q} = \left(\frac{d-c}{2}\right)\bar{D}_{j,q},\ j,q = 0,1,2,\ldots,N_y$, are entries of a standard first order Chebyshev differentiation matrix of size $(N_y+1)\times(N_y+1)$. Higher order differentiation matrix with respect to y can be obtained using matrix multiplication. Finally, we approximate the differentiation matrix in t at the collocation points $(\hat{x}_i, \hat{y}_j, \hat{t}_k)$, for $i = 0, 1, 2, \ldots, N_x$, and $j = 0, 1, 2, \ldots, N_y$, as;

$$\frac{\partial u}{\partial t}(\hat{x}_i,\hat{y}_j,\hat{t}_k) \approx \sum_{r=0}^{N_t}\sum_{q=0}^{N_y} U(x_i,y_j,t_r)L_r'(\hat{t}_k) = \sum_{r=0}^{N_t}\sum_{q=0}^{N_y} \bar{\bar{D}}_{k,r}\overset{q}{\mathbf{U}}_r = \sum_{r=0}^{N_t}\sum_{q=0}^{N_y}\left(\frac{2}{T}\right)\hat{\bar{\bar{D}}}_{k,r}\overset{q}{\mathbf{U}}_r, \tag{11}$$

where $\hat{\bar{\bar{D}}}_{k,r} = \left(\frac{T}{2}\right)\bar{\bar{D}}_{k,r},\ k,r = 0,1,2,\ldots,N_t$, are entries of a standard first order Chebyshev differentiation matrix of size $(N_t+1)\times(N_t+1)$. It must be noted that the factors $\frac{2}{d-c}$ and $\frac{2}{T}$ in Eq. (10) and Eq. (11), respectively, are as a result of applying chain rule of differentiation due to the transformation used in Eq. (6) to map the respective computational domain to $[-1,1]$. We remark that the bar in $\bar{D}$ at Eq. (10) and double bar in $\bar{\bar{D}}$ at Eq. (11) distinguishes the differentiation matrix in y and t, respectively, from that in x. We note that in generating the sequence of vectors $\mathbf{U}_k^j$, $j = 0, 1, 2, \ldots, N_y$, $k = 0, 1, 2, \ldots, N_t$, the superscript j is varied of each subscript k. In general, the vector of unknowns $\mathbf{U}$ is built in such a manner that, for every

grid point in t, the grid points in y are varied, and for every grid point in y, the grid points in y, the grid points in x are varied. Indeed, this arrangement of grid points, explains the presence of single and double sum in the last expression at Eq. (10) and Eq. (11), respectively. Such a pattern will be useful when assembling the system of linear algebraic equations to obtain coefficient matrices.

Using Eqs. (8), (10) and (11) in the QLM scheme Eq. (4), we obtain a $(N_t+1)(N_y+1)(N_x+1)$ system of linear algebraic equations given by;

$$\left[\boldsymbol{\delta_{4,s}}\mathbf{D}^2+\boldsymbol{\delta_{2,s}}\mathbf{D}+\boldsymbol{\delta_{0,s}}\mathbf{I}\right]\overset{j}{\mathbf{U}}_k+\sum_{q=0}^{N_y}\left[\boldsymbol{\delta_{3,s}}\bar{D}^2_{j,q}+\boldsymbol{\delta_{1,s}}\bar{D}_{j,q}\right]\overset{q}{\mathbf{U}}_k-\sum_{r=0}^{N_t}\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,r}\overset{q}{\mathbf{U}}_r=\overset{j}{\mathbf{R}}_k, \tag{12}$$

for $j=0,1,2,\ldots,N_y$, $k=0,1,2,\ldots,N_t$ and an identity matrix $\mathbf{I}$ of size $(N_x+1)\times(N_x+1)$. The initial condition evaluated at the collocation points yields

$$u(x_i,y_j,0)=u(x_i,y_j,t_{N_t})=h(x_i,y_j)=\overset{j}{\mathbf{U}}_{N_t}. \tag{13}$$

Using the initial condition Eq. (13), we can reduce Eq. (12) to

$$\left[\boldsymbol{\delta_{4,s}}\mathbf{D}^2+\boldsymbol{\delta_{2,s}}\mathbf{D}+\boldsymbol{\delta_{0,s}}\mathbf{I}\right]\overset{j}{\mathbf{U}}_k+\sum_{q=0}^{N_y}\left[\boldsymbol{\delta_{3,s}}\bar{D}^2_{j,q}+\boldsymbol{\delta_{1,s}}\bar{D}_{j,q}\right]\overset{q}{\mathbf{U}}_k-\sum_{r=0}^{N_t-1}\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,r}\overset{q}{\mathbf{U}}_r=\overset{j}{\bar{\mathbf{R}}}_k, \tag{14}$$

where $j=0,1,2,\ldots,N_y$, $k=0,1,2,\ldots,N_t-1$ and

$$\overset{j}{\bar{\mathbf{R}}}_k=\overset{j}{\mathbf{R}}_k+\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,N_t}\overset{q}{\mathbf{U}}_{N_t},\ \ k=0,1,2,\ldots,N_t-1. \tag{15}$$

The linear system of equations in Eq. (14) can be expanded into $N_t(N_y+1)(N_x+1)\times N_t(N_y+1)(N_x+1)$ matrix system given by;

$$\begin{bmatrix}\mathbf{A}_{0,0} & \mathbf{A}_{0,1} & \mathbf{A}_{0,2} & \cdots & \mathbf{A}_{0,N_t-1}\\ \mathbf{A}_{1,0} & \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \cdots & \mathbf{A}_{1,N_t-1}\\ \vdots & \vdots & \vdots & \cdots & \vdots\\ \mathbf{A}_{N_t-1,0} & \mathbf{A}_{N_t-1,1} & \mathbf{A}_{N_t-1,2} & \cdots & \mathbf{A}_{N_t-1,N_t-1}\end{bmatrix}\begin{bmatrix}\overset{j}{\mathbf{U}}_0\\ \overset{j}{\mathbf{U}}_1\\ \vdots\\ \overset{j}{\mathbf{U}}_{N_t-1}\end{bmatrix}=\begin{bmatrix}\overset{j}{\mathbf{R}}_0\\ \overset{j}{\mathbf{R}}_1\\ \vdots\\ \overset{j}{\mathbf{R}}_{N_t-1}\end{bmatrix}. \tag{16}$$

The matrix system Eq. (16) can be written compactly as

$$\left[\mathbf{A}\right]\left[\mathbf{U}\right]=\left[\mathbf{R}\right], \tag{17}$$

where

$$\begin{aligned}&\mathbf{A}_{k,k}=\mathbf{B}-\bar{\bar{D}}_{k,k}\mathbf{I}_{xy},\ \ \mathbf{A}_{k,r}=-\bar{\bar{D}}_{k,r}\mathbf{I}_{xy},\ \ k\neq r,\ \ k,r=0,1,2,\ldots,N_t-1,\\ &\mathbf{B}_{i,i}=\boldsymbol{\delta_{4,s}}\mathbf{D}^2+\boldsymbol{\delta_{2,s}}\mathbf{D}+\boldsymbol{\delta_{0,s}}\mathbf{I}_x+\boldsymbol{\delta_{3,s}}\bar{D}^2_{i,i}\mathbf{I}_x+\boldsymbol{\delta_{1,s}}\bar{D}_{i,i}\mathbf{I}_x,\\ &\mathbf{B}_{i,j}=\boldsymbol{\delta_{3,s}}\bar{D}^2_{i,j}\mathbf{I}_x+\boldsymbol{\delta_{1,s}}\bar{D}_{i,j}\mathbf{I}_x,\quad i\neq j,\ \ i,j=0,1,2,\ldots,N_y,\\ &\mathbf{U}=\left[u(x_0,y_0,t_0),\ldots,u(x_{N_x},y_0,t_0),u(x_0,y_1,t_0),\ldots,u(x_{N_x},y_{N_y},t_0),\ldots,u(x_{N_x},y_{N_y},t_{N_t})\right]^T,\end{aligned} \tag{18}$$

and $\mathbf{R}$ is the right hand side corresponding to the unknown vector $\mathbf{U}$. The quantities $\mathbf{I}_{xy}$ and $\mathbf{I}_x$, are identity matrices of size $(N_y+1)(N_x+1)\times(N_y+1)(N_x+1)$, and $(N_x+1)\times(N_x+1)$, respectively. We remark that subscripts in Eq. (18) have been used in dummy sense to denote the rows and columns of sub-matrices ib Eq. (16) and do not preserve their usual meaning of indexing the grid points. The boundary conditions Eq. (2) are evaluated at the collocation nodes as;

$$\begin{aligned}&\alpha_1^a\sum_{p=0}^{N_x}\mathbf{D}_{N_x,p}u(x_p,y_j,t_k)+\alpha_1^a u(x_{N_x},y_j,t_k)=f_a(y_j,t_k),\\ &\alpha_1^b\sum_{p=0}^{N_x}\mathbf{D}_{0,p}u(x_p,y_j,t_k)+\alpha_0^b u(x_0,y_j,t_k)=f_b(y_j,t_k),\ \ j=0,1,2,\ldots,N_y,\ \ k=0,1,2,\ldots,N_t,\end{aligned} \tag{19}$$

and

$$\beta_1^c \sum_{q=0}^{N_y} \bar{D}_{N_y,q} u(x_i, y_q, t_k) + \beta_0^c u(x_i, y_{N_y}, t_k) = g_c(x_i, t_k),$$
$$\beta_1^d \sum_{q=0}^{N_y} \bar{D}_{0,q} u(x_i, y_q, t_k) + \beta_0^d u(x_i, y_0, t_k) = g_d(x_i, t_k),\ i = 0, 1, 2, \ldots, N_x,\ k = 0, 1, 2, \ldots, N_t. \tag{20}$$

The boundary conditions are imposed on the main diagonal sub-blocks of matrices in Eq. (16) to yield a new consistent system of linear algebraic equations with a unique solution.

2.2. Method of solution for systems of nonlinear PDEs

In this section, we extend the algorithm described in the previous section to the solution of systems of nonlinear PDEs. To demonstrate the construction of the algorithm, we consider a system of two nonlinear PDEs that is expressible in the form;

$$\frac{\partial u}{\partial t} = F_1\left(\frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 v}{\partial x^2}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial^2 v}{\partial y^2}, \frac{\partial u}{\partial x}, \frac{\partial v}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial y}, u, v\right),$$
$$\frac{\partial v}{\partial t} = F_2\left(\frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 v}{\partial x^2}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial^2 v}{\partial y^2}, \frac{\partial u}{\partial x}, \frac{\partial v}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial y}, u, v\right), \tag{21}$$

where F_1 and F_2 are nonlinear operators acting on the unknown functions u and v and their derivatives with respect to x and y as illustrated. The system Eq. (21) is solved subject to boundary conditions

$$\alpha_1^a \frac{\partial u}{\partial x}(a, y, t) + \alpha_0^a u(a, y, t) = f_a(y, t),\ \alpha_1^b \frac{\partial u}{\partial x}(b, y, t) + \alpha_0^b u(b, y, t) = f_b(y, t),$$
$$\beta_1^c \frac{\partial u}{\partial y}(x, c, t) + \beta_0^c u(x, c, t) = g_c(x, t),\ \beta_1^d \frac{\partial u}{\partial y}(x, d, t) + \beta_0^d u(x, d, t) = g_d(x, t), \tag{22}$$

and

$$\alpha_1^{a*} \frac{\partial v}{\partial x}(a, y, t) + \alpha_0^{a*} v(a, y, t) = f_{a*}(y, t),\ \alpha_1^{b*} \frac{\partial v}{\partial x}(b, y, t) + \alpha_0^{b*} v(b, y, t) = f_{b*}(y, t),$$
$$\beta_1^{c*} \frac{\partial v}{\partial y}(x, c, t) + \beta_0^{c*} v(x, c, t) = g_{c*}(x, t),\ \beta_1^{d*} \frac{\partial v}{\partial y}(x, d, t) + \beta_0^{d*} v(x, d, t) = g_{d*}(x, t). \tag{23}$$

The star * in Eq. (23) distinguishes the boundary conditions in v from those of u. Applying the QLM, Eq. (21) can be expressed as the linearized coupled system

$$\delta_{4,s}^1 \frac{\partial^2 u_{s+1}}{\partial x^2} + \delta_{3,s}^1 \frac{\partial^2 u_{s+1}}{\partial y^2} + \delta_{2,s}^1 \frac{\partial u_{s+1}}{\partial x} + \delta_{1,s}^1 \frac{\partial u_{s+1}}{\partial y} + \delta_{0,s}^1 u_{s+1}$$
$$- \dot{u}_{s+1} + \sigma_{4,s}^1 \frac{\partial^2 v_{s+1}}{\partial x^2} + \sigma_{3,s}^1 \frac{\partial^2 v_{s+1}}{\partial y^2} + \sigma_{2,s}^1 \frac{\partial v_{s+1}}{\partial x} + \sigma_{1,s}^1 \frac{\partial v_{s+1}}{\partial y} + \sigma_{0,s}^1 v_{s+1} = R_{1,s}, \tag{24}$$

$$\delta_{4,s}^2 \frac{\partial^2 u_{s+1}}{\partial x^2} + \delta_{3,s}^2 \frac{\partial^2 u_{s+1}}{\partial y^2} + \delta_{2,s}^2 \frac{\partial u_{s+1}}{\partial x} + \delta_{1,s}^2 \frac{\partial u_{s+1}}{\partial y} + \delta_{0,s}^2 u_{s+1}$$
$$+ \sigma_{4,s}^2 \frac{\partial^2 v_{s+1}}{\partial x^2} + \sigma_{3,s}^2 \frac{\partial^2 v_{s+1}}{\partial y^2} + \sigma_{2,s}^2 \frac{\partial v_{s+1}}{\partial x} + \sigma_{1,s}^2 \frac{\partial v_{s+1}}{\partial y} + \sigma_{0,s}^2 v_{s+1} - \dot{v}_{s+1} = R_{2,s}, \tag{25}$$

where

$$\delta_{4,s}^\nu = \frac{\partial F_\nu}{\partial (u_{xx})_s},\ \delta_{3,s}^\nu = \frac{\partial F_\nu}{\partial (u_{yy})_s},\ \delta_{2,s}^\nu = \frac{\partial F_\nu}{\partial (u_x)_s},\ \delta_{1,s}^\nu = \frac{\partial F_\nu}{\partial (u_y)_s},\ \delta_{0,s}^\nu = \frac{\partial F_\nu}{\partial (u)_s},$$
$$\sigma_{4,s}^\nu = \frac{\partial F_\nu}{\partial (u_{xx})_s},\ \sigma_{3,s}^\nu = \frac{\partial F_\nu}{\partial (u_{yy})_s},\ \sigma_{2,s}^\nu = \frac{\partial F_\nu}{\partial (u_x)_s},\ \sigma_{1,s}^\nu = \frac{\partial F_\nu}{\partial (u_y)_s},\ \sigma_{0,s}^\nu = \frac{\partial F_\nu}{\partial (u)_s},\ \nu = 1, 2, \tag{26}$$

$$R_{\nu,s} = \delta_{4,s}^\nu (u_{xx})_s + \delta_{3,s}^\nu (u_{yy})_s + \delta_{2,s}^\nu (u_x)_s + \delta_{1,s}^\nu (u_y)_s + \delta_{0,s}^\nu u_s$$
$$+ \sigma_{4,s}^\nu (v_{xx})_s + \sigma_{3,s}^\nu (v_{yy})_s + \sigma_{2,s}^\nu (v_x)_s + \sigma_{1,s}^\nu (v_y)_s + \sigma_{0,s}^\nu v_s - F_{\nu s},\ \nu = 1, 2. \tag{27}$$

Application of spectral collocation method to the linearized QLM scheme Eqs. (24) and (25) and in account of the effect of initial condition yields a $2N_t(N_y + 1)(N_x + 1)$ system of linear algebraic equations given by;

$$\left[\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{4},s}\mathbf{D}^2+\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{2},s}\mathbf{D}+\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{0},s}\mathbf{I}\right]\overset{j}{\mathbf{U}}_k+\sum_{q=0}^{N_y}\left[\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{3},s}\bar{D}^2_{j,q}+\boldsymbol{\delta}^{\mathbf{1}}_{\mathbf{1},s}\bar{D}_{j,q}\right]\overset{q}{\mathbf{U}}_k-\sum_{r=0}^{N_t-1}\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,r}\overset{q}{\mathbf{U}}_r$$
$$+\left[\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{4},s}\mathbf{D}^2+\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{2},s}\mathbf{D}+\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{0},s}\mathbf{I}\right]\overset{j}{\mathbf{V}}_k+\sum_{q=0}^{N_y}\left[\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{3},s}\bar{D}^2_{j,q}+\boldsymbol{\sigma}^{\mathbf{1}}_{\mathbf{1},s}\bar{D}_{j,q}\right]\overset{q}{\mathbf{V}}_k=\overset{j}{\bar{\mathbf{R}}}_{1,k}, \tag{28}$$

$$\left[\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{4},s}\mathbf{D}^2+\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{2},s}\mathbf{D}+\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{0},s}\mathbf{I}\right]\overset{j}{\mathbf{U}}_k+\sum_{q=0}^{N_y}\left[\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{3},s}\bar{D}^2_{j,q}+\boldsymbol{\delta}^{\mathbf{2}}_{\mathbf{1},s}\bar{D}_{j,q}\right]\overset{q}{\mathbf{U}}_k+\left[\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{4},s}\mathbf{D}^2+\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{2},s}\mathbf{D}+\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{0},s}\mathbf{I}\right]\overset{j}{\mathbf{V}}_k$$
$$+\sum_{q=0}^{N_y}\left[\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{3},s}\bar{D}^2_{j,q}+\boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{1},s}\bar{D}_{j,q}\right]\overset{q}{\mathbf{V}}_k-\sum_{r=0}^{N_t-1}\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,r}\overset{q}{\mathbf{V}}_r=\overset{j}{\bar{\mathbf{R}}}_{2,k}, \tag{29}$$

Eqs. (28) and (29) can be expressed into $2N_t(N_y+1)(N_x+1)\times 2N_t(N_y+1)(N_x+1)$ matrix system given by

$$\begin{bmatrix}\mathbf{A}_{1,1} & \mathbf{A}_{1,2}\\ \mathbf{A}_{2,1} & \mathbf{A}_{2,2}\end{bmatrix}\begin{bmatrix}\mathbf{U}\\ \mathbf{V}\end{bmatrix}=\begin{bmatrix}\mathbf{R}_1\\ \mathbf{R}_2\end{bmatrix} \tag{30}$$

where

$$\mathbf{R}_1=\mathbf{R}_{1,s}+\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,N_t}\overset{q}{\mathbf{U}}_{N_t},\quad \mathbf{R}_2=\mathbf{R}_{2,s}+\sum_{q=0}^{N_y}\bar{\bar{D}}_{k,N_t}\overset{q}{\mathbf{V}}_{N_t},\quad k=0,1,2,\ldots,N_t-1. \tag{31}$$

The vector $\mathbf{V}$ is defined in similar manner to vector $\mathbf{U}$ given in Eq. (18). The right hand side vectors $\mathbf{R}_1$, and $\mathbf{R}_2$ correspond to vectors $\mathbf{U}$ and $\mathbf{V}$, respectively. The boundary conditions, Eqs. (22) and (23) are evaluated at the collocation points in a manner similar to that illustrated at Eqs. (19) and (20). We remark that if the initial conditions $u(x,\,y,\,0)$ and $v(x,y,0)$ are not available, the system is first solved for $t=0$ to approximate the initial conditions. In summary, we assert that the algorithms described so far can be easily adjusted and applied to PDEs of higher orders (>2) and to problems modeled by larger systems of nonlinear PDEs. An extension to numerical schemes of solutions of two-dimensional time-dependent PDEs exhibiting slightly different forms of nonlinearity can be achieved in a straightforward manner.

3. Error bounds theorems in a trivariate polynomial interpolation

In this section, we present new error bound theorems that govern polynomial interpolation error in a trivariate Lagrange interpolating polynomial constructed using Chebyshev Gauss-Lobatto nodes. Fundamental ingredients of this subject related to the construction of proofs of the theorems include; first, the understanding that CGL nodes are the relative extremes of the N_x-th degree Chebyshev polynomial of the first kind $T_{N_x}(\hat{x})=\cos[N_x\arccos(\hat{x})]$, $\hat{x}\in[-1,1]$, secondly, the general properties of Chebyshev polynomials, and thirdly, the understanding of the mean value theorem in calculus. Although to the best of our knowledge, there does not exist a well-known family of polynomials whose roots are the CGL nodes, it is easy to discern that the interior CGL nodes are roots of $T'_{N_x}(\hat{x})=0$. This fact leads to the discovery of a complete set of the CGL nodes as the roots of the N_x+1th degree polynomial given by;

$$L_{N_x+1}(\hat{x})=(1-\hat{x}^2)T'_{N_x}(\hat{x}). \tag{32}$$

Below, is the theorem that benchmarks formulation of the error bound theorems on trivariate polynomial interpolation;

Theorem 1. *[32] Let $u(x,y,t)\in C^{N_x+N_y+N_t+3}([a,b]\times[c,d]\times[0,T])$ be sufficiently smooth such that at least the (N_x+1)th partial derivative with respect to x, (N_y+1)th partial derivative with respect to y, (N_t+1)th partial derivative with respect to t, and the $(N_x+N_y+N_t+3)$th mixed partial derivative with respect to x, y, and t exists and are all continuous, then there exists values $\xi_x,\xi'_x\in(a,b)$, $\xi_y,\xi'_y\in(c,d)$, and $\xi_t,\xi'_t\in(0,T)$, such that*

$$u(x,y,t)-U(x,y,t)=\frac{\partial^{N_x+1}u(\xi_x,y,t)}{\partial x^{N_x+1}(N_x+1)!}\prod_{i=0}^{N_x}(x-x_i)+\frac{\partial^{N_y+1}u(x,\xi_y,t)}{\partial y^{N_y+1}(N_y+1)!}\prod_{j=0}^{N_y}(y-y_j)+\frac{\partial^{N_t+1}u(x,y,\xi_t)}{\partial t^{N_t+1}(N_t+1)!}\prod_{k=0}^{N_t}(t-t_k)$$
$$-\frac{\partial^{N_x+N_y+N_t+3}u(\xi'_x,\xi'_y,\xi'_t)}{\partial x^{N_x+1}\partial y^{N_y+1}\partial t^{N_t+1}(N_x+1)!(N_y+1)!(N_t+1)!}\prod_{i=0}^{N_x}(x-x_i)\prod_{j=0}^{N_y}(y-y_j)\prod_{k=0}^{N_t}(t-t_k), \tag{33}$$

where $U(x,\,y,\,t)$ is a trivariate interpolating polynomial of $u(x,\,y,\,t)$ at $\{x_i\}_{i=0}^{N_x}$ grid points in x-variable, $\{y_j\}_{j=0}^{N_y}$ grid points in y-variable, and $\{t_k\}_{k=0}^{N_t}$ grid points in t-variable.

The remainder formula Eq. (33) is based on the mean value theorem and is derived recursively from tshe corresponding univariate error formula given in [31] for a sufficiently smooth function $u(x,\,y,\,t)$. Taking the absolute value of Eq. (33) we

obtain

$$|u(x,y,t)-U(x,y,t)| \le \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_x+1}u(\xi_x,y,t)}{\partial x^{N_x+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x-x_i)\right|}{(N_x+1)!} + \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_y+1}u(x,\xi_y,t)}{\partial y^{N_y+1}}\right|\frac{\left|\prod_{j=0}^{N_y}(y-y_j)\right|}{(N_y+1)!} + \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x,y,\xi_t)}{\partial t^{N_t+1}}\right|\frac{\left|\prod_{k=0}^{N_t}(t-t_k)\right|}{(N_t+1)!} + \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_x+N_y+N_t+3}u(\xi_x',\xi_y',\xi_t')}{\partial x^{N_x+1}\partial y^{N_y+1}\partial t^{N_t+1}}\right|\frac{\left|\prod_{i=0}^{N_x}(x-x_i)\right|\left|\prod_{j=0}^{N_y}(y-y_j)\right|\left|\prod_{k=0}^{N_t}(t-t_k)\right|}{(N_x+1)!(N_y+1)!(N_t+1)!}, \tag{34}$$

where $\Omega=[a,b]\times[c,d]\times[0,T]$. Since the function $u(x,y,t)$ is assumed to be smooth on the interval of approximation, it follows that its derivatives are bounded and thus $\exists$ constants C_1, C_2, C_3 and C_4, such that

$$\max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_x+1}u(x,y,t)}{\partial x^{N_x+1}}\right|\le C_1,\quad \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_y+1}u(x,y,t)}{\partial y^{N_y+1}}\right|\le C_2,$$
$$\max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_t+1}u(x,y,t)}{\partial t^{N_t+1}}\right|\le C_3,\quad \max_{(x,y,t)\in\Omega}\left|\frac{\partial^{N_x+N_y+N_t+3}u(x,y,t)}{\partial x^{N_x+1}\partial y^{N_y+1}\partial t^{N_t+1}}\right|\le C_4. \tag{35}$$

The error bound for trivariate polynomial interpolation using Chebyshev Gauss-Lobatto nodes on a single domain is governed by the theorem below;

Theorem 2 (The error bound in a single domain)**.** *The resulting error bound when CGL grid points* $\{x_i\}_{i=0}^{N_x}\in[a,b]$, *in x-variable,* $\{y_j\}_{j=0}^{N_y}\in[c,d]$ *in y-variable, and* $\{t_k\}_{k=0}^{N_t}\in[0,T]$, *in t-variable are used in trivariate polynomial interpolation is given by*

$$E(x,y,t)\le C_1\frac{8\left(\frac{b-a}{4}\right)^{N_x+1}}{(N_x+1)!}+C_2\frac{8\left(\frac{d-c}{4}\right)^{N_y+1}}{(N_y+1)!}+C_3\frac{8\left(\frac{T}{4}\right)^{N_t+1}}{(N_t+1)!}+C_4\frac{8^3\left(\frac{b-a}{4}\right)^{N_x+1}\left(\frac{d-c}{4}\right)^{N_y+1}\left(\frac{T}{4}\right)^{N_t+1}}{(N_x+1)!(N_y+1)!(N_t+1)!}. \tag{36}$$

Proof. First, using the relation stated in [32] we express Eq. (32) as

$$L_{N_x+1}(\hat{x})=(1-\hat{x}^2)T'_{N_x}(\hat{x})=-N_x\hat{x}T_{N_x}(\hat{x})+N_xT_{N_x-1}(\hat{x}). \tag{37}$$

Using the triangle inequality and noting that $|T_{N_x}(\hat{x})|\le 1$, $\forall\,\hat{x}\in[-1,1]$, we have

$$|L_{N_x+1}(\hat{x})|=|-N_x\hat{x}T_{N_x}(\hat{x})+N_xT_{N_x-1}(\hat{x})|\le|-N_x\hat{x}T_{N_x}(\hat{x})|+|N_xT_{N_x-1}(\hat{x})|\le 2N_x. \tag{38}$$

The leading coefficient of $L_{N_x+1}(\hat{x})$ is $2^{N_x-1}N_x$, where the components 2^{N_x-1} and N_x comes from the leading coefficient of $T_{N_x}(\hat{x})$ and the application of N_xth rule of differentiation on $T_{N_x}(\hat{x})$, respectively. The product factor in the first term of the error bound expression given at Eq. (34) can therefore be taken as the factorized form of monic polynomial $\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}$. We write,

$$\prod_{i=0}^{N_x}(\hat{x}-\hat{x}_i)=\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x},\ \ \hat{x}\in[-1,1]. \tag{39}$$

Using Eq. (38), it is easy to establish that the monic polynomial Eq. (39) is bounded by

$$\left|\prod_{j=0}^{N_x}(x-\hat{x}_i)\right|=\left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right|\le\frac{2N_x}{2^{N_x-1}N_x}=\frac{4}{2^{N_x}}. \tag{40}$$

Considering a general interval $x\in[a,b]$, we can show that the first product factor in Eq. (34) is bounded by

$$\max_{a\le x\le b}\left|\prod_{i=0}^{N_x}(x-x_i)\right|=\max_{-1\le\hat{x}\le1}\left|\prod_{i=0}^{N_x}\frac{(b-a)}{2}(\hat{x}-\hat{x}_i)\right|=\left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\le\hat{x}\le1}\left|\prod_{i=0}^{N_x}(\hat{x}-\hat{x}_i)\right|$$
$$=\left(\frac{b-a}{2}\right)^{N_x+1}\max_{-1\le\hat{x}\le1}\left|\frac{L_{N_x+1}(\hat{x})}{2^{N_x-1}N_x}\right|\le\frac{4\left(\frac{b-a}{2}\right)^{N_x+1}}{2^{N_x}}=8\left(\frac{b-a}{4}\right)^{N_x+1}. \tag{41}$$

Similarly, we conclude that the second and the third product factor are bounded, respectively, by;

$$\max_{c\le y\le d}\left|\prod_{j=0}^{N_y}(y-y_j)\right|=\left(\tfrac{d-c}{2}\right)^{N_y+1}\max_{-1\le\hat{y}\le1}\left|\frac{L_{N_y+1}(\hat{y})}{2^{N_y-1}N_y}\right|\le\frac{4\left(\frac{d-c}{2}\right)^{N_y+1}}{2^{N_y}}=8\left(\tfrac{d-c}{4}\right)^{N_y+1}, \tag{42}$$

and

$$\max_{0\le t\le T}\left|\prod_{k=0}^{N_t}(t-t_k)\right| = \left(\frac{T}{2}\right)^{N_t+1}\max_{-1\le\hat{t}\le 1}\left|\frac{L_{N_t+1}(\hat{t})}{2^{N_t-1}N_t}\right| \le \frac{4\left(\frac{T}{2}\right)^{N_t+1}}{2^{N_t}} = 8\left(\frac{T}{4}\right)^{N_t+1}. \tag{43}$$

Using Eqs. (41)–(43), and Eq. (35) in Eq. (34) the proof is completed. □

We remark that the error bound for Chebyshev Gauss-Lobatto nodes is slightly larger than that of the optimal Chebyshev nodes [35]. Precisely, if we compare the results of the current theorems on error bounds with those extensions of error bound theorem presented in [31], we notice that error in interpolation errors due to Chebyshev Gauss-Lobatto notes is almost twice that of optimal Chebyshev nodes. However, Chebyshev Gauss-Lobatto nodes have been preferred as a choice of discretization nodes to Chebyshev nodes since Chebyshev Gauss-Lobatto points include the boundary point which is convenient in the treatment of the boundary conditions of the problem been solved.

4. Numerical experiment

In this section, the method described in the previous section is applied to a selected class of two-dimensional nonlinear initial-boundary value problems. The accuracy and efficiency of the proposed method are demonstrated by comparing the numerical results against the exact solution.

Example 1. Consider the problem of two-dimensional heat and mass transfer in a quiescent media with chemical reaction given in its general form by

$$\frac{\partial u}{\partial t} = a\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)+f(u), \tag{44}$$

where a is a real constant. We analyze special case of this problem where $a=1$ and $f(u)=-4u^3-2u^2$ and solve the problem subject to the boundary conditions

$$\begin{aligned} &u(2,y,t)=(2+y+2t)^{-1},\ \ u(4,y,t)=(4+y+2t)^{-1},\ \ 2\le y\le 3,\ \ t>0,\\ &u(x,2,t)=(2+x+2t)^{-1},\ \ u(x,3,t)=(3+x+2t)^{-1},\ \ 2\le x\le 4,\ \ t>0. \end{aligned} \tag{45}$$

The initial condition for this problem is given by

$$u(x,y,0)=(x+y)^{-1},\ \ 2\le x\le 4,\ \ 2\le y\le 3. \tag{46}$$

The exact solution is given in [7] as

$$u(x,y,t)=(x+y+2t)^{-1}. \tag{47}$$

Example 2. We consider the two-dimensional heat and mass transfer equation with power-law temperature-dependent thermal conductivity [8]

$$\frac{\partial u}{\partial t} = a\left[\frac{\partial}{\partial x}\left(u^n\frac{\partial u}{\partial x}\right)\right]+b\left[\frac{\partial}{\partial y}\left(u^m\frac{\partial u}{\partial y}\right)\right]+\beta u, \tag{48}$$

where n,m can be an integer or fraction and a, b, and β are some parameters. For simplicity, we consider a special case of Eq. (48) when $a=1$, $b=1$, $m=n=1$, and $\beta=0$ in which Eq. (48) reduces to Boussinesq equation

$$\frac{\partial u}{\partial t} = \left[\frac{\partial}{\partial x}\left(u\frac{\partial u}{\partial x}\right)\right]+\left[\frac{\partial}{\partial y}\left(u\frac{\partial u}{\partial y}\right)\right]. \tag{49}$$

The Boussinesq equation arises in nonlinear heat conduction and the theory of unsteady flow through porous media with a free surface. Eq. (49) is solved subject to boundary conditions

$$\begin{aligned} &u(0,y,t)=y+2t,\ \ u(5,y,t)=5+y+2t,\ \ 0\le y\le 2,\ \ t>0,\\ &u(x,0,t)=x+2t,\ \ u(x,2,t)=2+x+2t,\ \ 0\le x\le 5,\ \ t>0. \end{aligned} \tag{50}$$

The initial condition for this problem is given by

$$u(x,y,0)=x+y. \tag{51}$$

The exact solution is given by authors in [7] as

$$u(x,y,t)=x+y+2t. \tag{52}$$

Example 3. We consider the system two-dimensional Burgers' equations given by

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=\frac{1}{R}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right),$$

$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}=\frac{1}{R}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right), \tag{53}$$

where $u(x, y, t)$ and $v(x, y, t)$ are velocity components to be determined and R is the Reynolds number. Eq. (53) is subject to boundary conditions

$$\begin{aligned} u(0,y,t)&=\frac{3}{4}-\frac{1}{4\left[1+e^{(R(4y-t)/32)}\right]},\quad u(2,y,t)=\frac{3}{4}-\frac{1}{4\left[1+e^{(R(4y-8-t)/32)}\right]},\\ u(x,0,t)&=\frac{3}{4}-\frac{1}{4\left[1+e^{(R(-4x-t)/32)}\right]},\quad u(x,2,t)=\frac{3}{4}-\frac{1}{4\left[1+e^{(R(8-4x-t)/32)}\right]}, \end{aligned} \tag{54}$$

$$\begin{aligned} v(0,y,t)&=\frac{3}{4}+\frac{1}{4\left[1+e^{(R(4y-t)/32)}\right]},\quad v(2,y,t)=\frac{3}{4}+\frac{1}{4\left[1+e^{(R(4y-8-t)/32)}\right]},\\ v(x,0,t)&=\frac{3}{4}+\frac{1}{4\left[1+e^{(R(-4x-t)/32)}\right]},\quad v(x,2,t)=\frac{3}{4}+\frac{1}{4\left[1+e^{(R(8-4x-t)/32)}\right]}. \end{aligned} \tag{55}$$

The initial condition are

$$u(x,y,0)=\frac{3}{4}-\frac{1}{4\left[1+e^{(R(y-x)/8)}\right]},\quad v(x,y,0)=\frac{3}{4}+\frac{1}{4\left[1+e^{(R(y-x)/8)}\right]}. \tag{56}$$

The exact solutions for this problem are given in [33] as

$$u(x,y,t)=\frac{3}{4}-\frac{1}{4\left[1+e^{(R(4y-4x-t)/32)}\right]},\quad v(x,y,t)=\frac{3}{4}+\frac{1}{4\left[1+e^{(R(4y-4x-t)/32)}\right]}. \tag{57}$$

Example 4. We consider the two-dimensional Brusselator system given by

$$\begin{aligned} \frac{\partial u}{\partial t}&=B+u^2v-(A+1)u+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right),\\ \frac{\partial v}{\partial t}&=Au-u^2v+\mu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right), \end{aligned} \tag{58}$$

where $u(x, y, t)$ and $v(x, y, t)$ represent dimensionless concentration of two reactants, A and B are constant concentrations of the two reactants, and μ is the diffusion coefficient. The exact solutions for this problem are given in [34] for $A = 1$, $B = 0$, $\mu = \frac{1}{4}$ as

$$u(x,y,t)=e^{-x-y-t/2},\quad v(x,y,t)=e^{x+y+t/2}. \tag{59}$$

Eq. (58) is solved subject to boundary conditions

$$u(0,y,t)=e^{-y-t/2},\quad u(2,y,t)=e^{-2-y-t/2},\quad u(x,0,t)=e^{-x-t/2},\quad u(x,2,t)=e^{-2-x-t/2}, \tag{60}$$

$$v(0,y,t)=e^{y+t/2},\quad v(2,y,t)=e^{2+y+t/2},\quad v(x,0,t)=e^{x+t/2},\quad v(x,2,t)=e^{2+x+t/2}. \tag{61}$$

The initial condition for this problem are

$$u(x,y,0)=e^{-x-y},\quad v(x,y,0)=e^{x+y}. \tag{62}$$

5. Results and discussion

In this section, numerical results obtained after solving the test Examples 1–4 using trivariate spectral collocation method are presented and discussed in tables and graphs. The current analysis particularly emphasizes the accuracy of the method and the ease to obtain the numerical results. Implementation of the numerical schemes was performed on Matlab 2017b platform. We investigate how the length of the computational domain and the number of collocation points influence the accuracy of the method. For the case of coupled two-dimensional Burger's system, we study the dependence of the accuracy on both the Reynolds number (Re) values and the size of the time interval. The accuracy of the method is demonstrated by presenting infinity error norms computed as;

$$||E||_\infty=\max_{\substack{0\le i\le N_x,\\ 0\le j\le N_y,\\ 0\le k\le N_t}}|U^N(x_i,y_j,t_k)-u^E(,x_i,y_j,t_k)|, \tag{63}$$

Table 1
Absolute error norms values for spectral approximation of Example 1 using different time intervals [0, T]: $N_x = 20$, $N_y = 10$, $N_t = 10$.

T	*Error norm*	CPU time(sec)	Condition Number
0.5	2.26291e-12	0.898684	2.4567e+04
1.0	2.95380e-12	0.905972	2.5045e+04
1.5	7.35126e-11	0.910378	2.5290e+04
2.0	6.01130e-10	0.912381	2.5468e+04
2.5	2.75105e-09	0.916514	2.5607e+04

Table 2
Absolute error norms values for spectral approximation of Example 2 using different time intervals [0, T]: $N_x = 10$, $N_y = 10$, $N_t = 10$.

T	*Error norm*	CPU time(sec)	Condition Number
0.5	2.29941e-11	0.230755	1.1804e+04
1.0	2.19984e-11	0.232808	1.3548e+04
1.5	2.30047e-11	0.234972	1.5161e+04
2.0	5.26619e-11	0.235249	1.6779e+04
2.5	3.48477e-11	0.236863	1.8372e+04

where U^N and u^E denotes the numerical and the exact solutions, respectively. In order to access the convergence of the iterative scheme, we keep track of the infinity error norm at different iterations. We shall call these values the solution error. The solution error at each iteration is calculated using the formula;

$$||S_{s+1}||_\infty = \max_{\substack{0\le i\le N_x,\\ 0\le j\le N_y,\\ 0\le k\le N_t}} |U^N_{s+1}(x_i, y_j, t_k) - u^E(, x_i, y_j, t_k)|. \tag{64}$$

In Table 1 and 2, we capture the dynamics of error norms as the length of the subinterval in t variable [0, T] is varied. The number of grids in the spatial domains and time were kept constant. Unless otherwise stated, the results presented are those obtained after the fifth iteration. Table 1 shows that very accurate results are obtained as error norms with small orders of magnitude are recorded. However, we notice from this table that the accuracy deteriorates as the length of the time interval increase. This can be explained by the error bound Theorem 2 given at Eq. (36) since its expected that the error grows with an increase in the size of the interval in t. There is a negligible effect on the condition number of the coefficient matrices and computation time that result from a change in the size of the time interval for a fixed number of grid points. On the contrary, as shown in Table 2, a corresponding increase in the length of the interval in t does not significantly influence the accuracy of the method in the numerical solution of Example 2. This is common expectation as the solution of the problem examined in this Example (given Eq. (52)) is smooth as compared to that of Example 1 (given Eq. (47)) which is not smooth and exhibit a discontinuity at the origin. In general, we can expect that as long as the interval in t is kept small, accurate numerical results will be guaranteed for problems with smooth solutions. Reducing the number of grid points in x space variable saves on the computational time and this evident when the third column of Table 2 is compared with that of Table 1.

The computational order of the iterative scheme is evaluated as

$$\rho = \ln\left|\frac{U^N_{s+2} - U^N_{s+1}}{U^N_{s+1} - U^N_s}\right|. \tag{65}$$

Table 3 shows the values of the computational order ρ at different iterations. From 3, we can infer that the rate of convergence of the iterative scheme is 2. These results are expected as the quasi-linearization method is based on Newton-Raphson scheme for approximating root of an equation which has been proven to exhibit quadratic convergence in many elementary books in numerical analysis.

For comparison purposes, Example 1 was also solved using 4th order Runge-Kutta method and the numerical results in terms of absolute error values obtained were given in Table 4. Comparing Tables 1 and 4 we can conclude that the trivariate spectral collocation method is more accurate and computationally efficient than 4th order Runge-Kutta method.

In Table 5 we investigate the effect of increasing the Re on the accuracy of the numerical scheme for different sizes of intervals in the time variable t. We notice that an increase in the value of Reynolds number beyond $Re = 4.0$ decreases the accuracy of the method. However, we can attest that with large values of Reynolds numbers, better accuracy can still be achieved as long as the length of the interval in t is small as seen in last column of Table 5. Infinity error norms obtained using $N_x = 10$, $N_y = 10$, $Re = 4.0$ for various lengths of the time interval are shown graphically in Figs. 1 and 2. Fig. 1 shows the graph of error norms in approximating the unknown velocity function u whereas Fig. 2 show the error norm obtained when approximating v. From both Figures we can see that for a sufficiently small interval in t, use of $N_t = 10$ grid points yields accurate results, however, for a large computational domain in t, there is need to increase the number of grid points in order to maintain highly accurate numerical approximations. In this particular case, an increase in number of grid points

Table 3
Values of computational order ρ at different iteration levels when approximating results of Example 1 using $[0, T] = [0, 1]$: $N_x = 20$, $N_y = 10$, $N_t = 10$.

Iteraton s	ρ
1	-
2	1.804
3	1.956
4	1.983
5	1.989
6	2.010
7	1.997
8	1.998
9	2.002
10	2.001

Table 4
Absolute error norms values obtained after approximation of Example 1 using 4th order Runge-Kutta method for different time intervals $[0, T]$: $N_x = 20$, $N_y = 10$, $N_t = 10$.

T	*Error norm*	CPU time(sec)	Condition Number
0.5	3.45346e-08	2.342466	4.34237e+06
1.0	7.32458e-08	2.436882	4.43821e+06
1.5	8.43256e-08	2.497213	4.56922e+06
2.0	5.04287e-07	2.600215	4.74521e+06
2.5	2.79546e-06	2.716534	4.79507e+06

Table 5
Error norm values obtained when Example 3 is solved for different values of Re : $N_x = 10$, $N_y = 10$, $N_t = 10$, $It = 5$.

	T=1.0		$T = 2.0$	
Re	Absolute Error u	Absolute Error v	Absolute Error u	Absolute Error v
1.0	5.65326e-13	5.86309e-13	4.71401e-13	7.32414e-13
2.0	2.22267e-13	2.58793e-13	2.38143e-13	2.94431e-13
3.0	1.23457e-13	1.08802e-13	1.32672e-13	3.01648e-13
4.0	3.29181e-13	3.26739e-13	3.28182e-13	3.28182e-13
5.0	4.21252e-12	4.21174e-12	4.22551e-12	4.22706e-12
10.0	8.08547e-10	8.08547e-10	9.39105e-09	9.39104e-09

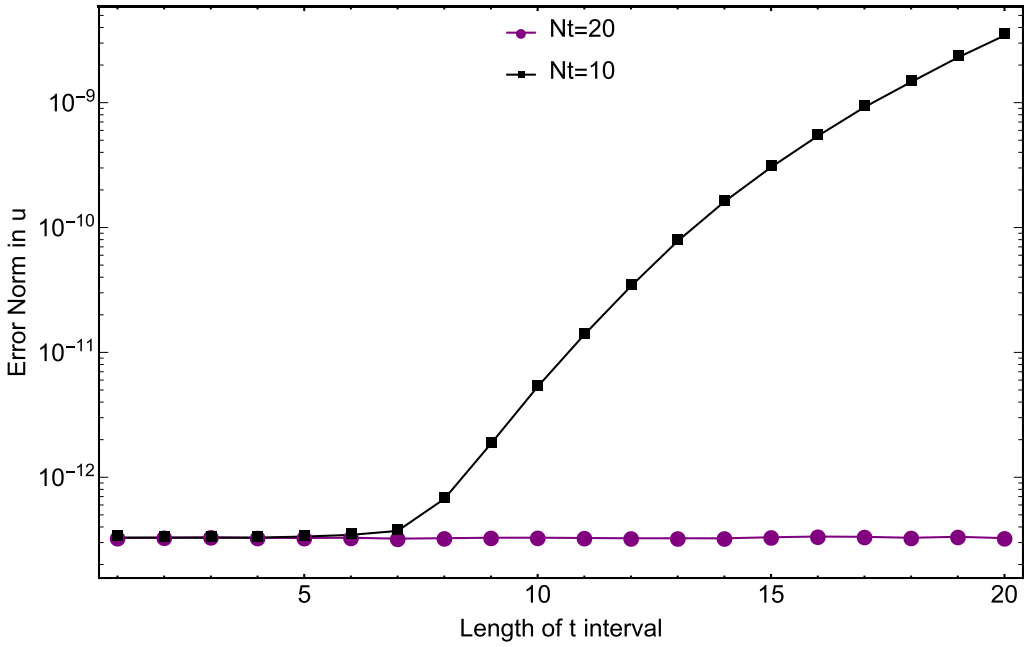


Fig. 1. Error norms in u for different time intervals $[0, T]$ for 2D coupled Burger's system.

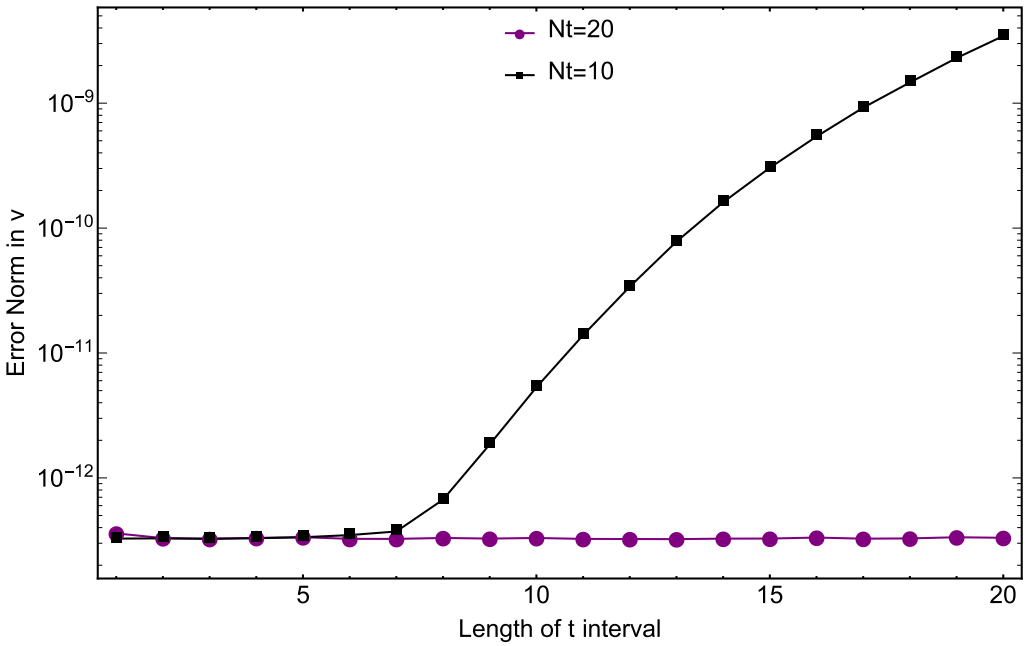


Fig. 2. Error norms in v for different time intervals [0, T] for 2D coupled Burger's system.

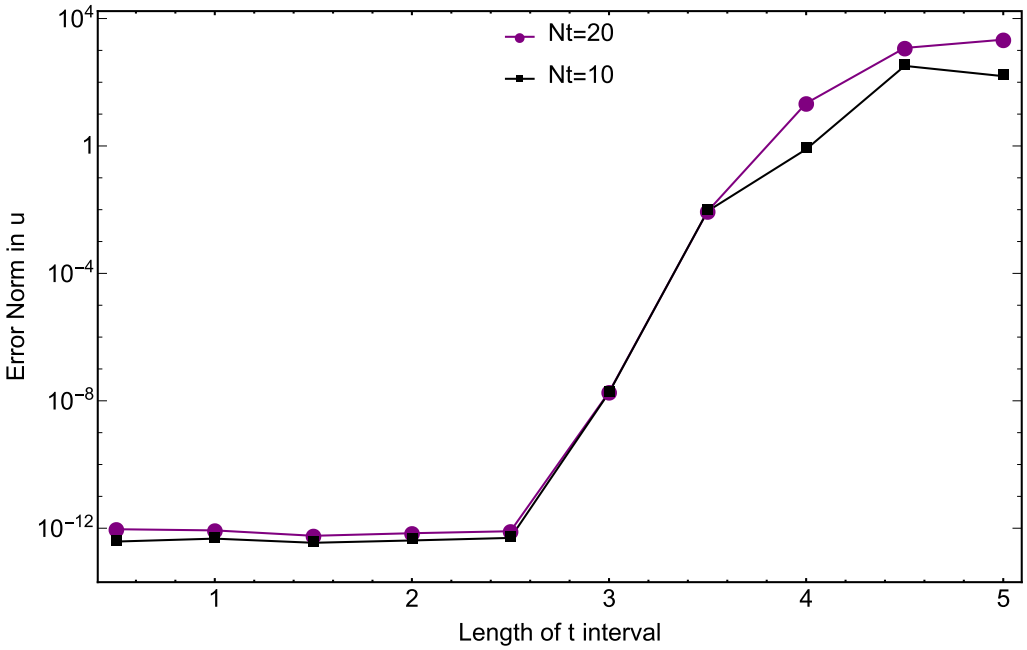


Fig. 3. Error norms in u for different time intervals [0, T] for 2D coupled Brusselator system.

from $N_t = 10$ to $N_t = 20$ was considered. We remark that this improvement in accuracy with an increase in the number of grid points is apparent only if the solution and its higher ordered derivatives are bounded within the interval of approximation which is indeed the case for the solutions of the system of 2D Burger's equation given in Eq. (57). This fact does not hold for functions whose higher ordered derivatives are unbounded within the interval of the computational domain. It must also be noted that an increase in the number of grid points is at the expense of the computational time.

Finally, in order to access the convergence of the method, the values of error norms obtained from the solution of Example 4 at different iteration levels are given in Table 6. We observe that after only 4 iterations, the numerical scheme converges and the best approximate results are obtained. In order to affirm the fact that an increase in the number of grid points for large interval in the time t variable does not always guarantee improved accuracy, error norms obtained after the fifth iteration are plotted against the length of the time interval for $N_t = 10$ and $N_t = 20$ grid points in t. We observe that for a large interval in t, increasing the number of grid points worsens the approximation errors. The loss in accuracy when a large number of grid points is used is an additional disadvantage to the increased computational time and intensive

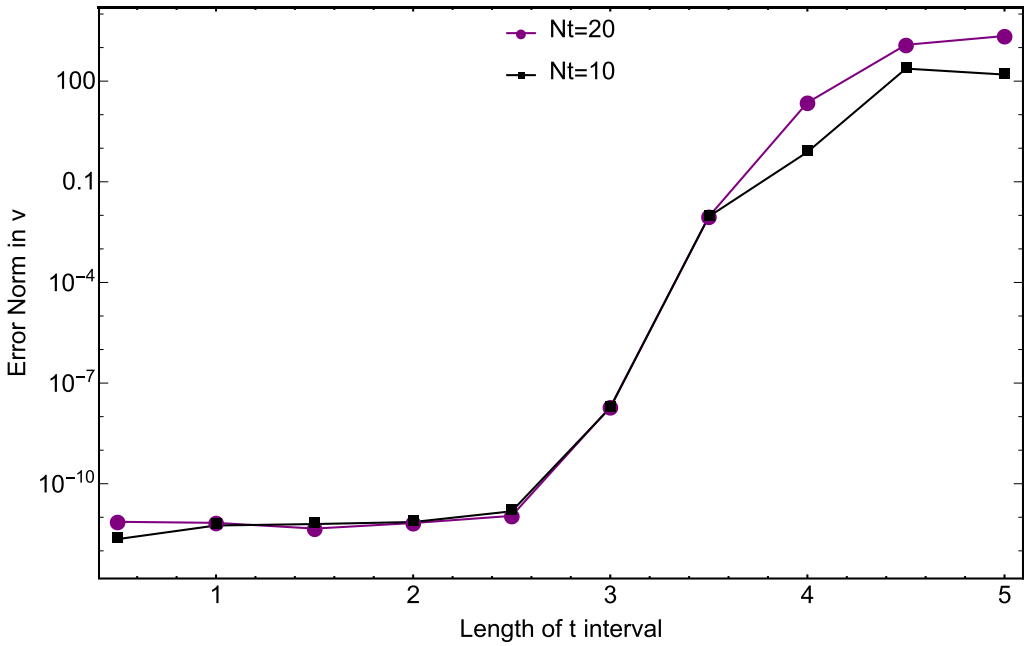


Fig. 4. Error norms in v for different time intervals $[0, T]$ for 2D coupled Brusselator system.

Table 6
Error norm values at different iterations obtained when Example 4 is solved: $N_x = 10$, $N_y = 10$, $N_t = 10$.

	T=1.0		$T = 2.0$	
Iterations	Absolute Error u	Absolute Error v	Absolute Error u	Absolute Error v
1.0	1.89722e-02	2.13396e-02	9.53246e-02	1.10242e-01
2.0	1.10363e-04	1.19616e-04	7.75372e-03	8.65135e-03
3.0	2.23119e-09	2.33731e-09	2.80567e-05	3.01118e-05
4.0	4.35874e-13	3.38041e-12	2.22283e-10	2.31627e-10
5.0	4.71456e-13	5.69322e-12	4.13003e-13	7.22800e-12
6.0	4.59466e-13	2.74447e-12	2.73559e-13	7.85416e-12
CPU time (sec)	1.305253		1.323658	
Cond Number	2.4249e+03		2.4493e+03	

memory requirements involved while handling large-sized matrices and thus must be avoided at all cost. We attribute the loss of accuracy with a large number of grid points to the unbounded nature of the solutions to 2D Brussellor system and their derivatives. In practice, since the solution to the differential equation is not known prior to the development of the numerical scheme, the proposed method will be well suited for problems defined over small intervals of the computational domain. It is worth noting that domain decomposition approaches can be employed to remedy this limitation.

6. Conclusion

In this work, a new numerical method namely, the trivariate spectral collocation method for solving nonlinear two-dimensional initial-boundary value problems has been proposed. The method employs the quasilinearization method to simplify the nonlinear differential equations and the spectral collocation to accomplish discretization in both space and time variables. The method has been described and successively applied on typical examples of two-dimensional initial-boundary value problems reported in the literature as a single nonlinear equation or systems of nonlinear equations. From the numerical simulations, we arrive at the following conclusions, the trivariate spectral collocation method yields highly accurate results when applied to solve nonlinear two-dimensional initial-boundary value problems defined on a small time interval. For fluid flow problems demonstrated in this case by the two-dimensional Burger's system, values of the governing flow parameters ought to be small to obtain accurate results. A significantly large time interval requires a large number of grid points to achieved results with stringent accuracy if the solution and its higher ordered derivatives are bounded within the computational domain. The new error bound theorems and proofs on trivariate interpolating polynomials that have been presented supports this claim. However, for problems with unbounded solutions, the application of the present method is limited to problems defined on a small time interval which requires a few grid points. The use of a large number of grid points is computationally expense and does not guarantee accuracy for problems with unbounded solutions which is not known at the onset of the application of the method. Under the suitable conditions, the current method is reliable as it gives

highly accurate results over short CPU time and the iterative scheme converges after a few iterations. Accuracy wise, the superiority of the method can be attributed to the purely spectral collocation discretization performed in all variables. Owing to the remarkable benefits the current method is a suitable alternative method for solving nonlinear two-dimensional initial-boundary value problems defined on small regular domains. Potential direction of future work would consider the extension of the current numerical method to solve problems defined on a large time interval by employing a domain decomposition technique.

Acknowledgments

The authors are thankful to the University of KwaZulu-Natal for providing essential research resources.

References

[1] B. Costa, Spectral methods for partial differential equations, A Math. J. 6 (2004) 1–32.
[2] M.G. Macaraeg, C.L. Street, Improvements in spectral collocation discretization through a multiple domain technique, Appl. Numer. Math. 2 (1986) 95–108.
[3] L. Yi, Z. Wang, Legendre spectral collocation method for second-order nonlinear ordinary/partial differential equations, Discrete Cont. Dyn. Syst. 19 (2014) 299–322.
[4] T. Zhao, C. Li, Z. Zang, Y. Wua, Chebyshev legendre pseudospectral method for the generalised Burger's-Fisher's equation, Appl. Math. Model. 36 (2012) 1046–1056.
[5] A. Tadmor, A review of numerical methods for nonlinear partial differential equations, Bull. Am. Math. Soc. 49 (2012) 507–554.
[6] M.R. Malik, T.A. Zang, M.Y. Hussaini, A spectral collocation method for the Navier-Stokes equations, Comput. Phys. 61 (1985) 64–88.
[7] A.D. Polyanin, V.F. Zaitsev, Handbook of Nonlinear Partial Differential Equations, Chapman and Hall/CRC Press, Boca Raton, 2004.
[8] S. Pamuk, N. Pamuk, Solution of two-dimensional heat and mass transfer equation with power-law temperature-dependent thermal conductivity, TWMS J. Appl. Eng. Math. 4 (2014) 199–208.
[9] J.M. Burgers, Application of a model system to illustrate some points to the statistical theory of free turbulance, R. Neth. Acad. Art Sci. 43 (1940) 2–12.
[10] I. Prigogine, R. Lefever, Symmetry breaking instabilities in dissipative systems II, Chem. Phys. 48 (1968) 1665–1700.
[11] A.R. Bahadir, A fully implicit finite-difference scheme for two-dimensional Burgers' equations, Appl. Math. Comput. 137 (2003) 131–137.
[12] A.W. Teong, The two dimensional reaction-diffusion Brusselator system: a dual-reciprocity boundary element solution, Eng. Anal. Bound. Elements 27 (2003) 897–903.
[13] A.J. Chamkha, A.M. Aly, Heat and mass transfer in stagnation-point flow of a polar fluid towards a stretching surface in porous media in the presence of soret, Dufour and chemical reaction effects, Chem. Eng. Commun. 198 (2010) 214–234.
[14] C.A.J. Fletcher, Generating exact solution of the two-dimensional Burger's equation, Numer. Methods Fluids 3 (1983) 213–216.
[15] R. Abazari, A. Borhanifar, Numerical study of the solution of the burgers and coupled burgers equations by a differential transformation method, Comput. Math. Appl. 59 (2010) 2711–2722.
[16] G. Zhao, X. Yub, R. Zhang, The new numerical method for solving the system of two-dimensional burgers equations, Comput. Math. Appl. 62 (2011) 3279–3291.
[17] S.U. Islam, B. Sarler, R. Vertnik, G. Kosec, Radial basis function collocation method for the numerical solution of the two-dimensional transient non-linear coupled burgers equations, Appl. Math. Model. 36 (2012) 1148–1160.
[18] E.H. Twizell, A.B. Gumel, Q. Cao, A second-order scheme for the Brusselator reaction diffusion system, Math. Chem. 26 (1999) 297–316.
[19] F. Khani, F. Samadi, S.H. Nezhad, New exact solutions of the Brusselator reaction-diffusion model using the exp-function method, Math. Probl. Eng. 2009 (2009) 9, doi:10.1155/2009/346461. Article ID 346461
[20] J. Biazar, Z. Ayati, An approximation to the solution of the Brusselator system by Adomian decomposition method and comparing the results with Runge-Kutta method, Contemp. Math. Sci. 2 (2007) 983–989.
[21] S. Abbasbandy, E. Shivanian, Multiple solutions of mixed convection in a porous medium on semi-infinite interval using pseudo-spectral collocation method, Commun. Nonlinear Sci. Numer. Simulat. 16 (2011) 2745–2752.
[22] R. Ellahi, E. Shivanian, S. Abbasbandy, T. Hayat, Numerical study of magnetohydrodynamics generalized couette flow of Eyring-Powell fluid with heat transfer and slip condition, Int. J. Numer. Methods Heat Fluid Flow 26 (2016) 1310–1324.
[23] R. Ellahi, E. Shivanian, S. Abbasbandy, S. U. Rahman, T. Hayat, Analysis of steady flows in viscous fluid with heat/mass transfer and slip effects, Int. J. Heat Mass Trans. 55 (2012) 6384–6390.
[24] E. Shivanian, A. Jafarabadi, Analysis of the spectral Meshless radial point interpolation for solving fractional reaction subdiffusion equation, Comput. Appl. Math. 336 (2018) 98–113.
[25] E. Shivanian, A. Jafarabadi, The spectral Meshless radial point interpolation method for solving an inverse source problem of the time-fractional diffusion equation, Appl. Numer. Math. 129 (2018) 1–25.
[26] E. Shivanian, A. Jafarabadi, An improved spectral Meshless radial point interpolation for a class of time-dependent fractional integral equations: 2D fractional evolution equation, Comput. Appl. Math. 325 (2017) 18–33.
[27] R.E. Bellman, R.E. Kalaba, Quasilinearization and Nonlinear Boundary-Value Problems, Elsevier Publishing Company, New York, 1965.
[28] L.N. Trefethen, Spectral Methods in MATLAB, SIAM, Philadelphia, 2000.
[29] M. Revers, On the approximation of certain functions by interpolating polynomials, Bull. Aust. Math. Soc. 58 (1998) 505–512.
[30] G. Mariano, S. Thomas, On the history of multivariate polynomial interpolation, Comput. Appl. Math. 122 (2000) 23–35.
[31] H.B. Ali, A highly accurate collocation algorithm for $1+1$ and $2+1$ fractional percolation equations, Vib. Control 5 (2015) 1–23.
[32] E.S. Herbert, Converting interpolation series into chebyshev series by recurrence formulas, Math. Comput. 30 (1976) 295–302.
[33] Z.B. Guozhong, Y. Xijum, Z. Rongpei, The new numerical method for solving the system of two-dimensional burgers equation, Comput. Math. Appl. 62 (2011) 3279–3291.
[34] Y. Kenan, A solution method for solving systems of nonlinear PDEs, World Appl. Sci. J. 18 (2012) 1527–1532.
[35] C.B. Dunham, J. Williams, Rate of convergence of discretization in chebyshev approximation, Math. Comput. 37 (1981) 1–5.