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

AN IMPLICIT-EXPLICIT LOCAL METHOD FOR STIFF DIFFERENTIAL EQUATIONS

Huseyin TUNC

DOCTOR OF PHILOSOPHY THESIS
Department of Mathematics
Program of Mathematics

Supervisor
Prof. Dr. Murat SARI

July, 2021

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

AN IMPLICIT-EXPLICIT LOCAL METHOD FOR STIFF DIFFERENTIAL EQUATIONS

A thesis submitted by Huseyin TUNC in partial fulfillment of the requirements for the degree of **DOCTOR OF PHILOSOPHY** is approved by the committee on 29.07.2021 in Department of Mathematics, Program of Mathematics.

Prof. Dr. Murat SARI
Yildiz Technical University
Supervisor

Approved By the Examining Committee

Prof. Dr. Murat SARI, Supervisor
Yildiz Technical University _____

Assoc. Prof. Dr. Ozgur YILDIRIM, Member
Yildiz Technical University _____

Prof. Dr. Dogan KAYA, Member
Istanbul Commerce University _____

Prof. Dr. Canan CELIK KARAASLANLI, Member
Yildiz Technical University _____

Prof. Dr. Idris DAG, Member
Eskisehir Osmangazi University _____

I hereby declare that I have obtained the required legal permissions during data collection and exploitation procedures, that I have made the in-text citations and cited the references properly, that I haven't falsified and/or fabricated research data and results of the study and that I have abided by the principles of the scientific research and ethics during my Thesis Study under the title of An implicit-explicit local method for stiff differential equations supervised by my supervisor, Prof. Dr. Murat SARI. In the case of a discovery of false statement, I am to acknowledge any legal consequence.

Huseyin TUNC

Signature

This study was supported by the Scientific and Technological Research Council of Turkey (TUBITAK) Grant No: 2211A.

Dedicated to my son, DUHA EREN

ACKNOWLEDGEMENTS

I wish to express my sincere appreciation to my supervisor, Prof. Dr. Murat SARI, who continuously guided and encouraged me to be professional and do the right thing even when the road got tough. Without his encouraging help, this work would not have achieved its purpose.

I would like to thank my thesis monitoring committee, Prof. Dr. Dogan KAYA and Assoc. Prof. Dr. Ozgur YILDIRIM, for their very important suggestions during this thesis study.

My sincere thanks also go to the Science Fellowships and Grant Programs Department of TUBITAK (TUBITAK BIDEB) for their financial support to this academic research with 2211A program.

At last but not least, I would like to thank my parents, my wife and my son, Duha Eren, for their enormous support and encouragements during this period. Their trust in me has been my biggest support.

Huseyin TUNC

TABLE OF CONTENTS

LIST OF SYMBOLS ix

LIST OF ABBREVIATIONS xi

LIST OF FIGURES xii

LIST OF TABLES xvi

ABSTRACT xviii

ÖZET xix

1 INTRODUCTION 1

1.1 Literature Review 1

1.2 Objective of the Thesis 7

1.3 Hypothesis 7

1.4 Thesis Outline 7

2 FUNDAMENTAL CONCEPTS 9

2.1 Polynomial Approximations 9

2.1.1 Multi-point Polynomial Approximations 10

2.1.2 Multi-derivative Polynomial Approximations 14

2.2 Stiff Differential Equations 17

2.3 Stiffness of the Advection-Diffusion Equations 19

3 THE IELDTM FOR STIFF IVPs 22

3.1 Implicit-Explicit Local Differential Transform Method 23

3.2 Error Analysis 25

3.3 Stability Analysis 28

3.3.1 Scalar Test Equation 28

3.3.2 Stability for Linear Systems 29

3.4 Adaptive Searching 33

3.5 Numerical Experiments 34

3.6 Summary 44

4 THE IELDTM FOR STIFF BVPs **45**
4.1 The Implicit-Explicit Local Differential Transform Method 45
4.2 Numerical Implementation and Error Analysis 49
4.3 Stability Analysis 55
4.4 Numerical Experiments 56
4.5 Summary 69

5 THE IELDTM FOR TIME INTEGRATION OF THE AD EQUATIONS **70**
5.1 Two-Dimensional Nonlinear Advection-Diffusion Equation 70
5.2 The ChSCM for Spatial Variation 71
5.3 The IELDTM for Time Integration 73
5.4 Error Analysis 74
5.5 Numerical Experiments 78
5.6 Summary 84

6 THE IELDTM FOR SPATIAL DISCRETIZATIONS OF ONE-DIMENSIONAL ADR EQUATIONS **85**
6.1 Space-Time IELDTM Treatment 85
6.2 The ChSCM for Time Integration 87
6.3 The IELDTM for Spatial Discretization 89
6.4 Error Analysis 91
6.5 Numerical Experiments 93
6.6 Summary 100

7 THE IELDTM FOR AD EQUATIONS IN TWO- AND THREE-DIMENSIONS **102**
7.1 IELDTM-ChSCM for (2+1)-dimensional AD equation 103
7.1.1 Temporal Variation for (2+1)-Dimensional AD Equation 103
7.1.2 The IELDTM for Two-Dimensional Problems 105
7.2 IELDTM-ChSCM for (3+1)-Dimensional AD Equation 108
7.2.1 Temporal Variation for (3+1)-Dimensional AD Equation 109
7.2.2 The IELDTM for Three-Dimensional Problems 110
7.3 Numerical Experiments 115
7.4 Summary 122

8 RESULTS AND DISCUSSION **125**

REFERENCES **128**

A BASICS OF THE DIFFERENTIAL TRANSFORMATION **139**

B ERROR ANALYSIS **143**

LIST OF SYMBOLS

P_N	Nth order approximation polynomial
$L_{N,i}$	Lagrange polynomial coefficients
E_N	Residual error function
I_N	Polynomial interpolation operator
Λ_N	Lebesgue constant
T_n	Chebyshev polynomials
δ_{ij}	Kronecker-delta
ρ	Radius of convergence
μ	Stiffness ratio
V	Advection constant
V_x	Advection constant in $x-$ direction
V_y	Advection constant in $y-$ direction
V_z	Advection constant in $z-$ direction
D	Diffusion constant
D_x	Diffusion constant in $x-$ direction
D_y	Diffusion constant in $y-$ direction
D_z	Diffusion constant in $z-$ direction
β	Reaction constant
Δt	Time increment
Δx	Spatial increment in $x-$direction
Δy	Spatial increment in $y-$direction
Δz	Spatial increment in $z-$direction
t_f	Final time

θ	Direction parameter
θ_x	Direction parameter in $x-$ direction
θ_y	Direction parameter in $y-$ direction
θ_z	Direction parameter in $z-$ direction
$\boldsymbol{\epsilon}$	Global discretization error
δ	Local discretization error
tol	Desired accuracy in adaptive schemes
K	Local approximation order
M_x	Number of spatial elements in $x-$direction
M_y	Number of spatial elements in $y-$direction
M_z	Number of spatial elements in $z-$direction

LIST OF ABBREVIATIONS

AD	Advection-Diffusion
ADR	Advection-Diffusion-Reaction
BDF	Backward Difference Formula
BVP	Boundary Value Problem
CGL	Chebyshev-Gauss-Lobatto
ChSCM	Chebyshev Spectral Collocation Method
CNM	Crank-Nicolson Method
CPU	Central Process Unit
DTM	Differential Transform Method
FDM	Finite Difference Method
FD6	Sixth-Order Finite Difference
FEM	Finite Element Method
FVM	Finite Volume Method
GFEM	Galerkin Finite Element Method
IELDTM	Implicit-Explicit Local Differential Transform Method
IVP	Initial Value Problem
LBRFDM	Linear Barycentric Rational Finite Difference Method
LDTM	Local Differential Transform Method
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
SEIR	Susceptible-Exposed-Infected-Recovered

LIST OF FIGURES

Figure 2.1 The stiffness ratio versus spatial increment for the advection-diffusion equation with the parameter values $V = 1$ and $D = 0.01$. 21
Figure 3.1 Stability region (blue) of the forward schemes, $\theta = 0$, for various orders on the complex plane. 30
Figure 3.2 Stability region (blue) of the central schemes, $\theta = 0.5$, for various orders on the complex plane. 30
Figure 3.3 Stability region (blue) of the backward schemes, $\theta = 1$, for various orders on the complex plane. 30
Figure 3.4 Comparison of the central IELDTM and LDTM with the reference *ode15s* solutions of the SEIR model (3.46) for $K = 6$ and $\Delta t = 3$. . 37
Figure 3.5 a) Comparison of the required number of time steps of the central adaptive IELDTM, *ode23s*, *ode15s*, *ode113* and *ode45* for solving the SEIR model (3.46) with the changing values of η and $tol = 10^{-5}$, b) Effect of the stiffness factor η on the dynamics of the susceptible population $S(t)$ is captured by the central adaptive IELDTM with an optimized degrees of freedom. 37
Figure 3.6 Order refinement results of the IELDTM at $t_f = 1$ for various values of the direction parameter θ with the time increment $\Delta t = 0.05$. . 39
Figure 3.7 Pointwise errors computed by the central adaptive IELDTM with various values of K and $tol = 10^{-15}$. 39
Figure 3.8 Comparison of the present central adaptive IELDTM and the *ode15s-ode23s* through pointwise errors with $tol = 10^{-12}$ for Problem 3.3. 42
Figure 3.9 Adaptive central solutions produced with $K = 7$, $\theta = 0.5$, $tol = 10^{-10}$ and various ε values for Problem 3.4. 43
Figure 3.10 Adaptive central solutions produced with $K = 5$, $\theta = 0.5$, $tol = 10^{-10}$, $\varepsilon = 10$ and $\varepsilon = 100$ for Problem 3.4. 44
Figure 4.1 $||A^*||_\infty$ norm values as a function of the problem parameters p and q. 55
Figure 4.2 Effects of K and θ values on $||A^*||_\infty$ norm for the problem parameter values $p = 3$ and $q = -3$. 56

Figure 4.3 Pointwise errors produced by the IELDTM and the DTM with $\Delta x = 0.1$ and $K = 9$ for Problem 4.1. 63
Figure 4.4 Pointwise errors produced by the IELDTM and the DTM with $\Delta x = 0.05$, $\varepsilon = 0.001$ and $K = 13$ for Problem 4.2. 64
Figure 4.5 Pointwise errors produced by the IELDTM and the DTM with $\Delta x = 0.05$ and $K = 7$ for Problem 4.3. 64
Figure 4.6 Pointwise errors produced by the IELDTM with $\Delta x = 0.1$ and $K = 13$ for Problem 4.4. 65
Figure 4.7 Pointwise errors produced by the IELDTM with $\Delta x = 0.1$ and $K = 9$ for Problem 4.5. 65
Figure 4.8 Comparison of numerical results produced by the IELDTM and the DTM with $\Delta x = 0.1$ and $K = 11$ for Problem 4.4. 66
Figure 4.9 Comparison of numerical results produced by the IELDTM and the DTM with $\Delta x = 0.1$ and $K = 9$ for Problem 4.5. 67
Figure 4.10 Effect of the direction parameter θ to maximum errors produced by the IELDTM with $\Delta x = 0.1$ and $K = 9$. 67
Figure 4.11 Processing times with respect to varying values of the transformation order K with $\Delta x = 0.1$ and $\theta = 0.5$. 67
Figure 5.1 Order refinement results of the ChSCM-IELDTM for Problem 5.1 at $t_f = 0.5$ for various values of the direction parameter θ with a) $D = 1$, $N = 10$ and $\Delta t = 0.0025$ b) $D = 0.1$, $N = 40$ and $\Delta t = 0.0025$. 80
Figure 5.2 The ChSCM-IELDTM solutions of Problem 5.1 at various times for the parameter values a) $D = 0.001$, $N = 80$, $\theta = 0.5$, $K = 4$ and $\Delta t = 0.001$ b) $D = 0.0005$, $N = 120$, $\theta = 0.5$, $K = 4$ and $\Delta t = 0.001$. 80
Figure 5.3 Order refinement results of the ChSCM-IELDTM for Problem 5.2 at $t_f = 0.5$ for various values of the direction parameter θ with a) $D = 1$, $N = M = 10$ and $\Delta t = 0.005$ b) $D = 0.1$, $N = M = 20$ and $\Delta t = 0.005$. 82
Figure 5.4 The ChSCM-IELDTM solutions of Problem 5.2 at various times for the parameter values a) $D = 0.01$, $N = M = 40$, $\theta = 0.5$, $K = 4$ and $\Delta t = 0.01$ b) $D = 0.005$, $N = M = 80$, $\theta = 0.5$, $K = 4$ and $\Delta t = 0.01$. 83
Figure 6.1 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter θ with $\Delta x = 0.01$, $N = 60$ and $t_f = 1$. 95

Figure 6.2 Convergence results of the present algorithm with respect to the spatial increment Δx for various values of the direction parameter θ with $K = 4$, $N = 60$ and $t_f = 1$. 95
Figure 6.3 a) Comparison of the IELDTM-ChSCM solution with the exact solution, b) Pointwise absolute errors produced by the IELDTM-ChSCM at various times with $\Delta x = 1$, $N = 80$, $\theta = 0.5$ and $K = 6$ for Problem 6.2. 96
Figure 6.4 Illustration of the advection dominated processes of Problem 6.2 solved by the IELDTM-ChSCM for a) $V = 0.02$, $D = 0.005$, $\Delta x = 0.25$, $N = 40$, $\theta = 0.5$, $K = 6$ and $t_f = 3000$ b) $V = 0.02$, $D = 0.001$, $\Delta x = 0.2$, $N = 40$, $\theta = 0.5$, $K = 6$ and $t_f = 3000$. 97
Figure 6.5 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter θ with $D = 0.1$, $\Delta x = 0.05$, $N = 50$ and $t_f = 2$ in Problem 6.3. 98
Figure 6.6 Convergence results of the present algorithm with respect to Δx for various values of the direction parameter θ with $D = 0.1$, $K = 4$, $N = 50$ and $t_f = 2$ (see Problem 6.3). 99
Figure 6.7 The IELDTM-ChSCM solution of Problem 6.4 and corresponding pointwise errors produced with $V = 1$, $\beta = 1$, $D = 2^{-8}$, $\Delta x = 0.003$, $N = 30$, $\theta = 0.5$ and $K = 2$. 100
Figure 7.1 Descriptive behaviour of pointwise absolute errors produced by the central IELDTM at various times with $\theta_x = \theta_y = 0.5$, $M_x = M_y = 5$, $K = 10$, $N = 20$, $S = 14$ and $t_f = 1$ (see Problem 7.1). 116
Figure 7.2 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter couples (θ_x, θ_y) with $x_f = y_f = 1$, $M_x = M_y = 2$, $N = 10$, $S = 10$ and $t_f = 0.1$ (see Problem 7.2). 118
Figure 7.3 Convergence results of the present algorithm with respect to the spatial element numbers $M_x = M_y$ for various values of the direction parameter couples (θ_x, θ_y) with $x_f = y_f = 1$, $K = 8$, $N = 10$, $S = 12$ and $t_f = 0.1$ (see Problem 7.2). 118
Figure 7.4 a) Effects of the mesh discretization parameter S to the maximum error norms $||E||_\infty$, b) Effects of the mesh discretization ratio ϕ to the maximum error norms $||E||_\infty$ with $x_f = y_f = 1$, $D_x = D_y = 1$, $M_x = M_y = 1$, $N = 10$, $K = 20$ and $t_f = 0.1$. 119

Figure 7.5 Numerical results and corresponding pointwise absolute errors for solving Problem 7.2 with the use of the central IELDTM for $\theta_x = \theta_y = 0.5$, $x_f = y_f = 1$, $K = 4$, $M_x = M_y = 18$, $N = 5$, $S = 6$ and $t_f = 0.05$. 119
Figure 7.6 Numerical results and corresponding pointwise absolute errors for solving Problem 7.2 with the use of the central IELDTM for $\theta_x = \theta_y = 0.5$, $x_f = y_f = 5$, $K = 8$, $M_x = M_y = 8$, $N = 16$, $S = 5$ and $t_f = 2$. 120
Figure 7.7 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter couples $(\theta_x,\ \theta_y)$ with $M_x = M_y = 2$, $N = 10$, $S = 10$ and $t_f = 0.5$ (see Problem 7.3). 121
Figure 7.8 Convergence results of the present algorithm with respect to the spatial element numbers $M_x = M_y$ for various values of the direction parameter couples $(\theta_x,\ \theta_y)$, $K = 6$, $N = 10$, $S = 10$ and $t_f = 0.5$ (see Problem 7.3). 121
Figure 7.9 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter triples $(\theta_x,\ \theta_y,\ \theta_z)$ with $M_x = M_y = M_z = 2$, $N = 10$, $S = 4$ and $t_f = 0.1$ (see Problem 7.4). 123
Figure 7.10 Convergence results of the present algorithm with respect to the spatial element numbers $M_x = M_y = M_z$ for various values of the direction parameter triples $(\theta_x,\ \theta_y,\ \theta_z)$ with $K = 6$, $N = 10$, $S = 4$ and $t_f = 0.5$ (see Problem 7.4). 123
Figure 7.11 Numerical results and corresponding pointwise absolute errors of Problem 7.4 with the use of the IELDTM for $K = 12$, $M_x = M_y = M_z = 1$, $N = 10$, $S = 4$ and $t_f = 0.25$. 124

LIST OF TABLES

Table 2.1 Some of the key properties of the Chebyshev polynomials. 12
Table 3.1 Some special A—stable cases of the IELDTM. 32
Table 3.2 Effect of the approximation order of the IELDTM with various direction parameters in comparative way for Problem 3.2 (NE: Numerical Experiment; TE:Theoretical Expectation). 40
Table 3.3 Comparison of the present IELDTM ($\theta = 0.5$) and the *ode15s-ode23s* solvers [142] with $tol = 10^{-10}$ for adaptive time steps N^* and maximum errors in Problem 3.2. 40
Table 3.4 Comparison of the present central IELDTM with the literature LBRFDM-BDF [31] through maximum errors at $t = 4$ for Problem 3.3. 41
Table 3.5 Comparison of the present central adaptive IELDTM and the MATLAB solvers [142], *ode15s-ode23s,* with the adaptive time step numbers for $tol = 10^{-10}$ in Problem 3.4. 43
Table 4.1 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.1. 63
Table 4.2 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.2. 64
Table 4.3 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.3. 65
Table 4.4 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.4 66
Table 4.5 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.5. 66
Table 4.6 Comparison of $||E||_\infty$ norms of the IELDTM, LDTM and DTM with $K = 6$ and $\Delta x = 0.01$ for Problem 4.2. 68
Table 4.7 Validation of the approximation orders of the present IELDTM with the direction parameter $\theta = 0.5$ for Problems 4.4-4.5. 68
Table 4.8 Comparison of the IELDTM and the rival higher order methods with $||E||_\infty$ norms for Problem 4.6. 68

Table 5.1 Number of time steps and the maximum pointwise errors of the time-adaptive ChSCM-IELDTM approach for various values of the viscosity constant D and time approximation order K with $\theta = 0.5$, $t_f = 0.5$ and $tol = 10^{-10}$. 81
Table 5.2 Comparison of the numerical results produced with $t_f = 0.1$, $D = 0.1$ and $\Delta t = 0.001$ for Problem 5.1 and a) The GFEM [43] with $N = 40$, b) The IELDTM with $\theta = 0.5$, $K = 1$ and $N = 20$, c) The IELDTM with $\theta = 0.5$, $K = 3$ and $N = 20$. 81
Table 5.3 Number of time steps and the maximum pointwise errors of the time-adaptive ChSCM-IELDTM method for various values of the viscosity constant D and time approximation order K with $\theta = 0.5$, $t_f = 0.5$ and $tol = 10^{-15}$. 83
Table 6.1 Comparison of the numerical results produced at various nodal points and $t_f = 1$ for Problem 6.1 and a) The FDM6/8/10 [107] with $M = 100$, $\Delta t = 0.001$ and $dof = 10^5$, b)The GFEM [151] with $M = 100$, $\Delta t = 0.001$ and $dof = 10^5$, c)The IELDTM with $\theta = 0.5$, $N = 10$, $M = 40$, $K = 10$ and $dof = 800$, d)The IELDTM with $\theta = 0.5$, $N = 10$, $M = 40$, $K = 8$ and $dof = 800$. 94
Table 6.2 Comparison of the numerical results produced with $D = 0.1$ and $t_f = 2.3$ for Problem 6.3 and a) The BDF3 [153] with $M = 100$, $dt = 0.01$ and $dof = 10^4$, b) The GFEM [43] with $M = 100$, $dt = 0.01$ and $dof = 10^4$, c) The IELDTM with $\theta = 0.5$, $N = 10$, $M = 25$, $K = 4$ and $dof = 500$, d) The IELDTM with $\theta = 0.5$, $N = 10$, $M = 25$, $K = 6$ and $dof = 500$. 98
Table 6.3 Maximum pointwise errors produced with $V = 0.001$, $\beta = 0.001$ and $t_f = 1$ in Problem 6.4 and a) The FEM [154] with $M = 32$, $\Delta t = 0.0002$ and $dof = 16 \times 10^4$, b) The FEM [154] with $M = 64$, $\Delta t = 0.0002$ and $dof = 32 \times 10^4$, c) The IELDTM with $N = 10$, $M = 32$ and $dof = 640$, d) The IELDTM with $N = 10$, $M = 64$ and $dof = 1280$. 100
Table 7.1 Comparison of various present schemes with the FDM-based results produced in references [108, 157] at $t_f = 0.25$ with $N = 15$, $S = 14$ and $K = 10$ for Problem 7.1. 116
Table 7.2 Comparison of various present schemes with the FEM and the ChSCM produced in [48, 98] at $t_f = 0.25$ with $N = 15$, $S = 16$ and $K = 10$ for Problem 7.3. 121

ABSTRACT

An implicit-explicit local method for stiff differential equations

Huseyin TUNC

Department of Mathematics
Doctor of Philosophy Thesis

Supervisor: Prof. Dr. Murat SARI

In this thesis, a new implicit-explicit local differential transform method (IELDTM) with an arbitrary order is produced for stiff differential equations. In the produced method, which is a stability-preserved numerical algorithm that accepts the idea of differential transformation as a starting point, all information about the numerical process is taken directly from the corresponding differential equations. The currently proposed method is derived for stiff initial value problems (IVPs), stiff boundary value problems (BVPs), and stiff parabolic initial-boundary value problems. All theoretical analyses including priori error estimations and stability analysis are provided and the theoretical order expectations are verified through various numerical experiments. The IELDTM is found to be a high order, stability preserved, and versatile numerical approach for both ordinary differential equations (ODEs) and parabolic partial differential equations (PDEs) up to three spatial dimensions. The IELDTM is proven to eliminate both some existing drawbacks of the differential transform-based methods and other numerical techniques such as the finite element method (FEM) and finite difference method (FDM) by providing optimized degrees of freedom.

Keywords: Advection-diffusion-reaction equation, stiff differential equation, numerical algorithm, Taylor series, initial value problem, boundary value problem

YILDIZ TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

ÖZET

Stiff diferansiyel denklemler için bir kapalı-açık lokal yöntem

Huseyin TUNC

Matematik Anabilim Dalı
Doktora Tezi

Danışman: Prof. Dr. Murat SARI

Bu tezde, stiff diferansiyel denklemler için keyfi mertebeden yeni bir kapalı-açık yerel diferansiyel dönüşüm yöntemi (IELDTM) üretilmiştir. Diferansiyel dönüşüm fikrini başlangıç noktası olarak kabul ederek kararlılığı koruyan bu sayısal yöntemde, hesaplama süreci ile ilgili tüm bilgiler doğrudan söz konusu diferansiyel denklemlerden alınır. Bu çalışmada önerilen yöntem, stiff başlangıç değer problemleri (BDP), stiff sınır değer problemleri (SDP) ve stiff parabolik başlangıç-sınır değer problemleri için üretilmiştir. Hata tahminleri ve kararlılık analizleri dahil olmak üzere tüm teorik çalışmalar sağlanıp teorik olarak beklenen mertebeler çeşitli sayısal deneylerle doğrulanmıştır. Sunulan yöntemin hem adi diferansiyel denklemler hem de üç boyutlu parabolik kısmi diferansiyel denklemlere kadar yüksek mertebeli, kararlılığı koruyan ve çok yönlü sayısal bir yaklaşım olduğu gösterilmiştir. Bu çalışmada üretilen yöntemin, optimize edilmiş serbestlik dereceleri sağlanarak hem diferansiyel dönüşüm tabanlı yöntemin mevcut dezavantajlarını hem de sonlu elemanlar yöntemi ve sonlu farklar yöntemi gibi diğer sayısal tekniklerin bazı dezavantajlarını ortadan kaldırdığı kanıtlanmıştır.

Anahtar Kelimeler: Adveksiyon-difüzyon-reaksiyon denklemi, stiff diferansiyel denklem, nümerik algoritma, Taylor serisi, başlangıç değer problemi, sınır değer problemi

YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

1
INTRODUCTION

1.1 Literature Review

Capturing numerical behaviour of differential equations encountered in various fields of science such as physics, chemistry, engineering, biology, and finance, is vital. In general, it is often not possible to obtain analytical solutions of the equations representing realistic behaviours in nature. Besides, analytical methods, which are available for only simple differential equations, yields time-consuming symbolic calculations for especially solving relatively large system of differential equations. These restrictions of analytical approaches can be handled by considering accurate and economic numerical methods that have a rich history dates back to 1768 with the well-known study of Euler [1]. In numerical techniques, it is, in general, easy to convert differential equations to algebraic equations. By the appropriate selection of parameters used in a numerical method, the accuracy of the method can generally be controlled by constructing suitable meshes. Thus, numerical methods for the differential equations eliminate the existing drawbacks of the analytical methods and yield quantitative results to observe the corresponding dynamics.

Performance of the numerical methods for solving differential equations strongly depends on the smoothness of the solution functions. Stiff differential equations, which are invented by Curtiss and Hirschfelder at the beginning of the 1950s [2], yield non-smooth solutions and it is not an easy task to find out reliable numerical solutions for these equations in the entire domain. Solutions of the stiff differential equations include both transient and smooth components that yield rapid and slow changes in the solutions, respectively. The areas of chemical kinetics, nonlinear mechanics, fluid dynamics and biochemistry are main sources of stiff problems [3]. Stiff behaviours are modelled by not only ordinary differential equations (ODEs) [4, 5] but also partial differential equations (PDEs) [6, 7] and arise in various phenomena. Numerical techniques are also classified according to the stiff and non-stiff problems in the literature [3, 5]. Since capturing behaviours of the fast transient components is not an easy task, the constructed numerical methods to solve stiff problems are

an event of special interest [3, 5]. Divergence, reduction of order and undesired oscillations are commonly faced in solving stiff problems under the consideration of some inappropriate numerical techniques.

Stiff initial value problems (IVPs) for ODEs can be solved by suitable time integration methods by constructing adaptive time-discretizations. Time integration methods are mainly divided into two subgroups, such as multi-point [8–13] and multi-derivative methods [14–22]. The multi-point methods are based on the idea of interpolation, such as finite element methods [8, 12, 13] and spectral methods [9–11]. Even if this group of methods is widely used for the spatial parts of PDEs, it is also possible to solve IVPs with these methods. The multi-point methods lead to large degrees of freedom and a large system of linear/nonlinear algebraic equations for solving ODEs due to their non-iterative nature. This is an essential drawback due to both computational cost and loss of accuracy over large time scales. The multi-derivative methods such as Runge-Kutta methods [14–16], Rosenbrock methods [17–19], Taylor series methods [20–23] have been used extensively to observe the behaviour of stiff and non-stiff IVPs. As for this group of methods, they are iterative in nature and have both explicit and implicit forms of their own. The Runge-Kutta and Rosenbrock methods are derived from the Taylor series expansions by introducing interior ghost nodes to get higher-order approximations. The Taylor series methods directly use the higher-order derivatives of given implicit/explicit functions by the procedure known as automatic differentiation or differential transformation [20–22, 24–27]. The idea of differential transformation leads us to a useful procedure for recursively evaluating higher-order Taylor coefficients [24–27].

In recent years, various numerical techniques have been derived in the literature to deal with challenging stiff IVPs for ODEs [13, 28–35]. A quasi-consistent fixed-step size numerical integration was used as an implicit stiff IVP solver [28]. The method was found to be stable and was seen to preserve accuracy but is not computationally optimal due to the use of fixed step sizes. Another numerical study [29] on solving stiff IVPs was presented as a one-leg explicit-implicit method, which is second-order and A-stable. Although this method offers an important numerical approach for stiff IVPs, the low order of the method and assuming fixed step sizes lead to a computationally ineffective algorithm. For solving stiff differential-algebraic equations, an adaptive implicit Euler scheme was proposed in literature [30]. Even if the method produces acceptable results for stiff differential-algebraic equations, the order of the method is one, and the adaptive procedure requires many time steps to achieve acceptable results over long intervals of time. Abdi et al. [31] proposed arbitrary sequential barycentric rational finite difference method regarding the backward difference formulae (BDF). This method was shown to be adaptable to stiff IVPs, but the A-stable cases are

obtained for at most second-order formulations as it is also the case in the BDFs. A multistep method based on the BDFs was derived for stiff linear problems in the literature [32]. The proposed method offers a different perspective on the construction of constrained optimization in terms of the coefficients of the BDFs rather than using predetermined coefficients. The authors tested their method on the linear equation, and the method seems to have a lack of computational efficiency for nonlinear stiff problems due to the extra cost of the optimization process. Moreover, it is known from the literature that the discontinuous Galerkin methods (DGM) are very effective for the differential equations representing shock behaviours. In the study of Fortin and Yakoubi [13], advantages of the DGM were discussed for stiff IVPs, and an adaptive formulation was presented. A considerable disadvantage of the DGM is the coupling of the degrees of freedom at common interior nodes, which makes the method a bit computationally ineffective. Recently, a modified approach based on radial basis functions that improve the quality of known numerical techniques was offered by Gu and Jung [33] for non-stiff problems. In addition, recently some techniques such as various versions of Runge-Kutta methods [14–16, 34, 35], backward differentiation methods [36–38] and collocation methods [39–41], have been in a competition to solve stiff IVPs of ODEs.

Stiff ODE systems do not only come out directly in applications but also are encountered in the discretization of some challenging processes described by PDEs, such as advection-diffusion equations [42, 43]. Time integrations of the reduced space-time partial differential equations are of great importance. Various versions of both explicit and implicit time integrators are constantly under development to create accurate and stable numerical approaches. For higher dimensional parabolic PDEs, the computational efficiency of the time integrator becomes more critical due to the large ODE systems obtained after spatial discretization [44]. In such cases, the performance of the effective explicit ODE solvers such as Taylor series methods or explicit Runge-Kutta methods is weakened due to instability [3, 5, 45]. Implicit time integration schemes are the best options in such situations but require more computational effort, especially for nonlinear problems [3, 5, 45].

When dealing with the time integration of the advection-diffusion processes, numerous numerical methods have been proposed in the literature [42, 44, 46–54]. Explicit Runge-Kutta schemes are more common in applications [44, 46–49] due to their efficient computational structures. It is well known that this group of time-integrators has a lack of stability preserving properties. The second-order and unconditionally stable Crank-Nicolson method (CNM) is another widely used time integrator in advection-diffusion processes, and the CNM is a special case of the well-known θ-method [42, 50–52]. Even if the stability properties of the CNM were

found to be excellent, the method is second order and needs small time-increments to get higher accuracy. This serious drawback of the θ-methods will be eliminated in this thesis with the derivation of a more general time-integration technique (see Chapters 3 and 5). Another time-integration technique for solving the reduced ODEs is the backward differentiation method (BDM), in which stability is preserved and has higher-order formulae [54, 55]. The BDM has A-stable formulae up to the second-order accuracy. Since the CNM is also a second-order A-stable method, the BDM is not of a considerable advantage over the CNM in this sense.

Other types of challenging problems in applied science are the stiff boundary value problems (BVPs) of ODEs. BVPs for ODEs arise in many disciplines such as physics, chemistry, engineering, finance and mathematical biology [56]. Stiff boundary value problems are also known as the singularly-perturbed BVPs, which are highly challenging problems for numerical methods [57]. Stiff BVPs involve significant local behaviours such as sharp discontinuities or boundary layers, and various notable difficulties are encountered in numerical methods [57–59].

Numerical methods for BVPs are mainly divided into two categories: direct methods and shooting methods [60]. The shooting strategies are important for using IVP algorithms to solve BVPs, but the computational mechanisms involve additional costs to find high-order initial conditions [61–63]. A well-known direct method for BVPs is the finite difference method widely used in literature [64–69]. Even if finite difference methods produce acceptable results for many BVPs, their local order refinement (p-refinement) is not an easy task due to direct disconnection of the higher-order formulations. Another kind of direct method is the collocation methods based on direct substitutions of approximate solutions and getting algebraic equations at collocation points [70–75]. This group of methods is suitable for local order refinement and can also be used in either local or global forms. Besides these widely used techniques, local discontinuous Galerkin method [76], spline interpolation based numerical methods [77–79], Adomian decomposition methods [80, 81] and reproducing kernel methods [81–83] have been applied to solve various kinds of BVPs for ODEs.

For solving BVPs for ODEs, the Taylor series-based differential transform method (DTM) provides semi-analytic series solutions [84, 85]. The DTM can solve the BVPs, even if the original structure of the method is designed for IVPs. Thus, DTM based methods for BVPs can be considered to be shooting methods [84–91]. Generally, those semi-analytic approaches produce divergent results for solving stiff BVPs. The idea of discretizing a finite domain and then applying the DTM procedure is known as local DTM (LDTM), or multistep DTM [27, 92]. Although the LDTM is better than the DTM

in terms of convergence, as shown in Chapter 4, the LDTM does not lead to stable results as a shooting method [**44**] for especially highly stiff BVPs. Thus, by eliminating the existing drawbacks of the DTM-based methods, the development of a higher order and stability preserved method will fill a serious gap in this area (see Chapter 4).

The stiff BVPs represented by parabolic PDEs are highly challenging for numerical methods. One of the prominent stiff parabolic PDE examples is the advection-diffusion-reaction (ADR) equations in various spatial dimensions. The ADR equations naturally appear in the modelling of many physical or biological problems such as the tumour angiogenesis model, the tumour invasion model, heat transfer in draining film, dispersion of tracers in porous media, the spread of pollutants in rivers and stream, the dispersion of dissolved material in estuaries and coastal sea and so on [4]. When the advection is dominant to the diffusion in the process, the sharp behaviours naturally occur at some spatial positions [42, 43, 46]. In these cases, numerical methods should be constructed in such a way that spurious oscillations do not occur. Since the advection-dominated processes need carefully constructed meshes, the minimization of the required degrees of freedom is vital for any numerical approach.

For solving stiff ADR equations, various versions of the finite element method (FEM), discontinuous Galerkin method (DGM), finite difference method (FDM) and finite volume method (FVM) have been presented in literature [42, 43, 46, 93–111]. Among these high capacity numerical approaches, finite element-based techniques have been dominantly applied for these extremely challenging problems [42, 43, 93–100]. The FEMs take advantage of geometric flexibility, weak formulation and interpolation. Even if the classical Galerkin FEM suffers from instability for advection-dominated problems, some stabilization methods together with the Galerkin FEM offer better numerical results [95–97]. Despite all these considerable advantages, the $p-$refinement procedure of the FEM increases the degrees of freedom and computational cost. Also, the discontinuous Galerkin methods (DGM) have several advantages: high geometric flexibility, mass conservation properties, and shock-capturing nature. In recent literature [101–106], various versions of the spatial DGM and space-time DGM have been presented for solving stiff ADR equations. Since the common nodes of the elements contribute two times in the DGM, both $h-$refinement and $p-$refinement procedures increase the degrees of freedom much more than the Galerkin FEM. In addition to these methods, the finite difference methods have been widely used for solving the ADR equations either in explicit or implicit forms [46, 107–111]. The FDMs take advantage of the $p-$refinement procedure by increasing the nodes used in the stencils. The required degrees of freedom are not affected by order of the FDM formulation. However, the FDMs

have crucial restrictions, such as the need for a structured geometric domain and the necessity of using low order approximations around the boundaries. One more disadvantage of the FDM is the high-order formulations are not directly related to each other, and all formulations must be redefined to increase the order of the method. Another spatial discretization method of the ADR equations is the finite volume method (FVM) or control volume method widely studied in literature [112–115]. The FVM is based on the spatial integrations of the given differential equations over the control volumes, and the method is more suitable for the problems that represent the conservation laws, such as the pure advection problems. Hybridization of the FVM with some flux limiting techniques produces physically acceptable numerical solutions around the shock positions [116]. The FVM has two crucial drawbacks. First, the known FVM can not take advantage of p—refinement, and thus the algorithms are low order [112, 113]. The second one is that the FVM is not a versatile method, i.e. the FVM works only for the differential equations representing the conservation laws.

All of the above mentioned methods for solving ADR equations are multi-point numerical methods, i.e. the derivatives encountered in the equations are approximated using local differential operators. This group of methods uses differential equations in various forms, such as integral forms in FEMs or collocational forms like in FDMs, to derive a system of algebraic equations. On the other hand, the Taylor series based methods use differential equations to evaluate higher order derivatives with a priori information such as positions or slopes of solution functions. These higher-order derivatives are used to represent the approximate local or global solutions. The global semi-analytic Taylor series-based methods are known to be highly restricted for some specific problems [117]. The Taylor series based multiple derivative techniques are widely used to solve higher dimensional elliptic partial differential equations that arise in elasticity problems [118–123]. The primary motivation of using local Taylor series expansions is to minimise required degrees of freedom with the use of recurrence relation obtained from the differential equation itself. The nature of the elliptic PDEs is suitable for the multi-derivative Taylor series-based methods since one has information at all boundaries to be used at interior nodes. Derivation of an effective implicit-explicit Taylor series method has been seen to be a necessity for solving parabolic PDEs by reducing them into nonhomogeneous elliptic systems (see Chapters 6 and 7). The space-time implementation of the Taylor series-based algorithms to the parabolic PDEs leads to a difference between the temporal and spatial order of the local approximations (see [117] and Chapter 6). This limitation can be handled with the use of an effective continuous time integration method or a one-step discrete time integration method (see Chapter 6). In this way, the multi-derivative and multi-centre Taylor series-based

algorithms can produce highly accurate results with optimized degrees of freedom for solving parabolic PDEs (see Chapters 6-7).

1.2 Objective of the Thesis

To overcome the aforementioned drawbacks, the main objective of this thesis is to produce an effective Taylor series-based numerical method for solving stiff differential equations that appear in the modelling of various real-life phenomena. First of all, we begin with constructing the implicit-explicit local differential transform method (IELDTM) for numerical solutions of the stiff IVPs for ODEs. Based on a crucial theorem of real analytic functions, the IELDTM is derived as a stability preserved, high order, adaptive and versatile numerical approach. The second concern of this study is how the IELDTM can be adopted to solve stiff BVPs for ODEs, which have a completely different nature than the IVPs. In this way, we will present the required theoretical explanations and various numerical implementations for stiff BVPs. Our next objective is to derive the IELDTM as a high order, stability preserved, and adaptive time-integrator for the reduced space-time parabolic PDEs. The final objective of this study is to derive the IELDTM for the spatial parts of the (1+1)-, (2+1)-, and (3+1)-dimensional parabolic PDEs. With the consideration of the Chebyshev spectral collocation method (ChSCM), the (1+1)-, (2+1)-, and (3+1)-dimensional parabolic PDEs will be reduced to a system of BVPs with one, two, and three spatial dimensions, respectively. The IELDTM will be constructed over these one-, two-, and three-dimensional meshes as a direction-free, higher-order, and optimal numerical algorithm.

1.3 Hypothesis

A highly stable, high-order, computationally efficient and versatile numerical method is found to solve stiff ODEs and stiff parabolic PDEs up to three spatial dimensions.

1.4 Thesis Outline

This thesis includes eight chapters. The literature review, objectives of the thesis, and hypothesis of the thesis are presented in Chapter 1. Chapter 2 introduces the polynomial approximation theory, stiff differential equations and stiffness of the advection-diffusion equations. The IELDTM for the numerical solutions of the stiff IVPs for ODEs with various real-life applications is derived in Chapter 3. The priori

error analysis, stability analysis, and adaptive construction of the IELDTM are also studied in this chapter. In Chapter 4, the IELDTM is derived for the numerical solutions of the stiff BVPs represented by ODEs. The priori error analysis of the IELDTM is provided, and order conditions of the method are determined in terms of the direction parameter θ. Various types of BVPs, particularly high-order BVPs, singular BVPs, singularly perturbed BVPs, and strongly nonlinear BVPs are solved by the IELDTM to test the performance of our algorithm. In Chapter 5, the IELDTM is derived as an adaptive, stability-preserved and high-order time-integration technique for the two-dimensional nonlinear advection-diffusion (AD) equations. The priori error analysis with order conditions are presented for the time-integration of the reduced matrix ODEs. The IELDTM is produced for the spatial discretizations of the one-dimensional advection-diffusion-reaction (ADR) equations with theoretical analysis in Chapter 6. Various linear and nonlinear ADR problems are numerically solved by the IELDTM and the quantitative observations are discussed in detail. In Chapter 7, the IELDTM is introduced for spatial discretizations of the two and three dimensional AD equations. The (2+1)- and (3+1)-dimensional AD equations are reduced to a system of nonhomogeneous elliptic PDEs using the Chebyshev spectral collocation method (ChSCM) as a continuous-time integrator. The IELDTM is derived over two- and three-dimensional computational meshes as a direction-free and optimal numerical technique. By considering various two- and three-dimensional test problems, both p—refinement and h—refinement performance of the IELDTM are presented quantitatively.

2
FUNDAMENTAL CONCEPTS

In this chapter, we present some fundamental concepts related to this thesis. First of all, some introductory properties of the polynomial approximation theory are presented. Secondly, the nature of the stiff differential equations is explained in detail. Finally, at the end of this chapter, stiffness of the advection-diffusion equations are discussed on a physical and mathematical basis.

2.1 Polynomial Approximations

The importance of approximation theory can be understood by addressing the following famous quote of Bertrand Russel:

"All exact science is dominated by the idea of approximation."

Determination of a quantity in any experiment depends on the sensitivity of the recorded instrument. More or less, the approximation of a given quantity is always inevitable. Similarly, observing the solutions of some problems with the computer programs depends heavily on the compute and memory capacity of machines. All these facts imply that the only option to understand the exact science is the use of approximation theory in an optimum way.

Polynomial approximations are an essential part of the approximation theory. Polynomials are known to have distinct features, such as easy computation of their integrals and derivatives. The most important question here is, can any function be approximated with a polynomial with the desired precision? This vital issue was entirely solved by Karl Weierstrass in 1855 by presenting the well-known Weierstrass approximation theorem.

Theorem 2.1. *(K. Weierstrass) Given $f : [a, b] \to R$ continuous and an arbitrary $\epsilon > 0$, there exists an algebraic polynomial p such that*

$$|f(x) - p(x)| \leq \epsilon, \quad \forall x \in [a, b]. \tag{2.1}$$

The Weierstrass approximation theorem implies that the space of all polynomial functions is dense in the space of all continuous functions. Thus, any continuous function $f(x)$ can be approximated with a polynomial function, and the approximation error can be chosen arbitrarily. The proof of this important theorem was given in several papers, and we suggest the proof given by Jackson in [124].

Even if existence of the best polynomial approximation is theoretically guaranteed, the determination of the best possible polynomial approximations is still under development. The methodological concepts can be divided into two main categories as multi-point polynomial approximations and multi-derivative polynomial approximations. The well-known interpolation theory provides an insight into the multi-point polynomial approximations. On the other hand, the theory of analytic functions and the Taylor series ensures good information about the multi-derivative methods. In the following subsections, we present the necessary procedures of both groups of ideas.

2.1.1 Multi-point Polynomial Approximations

Multi-point polynomial approximation deals with the determination of a polynomial approximation $P_N(x)$ for given set of discrete data $(x_i, f(x_i))$. The following key theorem provides the existence and uniqueness of these polynomials under some conditions [125].

Theorem 2.2. *(Existence and Uniqueness Theorem) Let the nodes $x_i \in I \subset R$ with $i = 0, 1, 2, ..., N$ be distinct.Then there exists a unique polynomial $P_N(x)$ of degree less than or equal to N which satisfies*

$$P_N(x_i) = y_i = f(x_i), \quad i = 0, 1, 2, ..., N \tag{2.2}$$

for a given function $f \in C(I)$.

The proof of this important theorem is straightforward and can be found in [125]. Determination of the polynomial approximations can be utilized by directly calculating the Vandermonde matrix of the given data. In practical problems such as the solution of the differential equations, one often needs to find out nodal forms of these polynomial representations using the Lagrange polynomials.

2.1.1.1 The Lagrange Interpolation

Construction of a polynomial $P_N(x)$ of degree at most N that passes through $N+1$ distinct points (x_0, y_0),(x_1, y_1), ..., (x_N, y_N) has the following form [126]

$$P_N(x) = \sum_{i=0}^{N} L_{N,i}(x) y_i \tag{2.3}$$

where the Lagrange coefficients $L_{N,i}(x)$ are defined as

$$L_{N,i}(x) = \frac{\prod_{j=1, j\neq i}^{N} (x - x_j)}{\prod_{j=1, j\neq i}^{N} (x_i - x_j)}. \tag{2.4}$$

Nodal formulation (2.4) provides several advantages for solving differential equations with finite element-based methods [127]. By increasing polynomial approximation order N in representation (2.4), p-refinement procedure can be used in finite element-based techniques [47, 127]. In the following theorem, we give the error bounds of the Lagrange polynomial interpolation [126].

Theorem 2.3. *Assume that $f \in C^{N+1}[a, b]$ and that $x_0, x_1, ..., x_N \in [a, b]$ are $N+1$ nodes. If $x \in [a, b]$, then*

$$f(x) = P_N(x) + E_N(x), \tag{2.5}$$

where $P_N(x)$ is defined in (2.3) *and the error term has the form*

$$E_N(x) = \frac{(x - x_0)(x - x_1)...(x - x_N) f^{N+1}(c)}{(N+1)!} \tag{2.6}$$

for some value $c = c(x)$ that lies in the interval $[a, b]$.

The proof of this key theorem is based on the Rolle theorem and can be found in the literature [126]. As observed from the error term of the Lagrange interpolation, the error term strongly depends on the distribution of the nodes and the smoothness of the function itself.

A fundamental problem of the polynomial approximation is the determination of optimal node distributions $(x_i)_{i=0}^{i=N}$ to minimize the error function defined in (2.6). To theoretically observe this issue , let us consider the Lebesgue constant [128] related to the Lagrange interpolation operator I_N

$$||f - I_N(f)||_\infty \leq (1 + \Lambda_N) ||f - I_N^*(f)||_\infty \tag{2.7}$$

where I_N^* denotes best possible approximation operator, I_N is the polynomial approximation operator related to the predetermined nodes $(x_i)_{i=0}^{i=N}$ and Λ_N is the

Lebesgue constant defined by

$$\Lambda_N = ||I_N(f)||_\infty = ||P_N(x)||_\infty = ||\sum_{i=0}^{N} |L_{N,i}(x)||_\infty. \tag{2.8}$$

The Lebesgue constant Λ_N determines how far we are from the best polynomial approximation. Thus, minimization of the Lebesgue constant leads to an optimal polynomial approximation. We present the Chebyshev polynomials and the minimax theorem to minimize the Lebesgue constant in the following subsection.

2.1.1.2 The Chebyshev Interpolation

Chebyshev polynomials are related to minimization of the error function $E_N(x)$ for any given function $f(x)$ defined on the interval $[-1,1]$. Definition of the Chebyshev polynomials is given in the following trigonometric form.

Definition 2.1. The Chebyshev polynomials of the first kind defined on the interval $[-1,1]$ have the following trigonometric form

$$T_n(x) = cos(ncos^{-1}(x)), \quad n = 0,1,2,.... \tag{2.9}$$

Table 2.1 Some of the key properties of the Chebyshev polynomials.

Property	**Description**
Recursive relation	$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$
Leading coefficient	$T_N(x) = 2^{N-1}x^N + ...,\ N \geq 1$
Symmetry	$T_{2N}(-x) = T_{2N}(x)$ and $T_{2N+1}(-x) = -T_{2N+1}(x)$
Distinct zeros	Roots of $T_N(x)$ are $x_k = cos\left(\frac{(2k+1)\pi}{2N}\right)$, $k = 0,1,..,N-1$
Extreme values	$\lvert T_N(x)\rvert \leq 1$ for $-1 \leq x \leq 1$

Some of the Chebyshev polynomials are $T_0(x) = 1$, $T_1(x) = x$, $T_2(x) = 2x^2 - 1$, $T_3(x) = 4x^3 - 3x$ and so on. In Table 2.1, some of the key properties of the Chebyshev polynomials are illustrated [126]. As mentioned earlier, invention of the Chebsyhev polynomials are directly related to the minimization of polynomial approximation error (2.6). Before giving this relation explicitly, we must observe the following inequality

$$|E_N(x)| \leq |Q(x)| \frac{\max_{-1\leq x\leq 1} f^{N+1}(x)}{(N+1)!} \tag{2.10}$$

where

$$Q(x) = (x - x_0)(x - x_1)...(x - x_N). \tag{2.11}$$

The following minimax theorem provides a way to select the distinct interpolation nodes related to the Chebyshev polynomials.

Theorem 2.4. *(Minimax Theorem [126]) Among all possible choices for $Q(x)$ in equation* (2.11) *for fixed values of N, and thus among all possible choices for the distinct nodes $(x_i)_{i=0}^{i=N}$ in $[-1,1]$, the polynomial $T(x) = T_{N+1}(x)/2^N$ is the unique choice which has the property*

$$\max_{-1\leq x\leq 1} |T(x)| \leq \max_{-1\leq x\leq 1} |Q(x)| \tag{2.12}$$

and also

$$\max_{-1\leq x\leq 1} |T(x)| = \frac{1}{2^N}. \tag{2.13}$$

Theorem 2.4 implies that the node distribution $(x_i)_{i=0}^{i=N}$ becomes an optimal selection if one chooses them as the roots of the Chebyshev polynomial presented in Table 2.1. This optimal interpolatory structure of the Chebyshev polynomials leads to a strong numerical method for solving differential equations. The Chebyshev spectral collocation method (ChSCM) has a great importance for solving ODEs and PDEs with optimal accuracy and exponential convergence [48, 129, 130]. Unlike the finite element methods, the ChSCM uses the modal forms of the polynomial approximations in terms of the Chebyshev polynomials as follows,

$$f(x) \cong P_N(x) = \sum_{n=0}^{N} c_n T_n(x), \quad x \in [-1,1] \tag{2.14}$$

where the coefficients c_n need to be determined with respect to function $f(x)$. If one aims to produce the polynomial approximation of the known function $f(x)$, determination of the coefficients can be completed with the use of orthogonality relations of the Chebyshev polynomials. The Chebyshev polynomials satisfy the following orthogonality relation [129]

$$\int_{-1}^{1} T_n(x)T_m(x)\frac{1}{\sqrt{1-x^2}}dx = \alpha_n \delta_{nm} \tag{2.15}$$

where

$$\alpha_n = \begin{cases} \pi, & n = 0, \\ \frac{\pi}{2}, & otherwise. \end{cases} \tag{2.16}$$

Thus, for any function $f(x)$ belonging to $L^2[-1,1]$ one can obtain the coefficients c_n as

$$c_n = \frac{1}{\alpha_n}\int_{-1}^{1} f(x)T_n(x)\frac{1}{\sqrt{1-x^2}}dx. \tag{2.17}$$

A discrete form of orthogonality relation (2.15) can be derived with the use of the following Chebyshev-Gauss-Lobatto (CGL) nodes

$$x_n = -cos\left(\frac{n\pi}{N}\right), \quad n = 0, 1, ..., N. \tag{2.18}$$

The use of the CGL quadrature nodes in (2.15) leads to the following discrete orthogonality relation

$$\sum_{i=0}^{N} \beta_i T_n(x_i) T_m(x_i) = \gamma_n \delta_{nm} \tag{2.19}$$

where

$$\gamma_n = \begin{cases} N, & n = 0, N \\ \frac{N}{2}, & otherwise \end{cases} \tag{2.20}$$

and

$$\beta_i = \begin{cases} \frac{1}{2}, & i = 0, N \\ 1, & otherwise. \end{cases} \tag{2.21}$$

The discrete orthogonality relation (2.19) is vital for the derivation of the Chebyshev spectral collocation method, and details of the method will be explained in the following chapters.

2.1.2 Multi-derivative Polynomial Approximations

Multi-derivative polynomial approximations are based on the theory of the real analytic functions and their algebraic properties. We explain the essential properties of the real analytic functions that allow us to produce an effective numerical method. Definition of the real analytic functions is presented below [131].

Definition 2.2. A function f, with domain an open set $U \subseteq R$ and range either the real or complex numbers is said to be real analytic at α if the function f may be represented by a convergent power series on some interval of positive radius centered at α:

$$f(x) = \sum_{j=0}^{\infty} c_j (x - \alpha)^j. \tag{2.22}$$

The function is said to be real analytic on $U \subseteq V$ if it is real analytic at each $\alpha \in V$.

All the required properties of representation (2.22) such as differentiation, integration, and multiplication can be seen in literature [131]. A primer role of the coefficients c_j will come out with the following Taylor approximation theorem [126].

Theorem 2.5. *Assume that $f \in C^{N+1}[a,b]$ and $x_0 \in [a,b]$ is a fixed real number. If $x \in [a,b]$, then*

$$f(x) = P_N(x) + E_N(x), \tag{2.23}$$

where

$$f(x) \cong P_N(x) = \sum_{k=0}^{N} \frac{f^{(k)}(x_0)}{k!}(x-x_0)^k, \tag{2.24}$$

and the error term has the following form

$$E_N(x) = \frac{f^{N+1}(c)}{(N+1)!}(x-x_0)^{N+1}. \tag{2.25}$$

The proof of this theorem can be found in the literature [132]. An essential property of the Taylor polynomial approximation is that the polynomial coefficients consist of the higher-order derivatives of the function itself. Thus, a multi-derivative polynomial approximation of a given continuous function is given in equation (2.24). The Taylor polynomial approximations can be used for different purposes such as numerical integration, numerical differentiation, or numerical solutions of differential equations [126].

The Taylor polynomial approximation provides a direct semi-analytic method for solving differential equations [90]. Taylor series-based methods for solving differential equations need some algebraic properties between the Taylor coefficients. The well-known definition of the Taylor coefficients can be given by the differential transformation idea [27]. Let us define the differential transform of a given function $f(x)$ at $x = x_0$ as

$$F(k) = \frac{f^{(k)}(x_0)}{k!} \tag{2.26}$$

where $k = 0, 1, 2, ..., N$. With the use of the differential transformation notation, the Taylor polynomial approximation of a given function $f(x)$ can be stated as

$$f(x) \cong P_N(x) = \sum_{k=0}^{N} F(k)(x-x_0)^k. \tag{2.27}$$

There exist various theorems and propositions related to the algebraic properties of the differential transformation, which are presented in Appendix A. For the sake of simplicity, the following most related proposition is selected here to explain further analysis.

Theorem 2.6. *Let $y(x)$ be analytic in domain S and $x_i \in S$. If $D(y_i(x), k) = Y_i(k)$ and*

$z_i(x) = \frac{d^m y_i(x)}{dx^m}$ *for* $m \in N$*, then the following equality holds*

$$Z_i(k) = D(z_i(x), k) = (k+1)(k+2)...(k+m)Y_i(k+m). \tag{2.28}$$

Proof of this theorem can be seen in Appendix A. Theorem 2.6 implies that the differential transform of the derivatives of any given function can be expressed in terms of the differential transform of the function itself. Let us consider the following linear second order initial value problem

$$y'' + py' + qy = 0, \quad y(0) = y_0 \quad and \quad y'(0) = y_0^* \tag{2.29}$$

where $x \in [0, X]$. According to Theorem 2.6, taking the differential transform of equation (2.29) at $x = 0$ yields

$$Y_0(k+2) = \frac{1}{(k+1)(k+2)}\left[(k+1)pY_0(k+1) + qY_0(k)\right] \tag{2.30}$$

where $k = 0, 1, ..., N-2$, $Y_0(k) = \frac{y^{(k)}(0)}{k!}$, $Y_0(0) = y_0$ and $Y_0(1) = y_0^*$. Consider the following global truncated representation

$$y(x) \cong \sum_{k=0}^{K} Y_0(k) x^k \tag{2.31}$$

where K is the approximation order. Thus, one can evaluate all required transform values $Y_0(k)$ in terms of $Y_0(0)$ and $Y_0(1)$ with the use of recursive relation (2.30). Inserting the obtained coefficient values into equation (2.31), one obtains the semi-analytic series solution of the IVP (2.29). Whether the global series approximation provides a convergent solution is a vital issue. For most of the differential equations, this approach fails to produce a convergent solutions (see Chapter 4).

An explicit numerical method based on the Taylor series can be derived, and it is known as the local differential transform method, or explicit Taylor series method [21, 27]. The following theorem provides a way of constructing a Taylor series-based numerical method [131].

Theorem 2.7. *Assume that the power series*

$$\sum_{k=0}^{\infty} Y_i(k)(x - x_i)^k \tag{2.32}$$

has the radius of convergence ρ*. For any point* x_j *satisfying* $\left|x_i - x_j\right| < \rho$*, then the power*

series

$$\sum_{k=0}^{\infty} Y_j(k)\left(x-x_j\right)^k \tag{2.33}$$

has the radius of convergence at least $\tau = \rho - \left|x_i - x_j\right|$ *and on the interval* $(x_j-\tau,\ x_j+\tau)$ *it converges to* $y(x)$.

According to Theorem 2.7, one can obtain a neighbour representation about $x = x_1$ with the use of the previous representation constructed over $x = x_0$. This process is also known as analytic continuation [22]. This numerical process has an explicit nature. It will be shown in later sections that this approach fails to preserve numerical stability over computational domains for solving stiff differential equations. Based on a similar idea, a generalized approach called as implicit-explicit local differential transform method (IELDTM) will be derived to eliminate existing drawbacks of the explicit Taylor series-based methods (see Chapters 3-4).

A crucial observation can be done over the natural differences between the multi-point and multi-derivative polynomial approximations. If one uses a multi-point polynomial approximation for the solution of a differential equation such as finite element-based methods [127], the local degrees of freedom of the representation increases with the increase of polynomial degrees. On the other hand, recursive relations like (2.30) provides an advantage for multi-derivative methods. Thus, the order-refinement procedure of the multi-derivative methods does not affect the local degrees of freedom of the polynomial representation in one-dimension. Additionally, in two- and three-dimensions, recursive relations like (2.30) provides optimization in degrees of freedom of the local two-dimensional and three-dimensional representations, as will be discussed in Chapter 7. The crucial difference between multi-point and multi-derivative approaches can be realized evidently for spatial discretizations of the partial differential equations (see Chapters 6-7).

2.2 Stiff Differential Equations

At the beginning of the 1950s, Curtiss & Hirschfelder [2] realized that there was a class of ordinary differential equations (ODEs), which is currently known as stiff equations, which presented a serious challenge to the existing numerical methods. Since then, many studies have been carried out in designing numerical methods for solving stiff equations. In particular, the text of Hairer and Wanner [5] presents a combination of theories and methods produced so far and has an important place in this field.

One of the significant difficulties associated with the study of stiff differential systems is

that a good mathematical definition of the concept of stiffness does not exist. However, definition of Lambert [133] may be the most acceptable one to understand the behaviours of the stiff differential systems. According to Lambert, the definition is "if a numerical method with a finite region of absolute stability, applied to a system with any initial conditions, is forced to use in a certain interval of integration a step-length which is excessively small in relation to the smoothness of the exact solution in that interval, then the system is said to be stiff in that interval". Another considerable definition was given by Curtiss and Hirschfelder [2] as "stiff equations are equations where certain implicit methods, and in particular backward-differentiation formulae (BDFs), perform better, usually tremendously better, than explicit ones".

In the case of linear differential equations with constant coefficients we can gain an idea of what the problem of stiffness is. Consider the linear constant coefficient nonhomogeneous system

$$y' = Ay + f(t) \tag{2.34}$$

where $y,\ f \in R^n$ and A is a constant, diagonalizable $n \times n$ matrix with eigenvalues $\lambda_i \in C$ and corresponding eigenvectors $c_i \in C^n$ for $i = 1, 2, \ldots,\ n$. General solution of (2.34) takes the form

$$y(t) = \sum_{i=1}^{n} \omega_i \exp(\lambda_i t) c_i + g(t) \tag{2.35}$$

where ω_i are arbitrary constants and $g(t)$ is a particular integral. Suppose that $Re(\lambda_i) < 0$ for all $i = 1, 2, \ldots,\ n$. In such cases, it is appropriate to call $\sum_{i=1}^{n} \omega_i \exp(\lambda_i t) c_i$ as the transient solution and $g(t)$ as the steady-state solution. If $|Re(\lambda_i)|$ is large, we reach a fast transient, if $|Re(\lambda_i)|$ is small we reach a slow transient. Defining the bounds as $\overline{\lambda}$ and $\overline{\overline{\lambda}}$ given by

$$\left|Re\left(\overline{\lambda}\right)\right| \leq \left|Re\left(\lambda_i\right)\right| \leq \left|Re\left(\overline{\overline{\lambda}}\right)\right|, \quad i = 1, 2, \ldots, n \tag{2.36}$$

so that we define the stiffness ratio as [133]

$$\mu = \frac{\left|Re\left(\overline{\overline{\lambda}}\right)\right|}{\left|Re(\overline{\lambda})\right|}. \tag{2.37}$$

By considering the literature [133], the following conclusions can be expressed

* A linear constant coefficient system is stiff if all of its eigenvalues have negative real parts and the stiffness ratio is large.

* Stiffness occurs when stability requirements, rather than those of accuracy, constrain

the steplength.

* Stiffness occurs when some components of the solution decay much more rapidly than others.

The stiff differential equations are naturally occurs in various fields of science such as the reaction kinetics [31], electric circuits [5], epidemiological modelling (see Chapter 3), fluid dynamics [134] and so on. Stiff behaviours are modelled by not only ODEs [5, 133] but also PDEs [6, 7] and arise for instance in advection dominated fluid flow problems. We discuss the stiff dynamics of the advection-diffusion equations in the following section.

2.3 Stiffness of the Advection-Diffusion Equations

Let us consider the following one-dimensional advection-diffusion equation [107],

$$u_t + Vu_x = Du_{xx}, \quad x \in [0,1] \quad and \quad t > 0 \tag{2.38}$$

with the initial condition

$$u(x,0) = g(x), \quad 0 \leq x \leq 1 \tag{2.39}$$

and the homogeneous Dirichlet boundary conditions

$$u(0,t) = u(1,t) = 0, \quad t > 0 \tag{2.40}$$

where V is the advection constant, D is the diffusion constant, $g(x)$ is a sufficiently smooth function. The subscripts x and t represent the differentiations with respect to space and time, respectively. Derivation of the advection-diffusion equation (2.38) with the consideration of related physical laws can be found in [135].

Many of the transport phenomena can be described by two fundamental mechanisms: advection and diffusion. Advection is described by the transport of a substance by the bulk motion of a fluid. Physical properties carried by the advected substance are called as conserved ones, such as energy or heat. Advection equations are also known as conservation equations in the presence of only bulk motion of a fluid and yield hyperbolic partial differential equation by simplifying equation (2.38) with $D = 0$. Another fundamental transport mechanism is diffusion, which is the transport of the properties thought molecular-level interactions. These molecular-level interactions lead to net movement of any quantity such as heat or energy from a region of higher concentration to a region of lower concentration. If the only molecular

interactions yield the transport of a substance, then the physical system is described by the diffusion equation by simplifying equation (2.38) with $V = 0$. The two fundamental transport mechanisms, advection and diffusion, contribute to the process when both bulk motion and molecular-level interactions co-occur as represented in advection-diffusion equation (2.38). Contributions of physical mechanisms play essential roles to observe the balanced advection-diffusion, advection dominated, or diffusion dominated processes.

Here we quantitatively show that how the advection-dominance leads to stiffness by evaluating the stiffness ratios. For the sake of simplicity, let us discretize the spatial part of advection-diffusion equation (2.38) with the central finite difference formulae as

$$(u_{xx})_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} \tag{2.41}$$

and

$$(u_x)_i = \frac{u_{i+1} - u_{i-1}}{2h} \tag{2.42}$$

where $i = 1, 2, ..., N-1$, $h = \frac{1}{N}$ and $x_i = ih$. By considering homogeneous Dirichlet boundary conditions (2.40) and initial condition (2.39), equation (2.38) reduces to following system of ODEs

$$\frac{d\boldsymbol{u}}{dt} = A\boldsymbol{u}, \quad \boldsymbol{u}(0) = \boldsymbol{u}_0 \tag{2.43}$$

where A is an $(N-1) \times (N-1)$ matrix having the form

$$A = \begin{pmatrix} -2\frac{D}{h^2} & \left(\frac{D}{h^2} + \frac{V}{2h}\right) & & & \\ \left(\frac{D}{h^2} - \frac{V}{2h}\right) & -2\frac{D}{h^2} & \left(\frac{D}{h^2} + \frac{V}{2h}\right) & & \\ & & \ddots & & \\ & & \left(\frac{D}{h^2} - \frac{V}{2h}\right) & -2\frac{D}{h^2} & \left(\frac{D}{h^2} + \frac{V}{2h}\right) \\ & & & \left(\frac{D}{h^2} - \frac{V}{2h}\right) & -2\frac{D}{h^2} \end{pmatrix}. \tag{2.44}$$

The dominance of advection implies that $D \ll V$. In Figure 2.1, the stiffness ratio of the ODE system (2.43) is presented for various values of the spatial increment h with $V = 1$ and $D = 0.001$. As observed from the figure, the stiffness ratio satisfies approximately the power law $\mu = \frac{D}{(\pi h)^2}$. A similar power law is also observed in literature [136] for the pure diffusion equation. Even the power law implies that the decreasing values of the diffusion constant D lead to lower stiffness, the Peclet number $Pe = \frac{Vh}{D}$ must be low enough to ensure physically acceptable results. Thus, because the term $1/h^2$ occurred in the power law, the low diffusion constant increases the stiffness considerably. Thus, advection dominated processes lead to stiff dynamical systems to be integrated with suitable numerical methods.

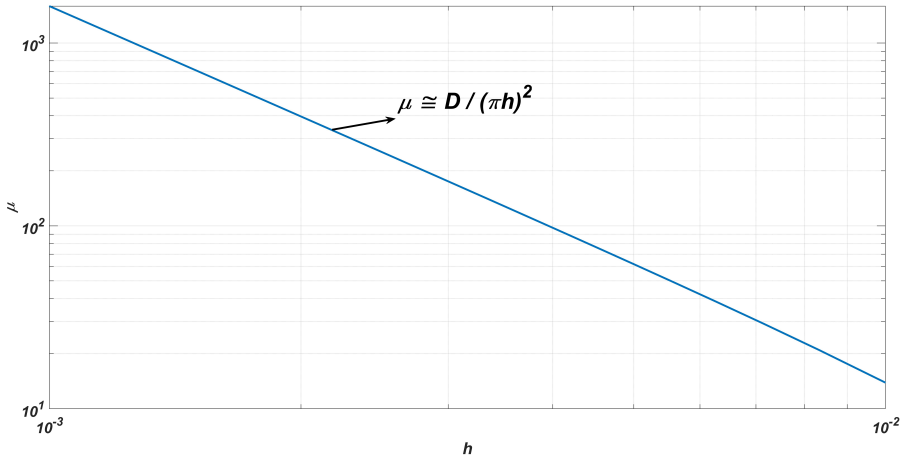

Figure 2.1 The stiffness ratio versus spatial increment for the advection-diffusion equation with the parameter values $V = 1$ and $D = 0.01$.

Two primary necessities exist in dealing with stiff differential equations that occur in real-life applications. One of them is to have an effective stiff time-integrator, and another one is to have an effective stiff boundary value problem solver. First of all, a time integration technique must be stability preserved, high order, computationally effective, and versatile to accurately capture the stiff behaviours that occur in the equations like (2.34). In the following Chapters 3-5, an implicit-explicit local differential transform method (IELDTM) is derived for solving these types of serious IVPs by overcoming the existing drawbacks of the Taylor series-based algorithms. The second main issue is how stiffness occurs in boundary value problems and what conditions must be satisfied by numerical methods to solve stiff BVPs. In Chapter 4, we will re-derive our method for the BVPs of ODEs in various orders, including stiff ones. In Chapters 6-7, the IELDTM is produced for the solutions of stiff parabolic one-dimensional, two-dimensional and three-dimensional PDEs.

3
THE IELDTM FOR STIFF IVPs

The stiff IVPs naturally occur in various fields of science and have extremely challenging structure for many existing numerical methods [5]. Numerical methods for solving stiff problems can be divided into two categories: explicit methods and implicit methods. Even if the explicit methods provide computational simplicity, their stability properties are not always satisfactory for especially stiff IVPs [3]. The implicit methods are known to be best options for stiff IVPs since the first invention of such problems by Curtiss and Hirschfelder [2]. However, the implicit algorithms are computationally costly especially for solving large nonlinear systems. Thus, to eliminate this disadvantage to some extent, an implicit algorithm must be optimized in terms of the local degrees of freedom of the algebraic equation system.

This chapter is devoted to the derivation of a Taylor series based numerical method for solving stiff IVPs for ODEs. The existing Taylor series based methods are the semi-analytic differential transform method (DTM) [137, 138] and the numerical LDTM [24, 27, 139]. The semi-analytical DTM lacks convergence for stiff differential equations whose exact solutions are not analytical. Similarly, the explicit nature of the LDTM yields instability for stiff IVPs for ODEs. By eliminating the existing drawbacks of the DTM based methods, the implicit-explicit local differential transform method (IELDTM) is here proven to provide improved stability properties for stiff problems.

This chapter begins with the derivation of the IELDTM for IVPs. The second phase is the priori error analysis of the IELDTM with the determination of the order conditions with respect to the local approximation order and the direction parameter. The stability analysis of the IELDTM is performed over a scalar test equation and linear equation systems. The A-stable and L-stable cases of the IELDTM are derived, and the stability regions are illustrated for various values of the direction parameter θ. The adaptivity equations are derived for stiff and non-stiff problems to minimize the computational cost of the present algorithm. In the last part of this chapter, the numerical experiments are performed over various real-life problems, including the SEIR-epidemiological model, the Duffing equation, the Van der Pol equation, and the

Robertson chemical system.

3.1 Implicit-Explicit Local Differential Transform Method

In this section, the IELDTM, which is a general multi-derivative time integration method, is derived for a first order IVP. How the IELDTM generalizes the existing differential transform based methods is shown here in details.

Consider the following IVP,

$$\boldsymbol{x}'(t) = G(\boldsymbol{x}(t), t), t > 0, \boldsymbol{x}(0) = \boldsymbol{C}, \tag{3.1}$$

where $\boldsymbol{x}(t) \in R^m$, $\boldsymbol{C} \in R^m$, $G : R^m \times R \to R^m$ and the function $G(\boldsymbol{x}(t), t)$ satisfies the Lipschitz condition with constant L. We assume that the IVP (3.1) is well-posed and the exact solution $\boldsymbol{x}(t)$ is analytic on the considered domain. Let us divide the interval $\left[0, t_f\right]$ into at most N time elements with $\Delta t_i = t_{i+1} - t_i$ and the partition of the interval as $\omega = \{0 = t_0 < t_1 < \ldots < t_{N^*} = t_f\}$ where $N^* \leq N$. Let us consider the convergent Taylor series representation of the function $x(t)$ about $t = t_i$ as

$$\boldsymbol{x}_i(t) = \sum_{k=0}^{K} X_i(k)(t - t_i)^k + O\left((t - t_i)^{K+1}\right), \quad t_i - \rho^i \leq t \leq t_i + \rho^i \tag{3.2}$$

where $i = 0, 1, \ldots, N^*, X_i(k) = \frac{x^{(k)}(t_i)}{k!}$ is the local differential transform of the function $x(t)$ about $t = t_i$ and ρ^i is the radius of convergence of the representation. From our assumption, the function $\boldsymbol{x}_i(t)$ is analytic and has the radius of convergence satisfying $\rho^i > \Delta t_i$. Thus, according to Theorem (2.7), the function $\boldsymbol{x}(t)$ has also a convergent Taylor series representation about $t = t_{i+1}$ with the radius of convergence at least $\rho^{i+1} = \rho^i - \Delta t_i$ and on the interval of convergence $\left(t_{i+1} - \rho^{i+1}, t_{i+1} + \rho^{i+1}\right)$. Then, the function $\boldsymbol{x}(t)$ can be written as

$$\boldsymbol{x}_{i+1}(t) = \sum_{k=0}^{K} X_{i+1}(k)(t - t_{i+1})^k + O\left((t - t_{i+1})^{K+1}\right), \; t_{i+1} - \rho^{i+1} \leq t \leq t_{i+1} + \rho^{i+1}. \tag{3.3}$$

Assuming $\rho^i > \Delta t_i$ and $\rho^{i+1} > \Delta t_i$, we can conclude the following conclusions;

* because of the convergence assumptions, two convergent representations in equations (3.2)-(3.3) need to give the same numerical result at any point of the interval $[t_i, t_{i+1}]$.

* any point in the interval $[t_i, t_{i+1}]$ can be written as $t^* = t_i + (1 - \theta)\Delta t_i$, where $0 \leq \theta \leq 1$.

* the IELDTM can be produced with the use of continuity conditions of the solutions at such interior points.

Convergent solutions $\boldsymbol{x}_i(t)$ and $\boldsymbol{x}_{i+1}(t)$ at the interior points $t^* = t_i + (1-\theta)\Delta t_i$ need to satisfy the following continuity relation

$$\boldsymbol{x}_{i+1}(t_i + (1-\theta)\Delta t_i) = \boldsymbol{x}_i(t_i + (1-\theta)\Delta t_i). \tag{3.4}$$

With the use of representations (3.2) and (3.3), the following equality needs to be hold

$$\sum_{k=0}^{K} X_{i+1}(k)(-\theta\Delta t_i)^k = \sum_{k=0}^{K} X_i(k)((1-\theta)\Delta t_i)^k + O\left((\Delta t_i)^{K+1}, \theta\right) \tag{3.5}$$

where $O\left((\Delta t_i)^{K+1}, \theta\right)$ represents the dependency of the local truncation error to time increment, transformation order and direction parameter. It is time to determine the algebraic relations between $X_i(k)$ and $X_i(0)$ for all $k = 1, 2, \ldots, K$. Taking differential transform of equation (3.1), the following relation

$$X_i(k+1) = \frac{1}{k+1} F(X_i(k), t_i) \tag{3.6}$$

is obtained where $i = 0, 1, \ldots, N^*$, $k = 0, 1, \ldots, K-1$ and F is the transformed form of the function $G(\boldsymbol{x}, t)$ at the local point $t = t_i$. Depending on the function $G(\boldsymbol{x}, t)$, the $F(X_i(k), t_i)$ can be obtained by imposing related propositions proposed in Appendix A. It is obvious that $X_i(k)$ can be written in terms of $X_i(0)$ for each i and k. Thus, using the algebraic relations between $X_i(k)$ and $X_i(0)$ and putting into equation (3.5), the implicit-explicit equation

$$g(X_{i+1}(0), \theta, \Delta t_i) = h(X_i(0), \theta, \Delta t_i) \tag{3.7}$$

is found. Here the functions g and h are obtained from the right and left hand sides of equation (3.5). In the right side of equation (3.7), all parameters $X_i(0)$, θ and Δt_i are known because of the previous step. The left side of equation (3.7) depends explicit/implicitly on the unknown $X_{i+1}(0)$. Thus, equation (3.7) is either linear or nonlinear equation(s) of $X_{i+1}(0)$ depending on the function $G(\boldsymbol{x}, t)$.

Then setting the initial condition $x_0(0) = X_0(0) = \boldsymbol{C}$ and predetermining adaptive parameters θ and Δt_i, equation (3.7) can be solved either directly or numerically for each step. Note that we consider the following global solutions

$$\boldsymbol{x}(t_i) = X_i(0) \tag{3.8}$$

for each $i = 0, 1, \ldots, N^*$. Therefore, the following remarks should be underlined because of its importance for the rest of the study:

* $\theta = 0$ with $N^* = 1$ leads to the classical semi-analytic DTM [90].

* $\theta = 0$ with arbitrary number of N^* leads to the explicit forward scheme and known as local or multi-step differential transform method [27, 140, 141].

* The rest of the selections of the parameter θ, i.e. $\theta \neq 0$, gives rise to the currently derived implicit schemes. To the best of the author's knowledge, this derivation has not been studied in the DTM.

* $\theta = 0.5$ yields the produced implicit central scheme with arbitrary order here, which gives us a stable and order preserved method for both stiff and non-stiff cases.

* $\theta = 1$ leads to a derivation of the implicit backward scheme which is both stable and order preserved method for both stiff and non-stiff cases.

* Stability preserving schemes can be obtained by considering $\theta \geq 0.5$.

* The classical θ-method including the Crank-Nicolson method is a special case of the IELDTM; $\theta = 0.5$ and $K = 1$.

In the following section, the priori error analysis of the derived numerical technique is done by illustrating the order conditions.

3.2 Error Analysis

Consider the following initial value problem,

$$\boldsymbol{x}'(t) = G(\boldsymbol{x}(t), t), t > 0, \boldsymbol{x}(0) = \boldsymbol{C}, \tag{3.9}$$

where $\boldsymbol{x}(t) \in R^m$, $\boldsymbol{C} \in R^m$ and $G : R^m \times R \to R^m$. Considering the procedure explained in the previous section and assuming the fixed time increment $\Delta t_n = \Delta t$, the present scheme leads to,

$$X_n(k+1) = \frac{1}{k+1}\left[F(X_n(k), t_n)\right], k = 0, 1, 2, \ldots, K-1, n = 0, 1, 2, \ldots, N, \tag{3.10}$$

and

$$\boldsymbol{x}(t_{n+1}) \cong \boldsymbol{x}_{n+1} = X_{n+1}(0) = \sum_{k=0}^{K} X_n(k)((1-\theta)\Delta t)^k - \sum_{k=1}^{K} X_{n+1}(k)(-\theta\Delta t)^k \tag{3.11}$$

where X_n and F are the transformed forms of the functions $\boldsymbol{x}_n(t)$ and $G(\boldsymbol{x}_n(t),t)$, respectively. The exact solution of (3.9) at point $t = t_{n+1}$ can be expressed in the Taylor expansion form as

$$\boldsymbol{x}(t_{n+1}) = \sum_{k=0}^{K} X_n(k)((1-\theta)\Delta t)^k - \sum_{k=1}^{K} X_{n+1}(k)(-\theta\Delta t)^k + \Delta t\rho_n \tag{3.12}$$

where $n = 0,1,2,\ldots,N-1$. Local truncation error can be obtained from the residuals of the Taylor expansions as follows

$$\rho_n = \left[(1-\theta)^{K+1} X_n(K+1) - (-\theta)^{K+1} X_{n+1}(K+1)\right]\Delta t^K + O(\Delta t^{K+1}). \tag{3.13}$$

Now using the expansion $X_{n+1}(K+1) = X_n(K+1) + \Delta t(K+2)X_n(K+2) + O(\Delta t^2)$ leads to

$$\rho_n = \left[(1-\theta)^{K+1} X_n(K+1) - (-\theta)^{K+1} X_n(K+1)\right]\Delta t^K + O(\Delta t^{K+1}). \tag{3.14}$$

Thus, the present scheme is of order $K+1$ if $\theta = 1/2$ and K is odd and is of order K for other selections of parameter θ and K. To have convergent numerical scheme, we also need to have bounded global error of the scheme. For further analysis about global discretization error, assume that the problem is linear, $G(\boldsymbol{x}(t),t) = A\boldsymbol{x}(t) + B(t)$ and let $\varepsilon_n = \boldsymbol{x}(t_n) - X_n(0)$ for $n = 0,1,\ldots,N$. The present method for this equation yields the following equation,

$$\sum_{k=0}^{K} X_{n+1}(k)(-\theta\Delta t)^k = \sum_{k=0}^{K} X_n(k)((1-\theta)\Delta t)^k. \tag{3.15}$$

With the use of recursive relation (3.6) with $G(\boldsymbol{x}(t),t) = A\boldsymbol{x}(t) + B(t)$, the general term $X_n(k)$ can be stated as

$$X_n(k) = \frac{1}{k!}A^k X_n(0) + \sum_{p=0}^{k-1} A^{k-p-1}\frac{p!}{k!}F(p,t_n) \tag{3.16}$$

where $k = 1,2,\ldots,K$ and $F(p,t_n) = \frac{d^p B(t)}{dt^p}\Big|_{t=t_n}$ is the local differential transform of the function $B(t)$. Then, substituting (3.16) into (3.15) gives the following recursive

form

$$\sum_{k=0}^{K}\frac{1}{k!}A^k(-\theta\Delta t)^k X_{n+1}(0)=\sum_{k=0}^{K}\frac{1}{k!}A^k((1-\theta)\Delta t)^k X_n(0)+$$
$$\sum_{k=1}^{K}\sum_{p=0}^{k-1}A^{k-p-1}\frac{p!}{k!}\left[F(p,t_n)((1-\theta)\Delta t)^k-F(p,t_{n+1})(-\theta\Delta t)^k\right]+ \tag{3.17}$$
$$[F(0,t_n)-F(0,t_{n+1})].$$

Let us define the following stability functions

$$R_1(A,\Delta t,\theta)=I+(1-\theta)\Delta tA+((1-\theta)\Delta t)^2A^2+\cdots+((1-\theta)\Delta t)^KA^K, \tag{3.18}$$

and

$$R_2(A,\Delta t,\theta)=I-\theta\Delta tA+(\theta\Delta t)^2A^2+\cdots+(-\theta\Delta t)^KA^K. \tag{3.19}$$

With the use of (3.16) into exact expansion (3.12) and subtraction of (3.17) from (3.12) gives the following relation

$$\varepsilon_{n+1}=R(A,\Delta t,\theta)\,\varepsilon_n+\delta_n \tag{3.20}$$

where $\delta_n=(R_2)^{-1}\Delta t\rho_n$ and $R(A,\Delta t,\theta)=(R_2)^{-1}R_1$. Here the important fact is that the scheme is of order p if $||\delta_n||=O\left(\Delta t^{p+1}\right)$. To bound global discretization error in terms of initial error ε_0 and local discretization error, the recursive relation (3.20) becomes,

$$\varepsilon_n=R^n\varepsilon_0+\sum_{p=0}^{n-1}R^{n-p-1}\delta_p. \tag{3.21}$$

Stability of the scheme is also depending on the bound of the norm estimate $||R^n||$ for all $n\Delta t\le t_f$, where t_f is the final time. So, assuming the following stability criteria,

$$||R^n||\le S,\quad \text{for}\quad \text{all}\quad n\ge 0\quad \text{and}\quad n\Delta t\le t_f, \tag{3.22}$$

and

$$\left|\left|(R_2)^{-1}\right|\right|\le P. \tag{3.23}$$

The bound of the global error norm becomes

$$\|\varepsilon_n\|\le S\,||\varepsilon_0||+S\sum_{p=0}^{n-1}\left|\left|\delta_p\right|\right|. \tag{3.24}$$

Then finally, using the definition of local discretization error $\left|\left|\delta_p\right|\right| \leq P\omega\Delta t^{K+1}$ for all p, ω is defined as

$$\omega = \begin{cases} \left| \frac{1}{(K+2)!} \underbrace{max}_{t\in[0,t_f]} \left(\left| \frac{dx^{K+2}\boldsymbol{x}}{dt^{K+2}} \right| \right) \left(\left(\frac{1}{2}\right)^{K+1} (K+1) \right) \right|, \text{ if } \theta = 1/2 \text{ and } K \text{ is odd} \\ \left| \frac{1}{(K+1)!} \underbrace{max}_{t\in[0,t_f]} \left(\left| \frac{d^{K+1}\boldsymbol{x}}{dt^{K+1}} \right| \right) \left[(1-\theta)^K - (-\theta)^K \right] \right|, \text{ if } \theta \neq \frac{1}{2} \text{ and } K \text{ is arbitrary.} \end{cases} \tag{3.25}$$

Assumption of $X_0(0) = \boldsymbol{x}(0)$ leads to the following error norm estimates

$$||\boldsymbol{x}(t_n) - X_n(0)|| \leq \begin{cases} \omega^* \Delta t^{K+1}, \text{ if } \theta = 1/2 \text{ and } K \text{ is odd} \\ \omega^* \Delta t^K, \text{ otherwise} \end{cases} \tag{3.26}$$

where $\omega^* = SP\omega t_n$ for all $n = 1, 2\ldots, N$. Thus, whenever the exact solution is smooth and stability criteria (3.22)-(3.23) are satisfied, then the present IELDTM converges to the exact solution with order $K+1$ or K depending on the selection of the parameters θ and K.

3.3 Stability Analysis

In order to visualize the behaviour of the currently presented IELDTM, we have analysed the stability properties over the scalar test equation and the system of linear equations.

3.3.1 Scalar Test Equation

At first, let us consider the following scalar-complex test equation

$$x'(t) = \lambda x(t), \quad x(0) = x_0 \tag{3.27}$$

with $\lambda \in C$. Application of the IELDTM, with order K, to this test problem gives

$$x_{n+1} = R(\Delta t\lambda)\, x_n, \tag{3.28}$$

$$R(z) = \frac{1 + (1-\theta)z + \frac{((1-\theta)z)^2}{2!} + \cdots + \frac{((1-\theta)z)^K}{K!}}{1 - \theta z + \frac{(\theta z)^2}{2!} + \cdots + \frac{(-\theta z)^K}{K!}} \cong e^z, \quad z \to 0, \tag{3.29}$$

where $x_n = X_n(0)$ and $z = \Delta t\lambda$. If we perturb the initial value x_0 to $\tilde{x}_0$, we get the recursion $\tilde{x}_{n+1} = R(\Delta t\lambda)\tilde{x}_n$ then the difference $\tilde{x}_n - x_n$ leads to the following stability

requirement

$$\tilde{x}_n - x_n = R(\Delta t\lambda)^n(\tilde{x}_0 - x_0) \tag{3.30}$$

and therefore, stability region of the present method is the set

$$S = \{z = \Delta t\lambda \in C : |R(z)| \leq 1 \tag{3.31}$$

where $S \subset C$. Notice that, if a numerical method has the stability set S that includes C^-, then the method is called as A—stable [4]. Additionally, if a numerical method is A—stable and has the stability function $R(z)$ that satisfies $\lim_{z\to\infty} R(z) = 0$, then the method is called as L—stable.

To analyse the current approach, we need to remind the maximum modulus theorem [4]. In the light of this theorem, we have concluded that:

* The central schemes, $\theta = 1/2$ of order $K = 1,\ 2,\ 3,\ 4$, are A-stable.

* The backward schemes, $\theta = 1$ of order $K = 1,\ 2$ are A-stable as well as L-stable.

* The schemes, with $\theta < 1/2$ of any order including LDTM ($\theta = 0$), have conditional stability.

* For higher order implicit schemes, $\theta \geq 1/2$, we have almost A-stability that means just the stability condition fails in a very little region in C^-.

The stability regions of the currently produced method, the IELDTM, have been illustrated with $\theta = 0$, $\theta = 1/2$ and $\theta = 1$ for various orders, respectively, as seen in Figures 1-3. As realized in the figures, stability properties of the central and backward cases are stronger than that of the forward cases.

3.3.2 Stability for Linear Systems

Consider the following linear homogeneous system

$$\boldsymbol{x}^{'}(t) = A\boldsymbol{x}(t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{3.32}$$

with $A \in R^{m\times m}$ and $\boldsymbol{x}_0 \in R^m$. Application of the IELDTM yields the following localized differential transformation coefficients

$$X_n(k) = \frac{1}{k!}A^k X_n(0). \tag{3.33}$$

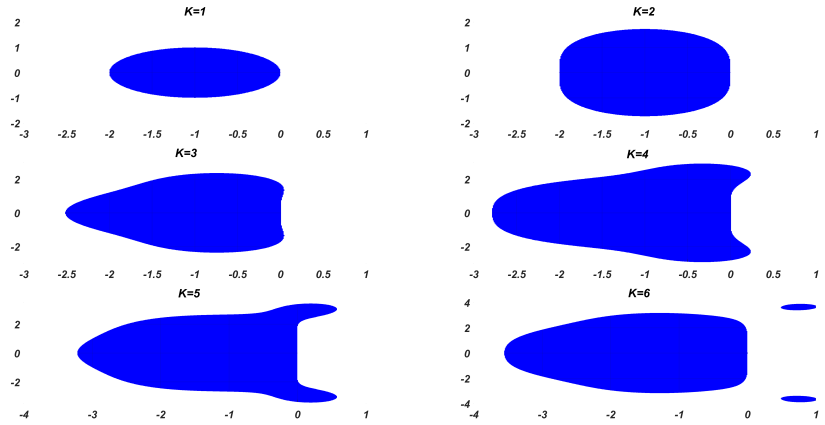

Figure 3.1 Stability region (blue) of the forward schemes, $\theta = 0$, for various orders on the complex plane.

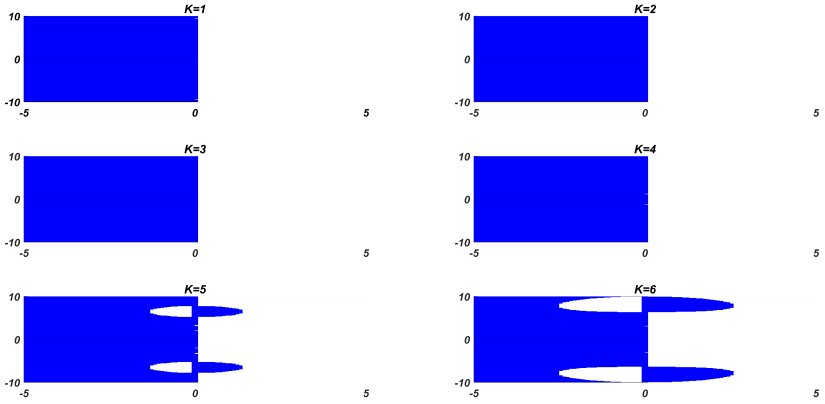

Figure 3.2 Stability region (blue) of the central schemes, $\theta = 0.5$, for various orders on the complex plane.

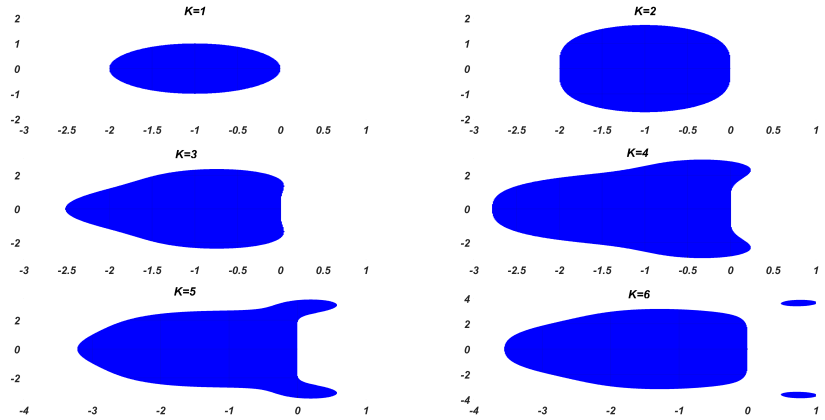

Figure 3.3 Stability region (blue) of the backward schemes, $\theta = 1$, for various orders on the complex plane.

Writing these coefficients into (3.5) leads to

$$\sum_{k=0}^{K}\left[\frac{1}{k!}A^k X_{n+1}(0)\right](-\theta\Delta t_i)^k = \sum_{k=0}^{K}\left[\frac{1}{k!}A^k X_n(0)\right]((1-\theta)\Delta t_i)^k \tag{3.34}$$

where X_n is the transformed form of $\boldsymbol{x}(t)$. We define the stability functions as

$$R(\theta,\Delta t,A) = R_1(\theta,\Delta t,A)R_2(\theta,\Delta t,A) \tag{3.35}$$

where

$$R_1(\theta,\Delta t,A) = \left[I - \theta\Delta tA + \frac{(-\theta\Delta tA)^2}{2} + \cdots + \frac{(-\theta\Delta tA)^K}{K!}\right]^{-1}$$

$$R_2(\theta,\Delta t,A) = I + (1-\theta)\Delta tA + \frac{((1-\theta)\Delta tA)^2}{2} + \cdots + \frac{((1-\theta)\Delta tA)^K}{K!}.$$

Recalling $\boldsymbol{x}(t_{n+1}) = X_{n+1}(0)$ and $\boldsymbol{x}(t_n) = X_n(0)$, then (3.34) becomes

$$\boldsymbol{x}(t_{n+1}) = R(\theta,\Delta tA)\boldsymbol{x}(t_n). \tag{3.36}$$

For a perturbed system with initial condition $\overline{\boldsymbol{x}}_0$, we get the same formula with (3.36). Subtraction of (3.36) from the corresponding perturbed system leads to

$$\boldsymbol{x}(t_n) - \overline{\boldsymbol{x}}(t_n) = R(\theta,\Delta tA)^n(\boldsymbol{x}_0 - \overline{\boldsymbol{x}}_0). \tag{3.37}$$

Thus, the stability function $R(\theta,\Delta tA)^n$ determines how much initial error will grow. Fortunately, we have a rich literature about stability conditions of like equation (3.35) as all details analysed in the work of Hairer and Wanner [5]. First, we assume the following important theorem related to the bound of $||R||$ [5].

Theorem 3.1. *Let the rational function $R(z)$ be bounded for $Re(z) \leq 0$ and assume that the matrix A satisfies*

$$Re(y,Ay) \leq 0 \quad for\ all\ y \in C^n \tag{3.38}$$

where $(.,.)$ denotes the Euclidean scalar product. Then, in the matrix norm corresponding to the scalar product we have

$$\|R(A)\| \leq \underbrace{sup}_{Re(z)\leq 0} |R(z)|. \tag{3.39}$$

Note that for the homogeneous system (3.32), the following diminishing property leads to condition (3.38) as follows

$$\frac{d}{dt}\|x\|^2 = \frac{d}{dt}(x,x) = 2Re\left(x,x'\right) = 2Re(x,Ax). \tag{3.40}$$

When the condition (3.38) is not satisfied, depending on the logarithmic norm of the matrix A, the following corollary becomes useful for the bound of stability function R [5].

Corollary 3.1. *If a matrix occurred in the system* (3.32) *satisfies* $Re(v,Av) \leq s||v||^2$ *for all* $v \in C^m$ *with* $\mu = \mu(A) \leq s$ *is the logarithmic norm of the matrix A, then*

$$||R(A)|| \leq \underbrace{sup}_{Re(z)\leq s} |R(z)|. \tag{3.41}$$

In the light of Corollary (3.1), the present A—stable methods will be unconditionally stable for linear systems (3.32) whenever the logarithmic norm of the matrix A satisfies $\mu(A) \leq 0$ where $\mu(A) = \lim_{\Delta t \to 0} \frac{||I+\Delta tA||-1}{\Delta t}$.

Corollary 3.2. *If* $\mu(A) \leq 0$*, the A—stable cases of the IELDTM are unconditionally stable for homogeneous system* (3.32).

Proof. Application of Corollary (3.1) to equation (3.41) gives

$$||\boldsymbol{x}(t_n) - \overline{\boldsymbol{x}}(t_n)|| \leq \left(\underbrace{sup}_{Re(z)\leq 0} |R(z)| \right)^n ||\boldsymbol{x}_0 - \overline{\boldsymbol{x}}_0||. \tag{3.42}$$

The A—stability assumption leads to the following estimate

$$\underbrace{sup}_{Re(z)\leq 0} |R(z)| \leq 1, \quad for\ all\ z \in C. \tag{3.43}$$

Thus, the A—stable cases of the IELDTM are seen to be unconditionally stable. ■

Table 3.1 Some special A—stable cases of the IELDTM.

θ	K
0.5	1
0.5	2
0.5	3
0.5	4
1	1
1	2

Table (3.1) provides information on some cases of A—stability of the IELDTM. Note that for the implicit schemes, $\theta \geq 0.5$, with the rest of the orders have not been mentioned in Table (3.1), are almost A—stable as illustrated in the scalar case. Except

that the central or backward cases of the IELDTM with $\theta \geq 0.5$ stated in Table (3.1), it is possible to find out various *A*—stable cases up to fourth-order approximation.

3.4 Adaptive Searching

The IELDTM needs the selection of suitable parameters Δt_i and θ to control both accuracy and stability. This situation also depends on the nature of the considered physical model. The established numerical approach is seen to be a general implicit-explicit method depending on the parameter θ. Since the stability properties of the implicit schemes and computational costs have been explained, an optimized technique needs to be obtained in terms of accuracy, stability and computational cost. The following adaptive techniques are produced depending on the nature of the problem.

Case 1:

In this case, we assume that the problem is nonlinear and non-stiff. In non-stiff problems, the current forward methods generally produce convergent and computationally effective results. Thus, the current explicit schemes are preferable for solving nonlinear and non-stiff problems by selecting the direction parameter $\theta = 0$. Even if the model equations are nonlinear, algebraic equations can directly be solved without using any symbolic nonlinear solver. An adaptive procedure is here proposed for the time increment Δt_i to reduce the computational cost. Considering the transformation of order K with the selection parameter $\theta = 0$ leads to the global error of order K as proven in error analysis. For a given tolerance, the following criteria need to be satisfied for scalar equations as also analysed in literature [26],

$$\Delta t_i < \left(\frac{tol}{|X_i(K+1)|} \right)^{\frac{1}{K}} \tag{3.44}$$

where tol is predetermined tolerance and $X_i(K+1)$ can be obtained from the recursive relation. For a system of equations, relation (3.44) can be modified to the related matrices and vectors by considering suitable norms. In this study, the norm $||\,.\,||_\infty$ is considered for the selection of adaptive Δt_i.

Case 2:

In this case, stiffness of the problem is assumed to be arbitrary. In such cases $\theta = 0.5$ and $\theta = 1$ will lead to accurate, stable and effective approaches. In Section 3.2, the error bounds of central schemes have been found to be optimal by taking odd values of the order K. Thus, the IELDTM with $\theta = 0.5$ and odd K values leads to the following

adaptivity equation

$$\Delta t_i < \left(\frac{tol}{\left| \left(\left(\frac{1}{2}\right)^{K+1} (K+1) \right) X_i (K+2) \right|} \right)^{\frac{1}{K+1}} \tag{3.45}$$

where tol is the predetermined tolerance and $X_i(K+2)$ is obtained from the recursion. For the backward adaptive cases, $\theta = 1$, adaptivity equation (3.44) is also valid for the arbitrary selection of the transformation order K. The scalar adaptive procedure given in equation (3.45) can be enlarged to vectors and matrices with the use of suitable norms.

3.5 Numerical Experiments

Considering the quantitative and qualitative results of various test problems, the numerical representation and efficiency of the derived method are presented here. To measure the effectiveness of the current method in terms of accuracy and stability, challenging physical processes represented by stiff differential equations such as the stiff SEIR equation system, cubic nonlinear Duffing equation, Robertson equation system, and Van der Pol equation have been dealt with. The computed results are compared with the results produced with the FDM [31], *ode23s*, *ode15s*, *ode45* and *ode113* [142] as well as the exact solutions. To evaluate the error norms of the current results, absolute pointwise errors E_i and maximum error norms $\|E\|_\infty$ have been preferred. The trust-region dogleg method [143] based *fsolve* built-in function of the MATLAB program has been applied to solve the resulting nonlinear algebraic equation system.

Problem 3.1

Consider the following Susceptible-Exposed-Infected-Recovered/Removed (SEIR) system of equations [144]

$$\begin{aligned}
\frac{dS}{dt} &= -\beta \left(\frac{S(t)}{N} \right) [P(t) + D(t) + \mu A(t)] \\
\frac{dE}{dt} &= \beta \left(\frac{S(t)}{N} \right) [P(t) + D(t) + \mu A(t)] - \frac{1}{d_1} E(t) \\
\frac{dP}{dt} &= \alpha \frac{E(t)}{d_1} - \frac{P(t)}{d_2} \\
\frac{dA}{dt} &= (1-\alpha) \frac{E(t)}{d_1} - \frac{A(t)}{d_3}
\end{aligned} \tag{3.46}$$

$$\frac{dD}{dt} = \frac{P(t)}{d_2} - \frac{D(t)}{p}$$
$$\frac{dR}{dt} = \frac{D(t)}{p} + \frac{A(t)}{d_3}$$

where $S(t) + E(t) + P(t) + A(t) + D(t) + R(t) = N$ is the total population, $S(t), E(t), P(t), A(t), D(t), R(t)$ denote susceptible, exposed, pre-symptomatic, asymptomatic, hospitalized and recovered/removed number of individuals in the population, respectively. The parameters β, μ and α are daily transmission rate, transmission reduction factor and pre-symptomatic ratio, respectively. d_1, d_2, d_3 and p represent the mean latency period, mean pre-symptomatic infectiousness period, mean asymptomatic infectiousness period, and mean hospitalization period, respectively. The SEIR model (3.46) is one of the key compartment models describing the spread of any infectious disease, such as the COVID-19 pandemic [144], in a population. If the basic reproduction number $R_0 = \beta[\alpha(d_2 + p) + (1-\alpha)d_3]$ is greater than one, then the number of susceptible individuals will asymptotically decrease to a threshold value with a stiff or non-stiff dynamic depending on the transmission rate β.

Taking differential transform of equation system (3.46) leads to

$$X_i(k+1) = \frac{1}{k+1}(AX_i(k) + B_i(k)) \tag{3.47}$$

where

$$X_i(k) = [S_i(k), E_i(k), P_i(k), A_i(k), D_i(k), R_i(k)]^T, \tag{3.48}$$

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{d_1} & 0 & 0 & 0 & 0 \\ 0 & \frac{\alpha}{d_1} & -\frac{1}{d_2} & 0 & 0 & 0 \\ 0 & \frac{1-\alpha}{d_1} & 0 & -\frac{1}{d_3} & 0 & 0 \\ 0 & 0 & \frac{1}{d_2} & 0 & -\frac{1}{p} & 0 \\ 0 & 0 & 0 & \frac{1}{d_3} & \frac{1}{p} & -\alpha \end{bmatrix} \tag{3.49}$$

and

$$B_i(k) = \begin{bmatrix} -\frac{\beta}{N}\sum_{j=0}^{k} S_i(k-j)[P_i(j) + D_i(j) + \mu A_i(j)] \\ \frac{\beta}{N}\sum_{j=0}^{k} S_i(k-j)[P_i(j) + D_i(j) + \mu A_i(j)] \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.50}$$

for all $k = 0, 1, \ldots, K-1$ and $i = 0, 1, \ldots, Nw$. The parameter values $d_1 = 3.69$, $d_2 = 3.47$, $d_3 = 3.47$, $p = 1.92$, $\alpha = 0.14$, $\beta = 1.12$ and $N = 3 \times 10^6$, which are obtained by Li et al. [144] for the dynamics of COVID 19 pandemics, considered for all simulations. The SEIR model describes a stiff behaviour for higher values of the transmission rate β.

The advantages of the IELDTM with $\theta = 0.5$ over the existing LDTM [24, 25, 27] can be seen in Figure 3.4 to solve the moderately-stiff SEIR model (3.46). The LDTM suffers from instability when a rapid decrease is observed in the susceptible population. Since the exact solution of the SEIR model (3.46) is not known, the *ode15s* solution obtained with $tol = 10^{-6}$ is considered as the reference solution. As compared with the reference *ode15s* solution of the system, the behaviours of the state variables are seen to be accurately captured by the central IELDTM using the same parameter values. The IELDTM with $\theta \geq 0.5$ destroys the instability drawback of the LDTM by solving the problem implicitly. To increase the stiffness of the SEIR model, the transmission rate is modified to have the following time-dependent form

$$\beta(t) = \begin{cases} 1.12, & t \leq t_c \\ 1.12\eta, & t > t_c \end{cases} \tag{3.51}$$

where $\eta \geq 1$ is the scaling factor of the transmission rate. If t_c is selected in an interval in which rapid decreasing of the susceptible population occurs, then the scaling factor η leads to increase the stiffness of the problem. The SEIR model is solved by the adaptive central IELDTM and various MATLAB *ode* solvers with $t_c = 66$ and $\eta \in [1,\ 12]$ and the results are illustrated in Figure 3.5. The required number of time steps is compared with respect to the changing stiffness of the problem in Figure 3.5a. It has been observed that the sixth and eighth order central IELDTMs need far less time elements than the adaptive MATLAB *ode23s*, *ode15s*, *ode113*, and *ode45*. Even if the number of required time steps of the MATLAB solvers increases with increasing η values, the IELDTM is not affected by increasing stiffness. The behaviour of the stiff dynamics of the susceptible population is illustrated in Figure 3.5b. It is observed that the central IELDTM accurately captures the dynamics at optimum cost.

Problem 3.2

Consider the following cubic nonlinear Duffing equation [27]

$$x'' + \alpha x' + \beta x + \gamma x^3 = 0, \quad for\ 0 \leq t \leq t_f \tag{3.52}$$

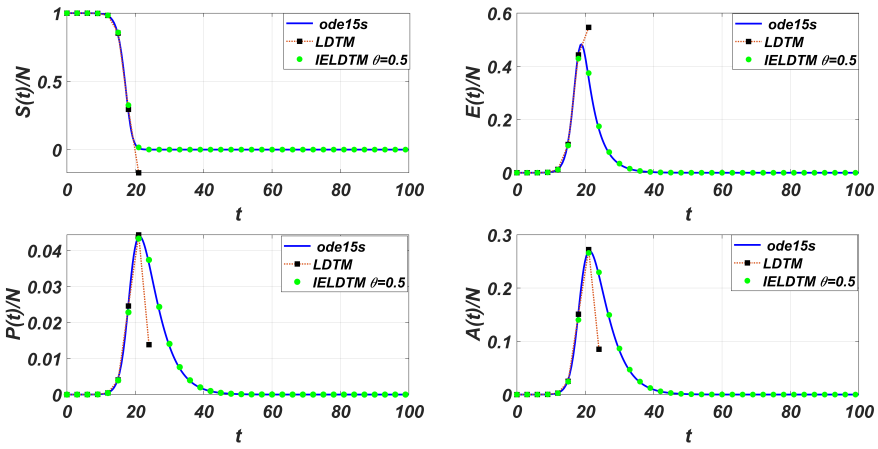

Figure 3.4 Comparison of the central IELDTM and LDTM with the reference *ode15s* solutions of the SEIR model (3.46) for $K = 6$ and $\Delta t = 3$.

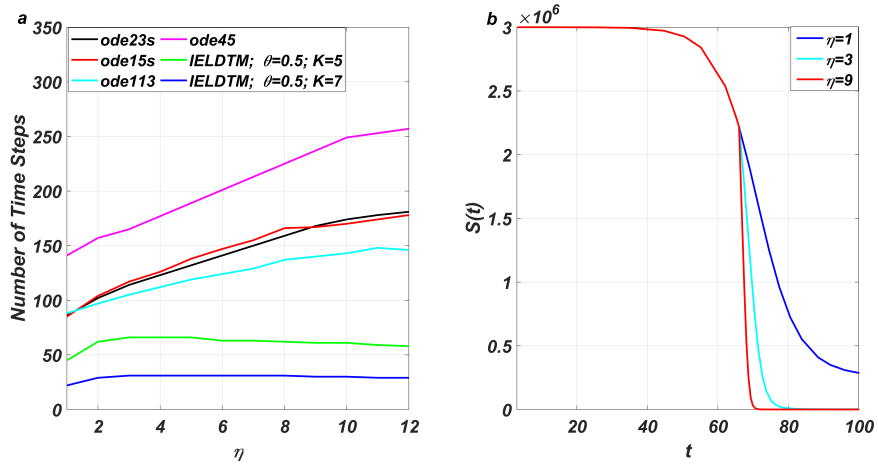

Figure 3.5 a) Comparison of the required number of time steps of the central adaptive IELDTM, *ode23s*, *ode15s*, *ode113* and *ode45* for solving the SEIR model (3.46) with the changing values of η and $tol = 10^{-5}$, b) Effect of the stiffness factor η on the dynamics of the susceptible population $S(t)$ is captured by the central adaptive IELDTM with an optimized degrees of freedom.

with initial displacement and velocity

$$x(0) = x_0 \quad and \quad x'(0) = x_0^*. \tag{3.53}$$

In the study of Tunc and Sari [27], it is proven that the logistic function $x(t) = \frac{1}{1+e^{-t}}$ is exact solution of equation (3.52) with the following parameters,

$$\alpha = -3, \quad \beta = 2, \quad \gamma = -2, \quad x_0 = 0.5, \quad x_0^* = 0.25. \tag{3.54}$$

With the use of $x_1(t) = x(t)$ and $x_2(t) = x'(t)$, the cubic nonlinear Duffing equation is transformed to the following nonlinear system

$$x_1'(t) = x_2(t) \tag{3.55}$$

$$x_2'(t) = -\alpha x_2(t) - \beta x_1(t) - \gamma x_1^3(t). \tag{3.56}$$

Taking differential transformation of (3.55)-(3.56), one can obtain

$$X_i(k+1) = \frac{1}{k+1}\left(AX_i(k) + B_i(k)\right) \tag{3.57}$$

where $k = 0, 1, \ldots, K-1$, $X_i(k) = \left[(X_1)_i(k), (X_2)_i(k)\right]^T$ is the differential transformation of $[x_1(t), x_2(t)]^T$, $B_i(k) = \left[0, \gamma\overline{(X_1)}_i(3;0,0;k)\right]^T$ (see Appendix A for $\overline{(X_1)}_i(m;n,p;k)$) and the constant matrix A can be defined as follows

$$A = \begin{bmatrix} 0 & 1 \\ -\beta & -\alpha \end{bmatrix}. \tag{3.58}$$

The IELDTM yields the following equation

$$\sum_{k=0}^{K} X_{i+1}(k)(-\theta\Delta t_i)^k = \sum_{k=0}^{K} X_i(k)((1-\theta)\Delta t_i)^k \tag{3.59}$$

where $X_0(0) = \left[x_0,\ x_0^*\right]^T$, $i = 0, 1, \ldots, N-1$ and the coefficients can be evaluated from equation (3.57). The performance of the current IELDTM is shown by comparing the maximum errors depending on the varying values of the direction parameter θ (see Figure (3.6)). As theoretically expected, the maximum errors are reduced exponentially by increasing the order of the method. It can be observed from Figure 3.6 that choosing $\theta = 0.5$ with odd-numbered K values leads to an $(K+1)-th$ order numerical method. The present adaptive central scheme is used for various transformation orders from $K = 3$ to $K = 9$ with the estimate $tol = 10^{-15}$ and the

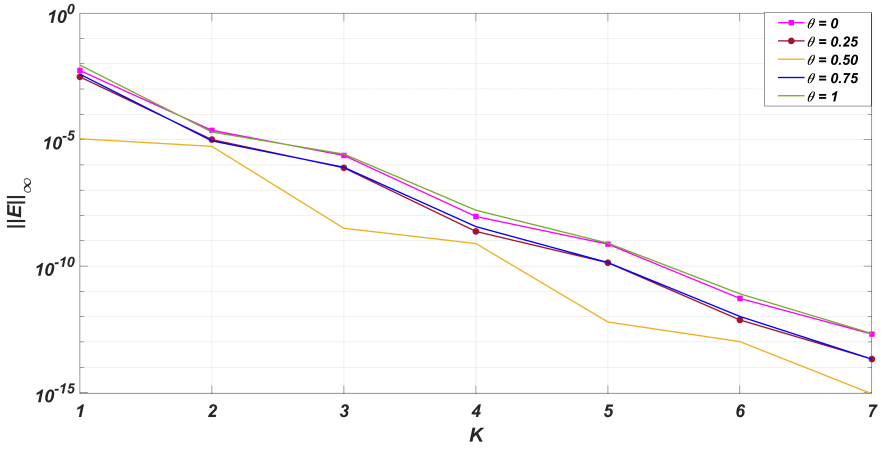

Figure 3.6 Order refinement results of the IELDTM at $t_f = 1$ for various values of the direction parameter θ with the time increment $\Delta t = 0.05$.

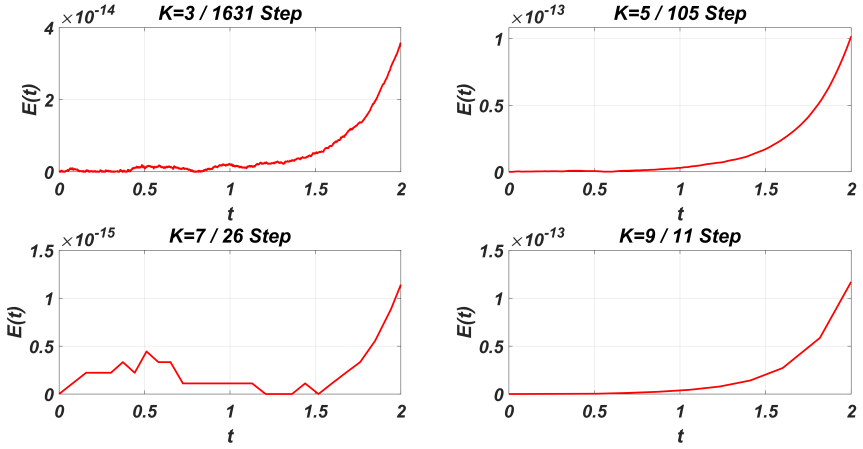

Figure 3.7 Pointwise errors computed by the central adaptive IELDTM with various values of K and $tol = 10^{-15}$.

numerical results are compared in terms of both accuracy and computational costs in Figure 3.7. As seen in the figure, increasing the local approximation order with K reduces the computational cost considerably. The theoretical order expectations and experimental order averages of the current IELDTM are compared with various values of the transformation order K and direction parameter θ as given in Table 3.2. In the table, the theoretical expectations are seen to be almost matching with the experiments. The current adaptive central scheme and the *ode15s-ode23s* solvers are compared with the maximum errors in Table 3.3 to measure the advantages of the IELDTM over the existing stiff *ode* solvers. In all comparisons, the central adaptive IELDTM appears to give better results with both high accuracy and using less number of time steps than literature [142].

Table 3.2 Effect of the approximation order of the IELDTM with various direction parameters in comparative way for Problem 3.2 (NE: Numerical Experiment; TE:Theoretical Expectation).

	$\theta = 0$		$\theta = 0.5$		$\theta = 1$	
K	NE	TE	NE	TE	NE	TE
$K = 1$	0.8348	1	1.9804	2	1.2583	1
$K = 2$	2.0603	2	1.9970	2	1.9319	2
$K = 3$	2.9458	3	4.0142	4	3.0477	3
$K = 4$	3.7102	4	4.0041	4	4.2008	4
$K = 5$	4.9576	5	5.9965	6	5.0311	5
$K = 6$	5.8559	6	6.0821	6	6.2854	6

Table 3.3 Comparison of the present IELDTM ($\theta = 0.5$) and the *ode15s-ode23s* solvers [142] with $tol = 10^{-10}$ for adaptive time steps N^* and maximum errors in Problem 3.2.

	IELDTM; $K = 3$		IELDTM; $K = 5$		*ode15s*		*ode23s*	
t_f	N^*	$\|\|E\|\|_\infty$	N^*	$\|\|E\|\|_\infty$	N^*	$\|\|E\|\|_\infty$	N^*	$\|\|E\|\|_\infty$
1	51	7.93E-11	9	2.38E-10	64	1.37E-09	825	8.61E-08
2	92	4.62E-09	16	8.45E-09	99	3.08E-08	1487	2.99E-06
4	152	1.01E-05	27	1.49E-05	161	6.53E-05	2595	6.47E-03

Problem 3.3

Consider the following modified Robertson chemical stiff system of nonlinear differential equations [31],

$$\begin{aligned} x_1' &= -0.04x_1 + 10^4 x_2 x_3 - 0.96e^{-t} \\ x_2' &= 0.04x_1 - 10^4 x_2 x_3 - 3.10^7 x_2^2 - 0.04e^{-t} \\ x_3' &= 3.10^7 x_2^2 + e^{-t} \end{aligned} \tag{3.60}$$

where $t \in [0,\ 4]$ and the initial conditions are

$$x_1(0) = 1,\ x_2(0) = x_3(0) = 0. \tag{3.61}$$

As stated in literature [31], the system (3.60) has the exact solution $x_1(t) = e^{-t}$, $x_2(t) = 0$ and $x_3(t) = 1 - e^{-t}$. Taking the differential transformation of (3.60) leads to the following relations,

$$\begin{aligned} (X_1)_i(k+1) = \frac{1}{k+1}\Bigg[&-0.04(X_1)(k)_i + 10^4 \sum_{s=0}^{k} (X_2)_i(s)(X_3)_i(k-s) - \\ &0.96(-1)^k \frac{e^{-t_i}}{k!}\Bigg], \end{aligned} \tag{3.62}$$

$$(X_2)_i(k+1)=\frac{1}{k+1}\Bigg[0.04(X_1)(k)_i-10^4\sum_{s=0}^{k}(X_2)_i(s)(X_3)_i(k-s)- \\ 3.10^7\left(\overline{X}_2\right)_i(2;0,0;k)-0.04(-1)^k\frac{e^{-t_i}}{k!}\Bigg], \tag{3.63}$$

$$(X_3)_i(k+1)=\frac{1}{k+1}\left[3.10^7\left(\overline{X}_2\right)_i(2;0,0;k)+(-1)^k\frac{e^{-t_i}}{k!}\right] \tag{3.64}$$

where $(X_1)_0(0)=1$, $(X_2)_0(0)=0$, $(X_3)_0(0)=0$, $k=0,1,\ldots,K-1$, $i=0,1,\ldots,N$ and $\overline{X}_i(m;n,p;k)$ is defined in Appendix A. Thus, the IELDTM yields the following equations

$$\begin{aligned}
\sum_{k=0}^{K}(X_1)_{i+1}(k)(-\theta\Delta t_i)^k&=\sum_{k=0}^{K}(X_1)_i(k)((1-\theta)\Delta t_i)^k,\\
\sum_{k=0}^{K}(X_2)_{i+1}(k)(-\theta\Delta t_i)^k&=\sum_{k=0}^{K}(X_2)_i(k)((1-\theta)\Delta t_i)^k,\\
\sum_{k=0}^{K}(X_3)_{i+1}(k)(-\theta\Delta t_i)^k&=\sum_{k=0}^{K}(X_3)_i(k)((1-\theta)\Delta t_i)^k
\end{aligned} \tag{3.65}$$

where $i=0,1,\ldots,N-1$ and the coefficients can be calculated from equations (3.62)-(3.64). The present central adaptive schemes and the MATLAB solvers, *ode15s-ode23s* [142], have been compared to demonstrate the efficiency of the present approach through the pointwise errors (Figure 3.8). The current adaptive IELDTM schemes of order 4 and 6 have been shown to produce more accurate results, using far less number of time elements, than the commonly used stiff solvers, *ode15s-ode23s*. Maximum errors of the current adaptive central scheme have been presented in comparison with the LBRFDM-BDF of the literature [31] (see Table 3.4). In the table, it is seen that the current schemes give more accurate results using the same number of time elements. It is also obvious from Table 3.4 that the IELDTM is far more accurate than various versions of the rational finite difference techniques presented for stiff behaviour in the literature [31].

Table 3.4 Comparison of the present central IELDTM with the literature LBRFDM-BDF [31] through maximum errors at $t=4$ for Problem 3.3.

Δt	IELDTM;$K=3$	IELDTM;$K=5$	LBRFDM;(6,3)	LBRFDM;(6,4)	BDF;4-Step
2^{-5}	2.69E-10	3.89E-13	3.11E-07	1.55E-07	1.41E-07
2^{-6}	4.97E-11	3.79E-13	2.13E-08	1.07E-08	1.02E-08
2^{-7}	4.97E-12	1.33E-15	1.40E-09	6.98E-10	6.83E-10
2^{-8}	4.76E-13	8.88E-16	8.94E-11	4.47E-11	4.42E-07

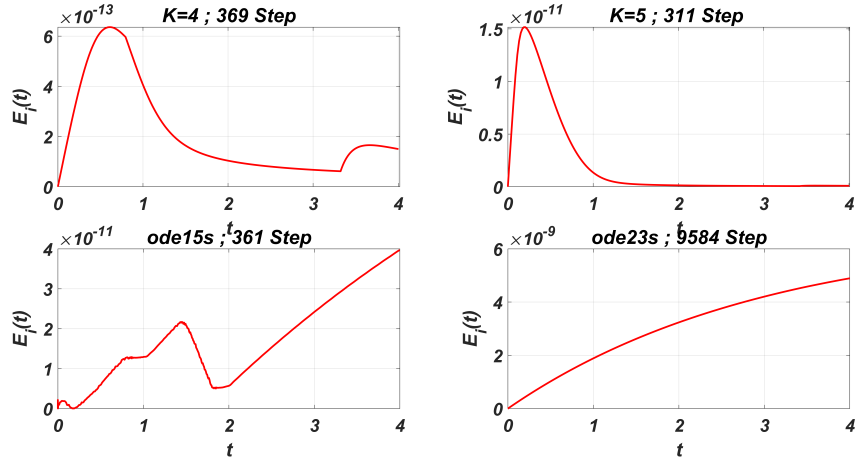

Figure 3.8 Comparison of the present central adaptive IELDTM and the *ode15s-ode23s* through pointwise errors with $tol = 10^{-12}$ for Problem 3.3.

Problem 3.4

Consider the Van der Pol equation in the first-order system form,

$$U^{'} = V \tag{3.66}$$

$$V^{'} = -U + \varepsilon\left(1 - U^{2}\right)V$$

where the initial conditions are taken to be $U(0) = 2$ and $V(0) = 0$. Note that the system (3.66) corresponds to the second-order nonlinear Van der Pol equation with dependent variable U. The Van der Pol equation is a model of self-sustained oscillations of a triode electric circuit with the current voltage characteristics. Stiffness of the Van der Pol system (3.66) strongly depends on the parameter value ε. If the parameter ε is not too large, then the equation is not stiff and can be solved with also non-stiff methods. However, in case of $\varepsilon \gg 1$, the system (3.66) is not easy to handle and requires more accurate numerical methods for its solution. Taking the differential transform of the system (3.66) yields

$$\boldsymbol{U}_i(k+1) = A\boldsymbol{U}_i(k) + \boldsymbol{F}(k) \tag{3.67}$$

where $\boldsymbol{U}_i(k) = [U_i(k),\ V_i(k)]^T$, $A = [0,\ 1;\ -1,\ \varepsilon]$, $\boldsymbol{F}(k) = \left[0,\ -\varepsilon\sum_{l=0}^{k}\sum_{n=0}^{l} U_i(n)U_i(l-n)V_i(k-l)\right]^T$ and $k = 0, 1, \ldots, K-1$. The IELDTM leads to the following equations

$$\sum_{k=0}^{K} \boldsymbol{U}_{i+1}(k)(-\theta\Delta t_i)^k = \sum_{k=0}^{K} \boldsymbol{U}_i(k)((1-\theta)\Delta t_i)^k \tag{3.68}$$

where $i = 0, 1, \ldots, N-1$ and $\boldsymbol{U}_i(k)$ can be evaluated from recursive relation (3.67).

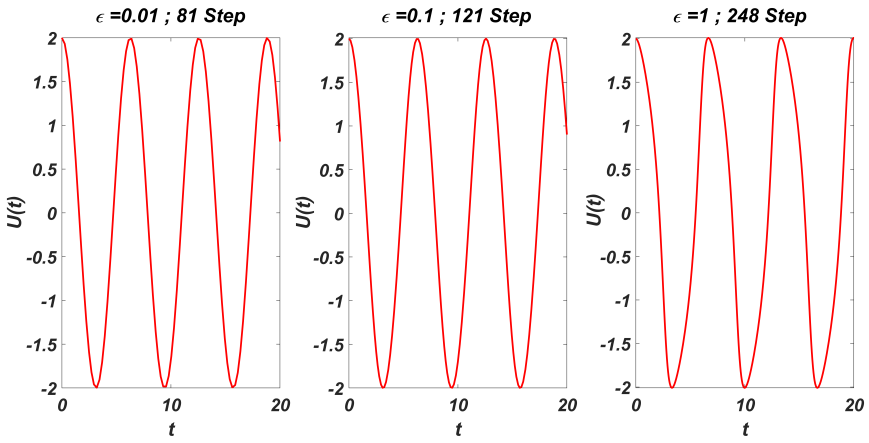

Figure 3.9 Adaptive central solutions produced with $K = 7$, $\theta = 0.5$, $tol = 10^{-10}$ and various ε values for Problem 3.4.

As seen in Figure 3.9, numerical behaviours of various non-stiff cases of the Van der Pol equation have been illustrated through the present central adaptive technique. The qualitative behaviour is produced by considering the parameter values $K = 7$, $\theta = 0.5$, $t_f = 20$ and $tol = 10^{-10}$. As seen in Figure 3.10, as the stiffness increases, the number of time steps required by the algorithm increases. The computed results of the stiff cases are presented as seen in the figure for the parameter values of $\varepsilon = 10$ and $\varepsilon = 100$, respectively, in Problem 3.4. The central adaptive technique is used for Figure 3.10 with the parameter values $K = 5$, $\theta = 0.5$ and $tol = 10^{-10}$. As shown in the figures, with the current techniques, it is seen that the stiff behaviours are correct and successfully captured. The obtained results showed that the effectiveness of the present IELDTM has been realized by comparing the MATLAB solvers, *ode15s-ode23s* [142] (see Table 3.5). In the table, it is understood that our method requires far less number of time steps than the *ode* solvers to accurately capture the physical behaviours.

Table 3.5 Comparison of the present central adaptive IELDTM and the MATLAB solvers [142], *ode15s-ode23s,* with the adaptive time step numbers for $tol = 10^{-10}$ in Problem 3.4.

ε/t_f	IELDTM;$K = 5$	IELDTM;$K = 7$	IELDTM;$K = 9$	*ode15s*	*ode23s*
0.1/1	254	95	53	700	10912
1/10	520	193	108	1256	15717
10/100	5339	1888	1068	9632	141653
100/1000	19012	15282	10820	17523	299643

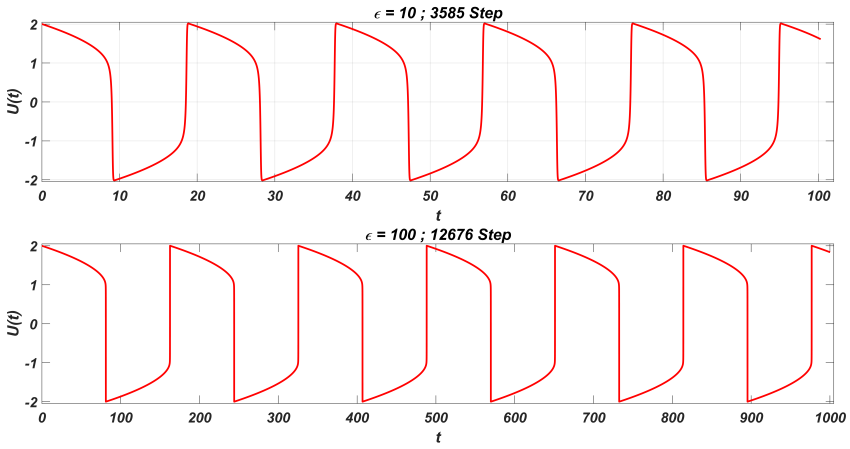

Figure 3.10 Adaptive central solutions produced with $K = 5$, $\theta = 0.5$, $tol = 10^{-10}$, $\varepsilon = 10$ and $\varepsilon = 100$ for Problem 3.4.

3.6 Summary

In this chapter, a new one-step implicit-explicit local differential transform method (IELDTM) has been developed with an arbitrary order for especially stiff initial value problems. The strong stability of the numerical method produced using the Taylor series has been proven to be preserved. A priori error estimates of the currently derived approaches has been constructed, and order conditions of the methods have been determined depending on direction parameters. Stability of the present methods has been discussed, and they have thus been examined by considering A- and L-stabilities. To reach optimal numerical methods, adaptive procedures have been produced for forward, central, and backward cases, respectively. The currently produced versatile methods have been seen to be effective for very challenging problems defined by stiff differential equations. To explain and analyse the challenging aspects of the problems, four striking stiff differential equations have been taken into account, and thus qualitative and quantitative results have been exhibited. In a comparative way, efficiency of the present methods in terms of accuracy, stability, versatility and computational cost has been proven. In the next chapter, the IELDTM will be derived for the numerical investigations of the stiff BVPs for ODEs.

4
THE IELDTM FOR STIFF BVPs

Boundary value problems (BVPs) of ordinary differential equations (ODEs) arise in many disciplines such as physics, chemistry, engineering, finance and mathematical biology [60]. Analytical solutions are often not available for most of those problems. While some series-based techniques are capable of producing semi-analytical solutions for BVPs, the convergence of those methods is mainly dependent on the global smoothness of exact solutions, as observed from examples discussed in the literature [145]. Besides, stiff BVPs involving significant local behaviours such as sharp discontinuities or boundary layers are areas where various notable difficulties are encountered [57–59]. Practical numerical algorithms are the only options for dealing with such challenging problems. Since the solutions of stiff BVPs have distinct local behaviours to be captured by numerical solvers, an effective numerical algorithm must satisfy both optimized p—refinement and h—refinement procedures.

In this chapter, we propose a new IELDTM to solve BVPs for ODEs. First of all, the IELDTM is derived for a general two-point BVP, and the mathematical foundations of the method are presented. The presented method here eliminates the well-known disadvantages of the DTMs, such as divergence and numerical instability. A priori error analysis of the present method has been done, and order conditions have been produced. Stability results are discussed together with the selection of the direction parameter that determines whether the method is explicit or implicit. In numerical experiments, various types of BVPs, particularly high-order BVPs, singular BVPs, singularly perturbed BVPs, and strongly nonlinear BVPs have been studied to test the derived method.

4.1 The Implicit-Explicit Local Differential Transform Method

This section is devoted to numerical foundations of the IELDTM for a general second-order BVP. A general description of the IELDTM is explained and some comments on this method are provided.

Let us consider the following general second order two-point boundary value problem,

$$y''(x) = G(y'(x),\ y(x), x), \quad x \in S \tag{4.1}$$

with the mixed boundary conditions

$$\alpha_1 y(a) + \alpha_2 y'(a) = \delta_1 \quad and \quad \beta_1 y(b) + \beta_2 y'(b) = \delta_2 \tag{4.2}$$

where $S = [a, b] \subset R$, $y(x) \in R$, $\alpha_1,\ \alpha_2,\ \beta_1,\ \beta_2,\ \delta_1,\ \delta_2 \in R$ *and* $G : R \times R \times S \to R$.

We assume that the BVP (4.1-4.2) is well-posed and the exact solution $y(x)$ is analytic on the considered domain S. Let us divide the interval $[a, b]$ into at most N spatial elements with $\Delta x_i = x_{i+1} - x_i$ and the partition of the interval is given by $\omega = \{a = x_0 < x_1 < \ldots < x_{N^*} = b\}$ where $N^* \leq N$. Let us consider the convergent local Taylor series representation of the function $y(x)$ about $x_i \in S$ as follows

$$y_i(x) = \sum_{k=0}^{K} Y_i(k)(x - x_i)^k + O\left((x - x_i)^{K+1}\right),\ \ x_i - \rho^i \leq x \leq x_i + \rho^i \tag{4.3}$$

where $i = 0, 1, \ldots,\ N^*$. The function $y_i(x)$ is assumed to be analytic for all i and has the radius of convergence $\rho^i > \Delta x_i$. This assumption leads us to search about the relations between neighbour solutions $y_i(x)$ and $y_{i+1}(x)$ in terms of their local behaviours. With Theorem 2.7, since the condition $|x_{i+1} - x_i| = \Delta x_i < \rho_i$ holds, the function $y(x)$ has also a local convergent Taylor series representation about $x = x_{i+1}$ with the radius of convergence at least $\rho^{i+1} = \rho^i - \Delta x_i$ and the interval of convergence is $(x_{i+1} - \rho^{i+1}, x_{i+1} + \rho^{i+1})$. Then, the function $y(x)$ can be locally approximated as

$$y_{i+1}(x) = \sum_{k=0}^{K} Y_{i+1}(k)(x - x_{i+1})^k + O\left((x - x_{i+1})^{K+1}\right), \tag{4.4}$$

where $x_{i+1} - \rho^{i+1} \leq x \leq x_{i+1} + \rho^{i+1}$. Let us define the intervals $P^i = [x_i - \rho^i, x_i + \rho^i]$, $P^{i+1} = [x_{i+1} - \rho^{i+1}, x_{i+1} + \rho^{i+1}]$ and $P = [x_i,\ x_{i+1}]$. Assume that the condition $P \subset \left(P^i \cap P^{i+1}\right)$ holds. Then we state the following conclusions;

* Due to the convergence assumptions, the two convergent representations (4.3) and (4.4) should give the same numerical result at any point in the interval $[x_i,\ x_{i+1}]$.

* If necessary, C^m continuity condition i.e. continuous slopes, continuous curvatures etc. can be applied in the interval $[x_i,\ x_{i+1}]$.

* Any point in the interval $[x_i,\ x_{i+1}]$ can be written as $x^* = x_i + (1 - \theta)\Delta x_i$, where $0 \leq \theta \leq 1$.

* We produce the IELDTM with the use of continuity conditions of the solutions at such interior points.

Convergent solutions $y_i(x)$ and $y_{i+1}(x)$ at the interior points $x^* = x_i + (1-\theta)\Delta x_i$ need to satisfy the following C^0 and C^1 continuity relations

$$y_{i+1}(x_i + (1-\theta)\Delta x_i) = y_i(x_i + (1-\theta)\Delta x_i), \tag{4.5}$$

$$y'_{i+1}(x_i + (1-\theta)\Delta x_i) = y'_i(x_i + (1-\theta)\Delta x_i). \tag{4.6}$$

Substituting interior node value $x^* = x_i + (1-\theta)\Delta x_i$ to approximate solutions (4.3-4.4) and using continuity relations (4.5-4.6) yield the following equations,

$$\sum_{k=0}^{K} Y_{i+1}(k)(-\theta\Delta x_i)^k = \sum_{k=0}^{K} Y_i(k)((1-\theta)\Delta x_i)^k + O\left((\Delta x_i)^{K+1}\right), \tag{4.7}$$

$$\sum_{k=1}^{K} Y_{i+1}(k)\,k(-\theta\Delta x_i)^{k-1} = \sum_{k=1}^{K} Y_i(k)\,k((1-\theta)\Delta x_i)^{k-1} + O\left((\Delta x_i)^{K}\right). \tag{4.8}$$

The next issue is to determine the algebraic relations between $Y_i(k)$ and $Y_{i+1}(k)$ for all $k = 0, 1, \ldots, K$. Taking differential transform of equation (4.1) leads to the following relation

$$Y_i(k+2) = \frac{1}{(k+1)(k+2)} F(Y_i(k+1),\ Y_i(k), \ldots,\ Y_i(1),\ Y_i(0), H(k)) \tag{4.9}$$

where $i = 0, 1, \ldots,\ N^*$, $k = 0, 1, \ldots,\ K-2$, F is the transformed form of the function arbitrary function $G(y'(x),\ y(x), x)$ and $H(k)$ is the transformed form of the non-autonomous part of the function $G(y'(x),\ y(x), x)$. It is obvious that $Y_i(k+2)$ can be written in terms of $Y_i(k+1)$ and $Y_i(k)$ for each i and k. Thus, to completely describe approximation functions (4.3-4.4), one should evaluate the initial position $Y_i(0)$ and the initial slope $Y_i(1)$ for all $i = 0, 1, \ldots,\ N^*$. For each spatial position, $Y_i(0)$ and $Y_i(1)$ are assumed to be unknown values to determine highly accurate local approximations (4.3-4.4). With the use of algebraic relations in (4.9), substitution of all algebraic coefficients $Y_i(k)$ into continuity equations (4.7-4.8) leads to the following explicit-implicit local equations

$$g_1(Y_{i+1}(0), Y_{i+1}(1), \theta, \Delta x_i) = h_1(Y_i(0), Y_i(1), \theta, \Delta x_i), \tag{4.10}$$

$$g_2(Y_{i+1}(0), Y_{i+1}(1), \theta, \Delta x_i) = h_2(Y_i(0), Y_i(1), \theta, \Delta x_i) \tag{4.11}$$

where $i = 0, 1, \ldots, N^* - 1$ and the functions g_1, g_2, h_1 and h_2 are obtained from the right- and left-hand sides of equations (4.7-4.8). If $\theta = 0$ then the equations (4.10-4.11) are explicit equations of $Y_{i+1}(0)$ and $Y_{i+1}(1)$ irrespective of the linearity-nonlinearity of the differential equation. On the other hand, if $\theta \neq 0$ and the differential equation is nonlinear then equations (4.10-4.11) are implicit equations of $Y_{i+1}(0)$ and $Y_{i+1}(1)$. The boundary conditions lead to the following equations

$$\alpha_1 Y_0(0) + \alpha_2 Y_0(1) = \delta_1, \tag{4.12}$$

$$\beta_1 Y_{N^*}(0) + \beta_2 Y_{N^*}(1) = \delta_2. \tag{4.13}$$

By eliminating $Y_0(0)$ and $Y_{N^*}(0)$ from the equations (4.12-4.13) and substituting into system of algebraic equations (4.10-4.11), equations (4.10-4.11) can be rewritten as follows

$$\boldsymbol{\phi}\left(Y_0(1), Y_1(0), Y_1(1), Y_2(0), Y_2(1), \ldots, Y_{N^*-1}(0), Y_{N^*-1}(1), Y_{N^*}(1)\right) = \boldsymbol{Z} \tag{4.14}$$

where the system is $(2N^*) \times (2N^*)$ including all local positions and slopes. Depending on the function $G(y'(x),\ y(x), x)$, $\boldsymbol{\phi}(\boldsymbol{Y}) = \boldsymbol{Z}$ is either a linear or nonlinear system of algebraic equations with

$$\boldsymbol{Y} = \left(Y_0(1), Y_1(0), Y_1(1), Y_2(0), Y_2(1), \ldots, Y_{N^*-1}(0), Y_{N^*-1}(1), Y_{N^*}(1)\right).$$

MATLAB built-in function *fsolve* has been used when equation system (4.14) is nonlinear. The function $\boldsymbol{\phi}$ also depends on the selection of the direction parameter θ. As will be demonstrated later on, this parameter is crucial in terms of accuracy, stability, and computational efficiency. Therefore, the following notes are important for the rest of this study

* $N^* = 1$ and $\theta = 0$ yields the classical semi-analytic DTM [84–89].

* $\theta = 0$ yields the explicit forward scheme and known as the local differential transform method, which is used in the literature [92].

* The rest of the selections of the parameter θ, i.e. $\theta \neq 0$, leads to the implicit schemes. To the best of the author's knowledge, this generalization for boundary value problems is carried out for the first time.

* $\theta = 0.5$ leads to an implicit central scheme which will give us stability and order preserved method for linear and nonlinear cases.

* $\theta = 1$ leads to the implicit backward scheme, which is also a stable and order preserved method for linear and nonlinear cases.

4.2 Numerical Implementation and Error Analysis

Consider BVP (4.1-4.2) with

$$G(y'(x),\ y(x),x) = -\left(py'(x) + qy\right) \tag{4.15}$$

where $p, q \in R$. Thus, problem (4.1-4.2) becomes a linear two-point BVP as a test problem to explain numerical implementation of the current IELDTM. Taking the differential transform of equation (4.1) yields,

$$Y_i(k+2) = -\frac{1}{(k+1)(k+2)}\left[p\,(k+1)\,Y_i(k+1) + qY_i(k)\right] \tag{4.16}$$

where $k = 0, 1, 2, \ldots,\ K-2$ and $i = 0, 1, 2, \ldots,\ N$. The following coefficients can be evaluated

$$Y_i(2) = -\frac{q}{2}Y_i(0) - \frac{p}{2}Y_i(1), \tag{4.17}$$

$$Y_i(3) = \frac{pq}{6}Y_i(0) + \frac{\left(p^2-q\right)}{6}Y_i(1), \tag{4.18}$$

$$Y_i(4) = \frac{\left(q^2-p^2q\right)}{24}Y_i(0) + \frac{\left(2pq-p^3\right)}{24}Y_i(1), \tag{4.19}$$

$$Y_i(5) = \frac{\left(7pq^2-3p^2q\right)}{480}Y_i(0) + \frac{\left(10p^2q-4q^2-3p^4\right)}{480}Y_i(1). \tag{4.20}$$

This evaluation can be increased up to desired approximation order. If we apply the IELDTM to the considered BVP, the following equations are found for each spatial element $[x_i\ ,\ x_{i+1}]$

$$\sum_{k=0}^{K} Y_{i+1}(k)(-\theta\Delta x)^k = \sum_{k=0}^{K} Y_i(k)((1-\theta)\Delta x)^k, \tag{4.21}$$

$$\sum_{k=1}^{K} Y_{i+1}(k)\,k(-\theta\Delta x)^{k-1} = \sum_{k=1}^{K} Y_i(k)\,k\,((1-\theta)\Delta x)^{k-1}, \tag{4.22}$$

where $i = 0, 1, \ldots, N-1$ and the fixed spatial step size $\Delta x = \frac{(b-a)}{N}$ is assumed. With the use of coefficients (4.17-4.20), equations (4.21-4.22) become

$$\begin{aligned} Y_{i+1}(0) + Y_{i+1}(1)(-\theta\Delta x) + \left[-\frac{q}{2}Y_{i+1}(0) - \frac{p}{2}Y_{i+1}(1)\right](-\theta\Delta x)^2 + \\ \left[\frac{pq}{6}Y_{i+1}(0) + \frac{\left(p^2-q\right)}{6}Y_{i+1}(1)\right](-\theta\Delta x)^3 + \\ \left[\frac{\left(q^2-p^2q\right)}{24}Y_{i+1}(0) + \frac{\left(2pq-p^3\right)}{24}Y_{i+1}(1)\right](-\theta\Delta x)^4 + \ldots = \\ Y_i(0) + Y_i(1)((1-\theta)\Delta x) + \left[-\frac{q}{2}Y_i(0) - \frac{p}{2}Y_i(1)\right] \\ ((1-\theta)\Delta x)^2 + \\ \left[\frac{pq}{6}Y_i(0) + \frac{\left(p^2-q\right)}{6}Y_i(1)\right]((1-\theta)\Delta x)^3 + \\ \left[\frac{\left(q^2-p^2q\right)}{24}Y_i(0) + \frac{\left(2pq-p^3\right)}{24}Y_i(1)\right]((1-\theta)\Delta x)^4 + \ldots, \end{aligned} \tag{4.23}$$

and

$$\begin{aligned} Y_{i+1}(1) + 2\left[-\frac{q}{2}Y_{i+1}(0) - \frac{p}{2}Y_{i+1}(1)\right](-\theta\Delta x) + \\ 3\left[\frac{pq}{6}Y_{i+1}(0) + \frac{\left(p^2-q\right)}{6}Y_{i+1}(1)\right](-\theta\Delta x)^2 + \\ 4\left[\frac{\left(q^2-p^2q\right)}{24}Y_{i+1}(0) + \frac{\left(2pq-p^3\right)}{24}Y_{i+1}(1)\right](-\theta\Delta x)^3 + \ldots = \\ Y_i(1) + 2\left[-\frac{q}{2}Y_i(0) - \frac{p}{2}Y_i(1)\right]((1-\theta)\Delta x) + \\ 3\left[\frac{pq}{6}Y_i(0) + \frac{\left(p^2-q\right)}{6}Y_i(1)\right]((1-\theta)\Delta x)^2 + \\ 4\left[\frac{\left(q^2-p^2q\right)}{24}Y_i(0) + \frac{\left(2pq-p^3\right)}{24}Y_i(1)\right]((1-\theta)\Delta x)^3 + \ldots, \end{aligned} \tag{4.24}$$

where $i = 0, 1, \ldots, N-1$. If equations (4.23-4.24) is rearranged in terms of $Y_{i+1}(0)$, $Y_{i+1}(1)$, $Y_i(0)$ and $Y_i(1)$, equations (4.23-4.24) turn into the following equations

$$\begin{aligned}
&\left[1-\frac{q}{2}(-\theta\Delta x)^2+\frac{pq}{6}(-\theta\Delta x)^3+\frac{\left(q^2-p^2q\right)}{24}(-\theta\Delta x)^4+\ldots\right]Y_{i+1}(0)+\\
&\left[-\theta\Delta x-\frac{p}{2}(-\theta\Delta x)^2+\frac{\left(p^2-q\right)}{6}(-\theta\Delta x)^3+\frac{\left(2pq-p^3\right)}{24}(-\theta\Delta x)^4+\ldots\right]Y_{i+1}(1)=\\
&\left[1-\frac{q}{2}((1-\theta)\Delta x)^2+\frac{pq}{6}((1-\theta)\Delta x)^3+\frac{\left(q^2-p^2q\right)}{24}((1-\theta)\Delta x)^4+\ldots\right]Y_i(0)+\\
&\left[(1-\theta)\Delta x-\frac{p}{2}((1-\theta)\Delta x)^2+\frac{\left(p^2-q\right)}{6}((1-\theta)\Delta x)^3+\right.\\
&\left.\frac{\left(2pq-p^3\right)}{24}((1-\theta)\Delta x)^4+\ldots\right]Y_i(1),
\end{aligned} \tag{4.25}$$

and

$$\begin{aligned}
&\left[-q(-\theta\Delta x)+\frac{pq}{2}(-\theta\Delta x)^2+\frac{\left(q^2-p^2q\right)}{6}(-\theta\Delta x)^3+\ldots\right]Y_{i+1}(0)+\\
&\left[1-p(-\theta\Delta x)+\frac{\left(p^2-q\right)}{2}(-\theta\Delta x)^2+\frac{\left(2pq-p^3\right)}{6}(-\theta\Delta x)^3+\ldots\right]Y_{i+1}(1)=\\
&\left[-q((1-\theta)\Delta x)+\frac{pq}{2}((1-\theta)\Delta x)^2+\frac{\left(q^2-p^2q\right)}{6}((1-\theta)\Delta x)^3+\ldots\right]Y_i(0)+\\
&\left[1-p((1-\theta)\Delta x)+\frac{\left(p^2-q\right)}{2}((1-\theta)\Delta x)^2+\frac{\left(2pq-p^3\right)}{6}((1-\theta)\Delta x)^3+\ldots\right]Y_i(1)
\end{aligned} \tag{4.26}$$

where $i = 0, 1, \ldots, N-1$. Equations (4.25-4.26) include $2N$ equations with $2N+2$ unknowns for N spatial elements. Thus, imposing mixed boundary conditions defined in (4.2) we can eliminate $Y_0(0)$ and $Y_N(0)$. Finally, we reach the following linear system

$$A\hat{Y} = \hat{F} \tag{4.27}$$

where $\hat{Y} = [Y_0(1),\ Y_1(0),\ Y_1(1),\ \ldots, Y_{N-1}(0), Y_{N-1}(1),\ Y_N(1)]^T$ including all unknown positions and slopes, A is a $((2N)\times(2N))$ known matrix and $\hat{F}$ is the residual vector coming from boundary conditions. With the use of any suitable linear solver for equation (4.27), we find the desired solutions at discrete positions. Now, it is time to analyse the global error of the numerical approximation. Complete form of continuity

equations (4.21-4.22) can be written as follows

$$\sum_{k=0}^{K} Y_{i+1}(k)(-\theta\Delta x)^k = \sum_{k=0}^{K} Y_i(k)((1-\theta)\Delta x)^k + \Delta x \rho_i^1, \tag{4.28}$$

$$\sum_{k=1}^{K} Y_{i+1}(k)\,k(-\theta\Delta x)^{k-1} = \sum_{k=1}^{K} Y_i(k)\,k\,((1-\theta)\Delta x)^{k-1} + \Delta x \rho_i^2, \tag{4.29}$$

where ρ_i^1 and ρ_i^2 are the local truncation errors defined as

$$\rho_i^1 = \left[Y_{i_*}(K+1)(1-\theta)^{K+1} - Y_{(i+1)_*}(K+1)(-\theta)^{K+1}\right]\Delta x^K, \tag{4.30}$$

and

$$\rho_i^2 = \left[Y_{i_*}(K+1)(K+1)(1-\theta)^{K} - Y_{(i+1)_*}(K+1)(K+1)(-\theta)^{K}\right]\Delta x^{K-1} \tag{4.31}$$

where $i = 0, 1, \ldots, N-1$ and Y_{i_*} defines the local transform at point $x_{i_*} \in [x_{i-1},\ x_{i+1}]$. Then exact equation system corresponding to equation (4.27) can be stated as

$$AY = \hat{F} + \Delta x G(\theta, \Delta x) \tag{4.32}$$

where Y is a $((2N)\times 1)$ column vector corresponds to exact form of $\hat{Y}$ and $G(\theta,\Delta x) = G_1(\theta,\Delta x) + G_2(\theta,\Delta x)$ is a $((2N)\times 1)$ column vector corresponds to the truncation errors as follows,

$$G_1(\theta,\Delta x) = \begin{bmatrix} \left[(1-\theta)^{K+1} Y_{0_*}(K+1) - (-\theta)^{K+1} Y_{1_*}(K+1)\right] \\ 0 \\ \left[(1-\theta)^{K+1} Y_{1_*}(K+1) - (-\theta)^{K+1} Y_{2_*}(K+1)\right] \\ 0 \\ . \\ . \\ . \\ \left[(1-\theta)^{K+1} Y_{(N-1)_*}(K+1) - (-\theta)^{K+1} Y_{N_*}(K+1)\right] \\ 0 \end{bmatrix} \Delta x^K,$$

$$G_2(\theta,\Delta x)=\begin{bmatrix} 0 \\ \left[(K+1)(1-\theta)^K Y_{0_*}(K+1)-(K+1)(-\theta)^K Y_{1_*}(K+1)\right] \\ 0 \\ \left[(K+1)(1-\theta)^K Y_{1_*}(K+1)-(K+1)(-\theta)^K Y_{2_*}(K+1)\right] \\ . \\ . \\ . \\ 0 \\ \left[(K+1)(1-\theta)^K Y_{(N-1)_*}(K+1)-(K+1)(-\theta)^K Y_{N_*}(K+1)\right] \end{bmatrix}\Delta x^{K-1}.$$

Now, defining the error column vector as $\varepsilon = Y - \hat{Y}$ and subtracting equation (4.32) from equation (4.27) yield

$$A\varepsilon = \Delta x G(\theta,\Delta x) \tag{4.33}$$

and defining new matrix $A^* = \Delta x A^{-1}$ leads to

$$||\varepsilon|| \leq ||A^*||\ ||G(\theta,\Delta x)|| \tag{4.34}$$

where $||.||$ is an arbitrary norm. Complete error analysis of the IELDTM requires the bound of $||G(\theta,\Delta x)||$ in terms of the direction parameter θ. Then, the following two cases are considered:

***Case*1 :** Assume that the direction parameter $\theta = 0.5$ and K is even. In this case, the following general conditions are reached for $G_1(0.5,\Delta x)$

$$\begin{aligned}(G_1(0.5,\Delta x))_i &= \Big[\,(0.5)^{K+1} Y_{i_*}(K+1)-(-0.5)^{K+1}\big(Y_{i_*}(K+1)+ \\ &\qquad \Delta x(K+2)Y_{i_*}(K+2)\big]\Delta x^K, \\ &= \left(\frac{1}{2}\right)^K Y_{i_*}(K+1)\,\Delta x^K + O(\Delta x^{K+1}).\end{aligned} \tag{4.35}$$

Defining $M_1 = \max_i \left|\left(\frac{1}{2}\right)^K Y_{i_*}(K+1)\right|$ and neglecting the higher order terms yield

$$||G_1(0.5,\Delta x)|| \leq\ M_1 \Delta x^K.$$

Then the general term of $G_2(0.5,\Delta x)$ becomes,

$$\begin{aligned}(G_2(0.5,\Delta x))_i &= \big[(K+1)(0.5)^K Y_{i_*}(K+1)-(K+1)(-0.5)^K\big(Y_{i_*}(K+1)+ \\ &\qquad \Delta x(K+2)Y_{i_*}(K+2)\big]\Delta x^{K-1} \\ &= -(K+1)(K+2)\left(\frac{1}{2}\right)^K Y_{i_*}(K+2)\,\Delta x^K + O(\Delta x^{K+1}).\end{aligned} \tag{4.36}$$

Defining $M_2 = \max_i \left|(K+1)(K+2)\left(\frac{1}{2}\right)^K Y_{i_*}(K+2)\right|$ and neglecting the higher order terms yield

$$||G_2(0.5, \Delta x)|| \leq M_2 \Delta x^K. \tag{4.37}$$

Remembering $G(0.5, \Delta x) = G_1(0.5, \Delta x) + G_2(0.5, \Delta x)$ and taking $L_2 = \max(M_1, M_2)$ lead to the estimate $||G(0.5, \Delta x)|| \leq L_2 \Delta x^K$.

Case2 : Assume that the conditions of Case 1 are not satisfied. Then, the following general terms are found for $G_1(\theta, \Delta x)$

$$\begin{aligned}(G_1(\theta, \Delta x))_i = \big[(1-\theta)^{K+1} Y_{i_*}(K+1) - (-\theta)^{K+1} \big(Y_{i_*}(K+1) + \\ \Delta x(K+2) Y_{i_*}(K+2)\big] \Delta x^K \\ = \left[(1-\theta)^{K+1} - (-\theta)^{K+1}\right] Y_{i_*}(K+1)\, \Delta x^K + O(\Delta x^{K+1}).\end{aligned} \tag{4.38}$$

Defining $M_3 = \max_i \left|\left[(1-\theta)^{K+1} - (-\theta)^{K+1}\right] Y_{i_*}(K+1)\right|$ and neglecting the higher order terms give rise to

$$||G_1(\theta, \Delta x)|| \leq M_3 \Delta x^K. \tag{4.39}$$

Thus the general term of $G_2(\theta, \Delta x)$ becomes,

$$\begin{aligned}(G_2(\theta, \Delta x))_i = \big[(K+1)(1-\theta)^K Y_{i_*}(K+1) - (K+1)(-\theta)^K \big(Y_{i_*}(K+1) + \\ \Delta x(K+2) Y_{i_*}(K+2)\big] \Delta x^{K-1} \\ = \left[(K+1)(1-\theta)^K - (K+1)(-\theta)^K\right] Y_{i_*}(K+1)\, \Delta x^{K-1} + O(\Delta x^K).\end{aligned} \tag{4.40}$$

Defining $M_4 = \max_i \left|\left[(K+1)(1-\theta)^K - (K+1)(-\theta)^K\right] Y_{i_*}(K+1)\right|$ and dropping the higher order terms lead to

$$||G_2(\theta, \Delta x)|| \leq M_4 \Delta x^{K-1}. \tag{4.41}$$

Since $G(\theta, \Delta x) = G_1(\theta, \Delta x) + G_2(\theta, \Delta x)$, taking $L_2 = \max\left(M_3 \Delta x,\ M_4\right)$ leads to the estimate $||G(\theta, \Delta x)|| \leq L_2 \Delta x^{K-1}$. Assume that the following inequalities are satisfied

$$||G(\theta, \Delta x)|| \leq L_2 \Delta x^S \quad \text{and} \quad ||A^*|| \leq L_1 \tag{4.42}$$

where L_2 is defined with respect to the direction parameter θ and $S = K-1,\ K$. Thus, we obtain the following error estimates

$$||\varepsilon|| \leq \begin{cases} L_1 L_2 \Delta x^K, & \theta = 0.5 \text{ and } K \text{ is even} \\ L_1 L_2 \Delta x^{K-1}, & otherwise \end{cases} \tag{4.43}$$

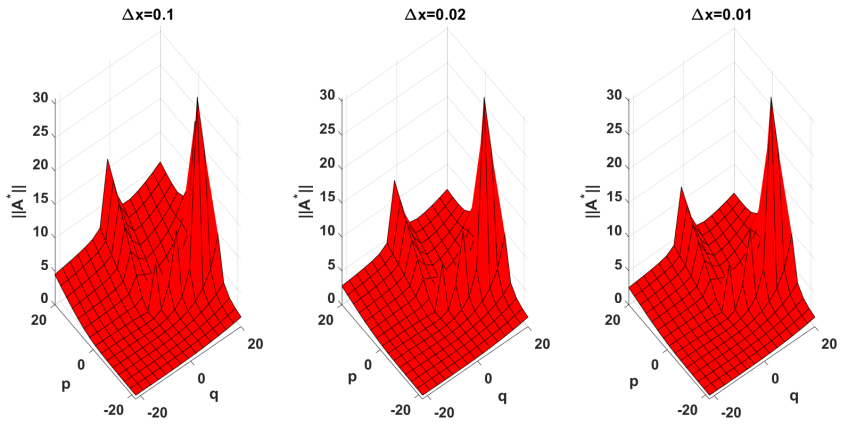

Figure 4.1 $||A^*||_\infty$ norm values as a function of the problem parameters p and q.

Note that, the norm $||A^*||$ is almost independent of the spatial increment Δx. Similar analysis can be executed for boundary value problems of various order. If the order of the differential equation is $2p$, then one needs $2p$ continuity equations to get consistent system of algebraic equations. Since one needs at least C^{2p-1} continuity, the truncation error terms will start with Δx^{K-2p+1} or Δx^{K-2p+2} depending on θ. As a conclusion, if the differential equation is $2p-th$ order and K is the transformation order, then the IELDTM leads to $(K-2p+1)-th$ or $(K-2p+2)-th$ order method depending on the selection of the direction parameter θ.

4.3 Stability Analysis

As explained in the last section, the stability and global errors of the present IELDTM for BVP (4.1-4.2) with the selection of (4.15) clearly depend on the structure of the matrix A defined in (4.27). Thus, the invertibility and the norm of the matrix A determine the stability of the present method. We analyse the bound value L_1 stated in the last subsection with the inequality $||A^*|| = ||\Delta x A^{-1}|| \leq L_1$ depending on p, q, $\Delta x, \theta$ and K. In Figure 4.1, we demonstrate the effect of the parameter values p and q to the matrix norm $||A^*||_\infty$. As seen in Figure 4.1, the bound value L_1 is finite and does not have much negative effect on the stability as clearly seen in equations (4.27). As stated in the error analysis, the norm bound of $||A^*||$ is almost independent of the spatial step size Δx. In Figure 4.2, we have demonstrated the effect of the various values of K and θ to the norm value $||A^*||_\infty$. Asymptotic behaviour can be seen with changing values of K and a little effect of the changing values of θ can be observed in Figure 4.2.

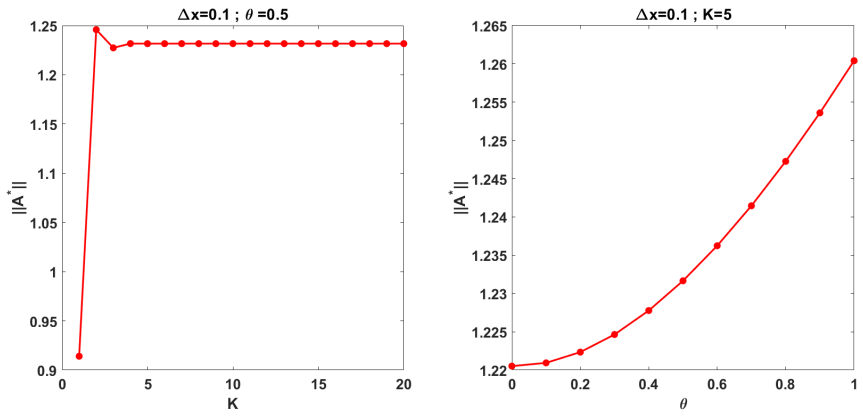

Figure 4.2 Effects of K and θ values on $||A^*||_\infty$ norm for the problem parameter values $p = 3$ and $q = -3$.

4.4 Numerical Experiments

Numerical illustrations of the current method through various boundary value problems are presented here. We considered linear, nonlinear, singular, singularly perturbed, and high order boundary value problems to illustrate the versatility of the IELDTM. The produced results are compared with the exact solutions, the DTM [87], the LDTM [92], the shooting method [58] and the FDM [59]. To measure errors of the present results, we prefer to use the following maximum error norm $||E||_\infty$ and absolute pointwise errors E_i

$$||E||_\infty = max_i|y_i^{numerical} - y_i^{exact}|,$$
$$E_i = |y_i^{numerical} - y_i^{exact}|.$$

Problem 4.1

Consider the following thick pressure vessel problem given in the reference [61]

$$\frac{d^2y}{dx^2} + \frac{1}{x}\frac{dy}{dx} - \frac{y}{x^2} = 0, \quad 5 \leq x \leq 8 \tag{4.44}$$

with the boundary conditions

$$y(5) = 0.0038731 \quad \text{and} \quad y(8) = 0.0030770 \tag{4.45}$$

where the exact solution is $y(x) = (10501x)/78000000 + 77999/(4875000x)$. The

differential transform of equation (4.44) is

$$Y_i(k+2) = \frac{1}{(k+1)(k+2)}\Big[\sum_{n=0}^{k}(-(k-n+1)F(n,x_i)Y_i(k-n+1)+ G(n,x_i)Y_i(k-n)\Big] \tag{4.46}$$

where Y_i is the transformed form of the local function $y_i(x)$. $F(n,x_i)$ and $G(n,x_i)$ can be defined as follows

$$F(n,x_i) = \frac{(-1)^n}{(x_i)^{n+1}} \quad and \quad G(n,x_i) = \frac{(-1)^n(n+1)}{(x_i)^{n+2}}, \quad i = 0,1,\ldots,N \tag{4.47}$$

where the nodes are defined as $x_i = 5 + i\Delta x$ and $\Delta x = \frac{3}{N}$. The boundary conditions lead to the following equations,

$$Y_0(0) = 0.0038731 \quad \text{and} \quad Y_N(0) = 0.0030770. \tag{4.48}$$

For a typical element $[x_i, x_{i+1}]$, the present IELDTM yields the following equations,

$$\sum_{k=0}^{K} Y_{i+1}(k)(-\theta\Delta x)^k = \sum_{k=0}^{K} Y_i(k)((1-\theta)\Delta x)^k, \tag{4.49}$$

$$\sum_{k=1}^{K} Y_{i+1}(k)\,k\,(-\theta\Delta x)^{k-1} = \sum_{k=1}^{K} Y_i(k)\,k\,((1-\theta)\Delta x)^{k-1}, \tag{4.50}$$

where $i = 0,1,\ldots,N-1$ and the coefficients $Y_i(k)$ can be calculated from the relation (4.46) in terms of $Y_i(0)$ and $Y_i(1)$.

Problem 4.2

Consider the following singularly perturbed nonlinear BVP [73],

$$-\varepsilon\frac{d^2y}{dx^2} + y + y^2 = e^{\frac{-2x}{\sqrt{\varepsilon}}}, \quad 0 \le x \le 1 \tag{4.51}$$

with the following boundary conditions

$$y(0) = 1 \quad \text{and} \quad y(1) = e^{\frac{-1}{\sqrt{\varepsilon}}} \tag{4.52}$$

where $\varepsilon \ll 1$ and the exact solution is $y(x) = e^{\frac{-x}{\sqrt{\varepsilon}}}$. The transform of equation (4.51) is

$$Y_i(k+2) = \frac{1}{\varepsilon(k+1)(k+2)}\left[Y_i(k) + \bar{Y}_i(2;0,0;k) - F(k,x_i)\right] \tag{4.53}$$

where Y_i is the transformed form of the local function $y_i(x)$ and $\bar{Y}_i$ is defined in Appendix A. $F(k, x_i)$ can be defined as follows

$$F(k, x_i) = \frac{1}{k!}\left(\frac{-2}{\sqrt{\varepsilon}}\right)^k e^{\frac{-2x_i}{\sqrt{\varepsilon}}}, \tag{4.54}$$

where the nodes are expressed by $x_i = i\Delta x$ and $\Delta x = \frac{1}{N}$. The boundary conditions lead to the following equations,

$$Y_0(0) = 1 \quad \text{and} \quad Y_N(0) = e^{\frac{-1}{\sqrt{\varepsilon}}}. \tag{4.55}$$

The present IELDTM yields the same expressions defined in (4.49-4.50) and the required coefficients can be obtained from equation (4.53) in terms of $Y_i(0)$ and $Y_i(1)$.

Problem 4.3

Consider the following fourth-order BVP [71]

$$\frac{d^4y}{dx^4} = \sin(x) + sin^2(x) - \left(\frac{d^2y}{dx^2}\right)^2, \quad 0 \leq x \leq 1 \tag{4.56}$$

with the following boundary conditions

$$y(0) = 0, \quad y'(0) = 1, \quad y(1) = sin(1) \quad \text{and} \quad y'(1) = cos(1), \tag{4.57}$$

where the exact solution is $y(x) = sin(x)$. The transform of equation (4.56) is

$$\begin{aligned} Y_i(k+4) = \frac{1}{(k+1)(k+2)(k+3)(k+4)}\Bigg[& F(k, x_i) + \\ & \sum_{n=0}^{k} F(n,, x_i) F(k-n,, x_i) - \bar{Y}_i(0; 2, 2; k) \Bigg] \end{aligned} \tag{4.58}$$

where Y_i is the transformed form of the local function $y_i(x)$ and $\bar{Y}_i$ is defined in Appendix A. $F(k, x_i)$ can be defined by

$$F(k, x_i) = \frac{1}{k!}\frac{d^k(sinx)}{dx^k}\bigg|_{x=x_i}, \tag{4.59}$$

where the nodes are defined by $x_i = i\Delta x$ and $\Delta x = \frac{1}{N}$. The boundary conditions lead to the following equations,

$$Y_0(0) = 0, \quad Y_0(1) = 1, \quad Y_N(0) = sin(1) \quad \text{and} \quad Y_N(1) = cos(1). \tag{4.60}$$

For a typical element $[x_i, x_{i+1}]$, the present IELDTM yields

$$\sum_{k=0}^{K} Y_{i+1}(k)(-\theta\Delta x)^k = \sum_{k=0}^{K} Y_i(k)((1-\theta)\Delta x)^k,$$

$$\sum_{k=1}^{K} Y_{i+1}(k)\,k(-\theta\Delta x)^{k-1} = \sum_{k=1}^{K} Y_i(k)\,k((1-\theta)\Delta x)^{k-1}, \tag{4.61}$$

$$\sum_{k=2}^{K} Y_{i+1}(k)\,k(k-1)(-\theta\Delta x)^{k-2} = \sum_{k=2}^{K} Y_i(k)\,k(k-1)((1-\theta)\Delta x)^{k-2},$$

$$\sum_{k=3}^{K} Y_{i+1}(k)\,k(k-1)(k-2)(-\theta\Delta x)^{k-3} = \sum_{k=3}^{K} Y_i(k)\,k(k-1)(k-2)((1-\theta)\Delta x)^{k-3},$$

where $i = 0, 1, \ldots, N-1$ and the coefficients can be calculated from equation (4.58) in terms of $Y_i(0)$, $Y_i(1)$, $Y_i(2)$ and $Y_i(3)$.

Problem 4.4

Consider the following second order nonlinear BVP [76]

$$\frac{d^2y}{dx^2} = \frac{1}{8}\left(32 + 2x^3 - y\frac{dy}{dx}\right), \quad 1 \leq x \leq 2 \tag{4.62}$$

with the boundary conditions

$$y(1) = 17 \quad \text{and} \quad y'(2) = 0 \tag{4.63}$$

where the exact solution is $y(x) = x^2 + \frac{16}{x}$. The transform of equation (4.62) is

$$Y_i(k+2) = \frac{1}{8(k+1)(k+2)}\left[32\delta_{0k} + 2\,F(k, x_i) - \bar{Y}_i(1;1,1;k)\right] \tag{4.64}$$

where Y_i is the transformed form of the local function $y_i(x)$, δ is Kronecker delta, $\bar{Y}_i$ is defined in Appendix A. $F(k, x_i)$ is defined by

$$F(k, x_i) = \begin{cases} \binom{3}{k} x_i^{3-k}, & k = 0, 1, 2, 3 \\ 0, & otherwise \end{cases} \tag{4.65}$$

where the nodes are defined by $x_i = 1 + i\Delta x$ and $\Delta x = \frac{1}{N}$. The boundary conditions lead to the following equations,

$$Y_0(0) = 17 \quad \text{and} \quad Y_N(1) = 0. \tag{4.66}$$

The present IELDTM yields the same expressions defined in (4.49-4.50) and the required coefficients can be obtained from equation (4.64) in terms of $Y_i(0)$ and $Y_i(1)$.

Problem 4.5

Consider the following nonlinear BVP [76],

$$\frac{d^2y}{dx^2} = \frac{1}{x^3}\left(\frac{dy}{dx}\right)^2 - 9\frac{y^2}{x^5} + 4x, \quad 1 \leq x \leq 2 \tag{4.67}$$

with the boundary conditions

$$y(1) = 0 \quad \text{and} \quad y'(2) = 4 + 12ln(2) \tag{4.68}$$

where the exact solution is $y(x) = x^3 ln(x)$. The transform of equation (4.62) is

$$Y_i(k+2) = \frac{1}{(k+1)(k+2)}\Bigg[\sum_{n=0}^{k}\Big(F(n,x_i)\,\bar{Y}_i(0;1,2;k-n) - 9G(n,x_i)\,\bar{Y}_i(2;0,0;k-n) + 4H(k,x_i)\Big)\Bigg] \tag{4.69}$$

where Y_i is the transformed form of the local function $y_i(x)$ and $\bar{Y}_i$ is defined in Appexdix A. $F(n,x_i)$, $G(n,x_i)$ and $H(k,x_i)$ are defined by

$$F(n,x_i) = \frac{(-1)^n(n+1)(n+2)}{2(x_i)^{n+3}}, \tag{4.70}$$

$$G(n,x_i) = \frac{(-1)^n(n+1)(n+2)(n+3)(n+4)}{24(x_i)^{n+5}}, \tag{4.71}$$

$$H(k,x_i) = \begin{cases} x_i^{1-k}, & k = 0,1 \\ 0, & otherwise \end{cases} \tag{4.72}$$

where the nodes are defined by $x_i = 1 + i\Delta x$ and $\Delta x = \frac{1}{N}$. The boundary conditions lead to the following equations,

$$Y_0(0) = 0 \quad \text{and} \quad Y_N(1) = 4 + 12ln(2). \tag{4.73}$$

The present IELDTM yields the same expressions defined in (4.49-4.50) and the coefficients can be evaluated from equation (4.69) in terms of $Y_i(0)$ and $Y_i(1)$.

Problem 4.6

Consider the following nonlinear singular BVP [58],

$$\frac{d^2y}{dx^2}+\frac{2}{x}\frac{dy}{dx}=-y^5,\quad 0<x<1 \tag{4.74}$$

with the boundary conditions

$$y'(0)=0\quad\text{and}\quad y(1)=\frac{\sqrt{3}}{2}. \tag{4.75}$$

where the exact solution is $y(x)=\left(1+x^2/3\right)^{-1/2}$. The transform of equation (4.4) is

$$Y_0(k+2)=\frac{1}{(k+1)(k+2)}\left[-\bar{Y}_0(5;0,0;k)\right] \tag{4.76}$$

and

$$Y_i(k+2)=-\frac{1}{(k+1)(k+2)}\left[2\sum_{n=0}^{k}F(n,x_i)\,Y_i(k-n)+\bar{Y}_i(5;0,0;k)\right], \tag{4.77}$$

where $i=1,2,...,N$, Y_i is the transformed form of the local function $y_i(x)$, $\bar{Y}_i$ is defined in Appendix A and $F(n,x_i)$ is defined as

$$F(n,x_i)=\frac{(-1)^n}{(x_i)^{n+1}}\quad i=0,1,\ldots,N \tag{4.78}$$

The nodes are defined with $x_i=i\Delta x$ and $\Delta x=\frac{1}{N}$. The boundary conditions lead to

$$Y_0(1)=0\quad\text{and}\quad Y_N(0)=\frac{\sqrt{3}}{2}. \tag{4.79}$$

The present IELDTM yields the same expressions defined in (4.49-4.50) and the coefficients can be evaluated from equation (4.76-4.77) in terms of $Y_i(0)$ and $Y_i(1)$.

In Figures 4.3-4.7, we demonstrate the effectiveness of the current IELDTM and the DTM on various BVPs, taking into account pointwise errors. As clearly seen in the figures, the present IELDTM with $\theta=0.5$ is more accurate than the LDTM ($\theta=0$) and the DTM for all the types of linear and nonlinear BVPs. With the consideration of the same transformation order K, numerically constructed IELDTM is seen to be far more efficient than the semi-analytical DTM. While the DTM produces acceptable results for Problems 4.1 and 4.3, the DTM results are seen to be divergent for Problem 4.2 and Problems 4.4-4.5 as shown in Figure 4.4, Figure 4.8 and Figure 4.9. Thus, the convergence, accuracy, and versatility of the IELDTM are far more reliable than that of

the DTM and the LDTM. To demonstrate the effect of the spatial step size Δx and the transformation order K, the maximum errors produced with various parameter values are presented in Tables 4.1–4.5. In line with our theoretical analysis, the accuracy of the present IELDTM is increasing with either Δx-refinement or K-refinement for all problems. Both the semi-analytical DTM [87] and the numerical LDTM [92] suffer from instability, especially in solving stiff BVPs. Just at this time, the IELDTM is here constructed for the elimination of this serious drawback. Since Problem 4.2 has stiff nature for $\varepsilon \ll 1$, the IELDTM and the rival methods are compared with the calculation of $||E||_\infty$ error norms for various challenging ε values in Table 4.6. As observed from the table, the DTM is completely divergent, and the LDTM produces divergent results for $\varepsilon < 0.001$. The IELDTM with $\theta = 0$ and $\theta = 0.5$ have ability to produce accurate numerical solutions up to $\varepsilon = 0.00001$ with $\Delta x = 0.01$. To produce the results in Table 4.6, the LDTM and the DTM are treated as a shooting method as previously considered in literature [92, 145] and the IELDTM with $\theta = 0$ and $\theta = 0.5$ are treated as a direct method as mathematically derived in Section 4.1.

The numerical nature of the current IELDTM depends heavily on the choice of the direction parameter value θ. Figure 4.10 has shown how the maximum error norms vary with respect to the direction parameter θ. It is obvious that the selections $\theta \cong 0.5$ provide more accurate results as already proven in error analysis. In Figure 4.11, processing times of the present IELDTM algorithms for solving Problems 4.4-4.5 are illustrated with the changing values of the transformation order K. It is possible to observe that the computational cost increases linearly according to the transformation order K. In Table 4.7, the theoretical order expectations and experimental order averages of the IELDTM are compared with various values of K. The highly matching experiment results are seen in the table, and the theoretical expectations stated in the error analysis are verified. The experimental order is calculated as $p = log(E_1/E_2)/log(\Delta x_1/\Delta x_2)$ where Δx_1 and Δx_2 are spatial increments and E_1 and E_2 are the corresponding absolute errors. By evaluating the experimental orders for various Δx_1 and Δx_2 values, the experimental average order is obtained for each problem. In addition to the above and those suggested here, also high order shooting methods and finite difference methods (FDM) have been widely used in the solution of various versions of BVPs in the literature [58, 59, 61–69]. To prove the efficiency of the proposed IELDTM, the error norms $||E||_\infty$ of the present algorithm, the sixth order shooting method [58] and the fourth-order FDM [59] are compared in Table 4.8. The IELDTM of orders four and six are used for computations with $\theta = 0.5$. As clearly observed from the table, our sixth order method produces far more accurate results than the sixth order shooting method [58]. Similarly, the present fourth-order IELDTM offers far more accurate results than that of the fourth-order FDM [59]. The

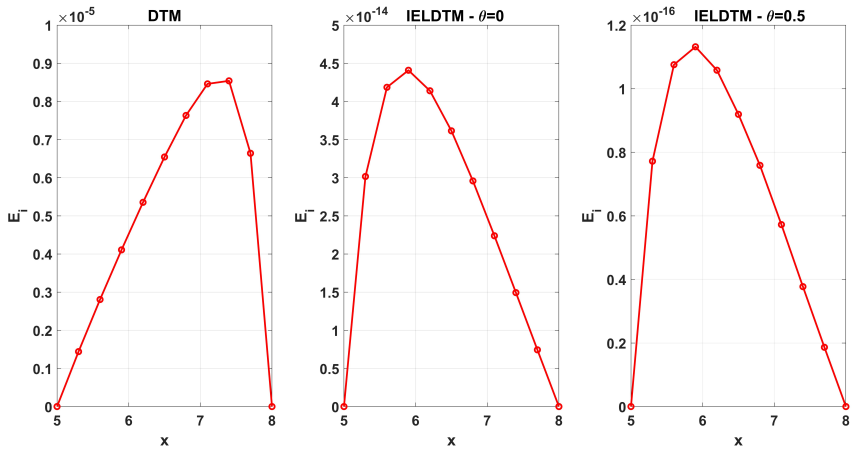

Figure 4.3 Pointwise errors produced by the IELDTM and the DTM with $\Delta x = 0.1$ and $K = 9$ for Problem 4.1.

CPU times required to produce the present results are also demonstrated in Table 4.8. As illustrated both qualitatively and quantitatively, the present IELDTM is an effective numerical method in terms of reliability, versatility, and computational efficiency.

Table 4.1 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.1.

K/N	N=5	N=10	N=15	N=20	N=25	N=30
K=3	6.62E-07	1.63E-07	7.22E-08	4.06E-08	2.60E-08	1.80E-08
K=4	7.84E-09	4.76E-10	9.46E-11	2.99E-11	1.22E-11	5.89E-12
K=5	2.51E-09	1.51E-10	2.99E-11	9.45E-12	3.86E-12	1.86E-12
K=6	4.39E-11	6.75E-13	5.87E-14	1.04E-14	2.73E-15	9.12E-16
K=7	8.89E-12	1.33E-13	1.15E-14	2.04E-15	5.33E-16	1.78E-16
K=8	2.06E-13	7.91E-16	3.12E-17	3.47E-18	2.17E-18	2.60E-18

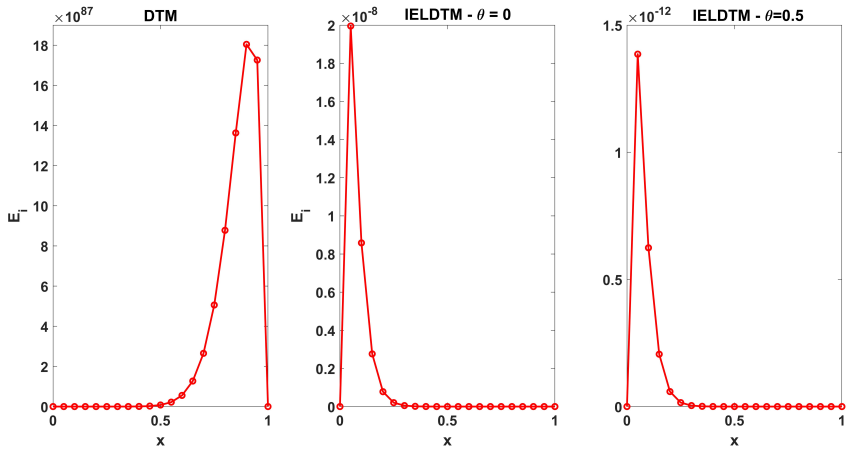

Figure 4.4 Pointwise errors produced by the IELDTM and the DTM with $\Delta x = 0.05$, $\varepsilon = 0.001$ and $K = 13$ for Problem 4.2.

Table 4.2 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.2.

K/N	N=20	N=25	N=30	N=35	N=40	N=45
K=5	4.26E-04	1.86E-04	9.12E-05	4.88E-05	2.80E-05	1.70E-05
K=6	2.93E-05	8.32E-06	2.87E-06	1.14E-06	5.05E-07	2.44E-07
K=7	6.63E-06	1.82E-06	6.14E-07	2.40E-07	1.05E-07	5.03E-08
K=8	3.60E-07	6.54E-08	1.57E-08	4.56E-09	1.54E-09	5.87E-10
K=9	5.91E-08	1.03E-08	2.40E-09	6.87E-10	2.30E-10	8.67E-11
K=10	2.66E-09	3.08E-10	5.11E-11	1.09E-11	2.83E-12	8.50E-13
K=11	3.42E-10	3.78E-11	6.10E-12	1.28E-12	3.28E-13	9.77E-14
K=12	1.31E-11	9.72E-13	1.12E-13	1.74E-14	3.44E-15	7.77E-16

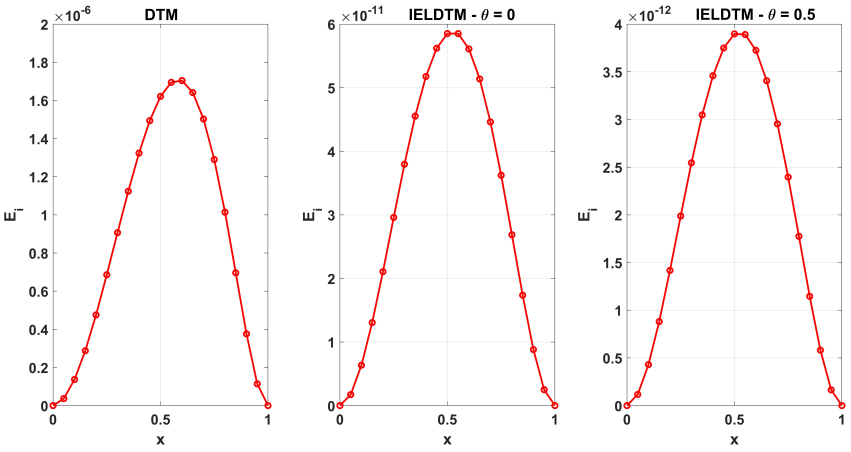

Figure 4.5 Pointwise errors produced by the IELDTM and the DTM with $\Delta x = 0.05$ and $K = 7$ for Problem 4.3.

Table 4.3 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.3.

K/N	N=5	N=10	N=15	N=20	N=25	N=30
K=5	1.90E-06	4.99E-07	2.22E-07	1.25E-07	8.02E-08	5.56E-08
K=6	2.83E-09	1.86E-10	3.70E-11	1.17E-11	4.81E-12	2.32E-12
K=7	9.51E-10	6.23E-11	1.24E-11	3.90E-12	1.60E-12	7.73E-13
K=8	1.13E-12	1.83E-14	1.67E-15	4.44E-16	2.22E-16	5.55E-16
K=9	2.26E-13	3.83E-15	3.33E-16	1.11E-16	2.22E-16	3.33E-16
K=10	3.33E-16	2.22E-16	2.22E-16	2.22E-16	2.22E-16	4.44E-16

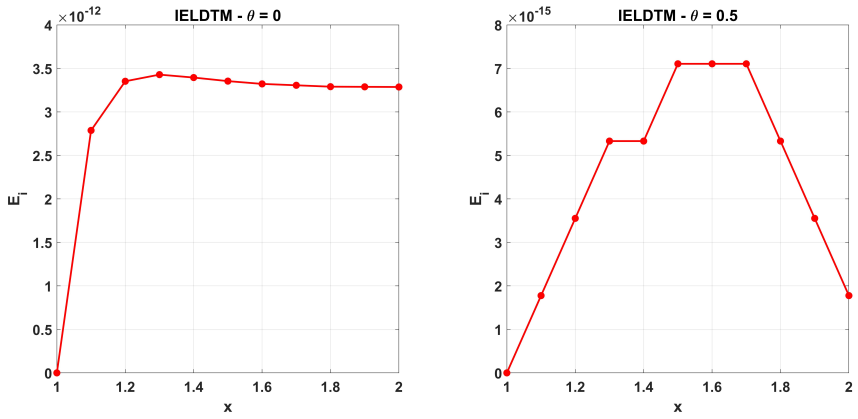

Figure 4.6 Pointwise errors produced by the IELDTM with $\Delta x = 0.1$ and $K = 13$ for Problem 4.4.

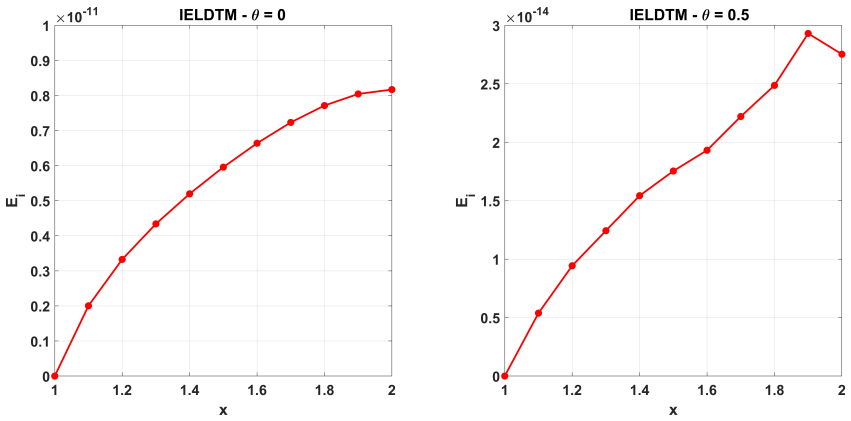

Figure 4.7 Pointwise errors produced by the IELDTM with $\Delta x = 0.1$ and $K = 9$ for Problem 4.5.

Table 4.4 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.4

K/N	N=5	N=10	N=15	N=20	N=25	N=30
K=3	4.84E-02	1.18E-02	5.20E-03	2.92E-03	1.87E-03	1.30E-03
K=4	1.16E-03	6.85E-05	1.34E-05	4.22E-06	1.73E-06	8.31E-07
K=5	3.71E-04	2.13E-05	4.14E-06	1.30E-06	5.32E-07	2.56E-07
K=6	1.41E-05	2.01E-07	1.73E-08	3.06E-09	7.99E-10	2.67E-10
K=7	2.96E-06	3.92E-08	3.33E-09	5.87E-10	1.53E-10	5.11E-11
K=8	1.56E-07	5.29E-10	2.00E-11	1.97E-12	3.53E-13	8.88E-14
K=9	2.57E-08	7.66E-11	2.82E-12	2.93E-13	4.09E-14	2.84E-14
K=10	1.73E-09	1.38E-12	2.31E-14	1.95E-14	1.95E-14	1.24E-14

Table 4.5 Maximum errors of the IELDTM with $\theta = 0.5$ and various values of the parameters K and N for Problem 4.5.

K/N	N=5	N=10	N=15	N=20	N=25	N=30
K=3	3.13E-03	7.77E-04	3.45E-04	1.94E-04	1.24E-04	8.62E-05
K=4	1.13E-06	6.29E-08	1.22E-08	3.83E-09	1.56E-09	7.53E-10
K=5	1.51E-06	9.23E-08	1.82E-08	5.74E-09	2.35E-09	1.13E-09
K=6	9.62E-09	1.41E-10	1.22E-11	2.16E-12	5.63E-13	1.94E-13
K=7	2.42E-09	3.54E-11	3.07E-12	5.53E-13	1.41E-13	4.88E-14
K=8	3.95E-11	1.38E-13	9.77E-15	7.11E-15	7.11E-15	4.44E-15
K=9	6.47E-12	2.93E-14	3.55E-15	4.44E-15	2.66E-15	3.55E-15

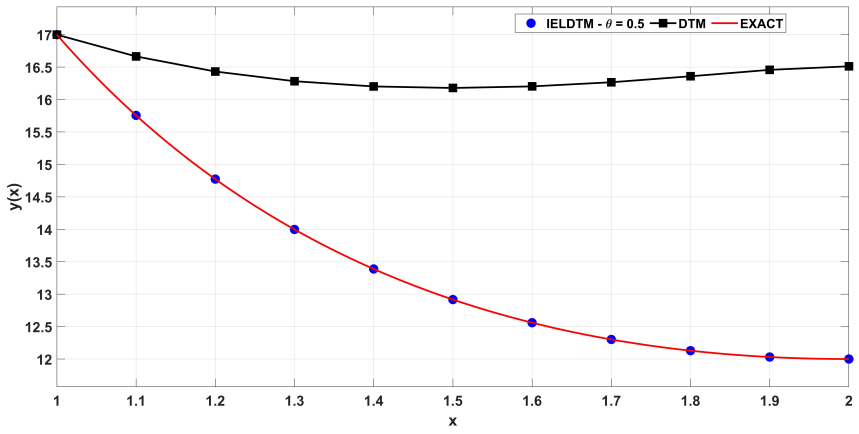

Figure 4.8 Comparison of numerical results produced by the IELDTM and the DTM with $\Delta x = 0.1$ and $K = 11$ for Problem 4.4.

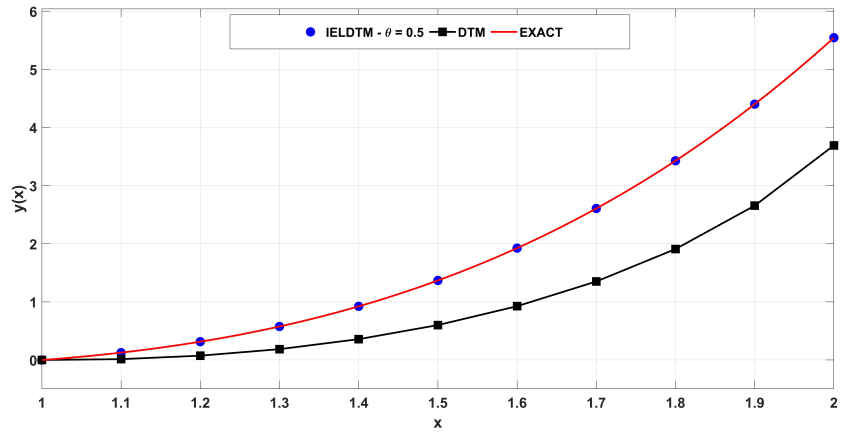

Figure 4.9 Comparison of numerical results produced by the IELDTM and the DTM with $\Delta x = 0.1$ and $K = 9$ for Problem 4.5.

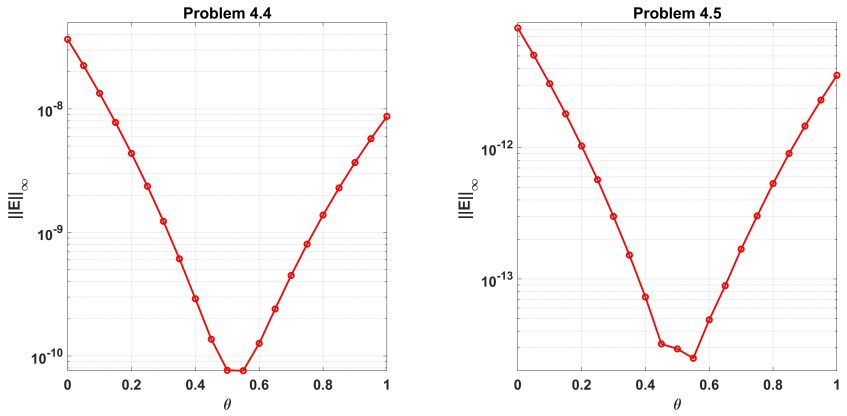

Figure 4.10 Effect of the direction parameter θ to maximum errors produced by the IELDTM with $\Delta x = 0.1$ and $K = 9$.

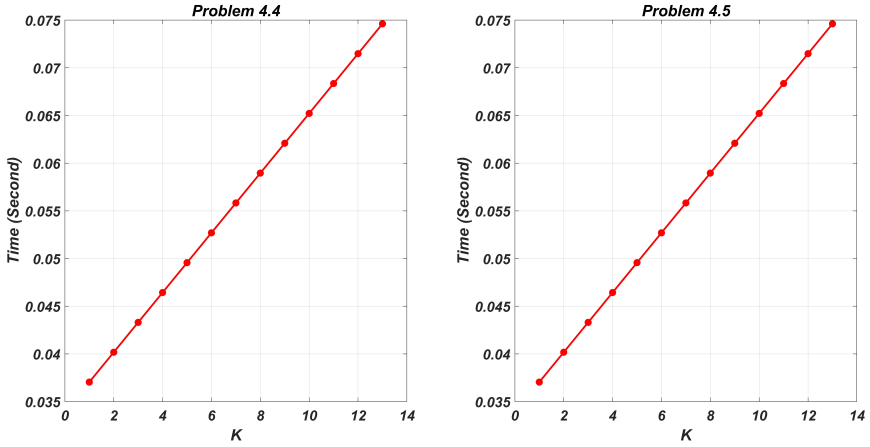

Figure 4.11 Processing times with respect to varying values of the transformation order K with $\Delta x = 0.1$ and $\theta = 0.5$.

Table 4.6 Comparison of $||E||_\infty$ norms of the IELDTM, LDTM and DTM with $K = 6$ and $\Delta x = 0.01$ for Problem 4.2.

ε	DTM	LDTM	IELDTM($\theta = 0$)	IELDTM($\theta = 0.5$)
1	divergent	2.15E-14	2.15E-14	4.44E-16
0.5	divergent	7.22E-14	7.22E-14	2.66E-15
0.4	divergent	1.38E-14	1.38E-14	8.88E-16
0.3	divergent	3.17E-13	3.17E-13	3.33E-16
0.2	divergent	9.76E-13	9.76E-13	6.66E-13
0.1	divergent	6.10E-12	6.10E-12	2.00E-15
0.01	divergent	2.08E-09	2.08E-09	2.16E-12
0.0015	divergent	6.29E-06	2.67E-07	6.38E-10
0.001	divergent	1.00E-02	7.67E-07	2.15E-09
0.0001	divergent	divergent	4.12E-04	2.10E-08
0.00001	divergent	divergent	4.09E-01	0.90E-02

Table 4.7 Validation of the approximation orders of the present IELDTM with the direction parameter $\theta = 0.5$ for Problems 4.4-4.5.

	Problem 4.4		Problem 4.5	
K	Experimental Average Order	Theoretical Expectation	Experimental Average Order	Theoretical Expectation
K=3	2.016	2	2.003	2
K=4	4.030	4	4.062	4
K=5	4.047	4	4.013	4
K=6	6.054	6	6.042	6
K=7	6.095	6	6.049	6

Table 4.8 Comparison of the IELDTM and the rival higher order methods with $||E||_\infty$ norms for Problem 4.6.

N	Shooting Method [58]	FDM [59]	IELDTM; $K = 4$	CPU Time (s)	IELDTM; $K = 6$	CPU Time (s)
8	1.23E-06		4.36E-07	0.025 s	9.85E-10	0.035 s
16	3.47E-08	4.78E-04	2.72E-08	0.048 s	1.54E-11	0.084 s
32	8.40E-10	3.31E-05	1.70E-09	0.127 s	2.43E-13	0.194 s
64	2.05E-11	2.12E-06	1.06E-10	0.539 s	4.21E-15	0.559 s

4.5 Summary

In this chapter, the IELDTM has been produced for solving BVPs. Based on the DTM and the local DTM idea, a high-order, direction free and stability-preserved numerical method has been presented. Priori global error analysis has been done for the second-order test problem, and the order conditions of the IELDTM have been determined according to the direction parameter. Stability properties of the IELDTM have been explained with the use of approximation matrices. Various linear and nonlinear BVPs have been considered as test problems, including singular and singularly perturbed BVPs.

As proven theoretically and experimentally throughout this chapter, the IELDTM is a numerically reliable, high-order, stable, and computationally efficient method for solving BVPs. One of the main advantages of the method proposed here is that the degrees of freedom are not affected by the order of the method, i.e. the IELDTM is an optimized numerical technique. It has been proven that the degrees of freedom depends only on the order of the differential equation concerned. If the differential equation is of order p and N nodes are used, then the degrees of freedom is $2pN$ irrespective of the local transformation order K. It has thus been proven that the convergence rate of the IELDTM increases with either $p-$ refinement or $h-$ refinement. If the number of terms used in local approximations increases, then the $p-$refinement procedure can be easily applied for the IELDTM. The central approach with $\theta = 0.5$ has been proven to be more accurate than both other explicit ($\theta = 0$) and implicit ($\theta \neq 0$) selections. With the completion of Chapters 3 and 4, the IELDTM has been produced for IVPs for ODEs and BVPs for ODEs. In the next chapter, the IELDTM is derived for time integrations of the partial differential equations.

5
THE IELDTM FOR TIME INTEGRATION OF THE AD EQUATIONS

In Chapter 3, the IELDTM was derived for the numerical investigations of the stiff IVPs of ODEs. Since IVPs of ODEs also arise in the reduced space-time parabolic PDEs [42, 46–48], it is crucial to observe the behaviour of the IELDTM for time-integrations of the reduced parabolic PDEs. Since the spatial discretization of the parabolic PDEs leads to large IVPs, the performance of the used time-integration technique is highly important for optimizing the computational cost. As proven in Chapter 3, the IELDTM has distinct numerical characteristics over stiff IVPs and it is important to test its efficiency over larger stiff systems.

The outline of this chapter is divided into six sections. First of all, the two-dimensional nonlinear advection-diffusion (AD) equation is introduced. In the second phase of this chapter, the Chebyshev spectral collocation method is derived for the spatial discretization of the two-dimensional AD equation. As discussed in Chapter 2, the ChSCM provides important interpolation characteristics for the numerical solutions of differential equations. The ChSCM reduces the two-dimensional AD equation into matrix differential equations. The IELDTM is then derived for the matrix differential equations with the consideration of similar theoretical basis introduced in Chapter 3. The priori error analysis together with the stability conditions of the IELDTM is theoretically derived, and the adaptivity equations are produced. In numerical experiments, one- and two-dimensional Burgers equations are investigated with their advection-dominated cases. At the end of this chapter, the conclusions are provided.

5.1 Two-Dimensional Nonlinear Advection-Diffusion Equation

Consider the following two-dimensional nonlinear advection-diffusion equation (Burgers equation) [48]

$$u_t + uu_x + uu_y = D\left(u_{xx} + u_{yy}\right), \quad (x, y) \in W,\ t > 0 \tag{5.1}$$

with the boundary conditions

$$u(a,y,t)=f_1(y,t),\ \ u(b,y,t)=f_2(y,t) \tag{5.2}$$

$$u(x,c,t)=h_1(x,t),\ \ u(x,d,t)=h_2(x,t) \tag{5.3}$$

and initial condition

$$u(x,y,0)=g(x,y) \tag{5.4}$$

where $W=\{(x,y)\,|\,a\le x\le b, c\le y\le d\ \}$, D is the kinematic viscosity constant with $D>0$ and f_1, f_2, h_1, h_2, and g are the known smooth functions. The subscripts x, y and t represent differentiations with respect to spaces x, y and time t, respectively.

The Burgers equation is one of the most encountered nonlinear parabolic PDE for diffusive waves in computational fluid dynamics. The corresponding equation has also many application areas including the theory of shock waves, sound waves in a viscous medium, mathematical modelling of turbulent fluid and so on. The Burgers equation was firstly introduced by Bateman [146] and proposed the steady-state solution of the equation. Burgers [147] introduced this equation to model turbulent fluid in a channel caused by the opposite effects of convection and diffusion. The structure of the Burgers equation is similar to that of the Navier-Stokes equations without stress term due to the presence of the nonlinear advection term and linear diffusion term. The Burgers equation was solved exactly by using the Hopf-Cole transformation [148, 149] which converts the equation to a heat diffusion equation. In most of those cases, the solutions involve infinite series which may diverge or converge very slowly for relatively small values of the kinematic viscosity constant D, which corresponds to steep wavefronts in the propagation of the dynamic wave forms. In such scenarios, it is inevitable to construct a space-time numerical method to observe the physical behaviour of the nonlinear mechanism.

5.2 The ChSCM for Spatial Variation

Let us approximate the solution $u(x,y,t)$ with $(N,\ M)$ order Chebyshev polynomials as follows [48]

$$u(x,y,t)=\sum_{n=0}^{N}\sum_{m=0}^{M}\alpha_{nm}\beta_{nm}(t)\overline{T}_n(x)\overline{T}_m(y) \tag{5.5}$$

where $\beta_{nm}(t)$ are time dependent parts of the representation, $\alpha_{nm}=1$ for interior points and $\alpha_{nm}=1/2$ for any boundary point. In equation (5.5), $\overline{T}_n(x)$ and $\overline{T}_m(y)$

are defined in terms of the first kind Chebyshev polynomials $T_n(x)$ and $T_m(y)$ as

$$\overline{T}_n(x)=T_n\left(\frac{2x-(a+b)}{b-a}\right)=cos\left(n\ arccos\left(\frac{2x-(a+b)}{b-a}\right)\right), \tag{5.6}$$

$$\overline{T}_m(y)=T_m\left(\frac{2y-(c+d)}{d-c}\right)=cos\left(m\ arccos\left(\frac{2y-(c+d)}{d-c}\right)\right). \tag{5.7}$$

Let us define the following CGL collocation points on $[a,\ b]$ and $[c,\ d]$ as

$$x_n=\frac{1}{2}\left((a+b)-(b-a)cos\left(\frac{\pi n}{N}\right)\right),\quad n=0,1,2,\ldots,\ N,$$

$$y_m=\frac{1}{2}\left((c+d)-(d-c)cos\left(\frac{\pi m}{M}\right)\right),\quad m=0,1,2,\ldots,\ M.$$

The discrete orthogonality relation stated in Chapter 2 is vital for the ChSCM defined by

$$\sum_{n=0}^{N}\alpha_n\overline{T}_i(x_n)\overline{T}_j(x_n)=\gamma_i\delta_{ij}, \tag{5.8}$$

$$\sum_{m=0}^{M}\alpha_m\overline{T}_i(y_m)\overline{T}_j(y_m)=\omega_i\delta_{ij} \tag{5.9}$$

with

$$\gamma_i=\begin{cases}\frac{N}{2}, & i\neq 0,\ N\\ N, & i=0,\ N,\end{cases} \tag{5.10}$$

$$\omega_i=\begin{cases}\frac{M}{2}, & i\neq 0,\ M\\ M, & i=0,\ M.\end{cases} \tag{5.11}$$

Thus, we reach the required derivatives at predetermined collocation points in equation (5.1) as follows:

$$u_x\left(x_i,y_j,\ t\right)=\sum_{n=0}^{N}[A_x]_{in}u_{nj}(t), \tag{5.12}$$

$$u_{xx}\left(x_i,y_j,t\right)=\sum_{n=0}^{N}[B_x]_{in}u_{nj}(t), \tag{5.13}$$

$$u_y\left(x_i,y_j,\ t\right)=\sum_{m=0}^{M}\left[A_y\right]_{jm}u_{im}(t), \tag{5.14}$$

$$u_{yy}\left(x_i,y_j,t\right)=\sum_{m=0}^{M}\left[B_y\right]_{jm}u_{im}(t), \tag{5.15}$$

where $i = 0, 1, 2, \ldots, N$, $j = 0, 1, 2, \ldots, M$, $u_{nm}(t) = u(x_n, y_m, t)$, $B_x = A_x^2$, $B_y = A_y^2$, A_x and A_y are $(N \times N)$ and $(M \times M)$ matrices are defined by

$$[A_x]_{in} = \begin{cases} \xi_n^1 \mu_1 \sum_{j=0}^{N} (-1)^{j+1} j^2 cos\left(j\left(\pi - \frac{n\pi}{N}\right)\right), & i = 0 \\ \xi_n^1 \mu_1 \sum_{j=0}^{N} \frac{jsin\left(j\left(\pi - \frac{i\pi}{N}\right)\right) cos\left(j\left(\pi - \frac{n\pi}{N}\right)\right)}{sin\left(\pi - \frac{i\pi}{N}\right)}, & i = 1, 2, \ldots, N-1 \\ \xi_n^1 \mu_1 \sum_{j=0}^{N} j^2 cos\left(j\left(\pi - \frac{n\pi}{N}\right)\right), & i = N \end{cases} \tag{5.16}$$

$$\left[A_y\right]_{im} = \begin{cases} \xi_m^2 \mu_2 \sum_{j=0}^{M} (-1)^{j+1} j^2 cos\left(j\left(\pi - \frac{m\pi}{M}\right)\right), & i = 0 \\ \xi_m^2 \mu_2 \sum_{j=0}^{M} \frac{jsin\left(j\left(\pi - \frac{i\pi}{M}\right)\right) cos\left(j\left(\pi - \frac{m\pi}{M}\right)\right)}{sin\left(\pi - \frac{i\pi}{M}\right)}, & i = 1, 2, \ldots, M-1 \\ \xi_m^2 \mu_2 \sum_{j=0}^{M} j^2 cos\left(j\left(\pi - \frac{m\pi}{M}\right)\right), & i = M \end{cases} \tag{5.17}$$

with

$$\xi_n^1 = \begin{cases} \frac{1}{N}, & n = 0, N \\ \frac{2}{N}, & n \neq 0, N \end{cases} \quad \text{and} \quad \xi_m^2 = \begin{cases} \frac{1}{M}, & m = 0, M \\ \frac{2}{M}, & m \neq 0, M \end{cases}$$

where $\mu_1 = \frac{2}{b-a}$, $\mu_2 = \frac{2}{d-c}$, $[A_x]_{in}$ and $\left[A_y\right]_{im}$ are called the Chebyshev differentiation matrices [48]. Writing equations (5.12)-(5.15) into equation (5.1) and imposing the boundary conditions leads to the following nonlinear ODE system,

$$\frac{d\boldsymbol{\beta}}{dt} = D\left(\overline{B}_x \boldsymbol{\beta} + \boldsymbol{\beta} \overline{B}_y^T\right) - \left\langle \boldsymbol{\beta}\ ,\ \left(\overline{A}_x \boldsymbol{\beta} + \boldsymbol{\beta} \overline{A}_y^T\right)\right\rangle + \boldsymbol{F}(t) \tag{5.18}$$

where $\langle ., . \rangle$ is the standard elementwise product, $\boldsymbol{\beta}(t) = [\boldsymbol{\beta}(t)]_{nm}$ is an $((N-1) \times (M-1))$ matrix of time variable and $\boldsymbol{F}(t) = [\boldsymbol{F}(t)]_{nm}$ is an $((N-1) \times (M-1))$ matrix of time variable related to boundary conditions (5.2)-(5.3). After applying the boundary conditions to matrices A_x, A_y, B_x and B_y, respectively, $\overline{A}_x$, $\overline{A}_y$, $\overline{B}_x$, and $\overline{B}_y$ are obtained. Note that the initial condition of the nonlinear ODE system is defined by

$$\boldsymbol{\beta}(0) = [\boldsymbol{\beta}(0)]_{nm} = g(x_n, y_m). \tag{5.19}$$

5.3 The IELDTM for Time Integration

Now, we construct the IELDTM to solve the IVP stated in (5.18)-(5.19). Detailed description of the IELDTM for the solution of IVPs can be found in Chapter 3. Let us divide the time interval $[0, t_f]$ into at most P time elements with $\Delta t_i = t_{i+1} - t_i$ and the partition of the interval as $\omega = \{0 = t_0 < t_1 < \cdots < t_{P^*} = t_f\}$. Let us then consider the convergent Taylor series representation of the function $\boldsymbol{\beta}(t)$ of order K

about $t = t_i$ as

$$\boldsymbol{\beta}_i(t) = \sum_{k=0}^{K} \overline{\boldsymbol{\beta}}_i(k)(t-t_i)^k + O\left((t-t_i)^{K+1}\right), \quad t_i - \rho^i \le t \le t_i + \rho^i \tag{5.20}$$

where $i = 0, 1, \ldots, P^*$, $\overline{\boldsymbol{\beta}}_i(k)$ is the local differential transformation of $\boldsymbol{\beta}_i(t)$ and ρ^i is the radius of convergence of the representation. Taking differential transform of equation (5.18) leads to,

$$\begin{aligned} \overline{\boldsymbol{\beta}}_i(k+1) = \frac{1}{k+1}\Big[D\left(\overline{B}_x \overline{\boldsymbol{\beta}}_i(k) + \overline{\boldsymbol{\beta}}_i(k)\overline{B}_y^T\right) - \\ \sum_{s=0}^{k} \overline{\boldsymbol{\beta}}_i(s)\left(\overline{A}_x \overline{\boldsymbol{\beta}}_i(k-s) + \overline{\boldsymbol{\beta}}_i(k-s)\overline{A}_y^T\right) + \overline{\boldsymbol{F}}_i(k) \Big] \end{aligned} \tag{5.21}$$

where $\overline{\boldsymbol{F}}_i(k)$ is the local differential transformation of $\boldsymbol{F}(t)$. By defining the parameter $\theta \in [0,1]$ as a direction parameter, continuity requirement of the two consecutive solutions yields the following equation system,

$$\sum_{k=0}^{K} \overline{\boldsymbol{\beta}}_{i+1}(k)(-\theta\Delta t_i)^k = \sum_{k=0}^{K} \overline{\boldsymbol{\beta}}_i(k)((1-\theta)\Delta t_i)^k + O((\Delta t_i)^{K+1}, \theta) \tag{5.22}$$

where $O((\Delta t_i)^{K+1}, \theta)$ represents the dependency of the local truncation error to time increment, transformation order and direction parameter. By defining $\overline{\boldsymbol{\beta}}_0(0) = \boldsymbol{\beta}(0)$, the rest of $\overline{\boldsymbol{\beta}}_i(k)$ can be calculated from the recursive relation (5.21). Since $\overline{\boldsymbol{\beta}}_i(0)$ is known from the previous step, equation (5.22) is an implicit/explicit algebraic equation system of $\overline{\boldsymbol{\beta}}_{i+1}(0)$. This numerical algorithm finds all local solutions $\boldsymbol{\beta}_i(t)$ by solving equation (5.22) at each time step. Finally, by obtaining all local solutions $\boldsymbol{\beta}_i(t)$ we reach

$$u^i(x,y,t) = \sum_{n=0}^{N}\sum_{m=0}^{M} \alpha_{nm}(\beta_{nm})_i(t)\overline{T}_n(x)\overline{T}_m(y) \tag{5.23}$$

where $i = 0, 1, \ldots, P^*$.

5.4 Error Analysis

In this section, priori error estimation of the IELDTM for solving the nonlinear ODE system (5.18) are presented. Let us first define the map $\phi : R^{(N-1)\times(M-1)} \rightarrow$

$R^{(N-1)(M-1)\times 1}$ as

$$\begin{aligned}\phi(\overline{\boldsymbol{\beta}}_i(k)) = \big[\big(\overline{\boldsymbol{\beta}}_i(k)\big)_{11}, \big(\overline{\boldsymbol{\beta}}_i(k)\big)_{12}, ..., \big(\overline{\boldsymbol{\beta}}_i(k)\big)_{21}, \\ ..., \big(\overline{\boldsymbol{\beta}}_i(k)\big)_{(N-1)(M-1)}\big]^T = \overline{\boldsymbol{\varphi}}_i(k).\end{aligned} \tag{5.24}$$

The map ϕ reshapes the matrix $\overline{\boldsymbol{\beta}}_i(k)$ as a related vector $\overline{\boldsymbol{\varphi}}_i(k)$ for further analysis. The exact form of the continuity equation (5.22) can be written as

$$\overline{\overline{\boldsymbol{\varphi}}}_{i+1}(0) = \overline{\overline{\boldsymbol{\varphi}}}_i(0) + \sum_{k=1}^{K} \overline{\overline{\boldsymbol{\varphi}}}_i(k)((1-\theta)\Delta t)^k - \sum_{k=1}^{K} \overline{\overline{\boldsymbol{\varphi}}}_{i+1}(k)(-\theta\Delta t)^k + \Delta t\boldsymbol{\rho}_i \tag{5.25}$$

where $\overline{\overline{\boldsymbol{\varphi}}}$ is the exact form of $\overline{\boldsymbol{\varphi}}$ and then the local truncation error $\boldsymbol{\rho}_i$ takes the following form

$$\boldsymbol{\rho}_i = \Big[\overline{\overline{\boldsymbol{\varphi}}}_{(i+1)_*}(K+1)(1-\theta)^{K+1} - \overline{\overline{\boldsymbol{\varphi}}}_{i_*}(K+1)(-\theta)^{K+1}\Big]\Delta t^K. \tag{5.26}$$

Here $\overline{\overline{\boldsymbol{\varphi}}}_{i_*}$ defines the local transform at local point $t_{i_*} \in [t_{i-1},\ t_{i+1}]$ for all $i = 0, 1, 2, ..., P-1$. With the use of recursive relation (5.21), equation (5.25) can be rewritten as

$$\begin{aligned}\overline{\overline{\boldsymbol{\varphi}}}_{i+1}(0) = \overline{\overline{\boldsymbol{\varphi}}}_i(0) + (1-\theta)\Delta t\boldsymbol{H}_1(\overline{\overline{\boldsymbol{\varphi}}}_i(0),\ \theta,\ \Delta t) \\ +\theta\Delta t\boldsymbol{H}_2(\overline{\overline{\boldsymbol{\varphi}}}_{i+1}(0),\ \theta,\ \Delta t) + \Delta t\boldsymbol{\rho}_i\end{aligned} \tag{5.27}$$

where $\boldsymbol{H}_1 : R^{(N-1)(M-1)\times 1} \times [0,1] \times R^+ \to R^{(N-1)(M-1)\times 1}$ and $\boldsymbol{H}_2 : R^{(N-1)(M-1)\times 1} \times [0,1] \times R^+ \to R^{(N-1)(M-1)\times 1}$ are defined by

$$\boldsymbol{H}_1(\overline{\overline{\boldsymbol{\varphi}}}_i(0),\ \theta,\ \Delta t) = \sum_{k=1}^{K} \overline{\overline{\boldsymbol{\varphi}}}_i(k)((1-\theta)\Delta t)^{k-1}, \tag{5.28}$$

$$\boldsymbol{H}_2(\overline{\overline{\boldsymbol{\varphi}}}_{i+1}(0),\ \theta,\ \Delta t) = \sum_{k=1}^{K} \overline{\overline{\boldsymbol{\varphi}}}_{i+1}(k)(-\theta\Delta t)^{k-1}. \tag{5.29}$$

The IELDTM form of the continuity equation (5.22) takes the form

$$\overline{\boldsymbol{\varphi}}_{i+1}(0) = \overline{\boldsymbol{\varphi}}_i(0) + (1-\theta)\Delta t\boldsymbol{H}_1(\overline{\boldsymbol{\varphi}}_i(0),\ \theta,\ \Delta t) + \theta\Delta t\boldsymbol{H}_2(\overline{\boldsymbol{\varphi}}_{i+1}(0),\ \theta,\ \Delta t). \tag{5.30}$$

Let us define the following Jacobian matrices

$$\boldsymbol{J}_1(\boldsymbol{\varphi},\ \theta,\ \Delta t) = \frac{\partial(\boldsymbol{H}_1)_i}{\partial\boldsymbol{\varphi}_j}, \tag{5.31}$$

$$\boldsymbol{J}_2(\boldsymbol{\varphi},\ \theta,\ \Delta t) = \frac{\partial (\boldsymbol{H}_2)_i}{\partial \boldsymbol{\varphi}_j}. \tag{5.32}$$

Assume that the conditions

$$\mu(\boldsymbol{J}_1(\boldsymbol{\varphi},\ \theta,\ \Delta t)) \leq 0 \quad \text{and} \quad \mu(\boldsymbol{J}_2(\boldsymbol{\varphi},\ \theta,\ \Delta t)) \leq 0 \tag{5.33}$$

hold for all $(\boldsymbol{\varphi},\ \theta,\ \Delta t) \in R^{(N-1)(M-1)\times 1} \times [0,1] \times R^{+}$ and $\mu(A) = \lim_{\tau \to 0} \frac{\|\boldsymbol{I}+\tau \boldsymbol{A}\|-1}{\tau}$ is the logarithmic norm [4]. Let $\boldsymbol{Z}_i^1 \in R^{(N-1)(M-1)\times(N-1)(M-1)}$ and $\boldsymbol{Z}_i^2 \in R^{(N-1)(M-1)\times(N-1)(M-1)}$ be defined for $i \geq 0$ with

$$\boldsymbol{Z}_i^1 = \int_0^1 \Delta t \boldsymbol{J}_1\left(\sigma \overline{\overline{\boldsymbol{\varphi}}}_i(0) + (1-\sigma)\overline{\boldsymbol{\varphi}}_i(0),\ \theta,\ \Delta t\right) d\sigma, \tag{5.34}$$

$$\boldsymbol{Z}_{i+1}^2 = \int_0^1 \Delta t \boldsymbol{J}_2\left(\sigma \overline{\overline{\boldsymbol{\varphi}}}_{i+1}(0) + (1-\sigma)\overline{\boldsymbol{\varphi}}_{i+1}(0),\ \theta,\ \Delta t\right) d\sigma. \tag{5.35}$$

Applying the mean value theorem to vector-valued functions $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ yields

$$\Delta t \boldsymbol{H}_1(\overline{\overline{\boldsymbol{\varphi}}}_i(0),\ \theta,\ \Delta t) - \Delta t \boldsymbol{H}_1(\overline{\boldsymbol{\varphi}}_i(0),\ \theta,\ \Delta t) = \boldsymbol{Z}_i^1\left(\overline{\overline{\boldsymbol{\varphi}}}_i(0) - \overline{\boldsymbol{\varphi}}_i(0)\right), \tag{5.36}$$

and

$$\begin{aligned} \Delta t \boldsymbol{H}_2(\overline{\overline{\boldsymbol{\varphi}}}_{i+1}(0),\ \theta,\ \Delta t) - \Delta t \boldsymbol{H}_2(\overline{\boldsymbol{\varphi}}_{i+1}(0),\ \theta,\ \Delta t) = \\ \boldsymbol{Z}_{i+1}^2\left(\overline{\overline{\boldsymbol{\varphi}}}_{i+1}(0) - \overline{\boldsymbol{\varphi}}_{i+1}(0)\right). \end{aligned} \tag{5.37}$$

Defining global discretization error $\boldsymbol{\varepsilon}_i = \boldsymbol{\varphi}(t_i) - \overline{\boldsymbol{\varphi}}_i(0) = \overline{\overline{\boldsymbol{\varphi}}}_i(0) - \overline{\boldsymbol{\varphi}}_i(0)$ and subtracting equation (5.30) from equation (5.27) give rise to

$$\boldsymbol{\varepsilon}_{i+1} = \boldsymbol{\varepsilon}_i + (1-\theta)\boldsymbol{Z}_i^1 \boldsymbol{\varepsilon}_i + \theta \boldsymbol{Z}_{i+1}^2 \boldsymbol{\varepsilon}_{i+1} + \Delta t \boldsymbol{\rho}_i. \tag{5.38}$$

From (5.34)-(5.35) and Lemma 2.6 in literature [4], the following inequalities hold

$$\mu\left(\boldsymbol{Z}_i^1\right) \leq 0 \quad \text{and} \quad \mu\left(\boldsymbol{Z}_i^2\right) \leq 0. \tag{5.39}$$

The analysis are considered in two main cases as follows:

Case 1: Assume that $\theta = 1$, i.e. a backward local differential transform method is obtained. Equation (5.38) gives

$$\boldsymbol{\varepsilon}_{i+1} = \left(\boldsymbol{I} - \boldsymbol{Z}_{i+1}^2\right)^{-1}\left(\boldsymbol{\varepsilon}_i + \Delta t \boldsymbol{\rho}_i\right). \tag{5.40}$$

Since $\left\|\left(\boldsymbol{I}-\boldsymbol{Z}_{i+1}^{2}\right)^{-1}\right\| \leq 1$ according to equation (5.39) and Theorem 2.13 of literature [4], we find

$$\|\boldsymbol{\varepsilon}_i\| \leq \|\boldsymbol{\varepsilon}_0\| + \sum_{j=0}^{i-1} \Delta t \left\|\boldsymbol{\rho}_j\right\| \leq \|\boldsymbol{\varepsilon}_0\| + t_i \underbrace{max}_{0\leq j\leq i-1} \left\|\boldsymbol{\rho}_j\right\|. \tag{5.41}$$

Assumption of $\boldsymbol{\varepsilon}_0 \to 0$ leads to the following global error estimate

$$||\boldsymbol{\varphi}(t_i) - \overline{\boldsymbol{\varphi}}_i(0)||_\infty = ||\boldsymbol{\beta}(t_i) - \overline{\boldsymbol{\beta}}_i(0)||_\infty \leq t_i \omega \Delta t^K \tag{5.42}$$

where

$$\omega = \left| \frac{1}{(K+1)!} \underbrace{max}_{t\in[0,t_f]} \left(\left\| \frac{d\boldsymbol{\varphi}^{K+1}}{dt^{K+1}} \right\| \right) \right|.$$

Case 2: Let us now consider the rest of the implicit cases for $\theta \geq 1/2$. The transformed errors are defined as

$$\overline{\boldsymbol{\varepsilon}}_i = \left(\boldsymbol{I} - \theta \boldsymbol{Z}_i^2\right) \boldsymbol{\varepsilon}_i \tag{5.43}$$

where $i = 0, 1, ..., P$. Then the following recursive relation can be obtained from equation (5.38)

$$\overline{\boldsymbol{\varepsilon}}_{i+1} = \left(\boldsymbol{I} + (1-\theta)\boldsymbol{Z}_i^1\right)\left(\boldsymbol{I} - \theta \boldsymbol{Z}_i^2\right)^{-1} \overline{\boldsymbol{\varepsilon}}_i + \Delta t \boldsymbol{\rho}_i = R(\boldsymbol{Z}_i^1, \boldsymbol{Z}_i^2)\overline{\boldsymbol{\varepsilon}}_i + \Delta t \boldsymbol{\rho}_i. \tag{5.44}$$

Assume also that the following stability condition holds

$$\left\|R(\boldsymbol{Z}_i^1, \boldsymbol{Z}_i^2)\right\| = \left\|\left(\boldsymbol{I} + (1-\theta)\boldsymbol{Z}_i^1\right)\left(\boldsymbol{I} - \theta \boldsymbol{Z}_i^2\right)^{-1}\right\| \leq 1. \tag{5.45}$$

Thus, equation (5.44) becomes

$$\|\overline{\boldsymbol{\varepsilon}}_i\| \leq \|\overline{\boldsymbol{\varepsilon}}_0\| + \sum_{j=0}^{i-1} \Delta t \left\|\boldsymbol{\rho}_j\right\| \leq \|\overline{\boldsymbol{\varepsilon}}_0\| + t_i \underbrace{max}_{0\leq j\leq i-1} \left\|\boldsymbol{\rho}_j\right\|. \tag{5.46}$$

Since $\left\|\left(\boldsymbol{I} - \theta \boldsymbol{Z}_{i+1}^2\right)^{-1}\right\| \leq 1$ according to equation (5.39) and Theorem 2.13 of literature [4] for $\theta \geq 1/2$, we have $\|\boldsymbol{\varepsilon}_i\| \leq \|\overline{\boldsymbol{\varepsilon}}_i\|$. Thus, the main global error estimate can be constructed from equation (5.46) as

$$\|\boldsymbol{\varepsilon}_i\| \leq \|\boldsymbol{\varepsilon}_0\| + t_i \underbrace{max}_{0\leq j\leq i-1} \left\|\boldsymbol{\rho}_j\right\|. \tag{5.47}$$

Assumption of $\boldsymbol{\varepsilon}_0 \to 0$ leads to the following global error estimate

$$||\boldsymbol{\varphi}(t_i) - \overline{\boldsymbol{\varphi}}_i(0)||_\infty = ||\boldsymbol{\beta}(t_i) - \overline{\boldsymbol{\beta}}_i(0)||_\infty \leq \begin{cases} t_i \omega \Delta t^{K+1}, \text{ if } \theta = 1/2 \text{ and } K \text{ is odd} \\ t_i \omega \Delta t^K, \quad \text{otherwise} \end{cases} \tag{5.48}$$

where

$$\omega = \begin{cases} \left| \frac{1}{(K+2)!} \underbrace{max}_{t\in[0,t_f]} \left(\left\| \frac{d\varphi^{K+2}}{dt^{K+2}} \right\| \right) \left(\left(\frac{1}{2}\right)^{K+1} (K+1) \right) \right|, & \text{if } \theta = 1/2 \text{ and } K \text{ is odd} \\ \left| \frac{1}{(K+1)!} \underbrace{max}_{t\in[0,t_f]} \left(\left\| \frac{d\varphi^{K+1}}{dt^{K+1}} \right\| \right) \left[(1-\theta)^K - (-\theta)^K \right] \right|, & \text{if } \theta \neq \frac{1}{2} \text{ and } K \text{ is arbitrary.} \end{cases}$$

Adaptive construction of the step-sizes is important to capture the accuracy of the solution and to reduce the computational cost. With the help of the local truncation error defined in equation (5.48), adaptive step-sizes of the algorithm can be determined with the following inequalities

$$\Delta t_i < \begin{cases} \left(\frac{tol}{\left| \left(\frac{1}{2}\right)^{K+1} (K+1) \right| \left\| \overline{\boldsymbol{\beta}}_i (K+2) \right\|_\infty} \right)^{\frac{1}{K+1}}, & if\ \theta = 0.5\ and\ K\ is\ odd \\ \left(\frac{tol}{\left| (1-\theta)^{K+1} - (-\theta)^{K+1} \right| \left\| \overline{\boldsymbol{\beta}}_i (K+1) \right\|_\infty} \right)^{\frac{1}{K}}, & otherwise \end{cases} \tag{5.49}$$

where K is the order of local polynomial approximation (5.20), tol is the predetermined tolerance and $\overline{\boldsymbol{\beta}}_i(K+1)$ or $\overline{\boldsymbol{\beta}}_i(K+2)$ can be obtained from the recursive relation (5.21).

5.5 Numerical Experiments

In this section, we have numerically analyzed the present ChSCM-IELDTM approach over the one- and two-dimensional Burgers equations. The exponential convergence of the IELDTM has been proven for various values of the direction parameter θ. The computational efficiency of the adaptive IELDTM has been shown in comparison with the literature and MATLAB solvers, *ode15s* and *ode45* [142]. To measure the accuracy of the current numerical method, we prefer to use the maximum error norm as follows:

$$||E||_\infty = max_i |u_i^{exact} - u_i^{numerical}|.$$

Problem 5.1

Consider the one-dimensional form of the Burgers equation with the initial condition (5.1)

$$u(x,0) = sin(\pi x), \quad 0 < x < 1 \tag{5.50}$$

and the homogeneous Dirichlet boundary conditions

$$u(0,t) = u(1,t) = 0, \quad t > 0. \tag{5.51}$$

The exact solution of the 1D form of the Burgers equation under the consideration of the conditions (5.50)-(5.51) is given by

$$u(x,t) = 2\pi D \frac{\sum_{n=1}^{\infty} a_n \exp\left(-n^2\pi^2 D t\right) nsin(n\pi x)}{a_0 + \sum_{n=1}^{\infty} a_n \exp\left(-n^2\pi^2 D t\right) cos(n\pi x)} \tag{5.52}$$

with the Fourier coefficients

$$a_0 = \int_0^1 exp\left\{-(2\pi D)^{-1}[1-\cos(\pi x)\ ]\right\} dx, \tag{5.53}$$

$$a_n = 2\int_0^1 exp\left\{-(2\pi D)^{-1}[1-\cos(\pi x)\ ]\right\} \cos(n\pi x) dx. \tag{5.54}$$

The performance of the current ChSCM-IELDTM is shown comparing the maximum errors depending on the varying values of the direction parameter θ (see Figure 5.1). As expected theoretically, the maximum errors are reduced exponentially by increasing the order of the method. It can be observed from Figure 5.1 that choosing $\theta = 0.5$ with odd number K values leads to the numerical method of order $K+1$. Note that the exponential convergence is verified for selections of both $D = 1$ and $D = 0.1$. The advection-dominated cases of the present example exhibit challenging behaviour as has been widely discussed in the literature [42, 46, 52]. The ChSCM-IELDTM hybridization produces solutions without undesired oscillations in such stiff cases as shown in Figure 5.2. The sharp behaviours are demonstrated by choosing the kinematic viscosity constants as $D = 0.001$ and $D = 0.0005$. Using the current central time integration method and high-degree spectral approach, challenging numerical solutions have been captured for both cases. The performance of time adaptive ChSCM-IELDTM hybridization has been illustrated based on the error norms and the required step sizes as seen in Table 5.1. It is obvious from the table that the IELDTM time-integration is more preferable than the *ode45-ode15s* of MATLAB [142]. The overall performance of the current approach is compared with the strong Galerkin finite element method (GFEM) [43] as shown in Table 5.2. In the temporal part of the FEM based algorithm, the Crank-Nicolson method (a special case of the IELDTM; $\theta = 0.5$ and $K = 1$) has been used as a time integration technique. The current approach, ChSCM-IELDTM, produces more reliable results than the FEM-based technique, even if their degrees of freedom are lower than the literature [43]. The effect of the IELDTM is easily seen from the results in the table, i.e. the case of $K = 3$ and $\theta = 0.5$, fourth-order

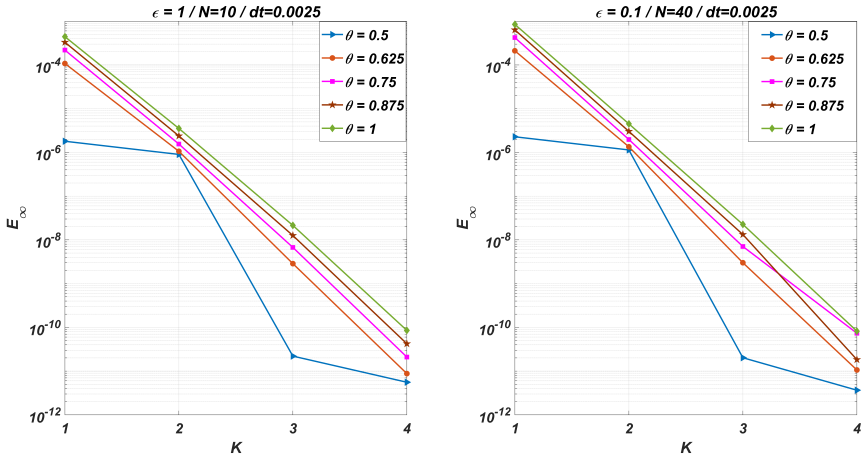

Figure 5.1 Order refinement results of the ChSCM-IELDTM for Problem 5.1 at $t_f = 0.5$ for various values of the direction parameter θ with a) $D = 1,\ N = 10$ and $\Delta t = 0.0025$ b) $D = 0.1,\ N = 40$ and $\Delta t = 0.0025$.

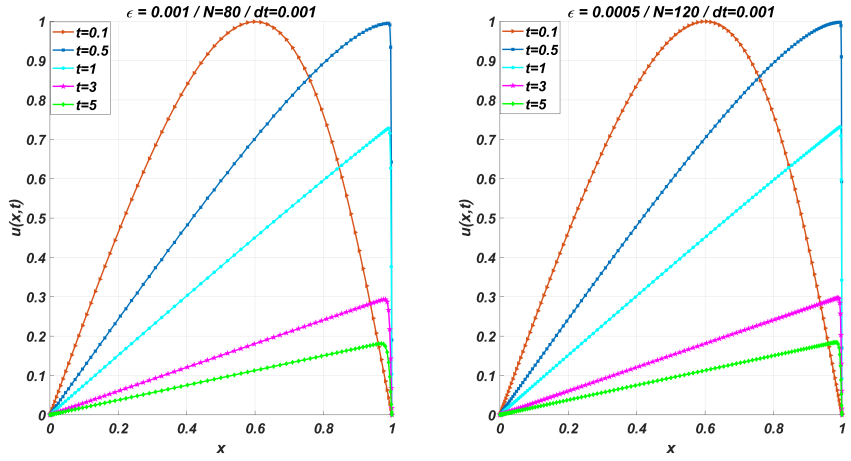

Figure 5.2 The ChSCM-IELDTM solutions of Problem 5.1 at various times for the parameter values a) $D = 0.001,\ N = 80,\ \theta = 0.5,\ K = 4$ and $\Delta t = 0.001$ b) $D = 0.0005,\ N = 120,\ \theta = 0.5,\ K = 4$ and $\Delta t = 0.001$.

in time, reduces the maximum error from $5.57E-07$ to $8.97E-10$.

Problem 5.2

Consider the two-dimensional form of the Burgers equation (5.1) with the following exact solution [44]

$$u(x,y,t) = \frac{1}{1+e^{(x+y-t)/2D}} \tag{5.55}$$

where the computational domain is $W = \left\{(x,y,t) \in R^2 \times R |\ 0 \le x,\ y \le 1 \text{ and } t \in [0,t_f]\right\}$. The initial and boundary conditions can be taken from the exact solution (5.55).

The direction parameter θ plays a critical role in terms of both accuracy and stability.

Table 5.1 Number of time steps and the maximum pointwise errors of the time-adaptive ChSCM-IELDTM approach for various values of the viscosity constant D and time approximation order K with $\theta = 0.5$, $t_f = 0.5$ and $tol = 10^{-10}$.

D	Method	Number of Steps	$\|\|E\|\|_\infty$
$D = 1$; $N = 10$	IELDTM-ChCSM; K=3	500	8.5E-09
	IELDTM-ChCSM; K=4	354	8.5E-09
	IELDTM-ChCSM; K=5	101	8.5E-09
	ode45-ChCSM	493	8.0E-09
	ode15s-ChCSM	243	8.6E-09
$D = 0.1$; $N = 20$	IELDTM-ChCSM; K=3	266	2.0E-08
	IELDTM-ChCSM; K=4	189	2.0E-08
	IELDTM-ChCSM; K=5	88	2.0E-08
	ode45-ChCSM	596	2.0E-08
	ode15s-ChCSM	163	2.0E-08
$D = 0.01$; $N = 40$	IELDTM-ChCSM; K=3	583	1.2E-05
	IELDTM-ChCSM; K=4	413	1.2E-05
	IELDTM-ChCSM; K=5	126	1.2E-05
	ode45-ChCSM	839	1.2E-05
	ode15s-ChCSM	253	1.2E-05

Table 5.2 Comparison of the numerical results produced with $t_f = 0.1$, $D = 0.1$ and $\Delta t = 0.001$ for Problem 5.1 and a) The GFEM [43] with $N = 40$, b) The IELDTM with $\theta = 0.5$, $K = 1$ and $N = 20$, c) The IELDTM with $\theta = 0.5$, $K = 3$ and $N = 20$.

x	$GFEM^a$	$ChSCM-IELDTM^b$	$ChSCM-IELDTM^c$	Exact
$x = 0.1$	0.2234492260	0.2234492160	0.2234495335	0.2234495335
$x = 0.3$	0.6251179353	0.6251179207	0.6251182341	0.6251182333
$x = 0.5$	0.8772800857	0.8772800822	0.8772796533	0.8772796530
$x = 0.7$	0.8369228762	0.8369228771	0.8369225592	0.8369225599
$x = 0.9$	0.3657542547	0.3657542612	0.3657544560	0.3657544557
$\|\|E\|\|_\infty$	5.5E-07	5.5E-07	8.9E-10	

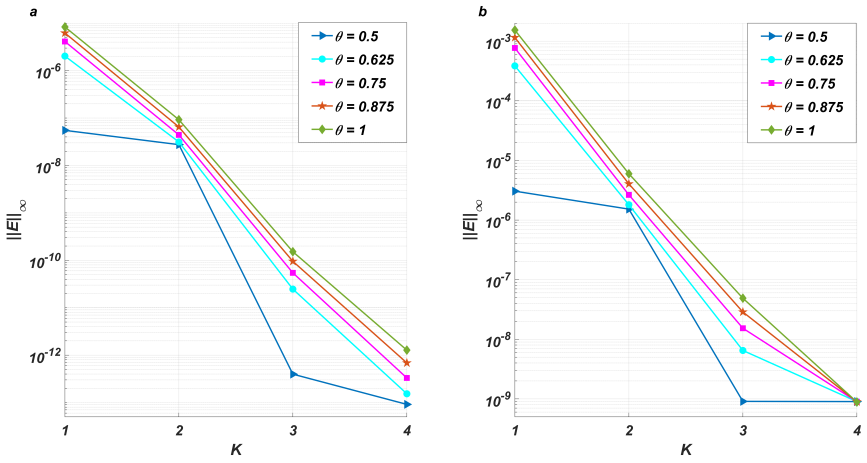

Figure 5.3 Order refinement results of the ChSCM-IELDTM for Problem 5.2 at $t_f = 0.5$ for various values of the direction parameter θ with a) $D = 1,\ N = M = 10$ and $\Delta t = 0.005$ b) $D = 0.1,\ N = M = 20$ and $\Delta t = 0.005$.

Exponential convergence of the current algorithm for both $D = 1$ and $D = 0.1$ is illustrated for various θ values as seen in Figure 5.3. We observe that all our theoretical error expectations are satisfied for all values of the direction parameter θ. Natural solution behaviour (5.55) drops rapidly from one to zero when the viscosity constant approaches zero. Thus, many numerical schemes fail to produce solutions without undesired oscillations in the regions in which rapid changes occur. In Figure 5.4, it has been shown that the present ChSCM-IELDTM technique has ability to accurately capture the shock behaviour for both $D = 0.01$ and $D = 0.005$. In Table 5.3, our motivation is to compare the current IELDTM integration method with the MATLAB solvers, *ode45* and *ode15s*, in terms of both accuracy and computational cost. As seen in the table, the present IELDTM-ChSCM provides optimal results with fewer time steps than the rival schemes.

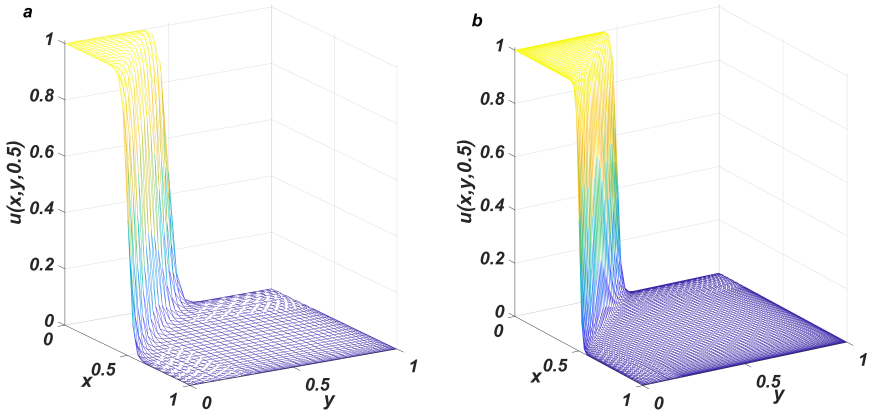

Figure 5.4 The ChSCM-IELDTM solutions of Problem 5.2 at various times for the parameter values a) $D = 0.01,\ N = M = 40,\ \theta = 0.5,\ K = 4$ and $\Delta t = 0.01$ b) $D = 0.005,\ N = M = 80,\ \theta = 0.5,\ K = 4$ and $\Delta t = 0.01$.

Table 5.3 Number of time steps and the maximum pointwise errors of the time-adaptive ChSCM-IELDTM method for various values of the viscosity constant D and time approximation order K with $\theta = 0.5$, $t_f = 0.5$ and $tol = 10^{-15}$.

D	Method	Number of Steps	$\|\|E\|\|_\infty$
$D = 1; N = M = 10$	IELDTM-ChCSM; K=3	222	1.84E-14
	IELDTM-ChCSM; K=4	178	1.48E-14
	IELDTM-ChCSM; K=5	149	2.35E-14
	ode45-ChCSM	7023	1.87E-14
	ode15s-ChCSM	94	2.25E-14
$D = 0.1; N = M = 20$	IELDTM-ChCSM; K=3	274	8.93E-10
	IELDTM-ChCSM; K=4	220	9.02E-10
	IELDTM-ChCSM; K=5	183	9.02E-10
	ode45-ChCSM	19919	9.02E-10
	ode15s-ChCSM	504	9.02E-10
$D = 0.01; N = M = 40$	IELDTM-ChCSM; K=3	378	2.50E-03
	IELDTM-ChCSM; K=4	311	2.60E-03
	IELDTM-ChCSM; K=5	268	2.60E-03
	ode45-ChCSM	55436	2.60E-03
	ode15s-ChCSM	4327	2.60E-03

5.6 Summary

In this chapter, the IELDTM has been derived for the temporal integration of problems represented by two-dimensional nonlinear advection-diffusion processes. The ChSCM has been considered for the spatial part of the considered PDEs due to the computational efficiency of the method. The global error analysis of the IELDTM together with the stability analysis has been utilized, and adaptivity equations have been proposed. The exponential convergence of the current numerical algorithm has been proven both theoretically and experimentally. The central case of the IELDTM, with the selection of $\theta = 0.5$, has been observed to be both more accurate and stability preserved. It has been shown that the IELDTM overcomes the shortcomings of current numerical techniques, such as the inaccuracy disadvantages of the well-known $\theta-$method and the instability drawbacks of the DTM-based methods. It has also been presented that the current time integration scheme achieves optimum accuracy by using fewer time steps than the commonly used time integrators of MATLAB such as *ode45* and *ode15s* [142]. It has also been shown here that the ChSCM-IELDTM has the ability to solve advection-dominated problems without producing any unwanted oscillation in both one- and two-dimensions. In the next chapter, the IELDTM is derived for the spatial discretizations of the one-dimensional parabolic PDEs.

6
THE IELDTM FOR SPATIAL DISCRETIZATIONS OF ONE-DIMENSIONAL ADR EQUATIONS

It has been proven in Chapter 4 that the IELDTM provides improved convergence properties for the numerical investigations of the BVPs for ODEs. The performance of the IELDTM encourages us to extend the scope of our method to solve time-discretized parabolic PDEs. The IELDTM is derived here for spatial discretizations of the one-dimensional advection-diffusion-reaction (ADR) equations encountered in various fields of science. The spatial discretization of the ADR equations has great importance due to the challenging physical structure of the equation, especially when advection dominates diffusion [43]. Since the IELDTM takes advantage of optimized $p-$ refinement, the proposed algorithm eliminates the drawbacks of the existing numerical methods.

The remainder of this chapter is outlined as follows: We explain why the IELDTM should be used together with a continuous time integrator in Section 6.1. In Section 6.2, the one-dimensional ADR equation is reduced to a system of nonhomogeneous BVPs with the use of the ChSCM in time. The IELDTM is produced for the resulted system of BVPs, and the mathematical background of the method is discussed in Section 6.3. The priori error analysis of the IELDTM together with stability conditions is provided in Section 6.4. In Section 6.5, four striking test problems are considered, and the quantitative results are illustrated to validate our theoretical results. At the end of this chapter, a summary of our investigations is given.

6.1 Space-Time IELDTM Treatment

In this section, we prove the difficulties encountered in the space-time IELDTM for solving the one-dimensional ADR equations. Let us consider the following linear ADR equation [107]

$$u_t + Vu_x + \beta u = Du_{xx}, \quad x \in [a,\ b], \quad t \in [0, t_f] \tag{6.1}$$

with boundary conditions

$$u(a,t) = f_1(t) \text{ and } u(b,t) = f_2(t) \tag{6.2}$$

and initial condition

$$u(x,0) = g(x) \tag{6.3}$$

where V, D, β stand for the advection constant, the diffusion constant and the reaction constant, respectively and while f_1, f_2 and g are known smooth functions. The subscripts x and t represent differentiations with respect to space x and time t, respectively.

Let us divide the space-time domain $[a,b] \times [0,t_f]$ into $M_x M_t$ element $[x_i, x_{i+1}] \times [t_j, t_{j+1}]$ with $\Delta x = x_{i+1} - x_i$ and $\Delta t = t_{j+1} - t_j$. The function $u(x,t)$ can be locally represented with the convergent local Taylor expansion about $\left(x_i,\ t_j\right)$ as follows

$$u_{ij}(x,t) = \sum_{k=0}^{K} \sum_{p=0}^{K-k} U_{ij}(k,p)(x-x_i)^k \left(t-t_j\right)^p + O\left((x-x_i)^{K+1} + \left(t-t_j\right)^{K+1}\right) \tag{6.4}$$

where $U_{ij}(k,p) = \frac{1}{k!p!} \frac{\partial^{k+p} u(x,t)}{\partial x^k t^p}\Big|_{x=x_i,\ t=t_j}$ is the local differential transform of the function $u(x,t)$ about $x = x_i$ and $t = t_j$ for $i = 0, 1, \ldots, M_x$ and $j = 0, 1, \ldots, M_t$. To completely describe the convergent local solution $u_{ij}(x,t)$ of order K, equation (6.4) includes $\frac{(K+1)(K+2)}{2}$ coefficients to be determined. Taking the differential transform of equation (6.1) yields

$$\begin{aligned} U_{ij}(k,p+1) = \frac{1}{(p+1)}\big[& D(k+1)(k+2)U_{ij}(k+2,p) - \\ & V(k+1)U_{ij}(k+1,p) - \beta U_{ij}(k,p)\big] \end{aligned} \tag{6.5}$$

where $k, p = 0, 1, ..., K-1$, $i = 0, 1, \ldots, M_x$ and $j = 0, 1, \ldots, M_t$.

Let us first consider that K is an even integer. Here one must notice the fact that if one has information about $U_{ij}(k,0)$ for all $k = 0, 1, \ldots, K$ then the next iteration only yields $U_{ij}(k,1)$ for $k = 0, 1, \ldots, K-2$ with the use of the recursive relation (6.5). By going on this way, one only gets $U_{ij}(0,K)$ for the highest time approximation term. In this scenario, it is not possible to obtain a complete local polynomial approximation (6.4) by optimizing the local degrees of freedom. Even if this crucial limitation has not been stated in semi-analytic studies, this situation can block the existing advantages of the IELDTM.

Then there exist two different ways of handling this inconvenience. One of them is the use of a one-step time integration to reduce the space-time PDE into a system

of BVPs. In this case, a special case of the present IELDTM ($\theta-$ method), including the Crank-Nicolson method (CNM), can be used. However, since the CNM is only a second-order method, the computational process needs a huge number of time elements. The second way is the consideration of an exponentially convergent and continuous time integration method to reduce the space-time PDE into a system of BVPs. As described in Chapter 1, the Chebyshev spectral collocation method has distinct properties, and the method is known to preserve exponential convergence [129]. Thus, it is more suitable to introduce a continuous time integration method for ADR equation (6.1) to reduce the equation into a system of boundary value problems to be solved by the IELDTM.

6.2 The ChSCM for Time Integration

In this section, we introduce the implementation procedure of the ChSCM to the temporal part of linear ADR equation (6.1). The ChSCM provides important interpolation characteristics over global domains as also described in subsection 2.1.1.2. To completely describe the space-time numerical solution for (6.1), the IELDTM is applied to the reduced system of BVPs as described in the following section.

Let us approximate the time part of $u(x,t)$ with Nth order Chebyshev polynomials as follows [48]

$$u(x,t) = \sum_{n=0}^{N} \alpha_n c_n(x) \overline{T}_n(t) \tag{6.6}$$

where $c_n(x)$ represent the spatial parts of $u(x,t)$, $\alpha_n = 1$ for interior points and $\alpha_0 = \alpha_N = 1/2$ for any boundary point. In equation (6.6), $\overline{T}_n(t)$ is defined in terms of the first kind Chebyshev polynomials $T_n(t)$ as

$$\overline{T}_n(t) = T_n\left(\frac{2t - t_f}{t_f}\right) = cos\left(ncos^{-1}\left(\frac{2t - t_f}{t_f}\right)\right). \tag{6.7}$$

We define the following CGL collocation points on $[0,\ t_f]$ as

$$t_n = \frac{t_f}{2}\left(1 - cos\left(\frac{\pi n}{N}\right)\right), \quad n = 0, 1, 2, \ldots,\ N. \tag{6.8}$$

The discrete orthogonality relation of the ChSCM yields

$$\sum_{n=0}^{N} \alpha_n \overline{T}_i(t_n) \overline{T}_j(t_n) = \gamma_i \delta_{ij} \tag{6.9}$$

with

$$\gamma_i = \begin{cases} \frac{N}{2}, & i \neq 0,\ N \\ N, & i = 0,\ N. \end{cases} \tag{6.10}$$

With the use of discrete orthogonality relation (6.9), we arrive at the following equality

$$c_j(x) = \kappa_j \sum_{n=0}^{N} \alpha_n \overline{T}_j(t_n) u(x, t_n) \tag{6.11}$$

where

$$\kappa_j = \begin{cases} \frac{2}{N}, & j \neq 0, N \\ \frac{1}{N}, & j = 0, N. \end{cases}$$

Thus, we can calculate $u_t(x,t)$ at the predetermined collocation points as follows,

$$u_t(x,\ t_i) = \sum_{n=0}^{N} [A_t]_{in} c_n(x) \tag{6.12}$$

where $i = 0, 1, 2, \ldots,\ N$, $c_n(x) = u(x, t_n)$ and A_t is an $(N \times N)$ matrix defined by

$$[A_t]_{in} = \begin{cases} \xi_n \mu \sum_{j=0}^{N} (-1)^{j+1} j^2 cos\left(j\left(\pi - \frac{n\pi}{N}\right)\right), & i = 0 \\ \xi_n \mu \sum_{j=0}^{N} \frac{j sin\left(j\left(\pi - \frac{i\pi}{N}\right)\right) cos\left(j\left(\pi - \frac{n\pi}{N}\right)\right)}{sin\left(\pi - \frac{i\pi}{N}\right)}, & i = 1, 2, \ldots, N-1 \\ \xi_n \mu \sum_{j=0}^{N} j^2 cos\left(j\left(\pi - \frac{n\pi}{N}\right)\right), & i = N \end{cases} \tag{6.13}$$

with

$$\xi_n = \begin{cases} \frac{1}{N}, & n = 0, N \\ \frac{2}{N}, & n \neq 0, N \end{cases}$$

where $\mu = \frac{2}{t_f}$ and $[A_t]_{in}$ is called as Chebyshev differentiation matrix. With the use of equation (6.12) for main equation (6.1) in collocation sense, we find the following system of boundary value problems

$$D\frac{d^2\boldsymbol{c}}{dx^2} - V\frac{d\boldsymbol{c}}{dx} - (\beta I + A_t)\boldsymbol{c} = \boldsymbol{0} \tag{6.14}$$

where $\boldsymbol{c}(x) = [c_0(x),\ c_1(x),\ c_2(x), \ldots,\ c_N(x)]^T$ is an $((N+1) \times 1)$ column vector and I is the $((N+1) \times (N+1))$ unit matrix. Applying the initial condition $u(x,0) = c_0(x) = g(x)$ reduces the system into the following form

$$D\frac{d^2\boldsymbol{c}}{dx^2} - V\frac{d\boldsymbol{c}}{dx} - (\beta I + \overline{A}_t)\boldsymbol{c} = \boldsymbol{G}(x) \tag{6.15}$$

where $\boldsymbol{G}(x)$ is an $(N \times 1)$ column vector defined as $(\boldsymbol{G}(x))_i = (A_t)_{(i+1)1}\, g(x)$ and $\overline{A}_t$ is an $(N \times N)$ matrix reduced from A_t. Thus, equation (6.15) represents the $(N \times N)$

system of boundary value problems with the following conditions,

$$\boldsymbol{c}(a) = f_1(\overline{\boldsymbol{t}}) \quad \text{and} \quad \boldsymbol{c}(b) = f_2(\overline{\boldsymbol{t}}) \tag{6.16}$$

where $\overline{\boldsymbol{t}} = [t_1,\ t_2,\ \ldots,\ t_N]^T$ defined in equation (6.8).

6.3 The IELDTM for Spatial Discretization

Let us divide the spatial interval $[a, b]$ into M spatial element with $\Delta x_i = x_{i+1} - x_i$ and the partition of the interval is defined as $\omega = \{a = x_0 < x_1 < \cdots < x_M = b\}$. The function $\boldsymbol{c}(x)$ can be locally represented with the convergent local Taylor expansion about $x_i \in \omega$ as follows

$$\boldsymbol{c}_i(x) = \sum_{k=0}^{K} \boldsymbol{C}_i(k)(x - x_i)^k + O\left((x - x_i)^{K+1}\right), \quad x_i - \rho^i \le x \le x_i + \rho^i \tag{6.17}$$

where $\boldsymbol{C}_i(k) = \frac{1}{k!}\frac{\boldsymbol{d}^k \boldsymbol{c}(x)}{dx^k}|_{x=x_i}$ is the differential transform of the function $\boldsymbol{c}(x)$ about $x = x_i$ and $i = 0, 1, \ldots,\ M$. The function $\boldsymbol{c_i}(x)$ is assumed to be analytic for all i and has the radius of convergence $\rho^i > \Delta x_i$. This assumption leads us to search about the relations between neighbourhood solutions $\boldsymbol{c}_i(x)$ and $\boldsymbol{c}_{i+1}(x)$ in terms of their local behaviours. Then, the function $\boldsymbol{c}(x)$ can also be approximated as

$$\boldsymbol{c}_{i+1}(x) = \sum_{k=0}^{K} \boldsymbol{C}_{i+1}(k)(x - x_{i+1})^k + O\left((x - x_{i+1})^{K+1}\right), \tag{6.18}$$

where $x_{i+1} - \rho^{i+1} \le x \le x_{i+1} + \rho^{i+1}$. Let us define the intervals $P^i = [x_i - \rho^i, x_i + \rho^i]$, $P^{i+1} = [x_{i+1} - \rho^{i+1}, x_{i+1} + \rho^{i+1}]$ and $P = [x_i,\ x_{i+1}]$. Assume that the condition $P \subset \left(P^i \cap P^{i+1}\right)$ holds, and then we state the following conclusions:

$*$ C^0 and C^1 continuity conditions can be applied on the interval $[x_i,\ x_{i+1}]$.

$*$ Any point on the interval $[x_i,\ x_{i+1}]$ can be written as $x^* = x_i + (1-\theta)\Delta x_i$, where $0 \le \theta \le 1$.

Convergent solutions $\boldsymbol{c}_i(x)$ and $\boldsymbol{c}_{i+1}(x)$ on the interior points $x^* = x_i + (1-\theta)\Delta x_i$ need to satisfy the following C^0 and C^1 continuity conditions

$$\boldsymbol{c}_{i+1}(x_i + (1-\theta)\Delta x_i) = \boldsymbol{c}_i(x_i + (1-\theta)\Delta x_i), \tag{6.19}$$

$$\boldsymbol{c}'_{i+1}(x_i + (1-\theta)\Delta x_i) = \boldsymbol{c}'_i(x_i + (1-\theta)\Delta x_i). \tag{6.20}$$

With the use of the continuity conditions (6.19)-(6.20), we reach the following equation system,

$$\sum_{k=0}^{K} \boldsymbol{C}_{i+1}(k)(-\theta\Delta x_i)^k = \sum_{k=0}^{K} \boldsymbol{C}_i(k)((1-\theta)\Delta x_i)^k + O\left((\Delta x_i)^{K+1}\right), \tag{6.21}$$

$$\sum_{k=1}^{K} \boldsymbol{C}_{i+1}(k)\,k(-\theta\Delta x_i)^{k-1} = \sum_{k=1}^{K} \boldsymbol{C}_i(k)\,k((1-\theta)\Delta x_i)^{k-1} + O((\Delta x_i)^K) \tag{6.22}$$

where $\boldsymbol{C}_i(0)$ and $\boldsymbol{C}_i(1)$ are unknowns for all $i = 0, 1, \ldots, M$. The parameter θ determines the meeting point of two consecutive representations and is called the direction parameter. Selecting $\theta = 0$ yields forward directional and explicit local continuity conditions (6.21)-(6.22) and $\theta \neq 0$ yields implicit equations that include central directed ($\theta = 0.5$) and backward directed ($\theta = 1$) forms. Thence, the continuity conditions C^0 and C^1 (6.21)-(6.22) are in the implicit-explicit forms depending on the direction parameter θ.

The main point here is to determine $\boldsymbol{C}_i(k)$ for $k \geq 2$ in terms of $\boldsymbol{C}_i(0)$ and $\boldsymbol{C}_i(1)$ with the use of local differential transformation. Taking differential transform of equation (6.15) leads to

$$\boldsymbol{C}_i(k+2) = \frac{1}{D(k+1)(k+2)}\left[V(k+1)\boldsymbol{C}_i(k+1) + \left(\beta I + \overline{A}_t\right)\boldsymbol{C}_i(k) + \overline{\boldsymbol{G}}_i(k)\right] \tag{6.23}$$

where $i = 0, 1, \ldots, M$, $k = 0, 1, \ldots, K-2$, $\overline{\boldsymbol{G}}_i$ is the local transformed form of function $\boldsymbol{G}(x)$ about the point $x = x_i$. Thus, to completely describe the approximate solutions (6.17)-(6.18), one needs to evaluate initial positions $\boldsymbol{C}_i(0)$ and initial slopes $\boldsymbol{C}_i(1)$ for all $i = 0, 1, \ldots, M$. For each spatial position, $\boldsymbol{C}_i(0)$ and $\boldsymbol{C}_i(1)$ are assumed to be unknown values to determine highly accurate local approximations (6.17)-(6.18). Having $2N$ local unknowns explains why we have applied the continuity conditions C^0 and C^1 to obtain $2N$ linearly independent local equations as widely used in quadratic spline interpolations [150]. With the use of algebraic relations in (6.23), plugging all algebraic coefficients $\boldsymbol{C}_i(k)$ into continuity conditions (6.21)-(6.22), and imposing the boundary conditions lead to the following system of linear equations

$$\boldsymbol{Z}(K, \theta, \Delta x)\,\boldsymbol{\zeta} = \boldsymbol{F} \tag{6.24}$$

where $\boldsymbol{Z}$ is a $(2NM \times 2NM)$ matrix, $\boldsymbol{\zeta} = [\boldsymbol{C}_0(1), \boldsymbol{C}_1(0), \boldsymbol{C}_1(1), \ldots, \boldsymbol{C}_M(1)]^T$ and $\boldsymbol{F}$ are $(2NM \times 1)$ column vectors. The approximation matrix $\boldsymbol{Z}(K, \theta, \Delta x)$ depends on the spatial increment Δx, the direction parameter θ and the local approximation order K. Increasing values of the local approximation order K leads to a more accurate representation (6.24) with the $p-$refinement procedure. The column vector $\boldsymbol{\zeta}$ consists

of all space-time approximate numerical values of $u(x,t)$ and $u_x(x,t)$, respectively, denoted by $\boldsymbol{C}_i(0)$ and $\boldsymbol{C}_i(1)$.

By solving system (6.24) for $\boldsymbol{\zeta}$, we find all numerical solutions at the required space-time points. Note that in numerical experiments, we also solve nonlinear equations such as the Burgers and the Burgers-Fisher equations involving the nonlinear advective flux as $c(x,t)\frac{\partial c(x,t)}{\partial x}$. In such problems, the trust-region dogleg method [143] based *fsolve* built-in function of the MATLAB program have been applied to solve the resulted nonlinear algebraic system of equations.

6.4 Error Analysis

By assuming fixed step sizes $\Delta x_i = \Delta x$, complete forms of C^0 and C^1 continuity conditions (6.21)-(6.22) can be stated as

$$\sum_{k=0}^{K} \boldsymbol{C}_{i+1}(k)(-\theta\Delta x)^k = \sum_{k=0}^{K} \boldsymbol{C}_i(k)((1-\theta)\Delta x)^k + \Delta x \boldsymbol{\rho}_i^1, \tag{6.25}$$

$$\sum_{k=1}^{K} \boldsymbol{C}_{i+1}(k)\,k(-\theta\Delta x)^{k-1} = \sum_{k=1}^{K} \boldsymbol{C}_i(k)\,k((1-\theta)\Delta x)^{k-1} + \Delta x \boldsymbol{\rho}_i^2 \tag{6.26}$$

where $\boldsymbol{\rho}_i^1$ and $\boldsymbol{\rho}_i^2$ are the local truncation errors expressed as follows

$$\boldsymbol{\rho}_i^1 = \left[\boldsymbol{C}_{i_*}(K+1)(1-\theta)^{K+1} - \boldsymbol{C}_{(i+1)_*}(K+1)(-\theta)^{K+1}\right]\Delta x^K, \tag{6.27}$$

$$\boldsymbol{\rho}_i^2 = \left[\boldsymbol{C}_{i_*}(K+1)(K+1)(1-\theta)^K - \boldsymbol{C}_{(i+1)_*}(K+1)(K+1)(-\theta)^K\right]\Delta x^{K-1} \tag{6.28}$$

where $i = 0, 1, \ldots, M-1$ and $\boldsymbol{C}_{i_*}$ defines the local differential transform defined at point $x_{i_*} \in [x_{i-1},\ x_{i+1}]$. Assembling all contributions in equations (6.25)-(6.26) leads to the following system of equations

$$\boldsymbol{Z}\boldsymbol{\zeta} = \boldsymbol{F} + \Delta x \boldsymbol{G}(\theta, \Delta x) \tag{6.29}$$

where $\boldsymbol{\zeta}$ is a $(2NM \times 1)$ column vector corresponds to exact form of $\overline{\boldsymbol{\zeta}}$ and $\boldsymbol{G}(\theta,\Delta x) = \boldsymbol{G}_1(\theta,\Delta x) + \boldsymbol{G}_2(\theta,\Delta x)$ is column vector $(2NM \times 1)$ corresponds to the truncation errors as follows,

$$\boldsymbol{G}_1(\theta,\Delta x)=\begin{bmatrix}\left[(1-\theta)^{K+1}\boldsymbol{C}_{0_*}(K+1)-(-\theta)^{K+1}\boldsymbol{C}_{1_*}(K+1)\right]\\0\\\left[(1-\theta)^{K+1}\boldsymbol{C}_{1_*}(K+1)-(-\theta)^{K+1}\boldsymbol{C}_{2_*}(K+1)\right]\\0\\.\\.\\.\\\left[(1-\theta)^{K+1}\boldsymbol{C}_{(N-1)_*}(K+1)-(-\theta)^{K+1}\boldsymbol{C}_{N_*}(K+1)\right]\\0\end{bmatrix}\Delta x^K,$$

$$\boldsymbol{G}_2(\theta,\Delta x)=\begin{bmatrix}0\\\left[(K+1)(1-\theta)^{K}\boldsymbol{C}_{0_*}(K+1)-(K+1)(-\theta)^{K}\boldsymbol{C}_{1_*}(K+1)\right]\\0\\\left[(K+1)(1-\theta)^{K}\boldsymbol{C}_{1_*}(K+1)-(K+1)(-\theta)^{K}\boldsymbol{C}_{2_*}(K+1)\right]\\.\\.\\.\\0\\\left[(K+1)(1-\theta)^{K}\boldsymbol{C}_{(N-1)_*}(K+1)-(K+1)(-\theta)^{K}\boldsymbol{C}_{N_*}(K+1)\right]\end{bmatrix}\Delta x^{K-1}.$$

Denoting the error column vector as $\boldsymbol{\varepsilon}=\boldsymbol{\zeta}-\overline{\boldsymbol{\zeta}}$ and subtracting equation (6.24) from equation (6.29) yields

$$\boldsymbol{Z}\boldsymbol{\varepsilon}=\Delta x\boldsymbol{G}(\theta,\Delta x) \tag{6.30}$$

and defining $\boldsymbol{Z}^*=\Delta x\boldsymbol{Z}^{-1}$ leads to

$$||\boldsymbol{\varepsilon}||\leq||\boldsymbol{Z}^*||\;||\boldsymbol{G}(\theta,\Delta x)||. \tag{6.31}$$

where $||.||$ is an arbitrary norm. The bound of $||\boldsymbol{G}(\theta,\Delta x)||$ can be found in Appendix B depending on the selection of the direction parameter θ and transformation order K. Assume that the following inequalities are satisfied

$$||\boldsymbol{G}(\theta,\Delta x)||\leq P_2\Delta x^S \text{ and } ||\boldsymbol{Z}^*||\leq P_1 \tag{6.32}$$

where the definition of P_2 can be found in Appendix B and $S=K-1,\ K$. Finally, the following global error estimates

$$||\varepsilon||\leq\begin{cases}P_1P_2\Delta x^K, & \theta=0.5 \text{ and } K \text{ is even}\\P_1P_2\Delta x^{K-1}, & otherwise.\end{cases} \tag{6.33}$$

are obtained.

6.5 Numerical Experiments

This section is devoted to numerical solutions of linear and nonlinear ADR equations by illustrating quantitative and qualitative results. Convergence results of the present IELDTM-ChSCM hybridization are presented by demonstrating error norms and pointwise solutions. The produced results are compared with the literature and exact solutions. To measure the accuracy of the current numerical method, we prefer to use the absolute pointwise error and maximum error norms as follows:

$$E_{abs}^{i} = \left| u_i^{exact} - u_i^{numerical} \right|,$$

$$||E||_{\infty} = max_i |u_i^{exact} - u_i^{numerical}|.$$

Our problem set includes stiff/non-stiff linear advection-diffusion equation, nonlinear advection-diffusion equation (the Burgers equation) and nonlinear advection-diffusion-reaction equation (the Burgers-Fisher equation).

Problem 6.1 [151]

Consider the advection-diffusion process with the choices of $V = 1$, $D = 0.01$ and $\beta = 0$ in the ADR equation (6.1) for which the exact solution is given by [151]

$$u(\,x,t) = \frac{0.025}{\sqrt{0.000625 + 0.02t}} exp\left(-\frac{(x + 0.5 - t)^2}{0.00125 + 0.04t} \right), \tag{6.34}$$

where $t > 0$ and $0 \leq x \leq 1$. The initial and boundary conditions of the problem are taken from the exact solution (6.34) as follows:

$$u(0,t) = \frac{0.025}{\sqrt{0.000625 + 0.02t}} exp\left(-\frac{(0.5 - t)^2}{0.00125 + 0.04t} \right), \tag{6.35}$$

$$u(1,t) = \frac{0.025}{\sqrt{0.000625 + 0.02t}} \exp\left(-\frac{(1.5 - t)^2}{0.00125 + 0.04t} \right), \tag{6.36}$$

$$u(x,0) = exp\left(-\frac{(x + 0.5)^2}{0.00125} \right). \tag{6.37}$$

In Table 6.1, the present IELDTM-ChSCM algorithm has been compared with the finite element method [151], the finite difference method [107] and the exact solution. The degrees of freedom (*dof*) of all numerical algorithms have been presented. As seen in the table, the tenth order IELDTM, $K = 10$, produces more accurate results with less *dof* than both the finite element methods and the high-order finite difference methods. Note that, to get better accuracy, the present IELTM needs 125 times less *dof* than the other methods. While the rival iterative algorithms solve a relatively small system of

sparse equations in each iteration, here we need to solve a relatively large system of $2N-$band sparse equations. It may not be correct to expect our algorithm to be 125 times faster than the competing algorithms because the computational mechanisms are different. In Figures 6.1-6.2, the numerical convergence of the IELDTM has been proven with respect to $K-$refinement (order refinement) and $\Delta x-$refinement for various values of the direction parameter θ. It can be seen in the figures that all our theoretical expectations are satisfied for both refinements and all direction parameters. By fitting the curve $\Delta x - ||E||_\infty$, we have determined the orders of the present algorithms for various values of direction parameter θ in Figure 6.2. The expected orders of the methods are three for $\theta \neq 0.5$ and four for $\theta = 0.5$, and all expectations are matched with the numerical results in Figure 6.2.

Table 6.1 Comparison of the numerical results produced at various nodal points and $t_f = 1$ for Problem 6.1 and a) The FDM6/8/10 [107] with $M = 100$, $\Delta t = 0.001$ and $dof = 10^5$, b)The GFEM [151] with $M = 100$, $\Delta t = 0.001$ and $dof = 10^5$, c)The IELDTM with $\theta = 0.5$, $N = 10$, $M = 40$, $K = 10$ and $dof = 800$, d)The IELDTM with $\theta = 0.5$, $N = 10$, $M = 40$, $K = 8$ and $dof = 800$.

x	*Exact*	$FDM6/8/10^a$ [107]	$GFEM^b$ [151]	$IELDTM^c$	$IELDTM^d$
0.1	0.003599207	0.0035992	0.0035991	0.003599207	0.003599200
0.2	0.019642259	0.0196423	0.0196419	0.019642259	0.019642289
0.3	0.066009887	0.0660099	0.0660105	0.066009887	0.066009865
0.4	0.136602774	0.1366028	0.1366054	0.136602775	0.136602693
0.5	0.174077656	0.1740777	0.1740782	0.174077654	0.174077816
0.6	0.136602774	0.1366028	0.1365989	0.136602776	0.136602697
0.7	0.066009887	0.0660099	0.0660075	0.066009886	0.066009837
0.8	0.019642259	0.0196423	0.0196433	0.019642258	0.019642322
0.9	0.003599207	0.0035992	0.0035953	0.003599207	0.003599193
$\|\|E\|\|_\infty$		$4.40E-08$	$3.90E-06$	$2.00E-09$	$1.75E-07$

Problem 6.2 [152]

We consider an advection dominated stiff process with $\beta = 0$ in an infinitely long channel represented by the ADR equation (6.1). The channel length is $L = 200\ m$ and divided into M uniform elements with the length $\Delta x = 200/M$. The discontinuous initial condition and boundary conditions of the problem are taken to be

$$u(x,0) = \begin{cases} 1, \ if \ x = 0 \\ 0, \ if \ x > 0, \end{cases} \tag{6.38}$$

$$u(0,t) = 1 \text{ and } \frac{\partial u}{\partial x}\Big|_{x=200} = 0. \tag{6.39}$$

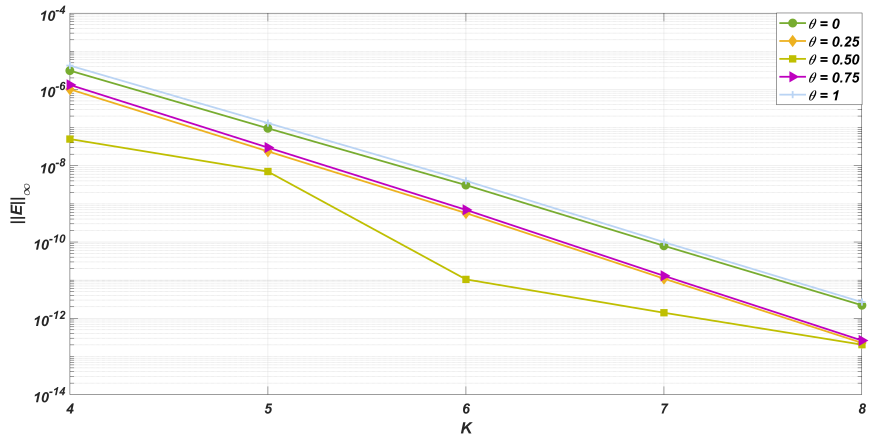

Figure 6.1 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter θ with $\Delta x = 0.01$, $N = 60$ and $t_f = 1$.

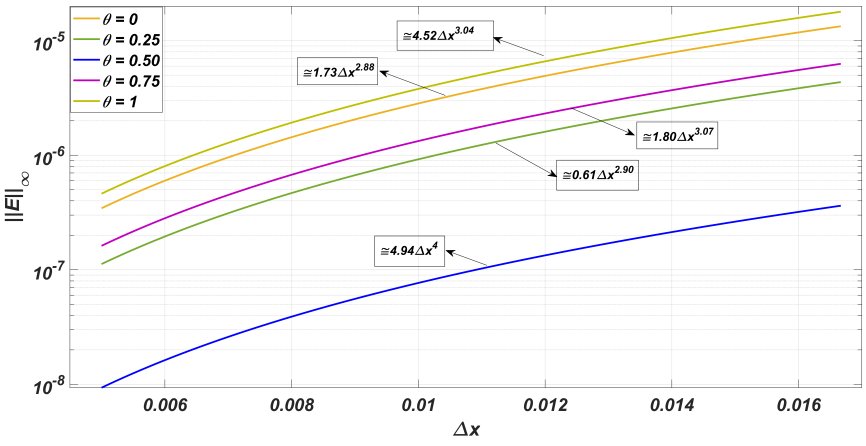

Figure 6.2 Convergence results of the present algorithm with respect to the spatial increment Δx for various values of the direction parameter θ with $K = 4$, $N = 60$ and $t_f = 1$.

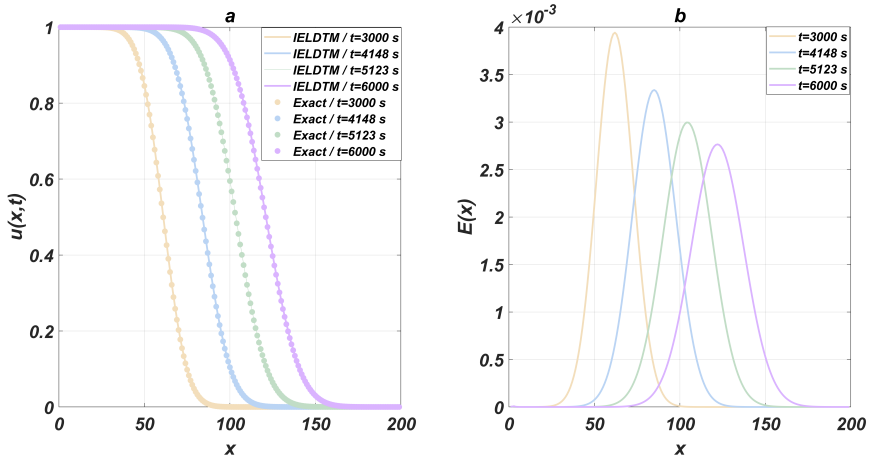

Figure 6.3 a) Comparison of the IELDTM-ChSCM solution with the exact solution, b) Pointwise absolute errors produced by the IELDTM-ChSCM at various times with $\Delta x = 1,\ N = 80,\ \theta = 0.5$ and $K = 6$ for Problem 6.2.

The exact solution of this problem is given in literature [152] by

$$u(x,t) = \frac{1}{2} erfc\left(\frac{x - Vt}{\sqrt{4tD}}\right) + \frac{1}{2} e^{(\frac{Vx}{D})} erfc\left(\frac{x + Vt}{\sqrt{4tD}}\right). \tag{6.40}$$

In Figure 6.3, we illustrate the performance of the present IELDTM-ChSCM for solving stiff advection-diffusion equation (6.1) with discontinuous initial condition (6.38). The parameter values $V = 0.02,\ D = 0.02,\ \theta = 0.5,\ \Delta x = 1,\ K = 6,\ N = 40$ and $t_f = 6000\ s$ are used for producing Figure 6.3. As seen in the figure, the present approach accurately captures the physical behaviour with an optimum cost. The stiffness of the problem is inversely proportional to the diffusion coefficient D. In Figure 6.4, the advection dominated cases of Problem 6.2 with $D = 0.005$ and $D = 0.001$ have been solved with the IELDTM-ChSCM to measure the potential of the current spatial method. The numerical results without undesired oscillations have been observed to capture the shock behaviour using the present sixth-order IELDTM (see Figure 6.4). Thus, the present algorithm provides reliable numerical results for the advection-dominated problems with discontinuous initial conditions.

Problem 6.3 [43]

Let us consider the following Burgers equation

$$u_t + uu_x = Du_{xx}, \quad x \in [0,1], \quad t \in [0, t_f] \tag{6.41}$$

with initial condition

$$u(x,0) = sin\ (\pi x), \quad 0 < x < 1 \tag{6.42}$$

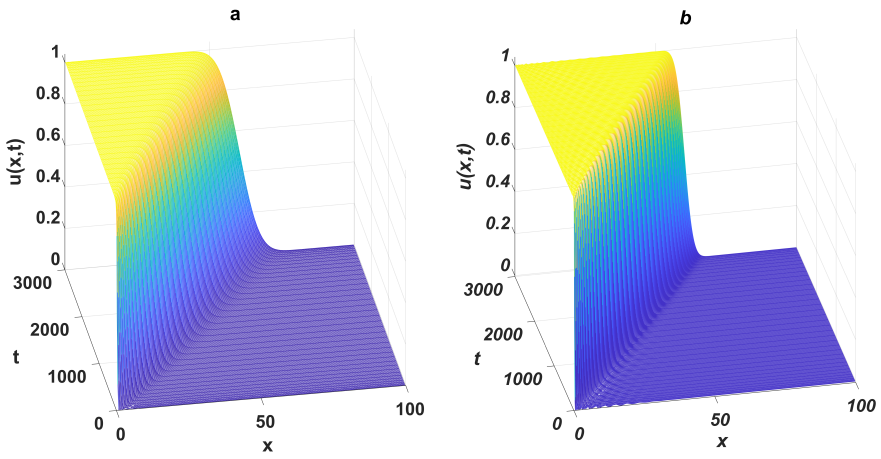

Figure 6.4 Illustration of the advection dominated processes of Problem 6.2 solved by the IELDTM-ChSCM for a) $V = 0.02$, $D = 0.005$, $\Delta x = 0.25$, $N = 40$, $\theta = 0.5$, $K = 6$ and $t_f = 3000$ b) $V = 0.02$, $D = 0.001$, $\Delta x = 0.2$, $N = 40$, $\theta = 0.5$, $K = 6$ and $t_f = 3000$.

and homogeneous Dirichlet boundary conditions

$$u(0,t) = u(1,t) = 0, \quad t > 0. \tag{6.43}$$

The exact solution of equation (6.41) under the consideration of cases (6.42)-(6.43) given in [43] is

$$u(x,t) = 2\pi D \frac{\sum_{n=1}^{\infty} a_n \exp\left(-n^2\pi^2 Dt\right) nsin(n\pi x)}{a_0 + \sum_{n=1}^{\infty} a_n \exp\left(-n^2\pi^2 Dt\right) cos(n\pi x)} \tag{6.44}$$

with Fourier coefficients

$$\begin{aligned} a_0 &= \int_0^1 exp\left\{-(2\pi D)^{-1}[1-\cos(\pi x)\,]\right\} dx, \\ a_n &= 2\int_0^1 exp\left\{-(2\pi D)^{-1}[1-\cos(\pi x)\,]\right\} \cos(n\pi x) dx. \end{aligned} \tag{6.45}$$

In Table 6.2, the fourth and sixth order IELDTM-ChSCM algorithm has been compared with the FEM [43], the FDM [153] and the exact solution for solving nonlinear advection-diffusion equation (6.41). As seen in Table 6.2, the proposed sixth-order method offers more accurate results with the use of 20 times less *dof* than both finite element and finite difference methods. Like Figures 6.1-6.2, the numerical convergence of the IELDTM for solving the corresponding nonlinear equation has been illustrated in Figures 6.5-6.6. For all values of the direction parameter θ, the theoretical expectations are verified. The curves $\Delta x - ||E||_\infty$ have been fitted with functions like Sh^p and all expected orders p are seen to be correct in Figure 6.6.

Table 6.2 Comparison of the numerical results produced with $D = 0.1$ and $t_f = 2.3$ for Problem 6.3 and a) The BDF3 [153] with $M = 100$, $dt = 0.01$ and $dof = 10^4$, b) The GFEM [43] with $M = 100$, $dt = 0.01$ and $dof = 10^4$, c) The IELDTM with $\theta = 0.5$, $N = 10$, $M = 25$, $K = 4$ and $dof = 500$, d) The IELDTM with $\theta = 0.5$, $N = 10$, $M = 25$, $K = 6$ and $dof = 500$.

x	Exact	$BDF3^a$ [153]	$GFEM^b$ [43]	$IELDTM^c$	$IELDTM^d$
$x = 0.1$	0.02213962	0.02253	0.02213891	0.02213812	0.02213965
$x = 0.2$	0.04279559	0.04357	0.04279420	0.04279269	0.04279559
$x = 0.3$	0.06043125	0.06155	0.06042922	0.06042731	0.06043122
$x = 0.4$	0.07344309	0.07485	0.07344051	0.07343881	0.07344313
$x = 0.5$	0.08023096	0.08182	0.08022800	0.08022713	0.08023105
$x = 0.6$	0.07939879	0.08104	0.07939570	0.07939603	0.07939873
$x = 0.7$	0.07010678	0.07161	0.07010391	0.07010567	0.07010671
$x = 0.8$	0.05251977	0.05368	0.05251753	0.05252019	0.05251991
$x = 0.9$	0.02817404	0.02881	0.02817281	0.02817457	0.02817397
$\|\|E\|\|_\infty$		$1.63E-03$	$3.09E-06$	4.28E-06	1.41E-07

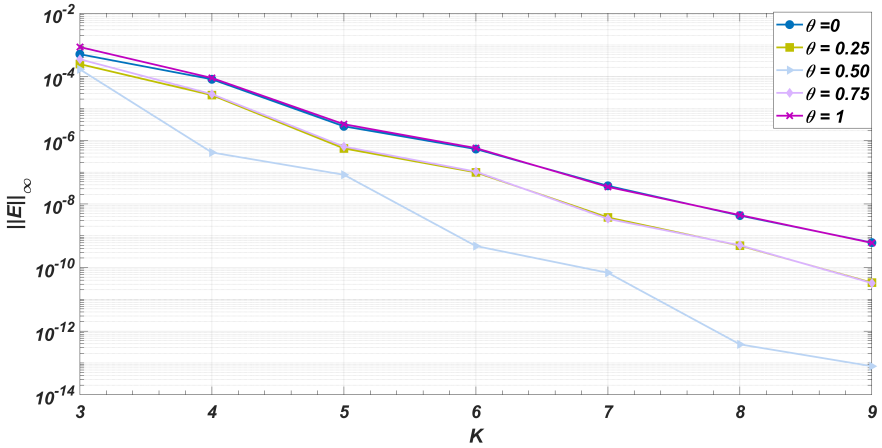

Figure 6.5 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter θ with $D = 0.1$, $\Delta x = 0.05$, $N = 50$ and $t_f = 2$ in Problem 6.3.

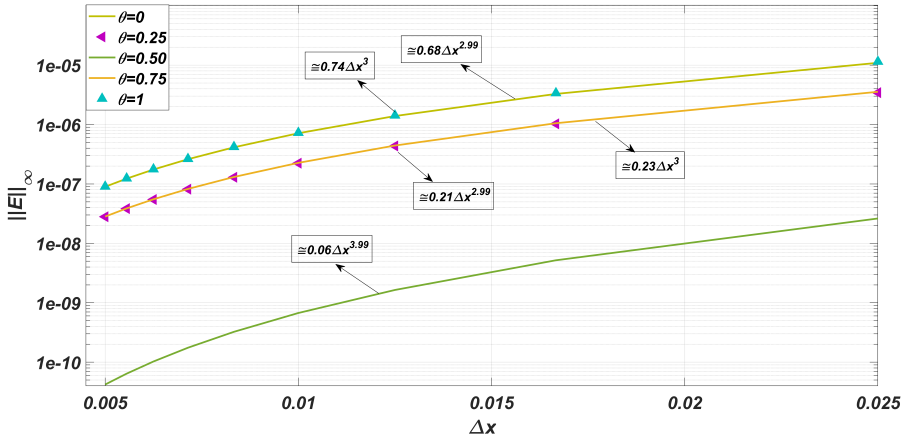

Figure 6.6 Convergence results of the present algorithm with respect to Δx for various values of the direction parameter θ with $D = 0.1$, $K = 4$, $N = 50$ and $t_f = 2$ (see Problem 6.3).

Problem 6.4 [154]

Let us consider the following singularly perturbed Burgers-Fisher equation [154]

$$u_t + Vuu_x = Du_{xx} + \beta u(1-u)\,, \quad x \in [0,1], \quad t \in [0, t_f] \tag{6.46}$$

with initial condition

$$u(x,0) = \frac{1}{2} + \frac{1}{2}tanh(\delta_1 x) \tag{6.47}$$

and Dirichlet boundary conditions

$$u(0,t) = \frac{1}{2} + \frac{1}{2}tanh(-\delta_1\delta_2 t) \tag{6.48}$$

$$u(1,t) = \frac{1}{2} + \frac{1}{2}tanh(\delta_1 - \delta_1\delta_2 t)\,. \tag{6.49}$$

where $\delta_1 = -V/4D$ and $\delta_1 = V/2 + 2D\beta/V$. The Burgers-Fisher equation with the conditions (6.47)-(6.49) has the following exact solution

$$u(x,t) = \frac{1}{2} + \frac{1}{2}tanh(\delta_1 x - \delta_1\delta_2 t)\,. \tag{6.50}$$

The present IELDTM and the FEM solutions [154] have been compared in the presence of weak advection and reaction effects with various values of diffusion coefficient D in Table 6.3. As observed from the table, the IELTDM offers more accurate results with far less *dof*. Even the problem behaves highly stiff with $D \ll 1$, the IELDTM produces far more accurate results than the FEM [154]. Advection-dominated cases of the Burgers-Fisher equation with $D \ll V$ are highly challenging due to the sharp

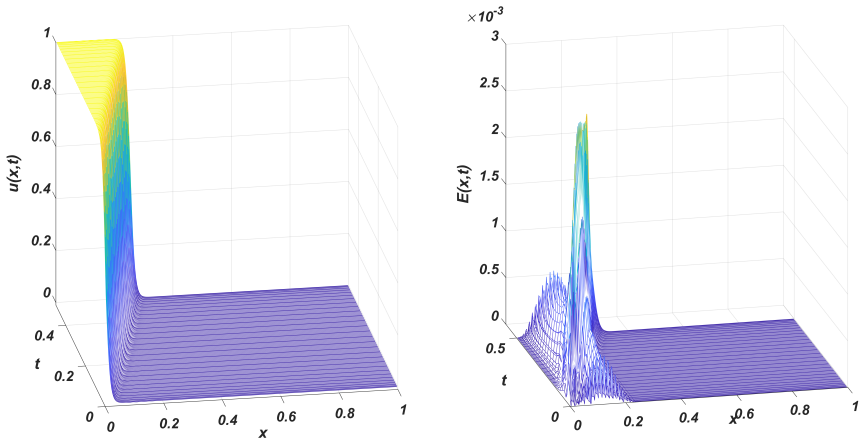

Figure 6.7 The IELDTM-ChSCM solution of Problem 6.4 and corresponding pointwise errors produced with $V = 1$, $\beta = 1$, $D = 2^{-8}$, $\Delta x = 0.003$, $N = 30$, $\theta = 0.5$ and $K = 2$.

behaviours of the solutions. With the selection of parameter values $V = 1$, $\beta = 1$ and $D = 2^{-8}$, the IELTM solution of the problem and the corresponding pointwise errors have been illustrated in Figure 6.7. As seen in the figure, the present method has the ability to produce accurate numerical results without producing any unwanted oscillation.

Table 6.3 Maximum pointwise errors produced with $V = 0.001$, $\beta = 0.001$ and $t_f = 1$ in Problem 6.4 and a) The FEM [154] with $M = 32$, $\Delta t = 0.0002$ and $dof = 16 \times 10^4$, b) The FEM [154] with $M = 64$, $\Delta t = 0.0002$ and $dof = 32 \times 10^4$, c) The IELDTM with $N = 10$, $M = 32$ and $dof = 640$, d) The IELDTM with $N = 10$, $M = 64$ and $dof = 1280$.

D	FEM^a [154]	FEM^b [154]	$IELDTM^c; K = 3$	$IELDTM^d; K = 4$
$D = 2^{-6}$	3.03E-06	2.97E-06	4.99E-16	5.55E-16
$D = 2^{-7}$	2.80E-06	2.49E-06	4.99E-16	6.10E-16
$D = 2^{-8}$	2.10E-06	1.52E-06	4.99E-16	5.55E-16
$D = 2^{-9}$	5.28E-06	2.65E-06	5.55E-16	5.55E-16
$D = 2^{-10}$	1.76E-05	1.69E-05	8.60E-15	9.99E-16
$D = 2^{-11}$	6.04E-05	5.29E-05	2.80E-13	2.00E-14
$D = 2^{-12}$	7.23E-04	2.73E-04	7.46E-12	6.07E-13
$D = 2^{-13}$	2.40E-03	7.24E-04	1.61E-10	1.78E-11
$D = 2^{-14}$	5.70E-03	4.10E-03	2.79E-09	4.67E-10

6.6 Summary

In this chapter, the IELDTM has been derived for the spatial parts of problems representing the advection-diffusion-reaction equations. Time integration of the ADR equations has been utilized with the Chebyshev collocation method. Spectral convergence analysis of the proposed method has been done, and convergence orders

have been determined with respect to the direction parameter θ. Various linear and nonlinear ADR equations have been considered as test problems to measure the potentiality of the derived method. The IELDTM has been proven to have effective convergence properties with both h—refinement and p-refinement strategies. All theoretical expectations have been verified through various numerical experiments. Advection-dominated linear and nonlinear processes have been successfully simulated with the IELDTM, and the method has been shown to produce accurate results for these challenging situations. Another essential advantage of the IELDTM is that the order refinement procedure of the method does not affect the degrees of freedom. The IELDTM, even with the use of far less degrees of freedom, has been proven to provide better accuracy than the FEM and FDM. In the next chapter, the IELDTM is derived for the spatial discretizations of the two- and three-dimensional advection-diffusion equations.

7
THE IELDTM FOR AD EQUATIONS IN TWO- AND THREE-DIMENSIONS

Various physical quantities such as mass, heat, energy, velocity, and concentration are represented in the AD equation [155]. (2+1) and (3+1) dimensional AD equations come out in the modelling of various real-life problems such as the tumour angiogenesis model, the tumour invasion model, heat transfer in draining film, dispersion of tracers in porous media, the spread of pollutants in rivers and stream, the dispersion of dissolved material in estuaries and coastal sea [4]. When the advection is dominant to the diffusion in the equation, the exact series-based solutions mostly fail and thus diverge [156]. Moreover, it is not easy to find out analytical solutions in nonlinear problems even the problem is not advection-dominated. In such scenarios, effective numerical methods need to be constructed over two- or three-dimensional meshes to observe the behaviour of the advection-diffusion mechanisms.

In this chapter, the IELDTM is derived for multidimensional advection-diffusion equations, which are more realistic than the one-dimensional form described in Chapter 6. The IELDTM is here constructed for two- and three-dimensional meshes, and the total degrees of freedom of the representations are minimized with the use of local differential transforms. Both (2+1)- and (3+1)-dimensional parabolic AD equations are reduced to a system of elliptic boundary value problems with the use of the ChSCM in time. Various realistic test equations including two-dimensional pure diffusion equation, two-dimensional linear AD equation, two-dimensional nonlinear AD equation, and three-dimensional linear AD equation are considered in numerical experiments. Both h-convergence and p-convergence of the present IELDTM are presented for two- and three-dimensional problems. Quantitative results are exhibited in comparison with the FEM, spectral method, and FDM.

7.1 IELDTM-ChSCM for (2+1)-dimensional AD equation

In this section, we introduce the IELDTM-ChSCM hybridization for solving the following (2+1)-dimensional AD equation

$$u_t + V_x u_x + V_y u_y = D_x u_{xx} + D_y u_{yy}, \quad (x, y) \in [a,\ b] \times [c,\ d], \quad t \in [0, t_f] \tag{7.1}$$

with Dirichlet boundary conditions

$$u(a, y, t) = f_1(y, t) \text{ and } u(b, y, t) = f_2(y, t) \tag{7.2}$$

$$u(x, c, t) = h_1(x, t) \text{ and } u(x, d, t) = h_2(x, t) \tag{7.3}$$

and initial condition

$$u(x, y, 0) = g(x, y) \tag{7.4}$$

where V_x and V_y are the advection constants, D_x and D_y are the diffusion constants and f_1, f_2, h_1, h_2 and g are the known smooth functions. The subscripts x, y and t represent differentiations with respect to spaces x, y and time t, respectively. In the following subsection, (2+1)-dimensional AD equation (7.1) is reduced to system of elliptic boundary value problems with the use of the Chebyshev spectral collocation method (ChSCM) in time.

7.1.1 Temporal Variation for (2+1)-Dimensional AD Equation

Let us approximate the time part of $u(x, y, t)$ with Nth order Chebyshev polynomials as follows [48]

$$u(x, y, t) = \sum_{n=0}^{N} \alpha_n c_n(x, y) \overline{T}_n(t) \tag{7.5}$$

where $c_n(x, y)$ represent the spatial parts of $u(x, y, t)$, $\alpha_n = 1$ for interior points and $\alpha_0 = \alpha_N = 1/2$ for boundary points. In equation (7.5), $\overline{T}_n(t)$ is defined in terms of the first kind Chebyshev polynomials $T_n(t)$ as

$$\overline{T}_n(t) = T_n\left(\frac{2t - t_f}{t_f}\right) = cos\left(ncos^{-1}\left(\frac{2t - t_f}{t_f}\right)\right). \tag{7.6}$$

We define the following CGL collocation points on $[0,\ t_f]$ as

$$t_n = \frac{t_f}{2}\left(1 - cos\left(\frac{\pi n}{N}\right)\right), \quad n = 0, 1, 2, \ldots,\ N. \tag{7.7}$$

The discrete orthogonality relation of the Chebyshev polynomials can be stated as

$$\sum_{n=0}^{N} \alpha_n \overline{T}_i(t_n)\overline{T}_j(t_n) = \gamma_i \delta_{ij} \tag{7.8}$$

with

$$\gamma_i = \begin{cases} \frac{N}{2}, & i \neq 0,\ N \\ N, & i = 0,\ N. \end{cases} \tag{7.9}$$

With the use of discrete orthogonality relations (7.7), we find that

$$c_j(x,y) = \kappa_j \sum_{n=0}^{N} \alpha_n \overline{T}_j(t_n) u(x,y,t_n). \tag{7.10}$$

where

$$\kappa_j = \begin{cases} \frac{2}{N}, & j \neq 0, N \\ \frac{1}{N}, & j = 0, N. \end{cases}$$

Thus, we can compute $u_t(x,y,t)$ at the predetermined collocation points as follows,

$$u_t\left(x,\ y, t_i\right) = \sum_{n=0}^{N} [A_t]_{in} c_n(x,y), \tag{7.11}$$

where $i = 0, 1, 2, \ldots,\ N$, $c_n(x,y) = u(x,y,t_n)$ and A_t is an $(N \times N)$ matrix defined by

$$[A_t]_{in} = \begin{cases} \xi_n \mu \sum_{j=0}^{N} (-1)^{j+1} j^2 cos\left(j\left(\pi - \frac{n\pi}{N}\right)\right), & i = 0 \\ \xi_n \mu \sum_{j=0}^{N} \frac{j sin\left(j\left(\pi - \frac{i\pi}{N}\right)\right) cos\left(j\left(\pi - \frac{n\pi}{N}\right)\right)}{sin\left(\pi - \frac{i\pi}{N}\right)}, & i = 1, 2, \ldots, N-1 \\ \xi_n \mu \sum_{j=0}^{N} j^2 cos\left(j\left(\pi - \frac{n\pi}{N}\right)\right), & i = N \end{cases} \tag{7.12}$$

with

$$\xi_n = \begin{cases} \frac{1}{N}, & n = 0, N \\ \frac{2}{N}, & n \neq 0, N \end{cases}$$

where $\mu = \frac{2}{t_f}$ and $[A_t]_{in}$ is called as the Chebyshev differentiation matrix. Inserting the approximation function (7.5) into main equation (7.1) in time collocation sense, we obtain the following two-dimensional system of elliptic PDEs

$$D_x \frac{\partial^2 \mathbf{c}}{\partial x^2} + D_y \frac{\partial^2 \mathbf{c}}{\partial y^2} - V_x \frac{\partial \mathbf{c}}{\partial x} - V_y \frac{\partial \mathbf{c}}{\partial y} - A_t \mathbf{c} = \mathbf{0} \tag{7.13}$$

where $\mathbf{c}(x,y) = [c_0(x,y),\ c_1(x,y),\ c_2(x,y), \ldots,\ c_N(x,y)]^T$ is $((N+1) \times 1)$ column vector. Applying the initial condition $u(x,y,0) = c_0(x,y) = g(x,y)$ reduces the

system into the following form

$$D_x \frac{\partial^2 \boldsymbol{c}}{\partial x^2} + D_y \frac{\partial^2 \boldsymbol{c}}{\partial y^2} - V_x \frac{\partial \boldsymbol{c}}{\partial x} - V_y \frac{\partial \boldsymbol{c}}{\partial y} - \overline{A}_t \boldsymbol{c} = \boldsymbol{G}(x, y) \tag{7.14}$$

where $\boldsymbol{G}(x, y)$ comes out due to the initial condition (7.4) and $\overline{A}_t$ is an $(N \times N)$ matrix reduced from A_t. Thus, equation (7.14) represents the $(N \times N)$ system of boundary value problems with the following conditions,

$$\boldsymbol{c}(a, y) = f_1(y, \overline{\boldsymbol{t}}) \quad \text{and} \quad \boldsymbol{c}(b, y) = f_2(y, \overline{\boldsymbol{t}}), \tag{7.15}$$

$$\boldsymbol{c}(x, c) = h_1(x, \overline{\boldsymbol{t}}) \quad \text{and} \quad \boldsymbol{c}(x, d) = h_2(x, \overline{\boldsymbol{t}}), \tag{7.16}$$

where $\overline{\boldsymbol{t}} = [t_1,\ t_2,\ \ldots,\ t_N]^T$ defined in equation (7.6).

7.1.2 The IELDTM for Two-Dimensional Problems

In this subsection, the IELDTM is derived to solve two-dimensional system of elliptic PDE (7.14). The detailed background of the differential transformation for multi-dimensional analytic functions can be seen in literature [117].

Let us divide the spatial domain $[a, b] \times [c, d]$ into $M_x M_y$ spatial element $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$ with $\Delta x = x_{i+1} - x_i$, $\Delta y = y_{j+1} - y_j$ and the centre of each element is defined at $\left(x_{i+\frac{1}{2}},\ y_{j+\frac{1}{2}}\right)$. The function $\boldsymbol{c}(x, y)$ can be locally represented by the convergent local Taylor expansion about $\left(x_{i+\frac{1}{2}},\ y_{j+\frac{1}{2}}\right)$ as follows

$$\begin{aligned} \boldsymbol{c}_{ij}(x, y) = & \sum_{k=0}^{K} \sum_{p=0}^{K-k} \boldsymbol{C}_{ij}(k, p) \left(x - x_{i+\frac{1}{2}}\right)^k \left(y - y_{j+\frac{1}{2}}\right)^p + \\ & O\left(\left(x - x_{i+\frac{1}{2}}\right)^{K+1} + \left(y - y_{j+\frac{1}{2}}\right)^{K+1}\right) \end{aligned} \tag{7.17}$$

where $\boldsymbol{C}_{ij}(k, p) = \frac{1}{k!p!} \frac{\partial^{k+p} \boldsymbol{c}(x,y)}{\partial x^k \partial y^p}\Big|_{x = x_{i+\frac{1}{2}},\ y = y_{j+\frac{1}{2}}}$ is the local differential transform of the function $\boldsymbol{c}(x, y)$ about $x = x_{i+\frac{1}{2}}$ and $y = y_{j+\frac{1}{2}}$ for $i = 0, 1, \ldots, M_x - 1$ and $j = 0, 1, \ldots, M_y - 1$. To completely describe the convergent local solution $\boldsymbol{c}_{ij}(x, y)$ of order K, equation (7.17) includes $N\frac{(K+1)(K+2)}{2}$ coefficients to be determined. The crucial advantage of the differential transformation leads to the reduction of these unknown coefficients to $N(2K + 1)$. Taking the differential transform of equation (7.14) yields

$$\begin{aligned} \boldsymbol{C}_{ij}(k, p+2) = & \frac{1}{D_y(p+1)(p+2)} \Big[-D_x(k+1)(k+2)\boldsymbol{C}_{ij}(k+2, p) + \\ & V_x(k+1)\boldsymbol{C}_{ij}(k+1, p) + V_y(p+1)\boldsymbol{C}_{ij}(k, p+1) + \overline{A}_t \boldsymbol{C}_{ij}(k, p) + \boldsymbol{G}_{ij}(k, p) \Big] \end{aligned} \tag{7.18}$$

where $k,p=0,1,...,K-1$, $i=0,1,\ldots,\ M_x-1$, $j=0,1,\ldots,\ M_y-1$ and $\boldsymbol{G}_{ij}(k,p)$ is the local differential transform of $\boldsymbol{G}(x,y)$ at $x=x_{i+\frac{1}{2}}$ and $y=y_{j+\frac{1}{2}}$. Thus, with the use of recursion (7.18), one only needs to determine the following set of $N(2K+1)$ local values

$$V_{ij}=\{\boldsymbol{C}_{ij}(k,0)\,|\,k\leq K\ \wedge\ k\in N\}\cup\{\boldsymbol{C}_{ij}(k,1)\,|\,k\leq K-1\ \wedge\ k\in N\} \tag{7.19}$$

where $i=0,1,\ldots,\ M_x-1$ and $j=0,1,\ldots,\ M_y-1$. Thus, the total degrees of freedom is $NM_xM_y(2K+1)$. As proven in numerical experiments, the minimization of the local degrees of freedom leads to higher-order approximations with optimum costs.

The function $\boldsymbol{c}_{ij}(x,y)$ is assumed to be analytic for all (i,j) and has the radius of convergences $\rho_x^i>\Delta x$ and $\rho_y^j>\Delta y$. This assumption leads us to search about the relations between neighbour solutions $\boldsymbol{c}_{ij}(x,y)$, $\boldsymbol{c}_{i(j+1)}(x,y)$ and $\boldsymbol{c}_{(i+1)j}(x,y)$ in terms of their local behaviours. Let us define the intervals $P^i=[x_{i+\frac{1}{2}}-\ \rho_x^i, x_{i+\frac{1}{2}}+\rho_x^i]$, $P^{i+1}=[x_{i+\frac{3}{2}}-\rho_x^{i+1}, x_{i+\frac{3}{2}}+\rho_x^{i+1}]$, $W^j=[y_{j+\frac{1}{2}}-\ \rho_y^j, y_{j+\frac{1}{2}}+\ \rho_y^j]$, $W^{j+1}=[y_{j+\frac{3}{2}}-\rho_y^{j+1}, y_{j+\frac{3}{2}}+\rho_y^{j+1}]$, $P=[x_{i+\frac{1}{2}},\ x_{i+\frac{3}{2}}]$. and $W=[y_{j+\frac{1}{2}},\ y_{j+\frac{3}{2}}]$. With the assumptions of $P\subset\left(P^i\cap P^{i+1}\right)$ and $W\subset\left(W^j\cap W^{j+1}\right)$, we state the following conclusions;

* C^0 and C^1 continuity conditions can be applied on the region $[x_{i+\frac{1}{2}},\ x_{i+\frac{3}{2}}]\times[y_{j+\frac{1}{2}},\ y_{j+\frac{3}{2}}]$.
* Any point on the interval $[x_{i+\frac{1}{2}},\ x_{i+\frac{3}{2}}]$ can be written as $x^*=x_{i+\frac{1}{2}}+(1-\theta_x)\Delta x$, where $0\leq\theta_x\leq 1$.
* Any point on the interval $[y_{j+\frac{1}{2}},\ x_{j+\frac{3}{2}}]$ can be written as $y^*=y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y$, where $0\leq\theta_y\leq 1$.

We define the sets $H_i^x=\{x_i,x_i+dx,...,x_{i+1}\}$ and $H_j^y=\left\{y_j,y_j+dy,...,y_{j+1}\right\}$ with $dx=\frac{\Delta x}{S}$ and $dy=\frac{\Delta y}{S}$ where S denotes the partition number of each edge. The sets H_i^x and H_j^y include all possible collocation points in x and y directions, respectively. C^0 and C^1 continuity conditions in y direction lead to

$$\boldsymbol{c}_{ij}(x_i+qdx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y)=\boldsymbol{c}_{i(j+1)}(x_i+qdx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y), \tag{7.20}$$

$$\left(\boldsymbol{c}_{ij}\right)_x(x_i+qdx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y)=\left(\boldsymbol{c}_{i(j+1)}\right)_x(x_i+qdx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y), \tag{7.21}$$

$$\left(\boldsymbol{c}_{ij}\right)_y(x_i+qdx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y)=\left(\boldsymbol{c}_{i(j+1)}\right)_y(x_i+qdx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y), \tag{7.22}$$

where $i = 0, 1, \ldots, M_x - 1$, $j = 0, 1, \ldots, M_y - 2$ and $q = 0, 1, ..., S$. Similarly, C^0 and C^1 continuity conditions in x direction lead to

$$\boldsymbol{c}_{ij}(x_{i+\frac{1}{2}} + (1-\theta_x)\Delta x, y_j + qdy) = \boldsymbol{c}_{(i+1)j}(x_{i+\frac{1}{2}} + (1-\theta_x)\Delta x, y_j + qdy), \quad (7.23)$$

$$\left(\boldsymbol{c}_{ij}\right)_x(x_{i+\frac{1}{2}} + (1-\theta_x)\Delta x, y_j + qdy) = \left(\boldsymbol{c}_{(i+1)j}\right)_x(x_{i+\frac{1}{2}} + (1-\theta_x)\Delta x, y_j + qdy), \quad (7.24)$$

$$\left(\boldsymbol{c}_{ij}\right)_y(x_{i+\frac{1}{2}} + (1-\theta_x)\Delta x, y_j + qdy) = \left(\boldsymbol{c}_{(i+1)j}\right)_y(x_{i+\frac{1}{2}} + (1-\theta_x)\Delta x, y_j + qdy), \quad (7.25)$$

where $i = 0, 1, \ldots, M_x - 2$, $j = 0, 1, \ldots, M_y - 1$ and $q = 0, 1, ..., S$. C^0 and C^1 continuity conditions for boundaries yield

$$\boldsymbol{c}_{0j}(x_0, y_j + qdy) = f_1(y_j + qdy, \bar{\boldsymbol{t}}),\ q = 0, 1, ..., S,\ j = 0, 1, ..., M_y - 1, \quad (7.26)$$

$$\left(\boldsymbol{c}_{0j}\right)_y(x_0, y_j + qdy) = (f_1)_y(y_j + qdy, \bar{\boldsymbol{t}}),\ q = 0, 1, ..., S,\ j = 0, 1, ..., M_y - 1, \quad (7.27)$$

$$\boldsymbol{c}_{i0}(x_i + qdx, y_0) = h_1(x_i + qdx, \bar{\boldsymbol{t}}),\ q = 0, 1, ..., S,\ i = 0, 1, ..., M_x - 1, \quad (7.28)$$

$$(\boldsymbol{c}_{i0})_x(x_i + qdx, y_0) = (h_1)_x(x_i + qdx, \bar{\boldsymbol{t}}),\ q = 0, 1, ..., S,\ i = 0, 1, ..., M_x - 1, \quad (7.29)$$

$$\boldsymbol{c}_{(M_x-1)j}(x_{M_x}, y_j + qdy) = f_2(y_j + qdy, \bar{\boldsymbol{t}}),\ q = 0, 1, ..., S,\ j = 0, 1, ..., M_y - 1, \quad (7.30)$$

$$\begin{aligned}\left(\boldsymbol{c}_{(M_x-1)j}\right)_y(x_{M_x}, y_j + qdy) &= (f_2)_y(y_j + qdy, \bar{\boldsymbol{t}}),\\ q = 0, 1, ..., S,\ j &= 0, 1, ..., M_y - 1,\end{aligned} \quad (7.31)$$

$$\boldsymbol{c}_{i(M_y-1)}(x_i + qdx, y_{M_y}) = h_2(x_i + qdx, \bar{\boldsymbol{t}}),\ q = 0, 1, ..., S,\ i = 0, 1, ..., M_x - 1, \quad (7.32)$$

$$\left(\boldsymbol{c}_{i(M_y-1)}\right)_x(x_i+qdx,y_{M_y})=(h_2)_x(x_i+qdx,\overline{\boldsymbol{t}}),\\ q=0,1,...,S,\ i=0,1,...,M_x-1. \tag{7.33}$$

Thus, combining all continuity conditions defined in (7.20)-(7.33) leads to the following linear algebraic system

$$M\zeta=P \tag{7.34}$$

where M is an $\left(N\left[6M_xM_y+M_x+M_y\right](S+1)\right)\times\left(NM_xM_y(2K+1)\right)$ matrix, P is an $\left(N\left[6M_xM_y+M_x+M_y\right](S+1)\right)\times 1$ column vector and ζ is an $\left(NM_xM_y(2K+1)\right)\times 1$ column vector defined by

$$\zeta=\left[V_{00},\ V_{01},...,V_{(M_x-1)(M_y-1)}\right]^T \tag{7.35}$$

where V_{ij} is a $1\times(N(2K+1))$ vector given in equation (7.19). By a suitable selection of S with $S>\frac{M_xM_y(2K+1)}{6M_xM_y+M_x+M_y}$, equation (7.34) describes an over-determined linear system. By minimizing the residual norm $\left\|M\zeta-P\right\|$ in least square sense, all unknown values V_{ij} are obtained. Note that θ_x and θ_y determine the direction of the continuity conditions, i.e. the schemes are

* forward directional with $\theta_x=0$ and $\theta_y=0$,
* backward directional with $\theta_x=1$ and $\theta_y=1$,
* central directional with $\theta_x=1/2$ and $\theta_y=1/2$.

As will be proven in the following sections, the central directional schemes are more accurate than the rest of them.

7.2 IELDTM-ChSCM for (3+1)-Dimensional AD Equation

In this section, we introduce the IELDTM-ChSCM hybridization for solving the following (3+1)-dimensional advection-diffusion AD equation

$$u_t+V_xu_x+V_yu_y+V_zu_z=D_xu_{xx}+D_yu_{yy}+D_zu_{zz},\\ (x,y,z)\in[a,\ b]\times[c,\ d]\times[e,\ f],\quad t\in[0,t_f] \tag{7.36}$$

with Dirichlet boundary conditions

$$u(a,y,z,t)=f_1(y,z,t)\ \text{ and }\ u(b,y,z,t)=f_2(y,z,t) \tag{7.37}$$

$$u(x,c,z,t) = h_1(x,z,t) \text{ and } u(x,d,z,t) = h_2(x,z,t) \tag{7.38}$$

$$u(x,y,e,t) = w_1(x,y,t) \text{ and } u(x,y,f,t) = w_2(x,y,t) \tag{7.39}$$

and initial condition

$$u(x,y,z,0) = g(x,y,z) \tag{7.40}$$

where V_x, V_y and V_z are the advection constants, D_x, D_y and D_z are the diffusion constants and f_1, f_2, h_1, h_2, w_1, w_2 and g are the known smooth functions. The subscripts x, y, z and t represent differentiations with respect to spaces x, y, z and time t, respectively. As constructed for the (2+1)-dimensional AD equation (7.1), (3+1)-dimensional AD equation will be reduced to a system of three-dimensional elliptic boundary value problems with the use of the ChSCM in time.

7.2.1 Temporal Variation for (3+1)-Dimensional AD Equation

Since all the details of the ChSCM have been given in subsection (7.1.1), just the essential formulae are presented here. Let us approximate the time part of $u(x,y,z,t)$ with Nth-order Chebyshev polynomials as

$$u(x,y,z,t) = \sum_{n=0}^{N} \alpha_n c_n(x,y,z) \overline{T}_n(t) \tag{7.41}$$

where $c_n(x,y,z)$ represent the spatial parts of $u(x,y,z,t)$. The use of approximation function (7.41) into (3+1)-dimensional ADR equation (7.36) leads to the following three-dimensional system of elliptic PDEs

$$D_x \frac{\partial^2 \mathbf{c}}{\partial x^2} + D_y \frac{\partial^2 \mathbf{c}}{\partial y^2} + D_y \frac{\partial^2 \mathbf{c}}{\partial z^2} - V_x \frac{\partial \mathbf{c}}{\partial x} - V_y \frac{\partial \mathbf{c}}{\partial y} - V_z \frac{\partial \mathbf{c}}{\partial z} - A_t \mathbf{c} = \mathbf{0} \tag{7.42}$$

where $\mathbf{c}(x,y,z) = [c_0(x,y,z),\ c_1(x,y,z),\ c_2(x,y,z),...,\ c_N(x,y,z)]^T$ is an $((N+1)\times 1)$ column vector. Applying the initial condition $u(x,y,z,0) = c_0(x,y,z) = g(x,y,z)$ reduces to

$$D_x \frac{\partial^2 \mathbf{c}}{\partial x^2} + D_y \frac{\partial^2 \mathbf{c}}{\partial y^2} + D_z \frac{\partial^2 \mathbf{c}}{\partial z^2} - V_x \frac{\partial \mathbf{c}}{\partial x} - V_y \frac{\partial \mathbf{c}}{\partial y} - V_z \frac{\partial \mathbf{c}}{\partial z} - \overline{A}_t \mathbf{c} = \boldsymbol{G}(x,y,z) \tag{7.43}$$

where $\boldsymbol{G}(x,y,z)$ occurs due to the initial condition (7.40) and $\overline{A}_t$ is an $(N\times N)$ matrix reduced from A_t. Thus, equation (7.43) represents the $(N\times N)$ system of nonhomogeneous three-dimensional elliptic PDEs with the following boundary

conditions,

$$\boldsymbol{c}(a,y,z) = f_1(y,z,\overline{\boldsymbol{t}}) \quad and \quad \boldsymbol{c}(b,y,z) = f_2(y,z,\overline{\boldsymbol{t}}), \tag{7.44}$$

$$\boldsymbol{c}(x,c,z) = h_1(x,z,\overline{\boldsymbol{t}}) \quad and \quad \boldsymbol{c}(x,d,z) = h_2(x,z,\overline{\boldsymbol{t}}), \tag{7.45}$$

$$\boldsymbol{c}(x,y,e) = w_1(x,y,\overline{\boldsymbol{t}}) \quad and \quad \boldsymbol{c}(x,y,f) = w_2(x,y,\overline{\boldsymbol{t}}), \tag{7.46}$$

where $\overline{\boldsymbol{t}} = [t_1,\ t_2,\ \ldots,\ t_N]^T$ defined in equation (7.6).

7.2.2 The IELDTM for Three-Dimensional Problems

The IELDTM is derived here to solve three-dimensional system of boundary value problems (7.43)-(7.46). In Section 7.2, the method was explained with all details, and here we omit those details for the sake of brevity.

Let us divide the spatial domain $[a,b]\times[c,d]\times[e,f]$ into $M_xM_yM_z$ spatial element $[x_i,x_{i+1}]\times[y_j,y_{j+1}]\times[z_r,z_{r+1}]$ with $\Delta x = x_{i+1}-x_i$, $\Delta y = y_{j+1}-y_j$, $\Delta z = z_{r+1}-z_r$ and the centre of each element is defined at $\left(x_{i+\frac{1}{2}},\ y_{j+\frac{1}{2}},z_{r+\frac{1}{2}}\right)$. The function $\boldsymbol{c}(x,y,z)$ can be locally represented with the convergent local Taylor expansion about $\left(x_{i+\frac{1}{2}},\ y_{j+\frac{1}{2}},z_{r+\frac{1}{2}}\right)$ as follows

$$\begin{aligned}\boldsymbol{c}_{ijr}(x,y,z) =& \sum_{h=0}^{K}\sum_{p=0}^{h}\sum_{k=0}^{h-p}\boldsymbol{C}_{ijr}(k,p,h-p-k)\left(x-x_{i+\frac{1}{2}}\right)^k\left(y-y_{j+\frac{1}{2}}\right)^p \\ &\left(z-z_{r+\frac{1}{2}}\right)^{h-p-k}+O\left(\left(x-x_{i+\frac{1}{2}}\right)^{K+1}+\left(y-y_{j+\frac{1}{2}}\right)^{K+1}+\left(z-y_{r+\frac{1}{2}}\right)^{K+1}\right)\end{aligned} \tag{7.47}$$

where $\boldsymbol{C}_{ijr}(k,p,h) = \frac{1}{k!p!h!}\frac{\partial^{k+p+h}\boldsymbol{c}(x,y,z)}{\partial x^k\partial y^p\partial z^h}\Big|_{x=x_{i+\frac{1}{2}},\ y=y_{j+\frac{1}{2}},\ z=z_{r+\frac{1}{2}}}$ is the local differential transform of the function $\boldsymbol{c}(x,y,z)$ about $x = x_{i+\frac{1}{2}}$, $y = y_{j+\frac{1}{2}}$ and $z = z_{r+\frac{1}{2}}$ for $i = 0,1,\ldots,M_x-1$, $j = 0,1,\ldots,M_y-1$ and $r = 0,1,\ldots,M_z-1$.The convergent local solution $\boldsymbol{c}_{ijr}(x,y,z)$ of order K includes $N\frac{(K+1)(K+2)(K+3)}{6}$ coefficients to be determined. With the use the differential transformation of three-dimensional equation (7.43), the number of unknowns is reduced to $N(K+1)^2$. Taking the differential transform of equation (7.43) yields

$$\begin{aligned}\boldsymbol{C}_{ijr}(k,p,h+2) = \frac{1}{D_z(h+1)(h+2)}\big[&-D_x(k+1)(k+2)\boldsymbol{C}_{ijr}(k+2,p,h)- \\ &D_y(p+1)(p+2)\boldsymbol{C}_{ijr}(k,p+2,h)+V_x(k+1)\boldsymbol{C}_{ijr}(k+1,p,h)+ \\ &V_y(p+1)\boldsymbol{C}_{ijr}(k,p+1,h)+V_z(h+1)\boldsymbol{C}_{ijr}(k,p,h+1)+ \\ &\overline{A}_t\boldsymbol{C}_{ijr}(k,p,h)+\boldsymbol{G}_{ijr}(k,p,h)\big]\end{aligned} \tag{7.48}$$

where $k,p,h=0,1,...,K-1$ and $\boldsymbol{G}_{ijr}(k,p,r)$ is the local differential transform of the $\boldsymbol{G}(x,y,z)$. Thus, with the use of recursion (7.48), one only needs to determine the following set of $N(K+1)^2$ local values

$$V_{ijr}=\{\boldsymbol{C}_{ijr}(k,p,0)\,|\,k+p\le K\ \wedge\ k,p\in N\}\cup\{\boldsymbol{C}_{ijr}(k,p,1)\,|\,k+p\le K-1\ \wedge\ k,p\in N\} \tag{7.49}$$

where $i=0,1,\ldots,\ M_x-1,\ j=0,1,\ldots,\ M_y-1$ and $r=0,1,\ldots,\ M_z-1$. The total degrees of freedom is $M_xM_yM_zN(K+1)^2$. We define the set of collocation points $H_i^x=\{x_i,x_i+dx,...,x_{i+1}\}$, $H_j^y=\{y_j,y_j+dy,...,y_{j+1}\}$ and $H_r^z=\{z_r,z_r+dz,...,z_{r+1}\}$ with $dx=\frac{\Delta x}{S}$, $dy=\frac{\Delta y}{S}$ and $dz=\frac{\Delta z}{S}$ where S denotes the partition number of each edge. C^0 and C^1 continuity conditions in x direction lead to

$$\begin{gathered}\boldsymbol{c}_{ijr}(x_{i+\frac{1}{2}}+(1-\theta_x)\Delta x,y_j+q_1dy,\ z_r+q_2dz)=\\ \boldsymbol{c}_{(i+1)jr}(x_{i+\frac{1}{2}}+(1-\theta_x)\Delta x,y_j+q_1dy,\ z_r+q_2dz),\end{gathered} \tag{7.50}$$

$$\begin{gathered}\left(\boldsymbol{c}_{ijr}\right)_x(x_{i+\frac{1}{2}}+(1-\theta_x)\Delta x,y_j+q_1dy,\ z_r+q_2dz)=\\ \left(\boldsymbol{c}_{(i+1)jr}\right)_x(x_{i+\frac{1}{2}}+(1-\theta_x)\Delta x,y_j+q_1dy,\ z_r+q_2dz),\end{gathered} \tag{7.51}$$

$$\begin{gathered}\left(\boldsymbol{c}_{ijr}\right)_y(x_{i+\frac{1}{2}}+(1-\theta_x)\Delta x,y_j+q_1dy,\ z_r+q_2dz)=\\ \left(\boldsymbol{c}_{(i+1)jr}\right)_y(x_{i+\frac{1}{2}}+(1-\theta_x)\Delta x,y_j+q_1dy,\ z_r+q_2dz),\end{gathered} \tag{7.52}$$

$$\begin{gathered}\left(\boldsymbol{c}_{ijr}\right)_z(x_{i+\frac{1}{2}}+(1-\theta_x)\Delta x,y_j+q_1dy,\ z_r+q_2dz)=\\ \left(\boldsymbol{c}_{(i+1)jr}\right)_z(x_{i+\frac{1}{2}}+(1-\theta_x)\Delta x,y_j+q_1dy,\ z_r+q_2dz),\end{gathered} \tag{7.53}$$

where $i=0,1,\ldots,\ M_x-2,\ j=0,1,\ldots,\ M_y-1,\ r=0,1,\ldots,\ M_z-1$ and $q_1,q_2=0,1,...,S$. C^0 and C^1 continuity conditions in y direction lead to

$$\begin{gathered}\boldsymbol{c}_{ijr}(x_i+q_1dx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y,\ \ z_r+q_2dz)=\\ \boldsymbol{c}_{i(j+1)r}(x_i+q_1dx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y,\ \ z_r+q_2dz),\end{gathered} \tag{7.54}$$

$$\begin{gathered}\left(\boldsymbol{c}_{ijr}\right)_x(x_i+q_1dx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y,\ \ z_r+q_2dz)=\\ \left(\boldsymbol{c}_{i(j+1)r}\right)_x(x_i+q_1dx,y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y,\ \ z_r+q_2dz),\end{gathered} \tag{7.55}$$

$$\begin{aligned}\left(\boldsymbol{c}_{ijr}\right)_y(x_i+q_1dx, y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y,\ \ z_r+q_2dz) =\\ \left(\boldsymbol{c}_{i(j+1)r}\right)_y(x_i+q_1dx, y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y,\ \ z_r+q_2dz),\end{aligned} \tag{7.56}$$

$$\begin{aligned}\left(\boldsymbol{c}_{ijr}\right)_z(x_i+q_1dx, y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y,\ \ z_r+q_2dz) =\\ \left(\boldsymbol{c}_{i(j+1)r}\right)_z(x_i+q_1dx, y_{j+\frac{1}{2}}+(1-\theta_y)\Delta y,\ \ z_r+q_2dz),\end{aligned} \tag{7.57}$$

where $i=0,1,\ldots,\ M_x-1$, $j=0,1,\ldots,\ M_y-2$, and $r=0,1,\ldots,\ M_z-1$ and $q_1,q_2=0,1,...,S$. Similarly, C^0 and C^1 continuity conditions in z direction lead to

$$\begin{aligned}\boldsymbol{c}_{ijr}(x_i+q_1dx, y_j+q_2dy,\ \ z_{r+\frac{1}{2}}+(1-\theta_z)\Delta z) =\\ \boldsymbol{c}_{ij(r+1)}(x_i+q_1dx, y_j+q_2dy,\ \ z_{r+\frac{1}{2}}+(1-\theta_z)\Delta z),\end{aligned} \tag{7.58}$$

$$\begin{aligned}\left(\boldsymbol{c}_{ijr}\right)_x(x_i+q_1dx, y_j+q_2dy,\ \ z_{r+\frac{1}{2}}+(1-\theta_z)\Delta z) =\\ \left(\boldsymbol{c}_{ij(r+1)}\right)_x(x_i+q_1dx, y_j+q_2dy,\ \ z_{r+\frac{1}{2}}+(1-\theta_z)\Delta z),\end{aligned} \tag{7.59}$$

$$\begin{aligned}\left(\boldsymbol{c}_{ijr}\right)_y(x_i+q_1dx, y_j+q_2dy,\ \ z_{r+\frac{1}{2}}+(1-\theta_z)\Delta z) =\\ \left(\boldsymbol{c}_{ij(r+1)}\right)_y(x_i+q_1dx, y_j+q_2dy,\ \ z_{r+\frac{1}{2}}+(1-\theta_z)\Delta z),\end{aligned} \tag{7.60}$$

$$\begin{aligned}\left(\boldsymbol{c}_{ijr}\right)_z(x_i+q_1dx, y_j+q_2dy,\ \ z_{r+\frac{1}{2}}+(1-\theta_z)\Delta z) =\\ \left(\boldsymbol{c}_{ij(r+1)}\right)_z(x_i+q_1dx, y_j+q_2dy,\ \ z_{r+\frac{1}{2}}+(1-\theta_z)\Delta z),\end{aligned} \tag{7.61}$$

where $i=0,1,\ldots,\ M_x-1$, $j=0,1,\ldots,\ M_y-1$, $r=0,1,\ldots,\ M_z-2$ and $q_1,q_2=0,1,...,S$. C^0 and C^1 continuity conditions for boundaries yield

$$\begin{aligned}\boldsymbol{c}_{0jr}(x_0, y_j+q_1dy, z_r+q_2dz) = f_1(y_j+q_1dy, z_r+q_2dz, \bar{\boldsymbol{t}}),\\ q_1,q_2=0,1,...,S\ ,\ j=0,1,...,M_y-1,\ r=0,1,\ldots,\ M_z-1\end{aligned} \tag{7.62}$$

$$\begin{aligned}\left(\boldsymbol{c}_{0jr}\right)_y(x_0, y_j+q_1dy, z_r+q_2dz) = (f_1)_y(y_j+q_1dy, z_r+q_2dz, \bar{\boldsymbol{t}}),\\ q_1,q_2=0,1,...,S\ ,\ j=0,1,...,M_y-1,\ r=0,1,\ldots,\ M_z-1\end{aligned} \tag{7.63}$$

$$\left(\boldsymbol{c}_{0jr}\right)_z(x_0, y_j + q_1 dy, z_r + q_2 dz) = (f_1)_z(y_j + q_1 dy, z_r + q_2 dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2 = 0, 1, ..., S\ ,\ j = 0, 1, ..., M_y - 1,\ r = 0, 1, \ldots,\ M_z - 1, \tag{7.64}$$

$$\boldsymbol{c}_{(M_x-1)jr}(x_{M_x}, y_j + q_1 dy, z_r + q_2 dz) = f_2(y_j + q_1 dy, z_r + q_2 dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2 = 0, 1, ..., S\ ,\ j = 0, 1, ..., M_y - 1,\ r = 0, 1, \ldots,\ M_z - 1, \tag{7.65}$$

$$\left(\boldsymbol{c}_{(M_x-1)jr}\right)_y(x_{M_x}, y_j + q_1 dy, z_r + q_2 dz) = (f_2)_y(y_j + q_1 dy, z_r + q_2 dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2 = 0, 1, ..., S\ ,\ j = 0, 1, ..., M_y - 1,\ r = 0, 1, \ldots,\ M_z - 1, \tag{7.66}$$

$$\left(\boldsymbol{c}_{(M_x-1)jr}\right)_z(x_{M_x}, y_j + q_1 dy, z_r + q_2 dz) = (f_2)_z(y_j + q_1 dy, z_r + q_2 dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2 = 0, 1, ..., S\ ,\ j = 0, 1, ..., M_y - 1,\ r = 0, 1, \ldots,\ M_z - 1, \tag{7.67}$$

$$\boldsymbol{c}_{i0r}(x_i + q_1 dx, y_0, z_r + q_2 dz) = h_1(x_i + q_1 dx, z_r + q_2 dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2 = 0, 1, ..., S\ ,\ i = 0, 1, ..., M_x - 1,\ r = 0, 1, \ldots,\ M_z - 1, \tag{7.68}$$

$$(\boldsymbol{c}_{i0r})_x(x_i + q_1 dx, y_0, z_r + q_2 dz) = (h_1)_x(x_i + q_1 dx, z_r + q_2 dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2 = 0, 1, ..., S\ ,\ i = 0, 1, ..., M_x - 1,\ r = 0, 1, \ldots,\ M_z - 1, \tag{7.69}$$

$$(\boldsymbol{c}_{i0r})_z(x_i + q_1 dx, y_0, z_r + q_2 dz) = (h_1)_z(x_i + q_1 dx, z_r + q_2 dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2 = 0, 1, ..., S\ ,\ i = 0, 1, ..., M_x - 1,\ r = 0, 1, \ldots,\ M_z - 1, \tag{7.70}$$

$$\boldsymbol{c}_{i(M_y-1)r}(x_i + q_1 dx, y_{M_y}, z_r + q_2 dz) = h_2(x_i + q_1 dx, z_r + q_2 dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2 = 0, 1, ..., S\ ,\ i = 0, 1, ..., M_x - 1,\ r = 0, 1, \ldots,\ M_z - 1, \tag{7.71}$$

$$\left(\boldsymbol{c}_{i(M_y-1)r}\right)_x(x_i + q_1 dx, y_{M_y}, z_r + q_2 dz) = (h_2)_x(x_i + q_1 dx, z_r + q_2 dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2 = 0, 1, ..., S\ ,\ i = 0, 1, ..., M_x - 1,\ r = 0, 1, \ldots,\ M_z - 1, \tag{7.72}$$

$$\left(\boldsymbol{c}_{i(M_y-1)r}\right)_z(x_i+q_1dx, y_{M_y}, z_r+q_2dz)=(h_2)_z(x_i+q_1dx, z_r+q_2dz, \bar{\boldsymbol{t}}),$$
$$q_1, q_2=0,1,...,S\ ,\ i=0,1,...,M_x-1,\ r=0,1,\ldots,\ M_z-1, \tag{7.73}$$

$$\boldsymbol{c}_{ij0}(x_i+q_1dx, y_j+q_2dy, z_0)=w_1(x_i+q_1dx, y_j+q_2dy, \bar{\boldsymbol{t}}),$$
$$q_1, q_2=0,1,...,S\ ,\ i=0,1,...,M_x-1,\ j=0,1,\ldots,\ M_y-1, \tag{7.74}$$

$$\left(\boldsymbol{c}_{ij0}\right)_x(x_i+q_1dx, y_j+q_2dy, z_0)=(w_1)_x(x_i+q_1dx, y_j+q_2dy, \bar{\boldsymbol{t}}),$$
$$q_1, q_2=0,1,...,S\ ,\ i=0,1,...,M_x-1,\ j=0,1,\ldots,\ M_y-1 \tag{7.75}$$

$$\left(\boldsymbol{c}_{ij0}\right)_y(x_i+q_1dx, y_j+q_2dy, z_0)=(w_1)_y(x_i+q_1dx, y_j+q_2dy, \bar{\boldsymbol{t}}),$$
$$q_1, q_2=0,1,...,S\ ,\ i=0,1,...,M_x-1,\ j=0,1,\ldots,\ M_y-1 \tag{7.76}$$

$$\boldsymbol{c}_{ij(M_z-1)}(x_i+q_1dx, y_j+q_2dy, z_{M_z})=w_2(x_i+q_1dx, y_j+q_2dy, \bar{\boldsymbol{t}}),$$
$$q_1, q_2=0,1,...,S\ ,\ i=0,1,...,M_x-1,\ j=0,1,\ldots,\ M_y-1, \tag{7.77}$$

$$\left(\boldsymbol{c}_{ij(M_z-1)}\right)_x(x_i+q_1dx, y_j+q_2dy, z_{M_z})=(w_2)_x(x_i+q_1dx, y_j+q_2dy, \bar{\boldsymbol{t}}),$$
$$q_1, q_2=0,1,...,S\ ,\ i=0,1,...,M_x-1,\ j=0,1,\ldots,\ M_y-1, \tag{7.78}$$

$$\left(\boldsymbol{c}_{ij(M_z-1)}\right)_y(x_i+q_1dx, y_j+q_2dy, z_{M_z})=(w_2)_y(x_i+q_1dx, y_j+q_2dy, \bar{\boldsymbol{t}}),$$
$$q_1, q_2=0,1,...,S\ ,\ i=0,1,...,M_x-1,\ j=0,1,\ldots,\ M_y-1. \tag{7.79}$$

Assembling all continuity conditions (7.50)-(7.79) yields to the following linear algebraic system

$$M\zeta=P \tag{7.80}$$

where M is an $\left(N\left[9M_xM_yM_z+M_xM_y+M_xM_z+M_yM_z\right](S+1)^2\right)\times\left(NM_xM_yM_z(K+1)^2\right)$ matrix, P is an $\left(N\left[9M_xM_yM_z+M_xM_y+M_xM_z+M_yM_z\right](S+1)^2\right)\times 1$ column vector and ζ is an $\left(NM_xM_yM_z(K+1)^2\right)\times 1$ column vector defined by

$$\zeta=\left[V_{000},\ V_{001},...,V_{(M_x-1)(M_y-1)(M_z-1)}\right]^T \tag{7.81}$$

where V_{ijr} is a $1\times\left(N(K+1)^2\right)$ vector defined in equation (7.49). If S is selected

as $(S+1)^2 > \frac{\left(NM_xM_yM_z(K+1)^2\right)}{9M_xM_yM_z+M_xM_y+M_xM_z+M_yM_z}$, equation (7.80) describes an over-determined linear system and minimizing the residual norm $\left\|M\zeta - P\right\|$ in least square sense will lead to unknown values V_{ijr}. Note that, $\theta_{x,}$ θ_y and θ_z are the direction parameters in x, y and z directions, respectively.

7.3 Numerical Experiments

In this section, the IELDTM derived in Sections 7.1-7.2 is tested over the linear/nonlinear (2+1)-dimensional AD problems and a linear (3+1)-dimensional AD problem. The convergence of the present algorithms is quantitatively demonstrated with the consideration of both h-refinement and p-refinement procedures. The produced numerical results are compared with the finite difference, finite element, and spectral methods presented in literature [48, 98, 108, 157]. To evaluate error norms of the present results, we prefer to use the following definitions,

$$E_{abs}^{ij} = \left|u_{ij}^{exact} - u_{ij}^{numerical}\right|,$$

$$\|E\|_\infty = \max_{(x,y)\in D,\ t\in\left[0,t_f\right]} \left|u^{exact}(x,y,t) - u^{numerical}(x,y,t)\right|.$$

Problem 7.1

Consider (2+1)-dimensional pure diffusion process with the choices of $V_x = V_y = 0$ and $D_x = D_y = 1$ in the AD equation (7.1) for which the exact solution is given by [108, 157]

$$u(\,x,y,t) = e^{-2\pi^2 t} sin(\pi x) sin(\pi y),\ t > 0 \text{ and } (x,y) \in [0,\ 1] \times [0,\ 1]. \tag{7.82}$$

The homogeneous Dirichlet boundary conditions and the initial condition are taken from the exact solution (7.47) as follows:

$$u(0,y,t) = u(1,y,t) = 0, \tag{7.83}$$

$$u(x,0,t) = u(x,1,t) = 0, \tag{7.84}$$

$$u(x,y,0) = sin(\pi x) sin(\pi y). \tag{7.85}$$

In Table 7.1, the present IELDTM-ChSCM algorithm has been compared with the finite difference-based methods [108, 157] with the consideration of maximum error

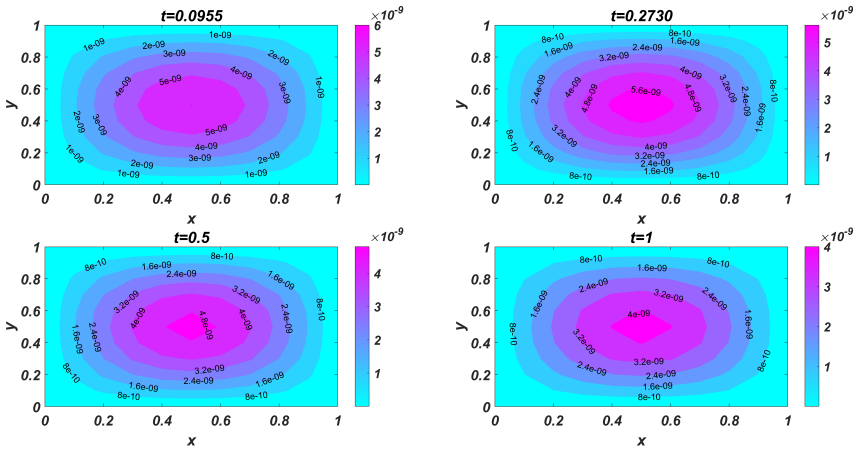

Figure 7.1 Descriptive behaviour of pointwise absolute errors produced by the central IELDTM at various times with $\theta_x = \theta_y = 0.5$, $M_x = M_y = 5$, $K = 10$, $N = 20$, $S = 14$ and $t_f = 1$ (see Problem 7.1).

norms. The spatial degrees of freedom (*dof*) of all numerical algorithms have been presented comparatively. As seen in Table 7.1, the IELDTM with forward, backward, and central directions produce more accurate results with far less spatial *dof* than finite difference-based methods presented in [108, 157]. In Figure 7.1, pointwise absolute errors produced by the central IELDTM are presented at various times with contour plots. As clearly observed from the figure, the IELDTM provides highly accurate results with optimized degrees of freedom. Even we consider only N = 20 elements in time, the ChSCM offers continuous time integration of the advection-diffusion equation up to $t = 1$ by maintaining roughly pointwise errors about 10^{-9}.

Table 7.1 Comparison of various present schemes with the FDM-based results produced in references [108, 157] at $t_f = 0.25$ with $N = 15$, $S = 14$ and $K = 10$ for Problem 7.1.

FDM			IELDTM			
$M_x \times M_y$ (*dof*)	[157]	[108]	$M_x \times M_y$ (*dof*)	Central	Backward	Forward
11×11 (121)	1.3E-05	5.6E-06	2×2 (84)	4.1E-08	6.9E-06	6.9E-06
21×21 (441)	8.5E-07	3.3E-07	3×3 (189)	1.6E-09	1.6E-06	1.6E-06
41×41 (1681)	5.3E-08	2.0E-08	4×4 (336)	8.7E-11	3.6E-08	3.6E-08
81×81 (6561)	3.3E-09	1.2E-09	5×5 (525)	9.9E-12	1.8E-09	1.8E-09

Problem 7.2

Consider the (2+1)-dimensional advection-diffusion process with $V_x = V_y = 1$ for which the exact solution is given by [108]

$$u(x,y,t) = \frac{1}{4t+1} exp\left(-\frac{(x-t-0.5)^2}{D_x(4t+1)} - \frac{(y-t-0.5)^2}{D_y(4t+1)}\right), \quad t > 0 \text{ and } (x,y) \in [0,\ x_f] \times [0,\ y_f]. \tag{7.86}$$

The nonhomogeneous time dependent Dirichlet boundary conditions can be taken from exact solution (7.86) and the initial condition is considered as

$$u(\,x,y,0) = exp\left(-\frac{(x-0.5)^2}{D_x} - \frac{(y-0.5)^2}{D_y}\right),\ (x,y) \in [0,\ x_f\,] \times [0,\ y_f\,]. \quad (7.87)$$

In Figures 7.2-7.3, spectral convergence of the IELDTM has been proven with respect to K-refinement (order refinement) and $\Delta x/\Delta y$—refinement for various values of the direction parameters θ_x and θ_y. As theoretically expected, K—refinement procedure leads to an exponential convergence and $\Delta x/\Delta y$—refinement procedure leads to a polynomial convergence like Δh^P when $\Delta x = \Delta y = \Delta h$. Since K—refinement procedure increases the local degrees of freedom (*ldof*) linearly as $ldof = 2K+1$, the IELDTM provides highly accurate results with optimized *dof*. The mesh discretization parameter S plays a vital role in both the accuracy and the computational cost of the IELDTM. In Figure 7.4a, the maximum error norms for the IELDTM have been presented with varying values of S. A rapid decrease in the corresponding error norm has been observed, and the curve behaves asymptotically for higher values of S. The mesh discretization ratio $\phi = Number\ of\ equations/Number\ of\ unknowns$ versus the maximum error norm curve has been illustrated with varying S values in Figure 7.4b. The curve indicates that the errors begin to be stable when the matrix that occurred in equation (7.34) is almost square ($\phi \cong 1$). Advection-dominated fluid flow problems are extremely challenging for all numerical methods [4]. With the consideration of various diffusion coefficients (D_x, D_y) in both spatial dimensions, the central IELDTM solutions and the corresponding pointwise absolute errors are illustrated in Figure 7.5. By considering the parameter values for $\theta_x = \theta_y = 0.5$, $x_f = y_f = 1$, $K = 4$, $M_x = M_y = 18$, $N = 5$, $S = 6$ and $t_f = 0.05$, it can be observed that the present algorithm produces accurate results for both non-stiff and stiff cases. The performance of the present IELDTM is illustrated over a relatively large space-time computational domain with $x_f = y_f = 5$ and $t_f = 2$ in Figure 7.6. As observed from the figure, the IELDTM accurately captures the physical behaviour with optimized degrees of freedom.

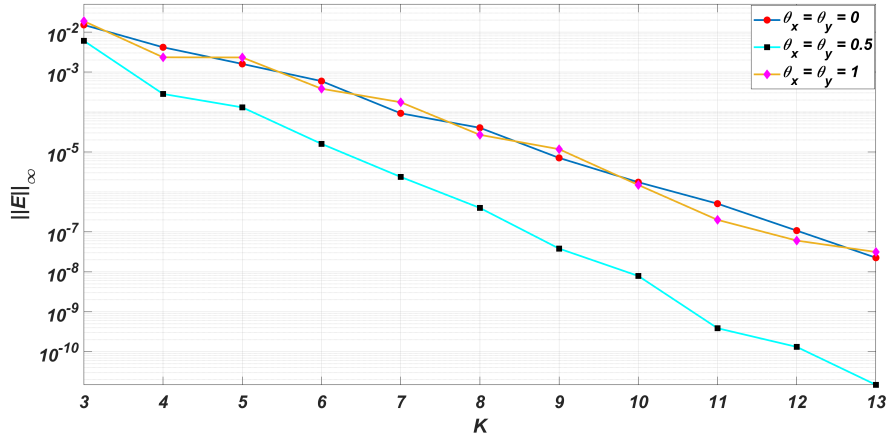

Figure 7.2 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter couples $(\theta_x,\ \theta_y)$ with $x_f = y_f = 1$, $M_x = M_y = 2$, $N = 10$, $S = 10$ and $t_f = 0.1$ (see Problem 7.2).

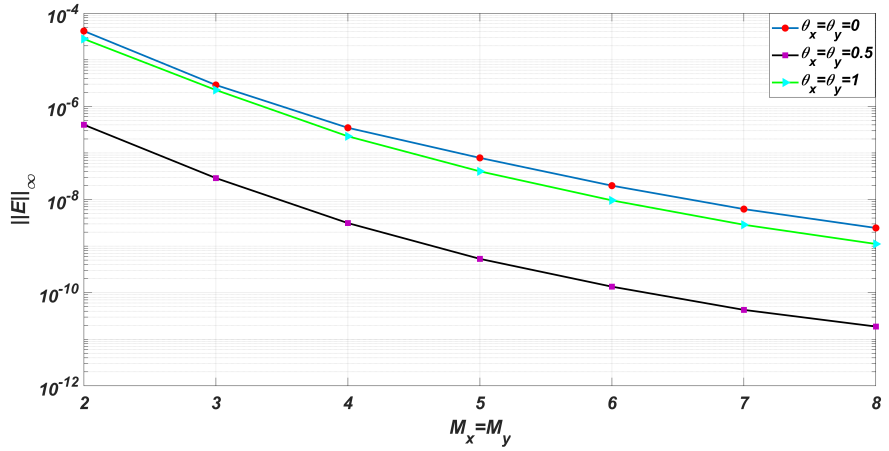

Figure 7.3 Convergence results of the present algorithm with respect to the spatial element numbers $M_x = M_y$ for various values of the direction parameter couples $(\theta_x,\ \theta_y)$ with $x_f = y_f = 1$, $K = 8$, $N = 10$, $S = 12$ and $t_f = 0.1$ (see Problem 7.2).

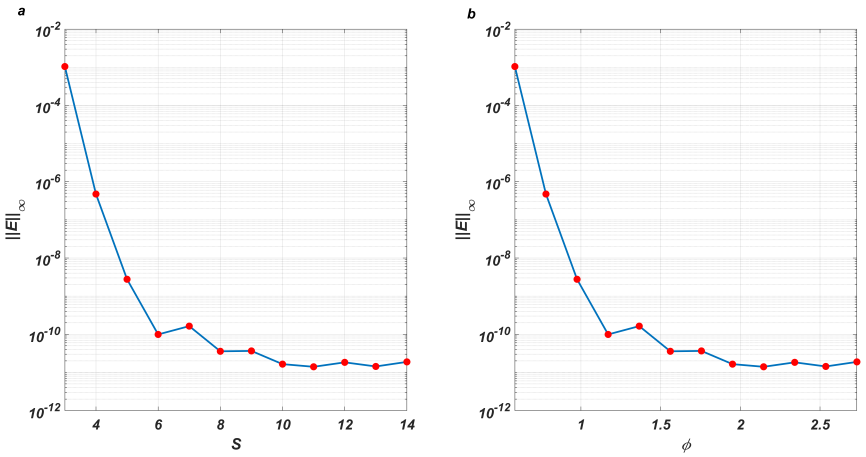

Figure 7.4 a) Effects of the mesh discretization parameter S to the maximum error norms $||E||_\infty$, b) Effects of the mesh discretization ratio ϕ to the maximum error norms $||E||_\infty$ with $x_f = y_f = 1$, $D_x = D_y = 1$, $M_x = M_y = 1$, $N = 10$, $K = 20$ and $t_f = 0.1$.

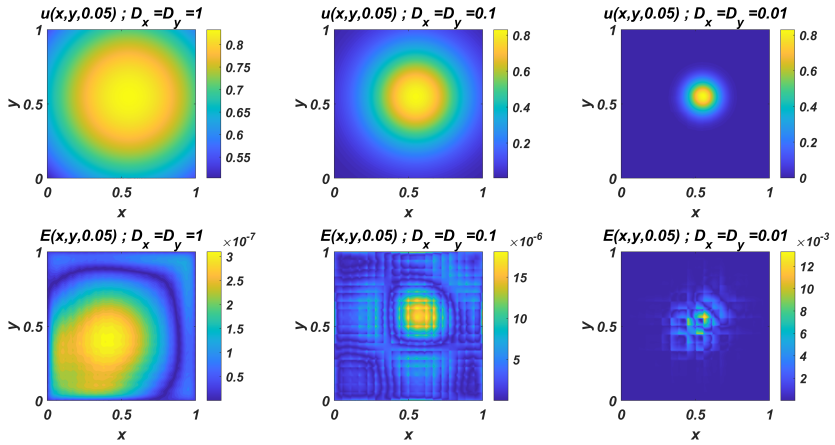

Figure 7.5 Numerical results and corresponding pointwise absolute errors for solving Problem 7.2 with the use of the central IELDTM for $\theta_x = \theta_y = 0.5$, $x_f = y_f = 1$, $K = 4$, $M_x = M_y = 18$, $N = 5$, $S = 6$ and $t_f = 0.05$.

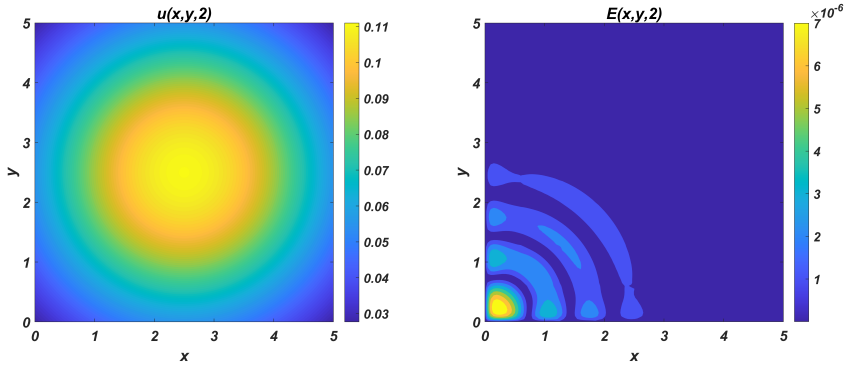

Figure 7.6 Numerical results and corresponding pointwise absolute errors for solving Problem 7.2 with the use of the central IELDTM for $\theta_x = \theta_y = 0.5$, $x_f = y_f = 5$, $K = 8$, $M_x = M_y = 8$, $N = 16$, $S = 5$ and $t_f = 2$.

Problem 7.3

Let us consider the (2+1)-dimensional Burgers equation [48, 98]

$$u_t + uu_x + uu_y = D(u_{xx} + u_{yy}), \quad (x, y) \in [0,\ 1] \times [0,\ 1], \quad t \in [0, t_f] \tag{7.88}$$

with the exact solution

$$u(x, y, t) = \frac{1}{1 + e^{(x+y-t)/2D}} \tag{7.89}$$

where D is the kinematic viscosity constant along the x and y directions. The initial and Dirichlet boundary conditions can be taken from the exact solution (7.89).

In Table 7.2, the present IELDTM with backward, central and forward directions has been compared with the FEM [98] and the Chebyshev spectral collocation method [48] with respect to the maximum error norms for solving (2+1)-dimensional nonlinear advection-diffusion equation (7.88). As seen in Table 7.2, the proposed method offers far more accurate results with the use of far less *dof* than both methods. The spectral convergence results for solving nonlinear equation (7.88) with respect to K-refinement and $\Delta x/\Delta y$—refinement procedures have been illustrated in Figures 7.7-7.8. The expected convergence behaviours are seen to be satisfied.

Problem 7.4

Consider the (3+1)-dimensional advection-diffusion equation with $V_x = V_y = V_z =$

Table 7.2 Comparison of various present schemes with the FEM and the ChSCM produced in [48, 98] at $t_f = 0.25$ with $N = 15,\ S = 16$ and $K = 10$ for Problem 7.3.

ChSCM/FEM			IELDTM			
$M_x \times M_y$ (dof)	ChSCM [48]	FEM [98]	$M_x \times M_y$ (dof)	Central	Backward	Forward
5×5 (25)	8.9E-08	4.6E-08	1×1 (21)	5.8E-10	5.8E-10	5.8E-10
10×10 (100)	7.4E-07	5.9E-09	2×2 (84)	1.1E-14	1.1E-12	6.4E-13
15×15 (225)	-	2.1E-09	3×3 (189)	5.5E-17	4.1E-14	4.0E-14
30×30 (900)	-	1.1E-09	5×5 (336)	1.6E-16	3.8E-15	4.2E-15

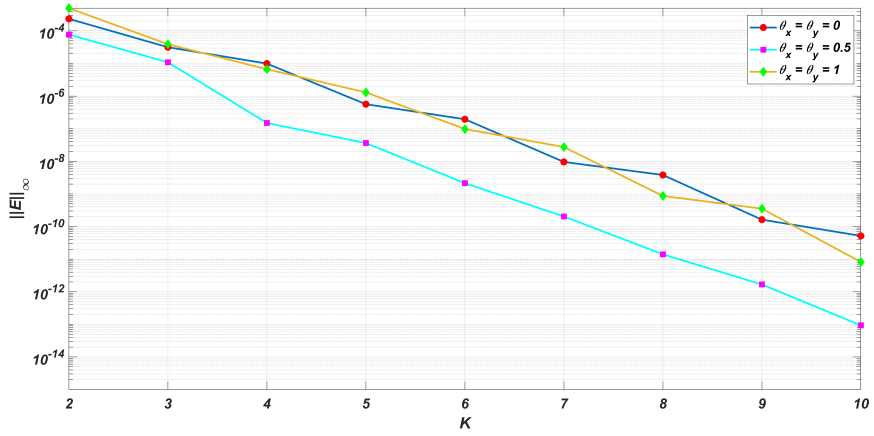

Figure 7.7 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter couples $(\theta_x,\ \theta_y)$ with $M_x = M_y = 2$, $N = 10$, $S = 10$ and $t_f = 0.5$ (see Problem 7.3).

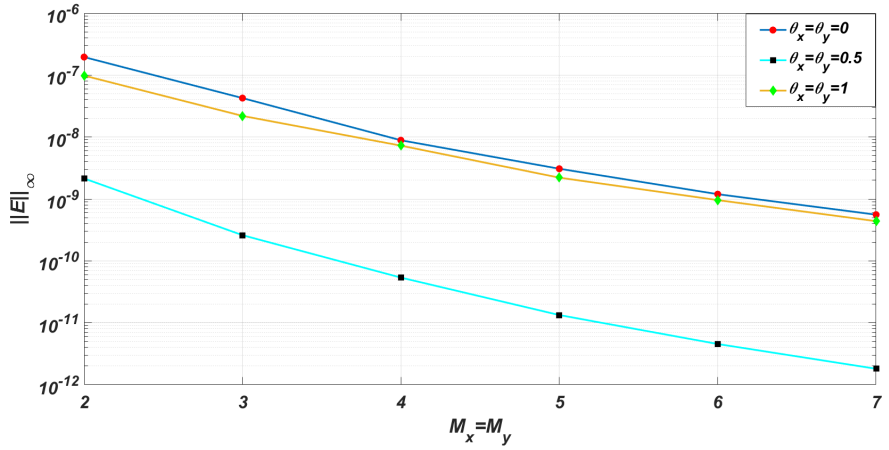

Figure 7.8 Convergence results of the present algorithm with respect to the spatial element numbers $M_x = M_y$ for various values of the direction parameter couples $(\theta_x,\ \theta_y)$, $K = 6$, $N = 10$, $S = 10$ and $t_f = 0.5$ (see Problem 7.3).

$D_x = D_y = D_z = 1$ for which the exact solution is given by [111]

$$u(x,y,z,t) = \frac{1}{(4t+1)^{3/2}} exp\Bigg(-\frac{(x-t-0.5)^2}{(4t+1)} - \frac{(y-t-0.5)^2}{(4t+1)} - \frac{(x-t-0.5)^2}{(4t+1)} \Bigg), \quad t > 0 \text{ and } (x,y,z) \in [0,1] \times [0,1] \times [0,1]. \tag{7.90}$$

The nonhomogeneous time-dependent Dirichlet boundary conditions can be taken from exact solution (7.90) and the initial condition is considered as

$$u(x,y,z,0) = exp\left(-(x-0.5)^2 - (y-0.5)^2 - (z-0.5)^2\right), \quad (x,y,z) \in [0,1] \times [0,1] \times [0,1]. \tag{7.91}$$

The order refinement procedure is more vital in three-dimensional equations to minimize the computational cost of the algorithms. It is proven that the forward, backward, and central directional IELDM provides exponential convergence with increasing order of the method in Figure 7.9. The parameter values $M_x = M_y = M_z = 2$, $N = 10$, $S = 4$ and $t_f = 0.1$ are used to produce the figure with increasing orders from $K = 5$ to $K = 10$. The h—refinement procedure is known to be another way of increasing accuracy with the use of more elements in computational domains. The h—refinement response of the IELDTM is illustrated in Figure 7.10 for various continuity directions and the parameter values $K = 6$, $N = 10$, $S = 4$ and $t_f = 0.5$. As observed from the figure, the IELDTM ensures power convergence with increasing values of spatial elements. The IELDTM solution of Problem 7.4 and the corresponding pointwise absolute errors are illustrated with the consideration of parameter values $K = 12$, $M_x = M_y = M_z = 1$, $N = 10$, $S = 4$ and $t_f = 0.25$ in Figure 7.11. As observed from the presented contour plots, the IELDTM accurately captures the physical behaviour using optimum degrees of freedom. As shown in Problem 7.2, the IELDTM has the ability to catch accurate physical behaviour of the advection-dominated problems in two dimensions. Since we perform the simulations on our personal computers, the advection-dominated three-dimensional cases lead to a storage trouble. It is believed that the IELDTM can overcome these challenging troubles with the optimized degrees of freedom with the computers that have relatively large RAM capacity.

7.4 Summary

In this chapter, the IELDTM has been derived for the (2+1)- and (3+1)-dimensional advection-diffusion equations. The parabolic PDEs have been reduced to a system of non-homogeneous elliptic PDEs with the Chebyshev spectral collocation method. The

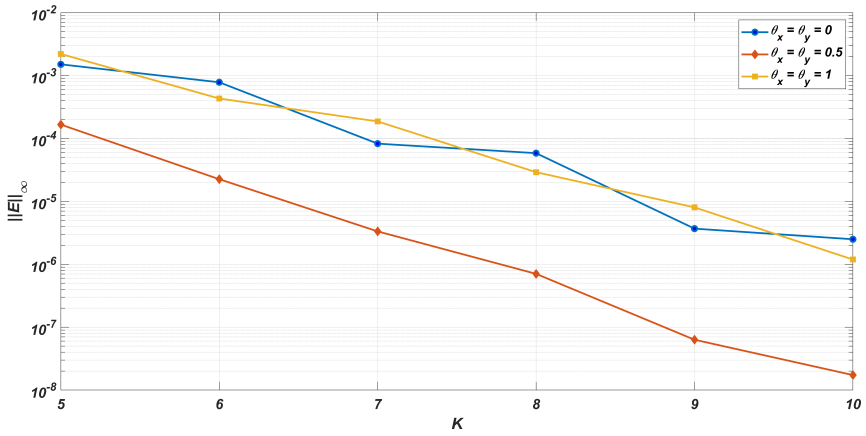

Figure 7.9 Convergence results of the present algorithm with respect to spectral convergence order K for various values of the direction parameter triples $(\theta_x,\ \theta_y,\ \theta_z)$ with $M_x = M_y = M_z = 2$, $N = 10$, $S = 4$ and $t_f = 0.1$ (see Problem 7.4).

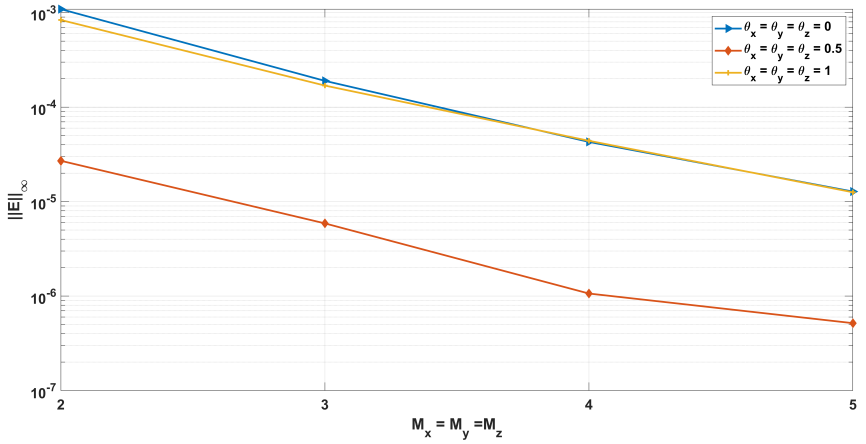

Figure 7.10 Convergence results of the present algorithm with respect to the spatial element numbers $M_x = M_y = M_z$ for various values of the direction parameter triples $(\theta_x,\ \theta_y,\ \theta_z)$ with $K = 6$, $N = 10$, $S = 4$ and $t_f = 0.5$ (see Problem 7.4).

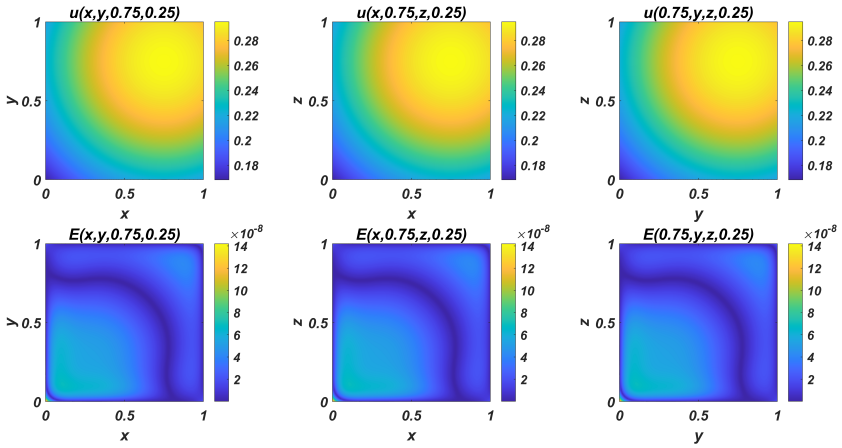

Figure 7.11 Numerical results and corresponding pointwise absolute errors of Problem 7.4 with the use of the IELDTM for $K = 12$, $M_x = M_y = M_z = 1$, $N = 10$, $S = 4$ and $t_f = 0.25$.

present IELDTM is observed to be a higher-order, stability-preserved, and versatile numerical technique for solving higher spatial dimensional AD equations. The experimental convergence analysis has revealed that the IELDTM provides excellent convergence properties with both h—refinement and p—refinement. The IELDTM leads to the reduction of the required degrees of freedom for achieving higher-order convergent numerical results. It is shown that the IELDTM needs only $2K + 1$ unknowns to represent a complete two-dimensional polynomial of order K while the complete polynomial includes $\frac{(K+1)(K+2)}{2}$ terms. Similarly, the IELDTM needs only $(K+1)^2$ unknowns to represent a complete three-dimensional polynomial of order K while the complete polynomial includes $\frac{(K+1)(K+2)(K+3)}{6}$ terms. By comparing with the existing finite difference methods, finite element methods, and spectral methods, the IELDTM has been proven to supply more accurate results with far less degrees of freedom than the three methods.

8
RESULTS AND DISCUSSION

Most real-life problems are represented by differential equations, and only a very limited number of these equations can be solved analytically. Approximate methods with discrete or continuous representations are constantly being developed to solve the corresponding differential equations. In this thesis, in order to realistically capture the behaviours of some natural and difficult processes represented by some stiff differential equations, a numerical method that is as efficient as possible in every respect has been produced by eliminating the well-known disadvantages in the literature. In this respect, an effective numerical approach called the implicit-explicit local differential transformation method (IELDTM), together with its mathematical and numerical analysis, has been presented to address the challenging behaviours of nature defined by stiff differential equations encountered in various fields of science. The IELDTM has been proven to be a high order, stability preserved, and versatile numerical approach for solving stiff IVPs for ODEs, stiff BVPs for ODEs, and stiff initial-boundary value problems for parabolic PDEs. The contributions of this thesis can be divided into four main parts: derivation of the IELDTM for stiff IVPs of ODEs, the IELDTM for stiff BVPs of ODEs, the IELDTM as an adaptive time integration method for parabolic PDEs, the IELDTM as a spatial discretization method for one-, two-, and three-dimensional advection-diffusion equations. Stiff IVPs of ODEs define how the system behaves over time for a given initial data and the situations encountered in emerging models in various fields such as physics, chemistry, biology, and engineering. The existing explicit-time integration methods, such as the LDTM and the Runge-Kutta methods, for solving stiff IVPs are known to lack stability-preserving features. The IELDTM overcomes the major disadvantages of the differential transform-based methods and has been shown to produce convergent solutions even for fairly stiff IVPs. The IELDTM is highly flexible for adaptive procedures because it uses the differential equation itself to generate highly accurate local approximations. It has been found that the IELDTM not only overcomes the disadvantages of the DTM-based methods but also provides better numerical characteristics than the well-established Taylor series based time-integration methods. The IELDTM has been shown to produce very well-suited

results with fewer time elements than that of the widely used MATLAB stiff solvers.

The stiff IVPs not only arise in direct applications but also in reduced space-time parabolic PDEs such as advection-diffusion equations. The explicit Runge-Kutta methods and the Crank-Nicolson method are commonly used time-integration techniques for the AD equations. The explicit Runge-Kutta methods are not stability preserved for especially advection dominated problems, and the Crank-Nicolson method is a second-order A-stable method. By providing an arbitrary order and direction-free mechanism, the IELDTM has been observed to improve the ability of the Crank-Nicolson approach. The IELDTM has been created as an adaptive time-integration method for the reduced two-dimensional nonlinear Burgers equation. The ChSCM has been used for the spatial parts of the two-dimensional Burgers equation and the resulting system of matrix IVPs has been integrated through the IELDTM. It has been seen that the ChSCM-IELDTM hybridization has been found to give more accurate results than the ChSCM-*ode15s* and ChSCM-*ode45* hybridizations. Thus, in addition, it has been shown that the IELDTM is a robust and efficient time integration method for reduced space-time PDEs and can be adapted to various physical problems facing the same difficulties.

Other striking problems that arise in various mathematical models are stiff or singularly perturbed BVPs of ODEs. Moreover, the IELDTM has been derived to solve stiff BVPs of ODEs with an arbitrary order and stability-preserved numerical approach. Here the DTM and LDTM have been shown to have poor convergence properties as a shooting method for singularly perturbed BVPs. Meanwhile, the IELDTM has proven to be able to produce accurate numerical solutions with optimum degrees of freedom for such difficult BVPs. With the use of the same approximation orders, the IELDTM has been observed to produce more accurate results than the FDM and the Runge-Kutta-based shooting methods. It has been shown that the p-refinement procedure of the IELDTM does not cause any increase in local degrees of freedom. This vital advantage in terms of computation and storage is strongly expected to make the IELDTM preferable over the FEM-based algorithms for solving BVPs.

Stiff initial-boundary value problems of PDEs arise in modelling various physical phenomena encountered in fields such as fluid dynamics, heat transfer, quantum mechanics, and materials science. Precisely in this context, another important contribution of this thesis is the derivation of the IELDTM for spatial discretizations of parabolic PDEs up to three-dimensional, representing advection-diffusion processes. The higher-order space-time IELDTM for solving AD equations has been proven to be hindered by a theoretical barrier. This limitation has been overcome by considering the ChSCM as a continuous time integrator in reducing space-time PDEs to a one-, two- or

three-dimensional system of nonhomogeneous BVPs to be solved by the IELDTM. Up to three-dimensional meshes, the IELDTM has been derived as an arbitrarily high-order, stable, and optimized numerical method for solving the reduced system of BVPs. It is important to note that the local degrees of freedom of the polynomial representations have been minimized with the consideration of the local differential transforms of the reduced AD equations. In this way, the IELDTM has been shown to produce more accurate results with far less degrees of freedom than the existing numerical methods such as the FEM, the FDM, and the spectral methods (see Chapters 6-7). It has been concluded that the IELDTM is capable of producing physically acceptable results for difficult advection-dominated problems with optimized degrees of freedom. The most noticeable property of a numerical method for solving PDEs is that having the effective *hp*-refinement property. The IELDTM has been proven to significantly improve both *h*-refinement and optimized *p*-refinement procedures for solving one-, two- or three-dimensional PDEs.

Even if the IELDTM has no noticeable disadvantage for solving ODEs, the IELDTM has no capable of reducing space-time PDEs into a system of ODEs to be integrated explicitly/implicitly. Thus, the IELDTM must be used as a last numerical algorithm when two numerical methods are applied for solving a space-time PDE. In this context, the IELDTM solves the reduced BVPs by applying a suitable time-integration method first instead of reducing the space-time initial-boundary value problems into a system of IVPs. This theoretical barrier seems to be the most considerable drawback of the IELDTM for solving space-time parabolic PDEs. As analysed throughout this thesis, it has been found that the IELDTM is a versatile numerical approach and can be adapted to solve a wide range of problems represented by differential equations. For further research in this area, the IELDTM can also be utilized successfully to solve:

* Integral equations and integro-differential equations,
* Elliptic PDEs occurred in the modelling of various engineering phenomena,
* Dispersive PDEs occurred in the modelling of various physical phenomena,
* Hyperbolic PDEs occurred in the computational fluid dynamics.

REFERENCES

[1] C. Brezinski, L. Wuytack, *Numerical analysis: Historical developments in the 20th century*. Elsevier, 2012.

[2] C. Curtiss, J. Hirschfelder, "Integration of stiff equations," *Proceedings of the National Academy of Sciences*, vol. 38, pp. 235–243, 1952.

[3] E. Hairer, S. Nørsett, G. Wanner, *Solving ordinary differential equations 1 - nonstiff problems*. Springer, 1993.

[4] W. Hundsdorfer, J. Verwer, *Numerical solution of time-dependent advection-diffusion-reaction equations*. Springer, 2003.

[5] E. Hairer, G. Wanner, *Solving ordinary differential equations 2- stiff and differential-algebraic equations*. Springer, 1996.

[6] B. Kleefeld, J. Vaqueron, "Serk2v3: Solving mildly stiff nonlinear partial differential equation," *Journal of Computational and Applied Mathematics*, vol. 299, pp. 94–206, 2016.

[7] S. Li, "Canonical euler splitting method for nonlinear composite stiff evolution equations," *Applied Mathematics and Computation*, vol. 289, pp. 220–236, 2016.

[8] M. Delfour, H. W., F. Trochu, "Discontinuous Galerkin methods for ordinary differential equations," *Mathematics of Computation*, vol. 36(154), pp. 455–473, 1981.

[9] E. Coutsias, T. Hagstrom, D. Torres, "An efficient spectral method for ordinary differential equations with rational function coefficients," *Mathematics of Computation*, vol. 65(214), pp. 611–635, 1996.

[10] L. Yi, Z. Wang, "Legendre-Gauss spectral collocation method for second order nonlinear delay differential equations," *Numerical Mathematics: Theory, Methods and Applications*, vol. 7(2), pp. 149–178, 2014.

[11] M. Saravi, E. Babolian, R. England, M. Bromilow, "System of linear ordinary differential and differential-algebraic equations and pseudo-spectral method," *Computers & Mathematics with Applications*, vol. 59, pp. 1524–1531, 2010.

[12] W. Tang, Y. Sun, W. Cai, "Discontinuous Galerkin methods for Hamiltonian ODEs and PDEs," *Journal of Computational Physics*, vol. 330, pp. 340–364, 2017.

[13] A. Fortin, D. Yakoubi, "An adaptive discontinuous galerkin method for very stiff systems of ordinary differential equations," *Journal of Computational and Applied Mathematics*, vol. 358, pp. 330–347, 2019.

[14] M. Narayanamurthi, Sandu, "Efficient implementation of partitioned stiff exponential Runge–Kutta methods," *Applied Numerical Mathematics*, vol. 152, pp. 141–158, 2020.

[15] B. Vermeire, "Paired explicit Runge–Kutta schemes for stiff systems of equations," *Journal of Computational Physics*, vol. 393, pp. 465–483, 2019.

[16] C. Kennedya, M. Carpenter, "Diagonally implicit Runge–Kutta methods for stiff odes," *Applied Numerical Mathematics*, vol. 146, pp. 221–244, 2019.

[17] V. Luan, A. Ostermann, "Parallel exponential Rosenbrock methods," *Computers & Mathematics with Applications*, vol. 71(5), pp. 1137–1150, 2016.

[18] D. Voss, "Fourth-order parallel Rosenbrock formulae for stiff systems," *Mathematical and Computer Modelling*, vol. 40(11-12), pp. 1193–1198, 2004.

[19] M. Caliaria, A. Ostermann, "Implementation of exponential Rosenbrock-type integrators," *Applied Numerical Mathematics*, vol. 59(3-4), pp. 568–581, 2009.

[20] J. Ernsthausen, N. Nedialkov, "Stepsize selection in the rigorous defect control of Taylor series methods," *Journal of Computational and Applied Mathematics*, vol. 368, pp. 112 483–, 2020.

[21] A. Abad, R. Barrio, M. Marco-Buzunariz, M. Rodríguez, "Automatic implementation of the numerical taylor series method: A MATHEMATICA and SAGE approach," *Applied Mathematics and Computation*, vol. 268, pp. 227–245, 2015.

[22] S. Abbasbandy, C. Bervillier, "Analytic continuation of Taylor series and the boundary value problems of some nonlinear ordinary differential equations," *Applied Mathematics and Computation*, vol. 218(5), pp. 2178–2199, 2011.

[23] J. Scott, "Solving ODE initial value problems with implicit Taylor series methods," *NASA/TM—2000-209400, Glenn Research Center, Cleveland, Ohio, USA*, 2000.

[24] Z. Odibat, C. Bertelle, M. Aziz-Alaoui, G. Duchamp, "A multi-step differential transform method and application to non-chaotic or chaotic systems," *Computers & Mathematics with Applications*, vol. 59(4), pp. 1462–1472, 2010.

[25] E. El-Zahar, "An adaptive step-size Taylor series based method and application to nonlinear biochemical reaction model," *Trends in Applied Sciences Research*, vol. 7(11), pp. 901–912, 2012.

[26] M. Jang, C. Chen, Y. Liy, "On solving the initial-value problems using the differential transformation method," *Applied Mathematics Computation*, vol. 115(2-3), pp. 145–160, 2000.

[27] H. Tunc, M. Sari, "A local differential transform approach for the cubic nonlinear Duffing oscillator with damping term," *Scientia Iranica*, vol. 26(2), pp. 879–886, 2019.

[28] G. Kulikov, R. Weiner, "Doubly quasi-consistent fixed-stepsize numerical integration of stiff ordinary differential equations with implicit two-step peer methods," *Journal of Computational and Applied Mathematics*, vol. 340, pp. 256–275, 2018.

[29] Z. Tan, C. Zhang, "Implicit-explicit one-leg methods for nonlinear stiff neutral equations," *Applied Mathematics and Computation*, vol. 335, pp. 196–210, 2018.

[30] N. Leterrier, "ARES: An efficient approach to adaptive time integration for stiff differential algebraic equations," *Computers & Chemical Engineering*, vol. 119, pp. 46–54, 2018.

[31] A. Abdi, S. Hosseini, H. Podhaisky, "ARES: Adaptive linear barycentric rational finite differences method for stiff ODEs," *Journal of Computational and Applied Mathematics*, vol. 357, pp. 204–214, 2019.

[32] B. Faleichik, "Minimal residual multistep methods for large stiff non-autonomous linear problems," *Journal of Computational and Applied Mathematics*, vol. 387, p. 112 498, 2021.

[33] J. Gu, J. Jung, "Adaptive radial basis function methods for initial value problems," *Journal of Scientific Computing*, vol. 82, p. 47, 2020.

[34] I. Higueras, T. Roldán, "Strong stability preserving properties of composition Runge–Kutta schemes," *Journal of Scientific Computing*, vol. 80, pp. 784–807, 2019.

[35] L. Isherwood, Z. Grant, S. Gottlieb, "Strong stability preserving integrating factor two-step Runge–Kutta methods," *Journal of Scientific Computing*, vol. 81, pp. 1446–1471, 2019.

[36] Z. Ibrahim, K. Othman, M. Suleiman, "Implicit r-point block backward differentiation formula for solving first-order stiff ODEs," *Applied Mathematics and Computation*, vol. 186, pp. 558–565, 2007.

[37] J. Cash, "Modifed extended backward differentiation formulae for the numerical solution of stiff initial value problems in ODEs and DAEs," *Journal of Computational and Applied Mathematics*, vol. 125, pp. 117–130, 2000.

[38] O. Akinfenwa, S. Jator, N. Yao, "Continuous block backward differentiation formula for solving stiff ordinary differential equations," *Computers & Mathematics with Applications*, vol. 65, pp. 996–1005, 2013.

[39] S. Pinto, D. Abreu, B. Simeon, "Strongly A-stable first stage explicit collocation methods with stepsize control for stiff and differential–algebraic equations," *Journal of Computational and Applied Mathematics*, vol. 259, pp. 138–152, 2014.

[40] X. Piao, S. Bu, D. Kim, P. Kim, "An embedded formula of the Chebyshev collocation method for stiff problems," *Journal of Computational Physics*, vol. 351, pp. 376–391, 2017.

[41] B. Wang, X. Wu, "Exponential collocation methods for conservative or dissipative systems," *Journal of Computational and Applied Mathematics*, vol. 360, pp. 99–116, 2019.

[42] M. Sari, H. Tunc, M. Seydaoglu, "Higher order splitting approaches in analysis of the Burgers equation," *Kuwait Journal of Science*, vol. 46(1), pp. 1–14, 2019.

[43] H. Tunc, M. Sari, "Simulations of nonlinear advection-diffusion models through various finite element techniques," *Scientia Iranica*, vol. 27(6), pp. 2853–2870, 2020.

[44] G. Zhao, X. Yu, R. Zhang, “The new numerical method for solving the system of two-dimensional Burgers’ equations,” *Computers & Mathematics with Applications*, vol. 62(8), pp. 3279–3291, 2011.

[45] G. Cooper, “The numerical solution of stiff differential equations,” *FEBS Letters*, vol. 2(1), S22–S29, 1969.

[46] M. Sari, G. Gurarslan, “A sixth-order compact finite difference scheme to the numerical solutions of Burgers’ equation,” *Applied Mathematics and Computation*, vol. 208, pp. 475–483, 2009.

[47] L. Shao, X. Feng, Y. He, “The local discontinuous Galerkin finite element method for Burgers equation,” *Mathematical and Computer Modelling*, vol. 154, pp. 2943–2954, 2011.

[48] A. Khater, R. Temsah, M. Hassan, “A Chebyshev spectral collocation method for solving Burgers’-type equations,” *Journal of Computational and Applied Mathematics*, vol. 222, pp. 333–350, 2008.

[49] M. Tamsir, V. Srivastava, R. Jiwari, “An algorithm based on exponential modified cubic B-spline differential quadrature method for nonlinear Burgers’ equation,” *Applied Mathematics and Computation*, vol. 290, pp. 111–124, 2016.

[50] B. Saka, I. Dag, “Quartic B-spline collocation method to the numerical solutions of the Burgers’ equation,” *Chaos, Solitons & Fractals*, vol. 32, pp. 1125–1137, 2007.

[51] R. Rodríguez, Y. Skiba, D. Filatov, “An implicit direct unconditionally stable numerical algorithm for the solution of advection-diffusion equation on a sphere,” *Applied Numerical Mathematics*, vol. 142, pp. 1–15, 2019.

[52] M. Sari, H. Tunc, “An optimization technique in analyzing the Burgers equation,” *Sigma Journal of Engineering and Natural Sciences*, vol. 35(3), pp. 369–386, 2017.

[53] T. Nazir, M. Abbas, A. Ismail, A. Majid, A. Rashid, “The numerical solution of advection-diffusion problems using new cubic trigonometric B-splines approach,” *Applied Mathematical Modelling*, vol. 40(7-8), pp. 4586–4611, 2016.

[54] S. Gowrisankar, S. Natesan, “An efficient robust numerical method for singularly perturbed Burgers’ equation,” *Applied Mathematics and Computation*, vol. 346, pp. 385–394, 2019.

[55] N. Egidi, P. Maponi, P. Quadrini, “An integral equation method for the numerical solution of the Burgers equation,” *Computers & Mathematics with Applications*, vol. 76(1), pp. 35–44, 2018.

[56] K. Rektorys, *Solving Ordinary and Partial Boundary Value Problems in Science and Engineering*. CRC Press, 1998.

[57] J. Lee, L. Greengard, “A fast adaptive numerical method for stiff two-point boundary value problems,” *SIAM Journal on Scientific Computing*, vol. 18(2), pp. 403–429, 1997.

[58] M. Elgindi, R. Langer, "On the shooting method for a class of two-point singular nonlinear boundary value problems," *International Journal of Computer Mathematics*, vol. 51(1-2), pp. 107–118, 1994.

[59] M. Chawla, R. Subramanian, H. Sathi, "A fourth order method for a singular two-point boundary value problem," *BIT Numerical Mathematics*, vol. 28, pp. 88–97, 1988.

[60] U. Ascher, R. Mattheij, R. Russell, *Numerical solution of boundary value problems for ordinary differential equations*. Society for Industrial and Applied Mathematics, 1995.

[61] M. Ahsan, S. Farrukh, "A new type of shooting method for nonlinear boundary value problems," *Alexandria Engineering Journal*, vol. 52(4), pp. 801–805, 2013.

[62] W. Auzinger, J. Burkotová, I. Rachůnková, W. V., "Shooting methods for state-dependent impulsive boundary value problems," *Applied Numerical Mathematics*, vol. 128, pp. 217–229, 2018.

[63] V. Makarov, D. Dragunov, "An efficient approach for solving stiff nonlinear boundary value problems," *Journal of Computational and Applied Mathematics*, vol. 345, pp. 452–470, 2019.

[64] R. Agarwal, C. Y.M., "Finite difference methods for boundary-value problems of differential equations with deviating arguments," *Computers & Mathematics with Applications*, vol. 12A(11), pp. 1143–1153, 1986.

[65] M. Kadalbajoo, K. Sharma, "A numerical method based on finite difference for boundary value problems for singularly perturbed delay differential equations," *Applied Mathematics and Computation*, vol. 197, pp. 692–707, 2008.

[66] D. Herceg, "Fourth-order finite-difference method for boundary value problems with two small parameters," *Applied Mathematics and Computation*, vol. 218, pp. 616–627, 2011.

[67] Y. Yirang, "The upwind finite difference method for moving boundary value problem of coupled system," *Acta Mathematica Scientia*, vol. 31(3), pp. 857–881, 2011.

[68] M. Król, A. Kunynets, K. M.V., "Exact three-point difference scheme for singular nonlinear boundary value problems," *Journal of Computational and Applied Mathematics*, vol. 298, pp. 175–189, 2016.

[69] P. Roul, V. Prasad Goura, R. Agarwal, "A compact finite difference method for a general class of nonlinear singular boundary value problems with Neumann and Robin boundary conditions," *Applied Mathematics and Computation*, vol. 350, pp. 283–304, 2019.

[70] E. Ideon, P. Oja, "Linear/linear rational spline collocation for linear boundary value problems," *Journal of Computational and Applied Mathematics*, vol. 263, pp. 32–44, 2014.

[71] F. Costabile, A. Napoli, "A class of Birkhoff–Lagrange-collocation methods for high order boundary value problems," *Applied Numerical Mathematics*, vol. 263, pp. 129–140, 2017.

[72] E. Ideon, P. Oja, "Quadratic/linear rational spline collocation for linear boundary value problems," *Applied Numerical Mathematics*, vol. 125, pp. 143–158, 2018.

[73] K. Ali, A. Hadhoud, M. Shaalan, "Numerical study of self-adjoint singularly perturbed two-point boundary value problems using collocation method with error estimation," *Journal of Ocean Engineering and Science*, vol. 3, pp. 237–243, 2018.

[74] P. Roul, K. Thula, "A new high-order numerical method for solving singular two-point boundary value problems," *Journal of Computational and Applied Mathematics*, vol. 343, pp. 556–574, 2018.

[75] M. Sohaib, S. Haq, S. Mukhtar, I. Khan, "Numerical solution of sixth-order boundary-value problems using Legendre wavelet collocation method," *Results in Physics*, vol. 8, pp. 1204–1208, 2018.

[76] M. Baccouch, "Analysis of optimal superconvergence of a local discontinuous Galerkin method for nonlinear second-order two-point boundary-value problems," *Applied Numerical Mathematics*, vol. 145, pp. 361–383, 2019.

[77] M. Sahlan, "Four computational approaches for solving a class of boundary value problems arising in chemical reactor industry," *Applied Mathematics and Computation*, vol. 355, pp. 253–268, 2019.

[78] R. Lodhi, H. Mishra, "Quintic b-spline method for solving second order linear and nonlinear singularly perturbed two-point boundary value problems," *Journal of Computational and Applied Mathematics*, vol. 319, pp. 170–187, 2017.

[79] A. Khan, Shahna, "Non-polynomial quadratic spline method for solving fourth order singularly perturbed boundary value problems," *Journal of King Saud University*, vol. 31(4), pp. 479–484, 2019.

[80] I. EL-Kalla, A. El Mhlawy, M. Botros, "A continuous solution of solving a class of nonlinear two point boundary value problem using Adomian decomposition method," *Ain Shams Engineering Journal*, vol. 10, pp. 211–216, 2019.

[81] G. Akram, I. Aslam, "Solution of fourth order three-point boundary value problem using ADM and RKM," *Journal of the Association of Arab Universities for Basic and Applied Sciences*, vol. 20, pp. 61–67, 2016.

[82] F. Geng, S. Qian, "An optimal reproducing kernel method for linear nonlocal boundary value problems," *Applied Mathematics Letters*, vol. 77, pp. 49–56, 2018.

[83] L. Yang, Y. Lin, "Reproducing kernel methods for solving linear initial-boundary-value problems," *Electronic Journal of Differential Equations*, vol. 29, pp. 1–11, 2008.

[84] A. Ravi Kanth, K. Aruna, "Solution of singular two-point boundary value problems using differential transformation method," *Physics Letters A*, vol. 372, pp. 4671–4673, 2008.

[85] S. Islam, S. Haq, J. Ali, "Numerical solution of special 12th-order boundary value problems using differential transform method," *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, pp. 1132–1138, 2009.

[86] V. Erturk, S. Momani, "Comparing numerical methods for solving fourth-order boundary value problems," *Applied Mathematics and Computation*, vol. 188, pp. 963–968, 2007.

[87] E. El-Zahar, "Approximate analytical solutions of singularly perturbed fourth order boundary value problems using differential transform method," *Journal of King Saud University*, vol. 25, pp. 257–265, 2013.

[88] A. Opanuga, H. Okagbue, S. Edeki, O. Agboola, "Differential transform technique for higher order boundary value problems," *Modern Applied Science*, vol. 9(13), pp. 224–230, 2013.

[89] L. Xie, C. Zhou, S. Xu, "An effective numerical method to solve a class of nonlinear singular boundary value problems using improved differential transform method," *Springerplus*, vol. 5(1), pp. 1066–, 2016.

[90] M. Rashidi, G. Domairry, "New analytical solution of the three-dimensional Navier-Stokes equations," *Modern Physics Letters B*, vol. 23, pp. 3147–3155, 2009.

[91] Y. Lin, H. Tang, C. Chen, "Modified differential transform method for two singular boundary values problems," *Journal of Applied Mathematics*, vol. 2014, pp. -, 2014.

[92] S. Mosayebidorcheh, "Solution of the boundary layer equation of the power-law pseudoplastic fluid using differential transform method," *Mathematical Problems in Engineering*, vol. 2013, pp. -, 2013.

[93] P. Huyakorn, "Solution of transient transport equation using an upstream finite element scheme," *Applied Mathematical Modelling*, vol. 3, pp. 7–17, 1978.

[94] T. Hughes, L. Franca, G. Hulbert, "A new finite element formulation for computational fluid dynamics: VIII. the Galerkin/least-squares method for advective-diffusive equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 73, pp. 173–189, 1989.

[95] S. Badia, J. Bonilla, "Monotonicity-preserving finite element schemes based on differentiable nonlinear stabilization," *Computer Methods in Applied Mechanics and Engineering*, vol. 313, pp. 133–158, 2017.

[96] C. Lohmann, D. Kuzmin, J. Shadid, S. Mabuza, "Flux-corrected transport algorithms for continuous Galerkin methods based on high order Bernstein finite elements," *Journal of Computational Physics*, vol. 344, pp. 151–186, 2017.

[97] D. Fenga, I. Neuweiler, U. Nackenhorst, T. Wick, "A time-space flux-corrected transport finite element formulation for solving multi-dimensional advection-diffusion-reaction equations," *Journal of Computational Physics*, vol. 396, pp. 31–53, 2019.

[98] R. Mittal, A. Tripathi, "Numerical solutions of two-dimensional Burgers' equations using modified bi-cubic B-spline finite elements," *Engineering Computations*, vol. 32(5), pp. 1275–1306, 2015.

[99] R. Mittal, S. Kumar, R. Jiwari, "A cubic B-spline quasi interpolation method for solving two-dimensional unsteady advection diffusion equations," *International Journal of Numerical Methods for Heat & Fluid Flow*, vol. 30(9), pp. 4281–4306, 2020.

[100] E. Tombarević, V. R. Voller, I. Vušanović, "Detailed CVFEM algorithm for three dimensional advection-diffusion problems," *CMES-Computer Modeling in Engineering & Sciences*, vol. 96(1), pp. 1–29, 2013.

[101] T. Chen, C. Shu, "Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws," *Journal of Computational Physics*, vol. 345, pp. 427–461, 2017.

[102] H. Dong, M. Lv, M. Li, "A reconstructed central discontinuous Galerkin method for conservation laws," *Computers and Fluids*, vol. 153, pp. 76–84, 2017.

[103] M. Uzunca, B. Karasozen, M. Manguoglu, "Adaptive discontinuous Galerkin methods for non-linear diffusion–convection–reaction equations," *Computers and Chemical Engineering*, vol. 68, pp. 24–37, 2014.

[104] R. Anderson, V. Dobrev, T. Kolev, D. Kuzmin, M. Quezada de Luna, R. Rieben, V. Tomov, "High-order local maximum principle preserving (MPP) discontinuous Galerkin finite element method for the transport equation," *Journal of Computational Physics*, vol. 334, pp. 102–124, 2017.

[105] J. Eshaghi, S. Kazem, H. Adibi, "The local discontinuous Galerkin method for 2D nonlinear time-fractional advection–diffusion equations," *Engineering with Computers*, vol. 35, pp. 1317–1332, 2019.

[106] L. Diosadya, S. Murman, "Scalable tensor-product preconditioners for high-order finite-element methods: Scalar equations," *Journal of Computational Physics*, vol. 394, pp. 759–776, 2019.

[107] M. Sari, G. Gurarslan, A. Zeytinoglu, "High-order finite difference schemes for solving the advection-diffusion equation," *Mathematical and Computational Applications*, vol. 15, pp. 449–460, 2010.

[108] N. Mishra, Y. Sanyasiraju, "Exponential compact higher-order schemes and their stability analysis for unsteady convection-diffusion equations," *International Journal of Computational Methods*, vol. 11(1), p. 1 350 053, 2014.

[109] J. Ramos, "A conservative method of lines for advection-reaction-diffusion equations," *International Journal of Numerical Methods for Heat & Fluid Flow*, vol. 30(11), pp. 4735–4763, 2020.

[110] S. Singh, D. Bansal, G. Kaur, S. Sircar, "Implicit-explicit-compact methods for advection diffusion reaction equations," *Computers and Fluids*, vol. 212, p. 104 709, 2020.

[111] M. Dehghan, "Numerical solution of the three-dimensional advection–diffusion equation," *Applied Mathematics and Computation*, vol. 150, pp. 5–19, 2004.

[112] G. Manzini, A. Russo, "A finite volume method for advection–diffusion problems in convection-dominated regimes," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, pp. 1242–1261, 2008.

[113] A. Hussain, Z. Zheng, E. Anley, "The finite volume-complete flux scheme for advection-diffusion-reaction equations," *Mathematics*, vol. 8, p. 1869, 2020.

[114] J. ten Thije Boonkkamp, "The finite volume-complete flux scheme for advection-diffusion-reaction equations," *Journal of Scientific Computing*, vol. 46, pp. 47–70, 2011.

[115] J. Schäfer, X. Huang, S. Kopecz, P. Birken, M. Gobbert, M. A.A., "A memory-efficient finite volume method for advection-diffusion-reaction systems with non-smooth sources," *Numerical Methods for Partial Differential Equations*, vol. 31(1), pp. 143–167, 2015.

[116] S. May, M. Berger, "Two-dimensional slope limiters for finite volume schemes on non-coordinate-aligned meshes," *SIAM Journal on Scientific Computing*, vol. 35(5), A2163–A2187, 2013.

[117] R. Abazari, A. Borhanifar, "Numerical study of the solution of the Burgers and coupled Burgers equations by a differential transformation method," *Computers & Mathematics with Applications*, vol. 59(8), pp. 2711–2722, 2010.

[118] A. Karamanli, A. Mugan, "Strong form meshless implementation of Taylor series method," *Applied Mathematics and Computation*, vol. 219, pp. 9069–9080, 2013.

[119] A. Karamanli, "Radial basis Taylor series method and its applications," *Engineering Computations*, vol. ahead-of-print, ahead-of–print, 2021.

[120] F. Yan, J. Lv, F. X.T., P. Pan, "A new hybrid boundary node method based on Taylor expansion and the Shepard interpolation method," *International Journal for Numerical Methods in Engineering*, vol. 102, pp. 1488–1506, 2015.

[121] J. Yang, H. Hu, Y. Koutsawa, M. Potier-Ferry, "Taylor meshless method for solving non-linear partial differential equations," *Journal of Computational Physics*, vol. 348, pp. 385–400, 2017.

[122] J. Yanga, H. Hu, M. Potier-Ferry, "Computing singular solutions to partial differential equations by Taylor series," *Comptes Rendus Mecanique*, vol. 346, pp. 603–614, 2018.

[123] J. Yang, Y. Koutsawa, M. Potier-Ferry, H. Hu, "Changing variables in Taylor series with applications to PDEs," *Engineering Analysis with Boundary Elements*, vol. 112, pp. 77–86, 2020.

[124] D. Jackson, "A proof of Weierstrass's theorem," *The American Mathematical Monthly*, vol. 41(5), pp. 309–312, 1934.

[125] J. Epperson, *An Introduction to Numerical Methods and Analysis*. Wiley, 2013.

[126] J. Mathews, K. Fink, *Numerical Methods Using MATLAB*. Pearson, 2003.

[127] J. Hesthaven, T. Warburton, *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis and Applications*. Springer, 2008.

[128] W. Gautschi, *Numerical Analysis*. Birkhäuser Basel, 2012.

[129] C. Canuto, M. Hussaini, A. Quarteroni, T. Zang, *Spectral Methods: Fundamentals in Single Domains*. Birkhäuser Basel, 2006.

[130] T. Meng, Z. Wang, L. Yi, "An h–p version of the Chebyshev spectral collocation method for nonlinear delay differential equations," *Numerical Methods for Partial Differential Equations*, vol. 35, pp. 664–680, 2019.

[131] S. Krantz, H. Parks, *A Primer of Real Analytic Functions*. Birkhäuser Basel, 2002.

[132] Z. Šikić, "Taylor's theorem," *International Journal of Mathematical Education in Science and Technology*, vol. 21(1), pp. 111–115, 1990.

[133] J. Lambert, *Numerical Methods for Ordinary Differential Systems*. Wiley, 1992.

[134] A. Hidalgo, M. Dumbser, "ADER schemes for nonlinear systems of stiff advection–diffusion–reaction equations," *Journal of Scientific Computing*, vol. 48, pp. 173–189, 2011.

[135] H. Tunc, "Various finite element techniques for advection-diffusion-reaction processes," M.S. thesis, Yildiz Technical University, 2017.

[136] N. Imamoglu, "Two numerical approaches for solving nonlinear stiff differential equations," M.S. thesis, Izmir Institute of Technology, 2014.

[137] I. Abdel-Halim Hassan, "Application to differential transformation method for solving systems of differential equations," *Applied Mathematical Modelling*, vol. 32, pp. 2552–2559, 2008.

[138] C. Baishya, "Solution of nonlinear singular initial value problem by differential transform method powered by Adomian polynomial," *Global Journal of Pure and Applied Mathematics*, vol. 13(2), pp. 617–625, 2017.

[139] H. Tunc, M. Sari, "An efficient local transform method for initial value problems," *Sigma Journal of Engineering and Natural Sciences*, vol. 39(1), pp. 163–174, 2019.

[140] I. Abdel-Halim Hassan, "Differential transformation technique for solving higher-order initial value problems," *Applied Mathematics and Computation*, vol. 154(2), pp. 299–311, 2004.

[141] B. Liu, X. Zhou, Q. Du, "Differential transform method for some delay differential equations," *Applied Mathematics*, vol. 6(3), pp. 585–593, 2015.

[142] L. Shampine, M. Reichelt, "The MATLAB ODE suite," *SIAM Journal on Scientific Computing*, vol. 18, pp. 1–22, 1997.

[143] J. Nocedal, S. Wright, *Numerical Optimization*. Springer, 2006.

[144] R. Li, S. Pei, B. Chen, Y. Song, T. Zhang, W. Yang, J. Shaman, "Substantial undocumented infection facilitates the rapid dissemination of novel coronavirus (SARS-COV-2)," *Science*, vol. 368, pp. 489–493, 2020.

[145] I. Abdel-Halim Hassan, "On solving some eigenvalue problems by using a differential transformation," *Applied Mathematics and Computation*, vol. 127(1), pp. 1–22, 2002.

[146] H. Bateman, "Some recent researches on the motion of fluids," *Monthly Weather Review*, vol. 43, pp. 163–170, 1915.

[147] J. Burgers, "A mathematical model illustrating the theory of turbulence," *Advances in Applied Mathematics*, vol. 1, pp. 171–199, 1948.

[148] E. Hopf, "The partial differential equation $ut+uux=\epsilon uxx$," *Communications on Pure and Applied Mathematics*, vol. 9, pp. 201–230, 1950.

[149] J. Cole, "On a quasi-linear parabolic equation in aerodynamics," *Advances in Applied Mathematics*, vol. 9, pp. 225–236, 1951.

[150] J. Wu, X. Zhang, "Integro quadratic spline interpolation," *Applied Mathematical Modelling*, vol. 39, pp. 2973–2980, 2015.

[151] M. Sari, H. Tunc, “Finite element-based hybrid techniques for advection-diffusion-reaction processes,” *An International Journal of Optimization and Control: Theories & Applications*, vol. 8(2), pp. 127–136, 2018.

[152] I. Dag, D. Irk, M. Tombul, “Least-squares finite element method for the advection-diffusion equation,” *Applied Mathematics and Computation*, vol. 173, 554:565, 2006.

[153] V. Mukundan, A. Awasthi, “Efficient numerical techniques for Burgers’ equation,” *Applied Mathematics and Computation*, vol. 262, pp. 282–297, 2015.

[154] V. Sangwan, B. Kaur, “An exponentially fitted numerical technique for singularly perturbed Burgers-Fisher equation on a layer adapted mesh,” *International Journal of Computer Mathematics*, vol. 96(7), pp. 1502–1513, 2019.

[155] T. Bennett, *Transport by Advection and Diffusion*. Wiley, 2012.

[156] E. Miller, “Predictor-corrector studies of Burger’s model of turbulent flow,” M.S. thesis, University of Delaware, 1966.

[157] Z. Tian, Y. Ge, “A fourth-order compact ADI method for solving two dimensional unsteady convection diffusion problems,” *Journal of Computational and Applied Mathematics*, vol. 198(1), pp. 168–286, 2007.

A
BASICS OF THE DIFFERENTIAL TRANSFORMATION

In this section, the required definitions and theorems of the differential transformation are introduced. The proposed properties of the differential transformation are frequently used in dealing with the differential transform of any given differential equation.

Definition A.1. Let $x(t)$ is analytic in the domain T. The function $\varphi(t,k)$ can be defined as follows

$$\frac{d^k x(t)}{dt^k} = \varphi(t,k), \quad for\ all\ t \in T, \tag{A.1}$$

where k is a non-negative integer.

By considering Definition A.1, the differential transform of function $x(t)$ at any time $t = t_i$ in the domain is locally defined as

$$X_i(k) = \frac{\varphi(t_i,k)}{k!} = \frac{1}{k!}\left[\frac{d^k x(t)}{dt^k}\right]\Bigg|_{t=t_i}. \tag{A.2}$$

Definition A.2. If $x(t)$ is analytic in the domain T, then $x(t)$ can be denoted by the Taylor series at $t = t_i$ as

$$x_i(t) = \sum_{k=0}^{\infty} X_i(k)(t-t_i)^k = D^{-1}X(k), \quad t_i - \rho \le t \le t_i + \rho \tag{A.3}$$

where D^{-1} denotes the inverse differential transform operator and ρ is the radius of convergence. By truncating series (A.3), $x(t)$ can be represented as

$$x_i(t) = \sum_{k=0}^{K} X_i(k)(t-t_i)^k + O\left((t-t_i)^{K+1}\right), \quad t_i - \rho \le t \le t_i + \rho. \tag{A.4}$$

Theorem A.1. *Let $x(t)$ and $y(t)$ be analytic in the domain T and $t_i \in T$. If $D(x_i(t),k) = X_i(k)$, $D(y_i(t),k) = Y_i(k)$ and $z_i(t) = x_i(t)\,y_i(t)$, then the following equality holds*

$$Z_i(k) = D(z_i(t),k) = \sum_{n=0}^{k} X_i(n)\,Y_i(k-n). \tag{A.5}$$

Proof. Since

$$x_i(t)\,y_i(t) = \left(\sum_{k=0}^{\infty} X_i(k)(t-t_i)^k\right)\left(\sum_{k=0}^{\infty} Y_i(k)(t-t_i)^k\right), \tag{A.6}$$

and

$$\begin{aligned} x_i(t)\,y_i(t) = X_i(0)\,Y(0) + [X_i(0)\,Y_i(1) + X_i(1)\,Y_i(0)](t-t_i) + \\ [X_i(0)\,Y_i(2) + X_i(1)\,Y_i(1) + X_i(2)\,Y_i(0)](t-t_i)^2 + \ldots \\ + [X_i(0)\,Y_i(N) + X_i(1)\,Y_i(N-1) + \cdots + X_i(N)\,Y_i(0)](t-t_i)^N + \ldots. \end{aligned}$$

Rearranging the terms yield

$$x_i(t)\,y_i(t) = \sum_{k=0}^{\infty}\left[\sum_{n=0}^{k} X_i(l)\,Y_i(k-n)\right](t-t_i)^k. \tag{A.7}$$

Taking differential transform of both sides leads to

$$Z_i(k) = D(z_i(t),k) = \sum_{n=0}^{k} X_i(n)\,Y_i(k-n).$$

■

Theorem A.2. *Let $x(t)$ be analytic in the domain T and $t_i \in T$. If $D(x_i(t)) = X_i(k)$ and $z_i(t) = x_i^m(t)$ for $m \in N$, then the following equality holds*

$$Z_i(k) = D(z_i(t),k) = \sum_{n_1=0}^{k}\sum_{n_2=0}^{k-n_1}\cdots\sum_{n_{m-1}=0}^{k-\sum_{i=1}^{m-2} n_i} X_i(n_1)\ldots X_i(n_{m-1})X_i\left(k-\sum_{i=1}^{m-1} n_i\right). \tag{A.8}$$

Proof. With the help of $z_i(t) = x_i^{m-1}(t)\,x_i(t)$, use of Theorem A.1 leads to

$$Z_i(k) = D(z_i(t),k) = \sum_{n_1=0}^{k} X_i(n_1)X_i^{m-1}(k-n_1). \tag{A.9}$$

Similarly, $X^{m-1}(k-n_1)$ can be written as

$$X_i^{m-1}(k-n_1) = \sum_{n_2=0}^{k-n_1} X_i(n_2)X_i^{m-2}(k-n_1-n_2). \tag{A.10}$$

In general, the following equality holds

$$X_i^{m-p}\left(k-\sum_{i=1}^{p} n_i\right) = \sum_{n_{p+1}=0}^{k-\sum_{i=1}^{p} n_i} X_i(n_{p+1})X_i^{m-p-1}\left(k-\sum_{i=1}^{p+1} n_i\right). \tag{A.11}$$

With the use of equality (A.11) in equation (A.9) gives the desired result (A.8). ■

Theorem A.3. *Let $x(t)$ is analytic in the domain T and $t_i \in T$. If $D(x_i(t)) = X_i(k)$*

and $z_i(t) = \frac{d^m x_i(t)}{dt^m}$ *for* $m \in N^+$*, then following equality holds*

$$Z_i(k) = D(z_i(t), k) = (k+1)(k+2)\dots(k+m)X_i(k+m). \tag{A.12}$$

Proof. Let us consider the Taylor expansion of the analytic function $x_i(t)$ about $t = t_i$ as follows

$$x_i(t) = \sum_{k=0}^{\infty} X_i(k)(t-t_i)^k. \tag{A.13}$$

$m-th$ time derivative of the previous expression is given by

$$z_i(t) = \frac{d^m x_i(t)}{dt^m} = \sum_{k=m}^{\infty} k(k-1)(k-2)\dots(k-m+1)X_i(k)(t-t_i)^{k-m}. \tag{A.14}$$

The series can also be written as

$$z_i(t) = \sum_{k=0}^{\infty} (k+1)(k+2)(k+3)\dots(k+m)X_i(k+m)(t-t_i)^k. \tag{A.15}$$

Taking differential transformation of both sides leads to the following desired result

$$Z_i(k) = D(z_i(t), k) = (k+1)(k+2)\dots(k+m)X_i(k+m). \tag{A.16}$$

■

Definition A.3. Let $x(t)$ be analytic in the domain T, $t_i \in T$ and m, n, $p \in N$. If $D(x_i(t), k) = X_i(k)$ and $z_i(t) = (x_i^m(t))\left(\frac{d^p x_i(t)}{dt^p}\right)^n$, we define the following differential transformation notation

$$Z_i(k) = D(z_i(t), k) = \overline{X}_i(m; p, n; k). \tag{A.17}$$

Throughout this thesis, we assume this notation is hold with given assumptions in Definition A.3. Let us consider the following special cases:

$$\overline{X}_i(m; 0, 0; k) = \sum_{n_1=0}^{k} \sum_{n_2=0}^{k-n_1} \sum_{n_3=0}^{k-n_1-n_2} \cdots \sum_{n_{m-1}=0}^{k-\sum_{i=1}^{m-2} n_i} X_i(n_1)\dots X_i(n_{m-1}) X_i\left(k - \sum_{i=1}^{m-1} n_i\right),$$

$$\overline{X}_i(0; p, 1; k) = (k+1)(k+2)\dots(k+p)X_i(k+p),$$

$$\overline{X}_i(0; p, n; k) = \sum_{n_1=0}^{k} \sum_{n_2=0}^{k-n_1} \dots \sum_{n_{m-1}=0}^{k-\sum_{i=1}^{m-2} n_i} \overline{X}_i(0, p, 0; n_1)\dots\overline{X}_i(0, p, 0; n_{m-1}) \overline{X}_i\left(0, p, 1; k - \sum_{i=1}^{m-1} n_i\right).$$

Theorem A.4. *Let* $x(t)$ *be analytic in the domain* T, $t_i \in T$ *and* m, n, $p \in N$. *If* $D(x_i(t), k) = X_i(k)$, *then the following equality holds,*

$$\overline{X}_i(m; p, n; k) = \sum_{j=0}^{k} \overline{X}_i(m; 0, 0; j)\overline{X}_i(0; p, n; k-j). \tag{A.18}$$

Proof. Considering the Theorem A.1 and the definitions of $\overline{X}_i(m; 0, 0; k)$ and $\overline{X}_i(0; p, n; k)$, the desired result (A.18) is seen to be obvious. ■

B
ERROR ANALYSIS

We analyse here the bound of $||G(\theta, dx)||$ occurred in Chapter 6.4 in terms of the direction parameter θ and local order of approximation K. We consider the following two cases;

Case 1:

Let us consider the direction parameter $\theta = 0.5$ and K is even. In this scenario, we get the following general terms for $G_1(0.5, \Delta x)$

$$\begin{aligned}(G_1(0.5,\Delta x))_i = \big[(0.5)^{K+1}\boldsymbol{C}_{i_*}(K+1) - (-0.5)^{K+1}\big(\boldsymbol{C}_{i_*}(K+1) + \\ \Delta x(K+2)\boldsymbol{C}_{i_*}(K+2)\big]\Delta x^K \\ = \left(\frac{1}{2}\right)^K \boldsymbol{C}_{i_*}(K+1)\,\Delta x^K + O(\Delta x^{K+1}).\end{aligned} \tag{B.1}$$

Defining $M_1 = \max_i \left\| \left(\frac{1}{2}\right)^K \boldsymbol{C}_{i_*}(K+1) \right\|$ and neglecting the higher order terms yield

$$||G_1(0.5, \Delta x)|| \leq M_1 \Delta x^K. \tag{B.2}$$

And the general term of $G_2(0.5, dx)$ becomes,

$$\begin{aligned}(G_2(0.5,dx))_i = \big[(K+1)(0.5)^K\boldsymbol{C}_{i_*}(K+1) - (K+1)(-0.5)^K\big(\boldsymbol{C}_{i_*}(K+1) + \\ \Delta x(K+2)\boldsymbol{C}_{i_*}(K+2)\big]\Delta x^{K-1} \\ = -(K+1)(K+2)\left(\frac{1}{2}\right)^K \boldsymbol{C}_{i_*}(K+2)\,\Delta x^K + O(\Delta x^{K+1}).\end{aligned} \tag{B.3}$$

Defining $M_2 = \max_i \left\| (K+1)(K+2)\left(\frac{1}{2}\right)^K \boldsymbol{C}_{i_*}(K+2) \right\|$ and neglecting the higher order terms lead to

$$||G_2(0.5, \Delta x)|| \leq M_2 \Delta x^K. \tag{B.4}$$

Reminding $G(0.5, \Delta x) = G_1(0.5, \Delta x) + G_2(0.5, \Delta x)$ and defining $P_2 = \max\{M_1,\ M_2\}$ give rise to the estimate $||G(0.5, \Delta x)|| \leq P_2 \Delta x^K$.

Case 2:

Assume that the conditions of the Case 1 are not hold. In this case, we get the following

general terms for $G_1(\theta, \Delta x)$

$$\begin{aligned}(G_1(\theta,\Delta x))_i = \big[(1-\theta)^{K+1}\boldsymbol{C}_{i_*}(K+1) - (-\theta)^{K+1}\big(\boldsymbol{C}_{i_*}(K+1) + \\ \Delta x(K+2)\boldsymbol{C}_{i_*}(K+2)\big]\Delta x^K, \\ = \big[(1-\theta)^{K+1} - (-\theta)^{K+1}\big]\boldsymbol{C}_{i_*}(K+1)\,\Delta x^K + O(\Delta x^{K+1}).\end{aligned} \tag{B.5}$$

Defining $M_3 = \max_i \left\|\big[(1-\theta)^{K+1} - (-\theta)^{K+1}\big]\boldsymbol{C}_{i_*}(K+1)\right\|$ and neglecting the higher order terms give rise to

$$||G_1(\theta,\Delta x)|| \leq M_3 \Delta x^K. \tag{B.6}$$

And the general term of $G_2(\theta, \Delta x)$ becomes,

$$\begin{aligned}(G_2(\theta,\Delta x))_i = \big[(K+1)(1-\theta)^K\boldsymbol{C}_{i_*}(K+1) - (K+1)(-\theta)^K\big(\boldsymbol{C}_{i_*}(K+1) + \\ \Delta x(K+2)\boldsymbol{C}_{i_*}(K+2)\big]\Delta x^{K-1} \\ = \big[(K+1)(1-\theta)^K - (K+1)(-\theta)^K\big]\boldsymbol{C}_{i_*}(K+1)\,\Delta x^{K-1} + O(\Delta x^K).\end{aligned} \tag{B.7}$$

Defining $M_4 = \max_i \left\|\big[(K+1)(1-\theta)^K - (K+1)(-\theta)^K\big]\boldsymbol{C}_{i_*}(K+1)\right\|$ and neglecting the higher order terms yield

$$||G_2(\theta,\Delta x)|| \leq M_4 \Delta x^{K-1}. \tag{B.8}$$

Since $G(\theta, \Delta x) = G_1(\theta, \Delta x) + G_2(\theta, \Delta x)$, taking $P_2 = \max\{M_3\Delta x,\ M_4\}$ leads to the estimate $||G(\theta,\Delta x)|| \leq P_2 \Delta x^{K-1}$.

PUBLICATIONS FROM THE THESIS

Contact Information: -

Papers

1. H. Tunc and M. Sarı, A new implicit-explicit local differential method for boundary value problems, Turkish Journal of Mathematics, (2021), 45(2), pp. 742-766.

2. H. Tunc and M. Sarı, A stability preserved time-integration method for nonlinear advection-diffusion-reaction processes, Journal of Mathematical Chemistry, (2021), 59, pp. 1917–1937.

3. H. Tunc and M. Sarı, An implicit-explicit local method for parabolic partial differential equations, Engineering Computations, (2021), https://doi.org/10.1108/EC-02-2021-0077.

4. H. Tunc and M. Sarı, A spatial local method for solving 2D and 3D advection-diffusion equations, Submitted.

5. H. Tunc and M. Sarı, A novel multi-derivative time integration method for stiff initial value problems, Submitted.

Conference Papers

1. H. Tunc and M. Sarı, An implicit-explicit local transform method for capturing stiff behavior of singularly perturbed boundary value problems, 4th International Conference on Mathematical Advances and Applications, 2021.