การแก้ระบบสมการเชิงปริพันธ์-อนุพันธ์เชิงเส้นด้วยระเบียบวิธีปริพันธ์อันตะร่วมกับพหุนามเชบีเชฟแบบเลื่อน

นางสาวเมธินี จุ้ยต่าย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาคณิตศาสตร์ประยุกต์และวิทยาการคณนา
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2562

SOLVING SYSTEM OF LINEAR INTEGRO-DIFFERENTIAL EQUATIONS BY FINITE INTEGRATION METHOD WITH SHIFTED CHEBYSHEV POLYNOMIALS

Miss Matinee Juytai

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science Program in Applied Mathematics and Computational Science

Department of Mathematics and Computer Science

Faculty of Science

Chulalongkorn University

Academic Year 2019

Thesis Title	SOLVING SYSTEM OF LINEAR INTEGRO-DIFFERENTIAL EQUATIONS BY FINITE INTEGRATION METHOD WITH SHIFTED CHEBYSHEV POLYNOMIALS
By	Miss Matinee Juytai
Field of Study	Applied Mathematics and Computational Science
Thesis Advisor	Associate Professor Ratinan Boonklurb, Ph.D.

Accepted by the Faculty of Science, Chulalongkorn University in Partial Fulfillment of the Requirements for the Master's Degree

. . . . Dean of the Faculty of Science

(Professor Polkit Sangvanich, Ph.D.)

THESIS COMMITTEE

. . . . Chairman

(Associate Professor Khamron Mekchay, Ph.D.)

. . . . Thesis Advisor

(Associate Professor Ratinan Boonklurb, Ph.D.)

. . . . Examiner

(Associate Professor Petarpa Boonserm, Ph.D.)

. . . . External Examiner

(Assistant Professor Tawikan Treeyaprasert, Ph.D.)

เมธินี จุ้ยต่าย : การแก้ระบบสมการเชิงปริพันธ์-อนุพันธ์เชิงเส้นด้วยระเบียบวิธีปริพันธ์อันตะร่วมกับพหุนามเชบีเชฟแบบเลื่อน. (SOLVING SYSTEM OF LINEAR INTEGRO-DIFFERENTIAL EQUATIONS BY FINITE INTEGRATION METHOD WITH SHIFTED CHEBYSHEV POLYNOMIALS) อ.ที่ปรึกษาวิทยานิพนธ์หลัก : รศ.ดร. รตินันท์ บุญเคลือบ, 84 หน้า.

ในวิทยานิพนธ์เล่มนี้ เราดัดแปลงระเบียบวิธีปริพันธ์อันตะโดยใช้พหุนามเชบีเชฟแบบเลื่อน ซึ่งเครื่องมือที่สำคัญของระเบียบวิธีปริพันธ์อันตะโดยใช้พหุนามเชบีเชฟแบบเลื่อนนี้ คือ เมทริกซ์ปริพันธ์เชบีเชฟแบบเลื่อน ซึ่งถูกสร้างขึ้นเพื่อเป็นตัวแทนเมทริกซ์สำหรับการอินทิเกรตบนจุดคำนวณที่สร้างจากศูนย์ของพหุนามเชบีเชฟแบบเลื่อนบางดีกรี จากนั้นจึงสร้างขั้นตอนวิธีเชิงตัวเลขที่มีประสิทธิภาพด้วยระเบียบวิธีปริพันธ์อันตะโดยใช้พหุนามเชบีเชฟแบบเลื่อนเพื่อหาผลเฉลยโดยประมาณของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นแบบสติฟ ระบบสมการเชิงปริพันธ์-อนุพันธ์เชิงเส้นแบบโวลเทอร์รา และระบบสมการเชิงปริพันธ์-อนุพันธ์เชิงเส้นแบบเฟรดโฮล์ม ภายใต้เงื่อนไขขอบบางประการ ยิ่งไปกว่านั้นยังได้ทำการทดสอบประสิทธิภาพของขั้นตอนวิธีทั้งสามของเราผ่านตัวอย่างที่หลากหลาย อีกทั้งยังได้นำเสนอการเปรียบเทียบค่าผิดพลาดสัมบูรณ์เฉลี่ยระหว่างผลเฉลยที่ได้จากขั้นตอนวิธีของเรา กับผลเฉลยเชิงวิเคราะห์ หรือผลเฉลยที่ได้จากวิธีอื่นๆ ซึ่งแสดงให้เห็นว่า ขั้นตอนวิธีเชิงตัวเลขของเรา ให้การปรับปรุงค่าความแม่นยำอย่างมีนัยสำคัญ

ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์	ลายมือชื่อนิสิต ....
		ลายมือชื่อ อ.ที่ปรึกษาหลัก ....
สาขาวิชา	คณิตศาสตร์ประยุกต์และวิทยาการคณนา	
ปีการศึกษา	2562	

6172042023 : MAJOR APPLIED MATHEMATICS AND COMPUTATIONAL SCIENCE

KEYWORDS : SHIFTED CHEBYSHEV POLYNOMIAL EXPANSION / FINITE INTEGRATION METHOD / SYSTEM OF LINEAR INTEGRO-DIFFERENTIAL EQUATIONS

MATINEE JUYTAI : SOLVING SYSTEM OF LINEAR INTEGRO-DIFFERENTIAL EQUATIONS BY FINITE INTEGRATION METHOD WITH SHIFTED CHEBYSHEV POLYNOMIALS. ADVISOR : ASSOC. PROF. RATINAN BOONKLURB, Ph.D., 84 pp.

In this thesis, we modify the finite integration method by using the shifted Chebyshev polynomial (FIM-SCP). The major tool of our FIM-SCP is the shifted Chebyshev integration matrix. It is constructed in order to be a matrix representation for integrating over interpolated points which are generated by the zeros of shifted Chebyshev polynomial of a certain degree. The efficiently numerical algorithms are then created by the modified FIM-SCP for seeking approximate solutions of a system of stiff linear ordinary differential equations, a system of linear Volterra integro-differential equations, and a system of linear Fredholm integro-differential equations under some given boundary conditions. Furthermore, our three proposed algorithms are examined the performance via the diversified numerical experiments. The comparisons of their analytical solutions or approximate solutions obtained by our proposed algorithms with other methods are also illustrated through the average absolute error. They provide that our numerical algorithms achieve a significantly accurate improvement.

Department	: Mathematics and Computer Science	Student's Signature	....
		Advisor's Signature	....
Field of Study	: Applied Mathematics and Computational Science		
Academic Year	: 2019		

ACKNOWLEDGEMENTS

First of all, I am deeply indebted to Associate Professor Dr. Ratinan Boonklurb, my thesis advisor, who gives me all the suggestions, guidance and encouragement throughout helps me to deal with many problems in my research. It would not have been completed without him. I also would like to thank Mr. Ampol Duangpan and Miss Phansphitcha Gugaew for helping me in this research.

Moreover, I would like to express many thanks, my financial sponsor, The His Royal Highness Crown Prince Maha Vajiralongkorn Scholarship from the Graduate School, Chulalongkorn University to commemorate the 72^{nd} anniversary of his Majesty King Bhumibala Aduladeja is gratefully acknowledged. I would like to express my sincere thanks to the professors from the Department of Mathematics and Computational Science at Chulalongkorn University for all the support and the opportunity. I am also thankful to my thesis committee, Associate Professor Dr. Khamron Mekchay, Associate Professor Dr. Petarpa Boonserm, and Assistant Professor Dr. Tawikan Treeyaprasert for their valuable advice and comment in this thesis.

Finally, I want to express my gratitude to my family for their kindness supports and encouragement throughout my study. I am so grateful for all of my best friends who helped me achieve here and I would like to thank my best partner for always understanding and making everything better.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Motivation and Literature Surveys

An integro-differential equation (IDE) is an equation that involves both integrals and derivatives of an unknown function. It can be distinguished into two types, namely, Volterra IDE (VIDE) and Fredholm IDE (FIDE) which each type is different depending on the limits of integration, that we will detail them in the next section. Moreover, they have many applications that can be found in various branches of science, engineering, physics, biology and etc., see [1–6] for details of each application. Actually, many problems of the IDE are often constructed to be a system. Anyway, the system of IDEs can be found in the fields of science and engineering. It has a lot of applications such as modeling of the competition between the tumor cell and the immune system [7], wind ripples in the desert [8], dropwise condensation [9], glass-forming process [10], examining the noise term phenomenon [11], nano-hydrodynamics [12] and so on.

The IDEs are usually difficult to solve analytically. Therefore, numerical methods are required to obtain a decent approximate solution. Several numerical methods for approximating either VIDEs or FIDEs are well-known. Zhao and Corless [13] used compact finite difference method (FDM) for IDEs. Brunner [14] applied a collocation-type method to Volterra-Hammerstein integral equation as well as IDEs. Sepehrian and Razzaghi [15] have been proposed a single term Walsh series method (STWS) for solving VIDEs. Pour-Mahmoud et al. [16] considered Ortiz and Samara's operational approach to the Tau method for the numerical solution of the system of FIDEs. Şuayip et al. [17] have been proposed the collocation method with Bessel polynomials for solving a system of FIDEs. Farshid and Seyede [18] also applied the collocation method to solve systems of linear FIDEs in terms of Fibonacci polynomials.

A few years ago, the finite integration method (FIM) was firstly introduced in 2013 by Wen et al. [19] which has been developed to solve the linear boundary value problems of differential equations. They use the linear approximation and radial basis functions to build the first order integration matrix for representing a single-layer integration and obtain directly the higher order integration matrix for a multi-layer integration. Their FIM can just solve the one-dimensional linear differential equations. After that in 2015, Li et al. [20] have been extended the FIM in order to solve multi-dimensional problems. Then, Li et al. [21] have been improved the FIM by consuming the numerical quadrature such as Simpson's rule, Newton Cotes and Lagrange interpolation instead of trapezoidal rule to handle the linear differential equations. Moreover, they demonstrate that their improved FIM give highly accurate solutions compared with the FDM and the traditional FIM. Recently, Boonklurb et al. [22] have been proposed the modified FIM using Chebyshev polynomial expansion (FIM-CPE) for solving the one- and two-dimensional linear differential equations. The modified FIM-CPE also provides a much higher accuracy than the FDM and those original FIMs with low computational nodes.

In this thesis, we apply the idea of FIM-CPE given by [22], but slightly modify it by using the shifted Chebyshev polynomials which is called the FIM-SCP. Henceforth, our idea will be referred to as FIM-SCP and use it to devise the efficiently numerical algorithms for solving the system of linear ordinary differential equations (ODEs), especially, the stiff system, the system of linear VIDEs, and the system of linear FIDEs. We assume that under some given boundary conditions, the three types of our considered systems of linear ODEs, VIDEs and FIDEs have a unique solution. Then, we express our approximate solution in form of the linear combination of shifted Chebyshev polynomials. We use the zeros of shifted Chebyshev polynomial of a certain degree to be the computational nodes and construct the shifted Chebyshev integration matrices which are the main ingredient for this devised algorithm. Finally, we implement our proposed algorithms with several numerical examples in order to demonstrate our accurate results when compared with the results obtained by other methods from literature or their analytical solutions.

1.2 Systems of Linear Differential Equations

In this section, we give the details of our three considered systems of linear differential equations in order to be the information for creating the numerical algorithms in this thesis. They consist of the system of linear ODEs, the system of linear VIDEs and the system of linear FIDEs. In addition, the facts and assumptions associated with each considered system are provided as follows.

► System of Linear ODEs

A system of linear ODEs is a system of linear differential equations in one-dimension which are equations containing a function of one independent variable and its derivatives. Our considered system of m linear ODEs is in the form of

$$\sum_{j=1}^{m} \mathcal{L}_{i,j} v_j(x) = f_i(x), \quad x \in (a, b) \tag{1.1}$$

for all $i \in \{1, 2, 3, \ldots, m\}$ and $a, b \in \mathbb{R}$ be such that $a < b$. The linear differential operator $\mathcal{L}_{i,j}$ of order $l_{i,j}$ is defined as

$$\mathcal{L}_{i,j} := p_{i,j}^{l_{i,j}}(x) D^{l_{i,j}} + p_{i,j}^{l_{i,j}-1}(x) D^{l_{i,j}-1} + p_{i,j}^{l_{i,j}-2}(x) D^{l_{i,j}-2} + \cdots + p_{i,j}^{1}(x) D + p_{i,j}^{0}(x), \tag{1.2}$$

where $D^k = \frac{d^k}{dx^k}$ is the k^{th} order derivative with respect to x for $k \in \{1, 2, 3, \ldots, l_{i,j}\}$, $p_{i,j}^k(x)$ for each $k \in \{0, 1, 2, \ldots, l_{i,j}\}$ are continuously differentiable functions up to the highest order of derivative contained in (1.1), $f_i(x)$ are given continuous functions and $v_j(x)$ are unknown functions to be solved. In this thesis, we assume that under some given boundary conditions, then the system (1.1) has a unique solution.

In this study, we are interested in a stiff system of ODEs which is a system of ODEs with a significant difference between the coefficients. There is no universally accepted definition for stiffness. However, the numerical methods for solving the stiff system of ODEs are numerically unstable. The numerical methods have to take small steps for solving this problem to obtain satisfactory results comparing with the analytical solutions.

The system of linear IDEs appears in many types of situation and depends mainly on the limits of integration appear therein. In this thesis, we study the system of linear IDEs in both types of Volterra and Fredholm. Next, we mention some details for our studied system of linear VIDEs and system of linear FIDEs that we study them as follows.

▶ System of Linear VIDEs

Next, we consider the system of linear VIDEs which contains both differential part and integration part. For the system of linear VIDEs, at least one of the limits of integration is a variable. The system of m linear VIDEs, that we study, is given by

$$\sum_{j=1}^{m} \mathcal{L}_{i,j} v_j(x) = f_i(x) + \sum_{j=1}^{m} \lambda_{i,j} \int_a^x \mathcal{K}_{i,j}(x,t) v_j(t) dt, \quad x \in (a,b) \tag{1.3}$$

for all $i \in \{1, 2, 3, \ldots, m\}$, where $a < b$ are arbitrary real constants. The linear differential operator $\mathcal{L}_{i,j}$ of order $l_{i,j}$ is defined as same as (1.2), $\lambda_{i,j}$ are real constant coefficients, $\mathcal{K}_{i,j}(x,t)$ are continuously integrable kernel functions, $f_i(x)$ are continuous functions and $v_j(x)$ are unknown functions to be solved. In this thesis, we assume that under some given boundary conditions, the system (1.3) has a unique solution.

▶ System of Linear FIDEs

Finally, we consider the system of linear FIDEs which contains both differential part and integration part. For the system of linear FIDEs, the limits of integration are fixed numbers. The system of m linear FIDEs, that we study, can be written as follows

$$\sum_{j=1}^{m} \mathcal{L}_{i,j} v_j(x) = f_i(x) + \sum_{j=1}^{m} \lambda_{i,j} \int_a^b \mathcal{K}_{i,j}(x,t) v_j(t) dt, \quad x \in (a,b) \tag{1.4}$$

for all $i \in \{1, 2, 3, \ldots, m\}$, where $a < b$ are any real constants. The linear differential operator $\mathcal{L}_{i,j}$ of order $l_{i,j}$ is defined as well as (1.2), $\lambda_{i,j}$ are real constant coefficients of the integration parts, $\mathcal{K}_{i,j}(x,t)$ are continuously integrable kernel functions, $f_i(x)$ are continuous functions and $v_j(x)$ are unknown functions to be solved. In this thesis, we assume that under some given boundary conditions, the system (1.4) has a unique solution.

1.3 Research Objectives

The goal of the research is to obtain numerical procedures based on the FIM-SCP for finding approximate solutions of the system of linear ODEs, the system of linear VIDEs and also the system of linear FIDEs.

1.4 Thesis Overview

We divide this thesis into five chapters. Chapter 1 is an introduction of this work including the motivation and literature surveys, the details of our considered systems of linear differential equations, the research objectives and the thesis overview. Next, the background knowledge concerning the shifted Chebyshev polynomial, including its definition and some important properties are presented in Chapter 2 in order to construct the shifted Chebyshev integration matrices. Chapter 3 presents the procedure for solving the stiff system of linear ODEs and numerical examples. Then, we propose the numerical procedures for solving the systems of linear IDEs which consist of VIDEs and FIDEs and numerical examples in Chapter 4. Finally, conclusions and some future works are presented in Chapter 5.

CHAPTER II

MODIFIED FIM-SCP

In this chapter, we provide the background knowledge on the definition and some basic properties of the shifted Chebyshev polynomials which are important in the part of the construction of our numerical algorithms. After that, we use these facts to construct the shifted Chebyshev integration matrices. We first introduce the shifted Chebyshev polynomials and some useful facts about them.

2.1 Shifted Chebyshev Polynomial

In some applications, the interval $[0, 1]$ is more convenient to use than $[-1, 1]$. Thus, we transform the independent variable of Chebyshev polynomial $T_n(x)$ for $n \geq 0$ from the interval $[-1, 1]$ into $[0, 1]$ by using the transformation $s = 2x-1$ or $x = \frac{1}{2}(s+1)$. Then, the polynomial obtained after transforming is called a shifted Chebyshev polynomial $T_n^*(x)$ for $x \in [0, 1]$. Their definitions are provided as follows.

Definition 2.1. ([23]) The Chebyshev polynomial of degree $n \geq 0$ is defined by

$$T_n(x) = \cos(n \arccos x) \quad \text{for} \quad x \in [-1, 1].$$

However, the shifted Chebyshev polynomial $T_n^*(x)$ of degree $n \geq 0$ can be defined by

$$T_n^*(x) = T_n(2x - 1) \quad \text{for} \quad x \in [0, 1]. \tag{2.1}$$

Moreover, the properties of shifted Chebyshev polynomial are given in Lemma 2.1 which will be used to construct the first and higher orders of the shifted Chebyshev integration matrices in the next section.

Lemma 2.1. ([23]) *The followings are properties of shifted Chebyshev polynomials.*

(i) The zeros of shifted Chebyshev polynomial $T_n^*(x)$ for $x \in [0,1]$ are

$$x_k = \frac{1}{2}\left(\cos\left(\frac{2k-1}{2n}\pi\right)+1\right), \ k \in \{1,2,3,\ldots,n\}. \tag{2.2}$$

(ii) The p^{th} order derivatives of $T_n^*(x)$ at $x=0$ and $x=1$ for $p \in \mathbb{N}$ are

$$\frac{d^p}{dx^p}T_n^*(x)\Big|_{x=0} = \prod_{k=0}^{p-1}\frac{(n^2-k^2)(-1)^{p+n}}{2k+1}, \tag{2.3}$$

$$\frac{d^p}{dx^p}T_n^*(x)\Big|_{x=1} = \prod_{k=0}^{p-1}\frac{n^2-k^2}{2k+1}. \tag{2.4}$$

(iii) The single integrations of shifted Chebyshev polynomial $T_n^*(x)$ for $x \in [0,1]$ are

$$\overline{T}_0^*(x) = \int_0^x T_0^*(\xi)\,d\xi = x, \tag{2.5}$$

$$\overline{T}_1^*(x) = \int_0^x T_1^*(\xi)\,d\xi = x^2 - x, \tag{2.6}$$

$$\overline{T}_n^*(x) = \int_0^x T_n^*(\xi)\,d\xi = \frac{1}{4}\left(\frac{T_{n+1}^*(x)}{n+1} - \frac{T_{n-1}^*(x)}{n-1}\right) - \frac{(-1)^n}{2(n^2-1)}, \ n \geq 2. \tag{2.7}$$

Moreover, the single integration of $T_n^*(x)$ at the upper bound can be written as

$$\overline{T}_n^*(1) = \int_0^1 T_n^*(\xi)\,d\xi = \begin{cases} \frac{1}{1-n^2} & \text{if } n \equiv 0 \pmod 2, \\ 0 & \text{if } n \equiv 1 \pmod 2. \end{cases} \tag{2.8}$$

(iv) The discrete orthogonality relation of shifted Chebyshev polynomials T_i^* and T_j^* is

$$\sum_{k=1}^{n} T_i^*(x_k)\,T_j^*(x_k) = \begin{cases} 0 & \text{if } i \neq j, \\ n & \text{if } i = j = 0, \\ \frac{n}{2} & \text{if } i = j \neq 0, \end{cases}$$

where x_k for $k \in \{1,2,3,\ldots,n\}$ is defined by (2.2) and $0 \leq i,j \leq n$.

(v) The shifted Chebyshev matrix $\mathbf{T}^*$ *at each node* $\{x_k\}_{k=1}^n$ *defined by* (2.2) *is*

$$\mathbf{T}^* = \begin{bmatrix} T_0^*(x_1) & T_1^*(x_1) & \cdots & T_{n-1}^*(x_1) \\ T_0^*(x_2) & T_1^*(x_2) & \cdots & T_{n-1}^*(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ T_0^*(x_n) & T_1^*(x_n) & \cdots & T_{n-1}^*(x_n) \end{bmatrix}. \tag{2.9}$$

Then, it has the multiplicative inverse

$$(\mathbf{T}^*)^{-1} = \frac{1}{n}\mathrm{diag}(1, 2, 2, \ldots, 2)(\mathbf{T}^*)^\top. \tag{2.10}$$

(vi) The recurrence relation of shifted Chebyshev polynomials T_{n-1}^**,* T_n^**, and* T_{n+1}^* *is*

$$T_{n+1}^*(x) = 2(2x-1)T_n^*(x) - T_{n-1}^*(x)$$

with the starting values $T_0^*(x) = 1$ *and* $T_1^*(x) = 2x - 1$*.*

Proof. The proofs of this lemma can similarly prove corresponding to the proofs of the properties of Chebyshev polynomial $T_n(x)$ which can be found in [23]. □

Next, we apply the idea of FIM-CPE which is described in [22] to construct the first order integration matrix based on the shifted Chebyshev polynomials. Then, the higher order shifted Chebyshev integration matrix can be obtained easily by using the same idea as the first order integration matrix.

2.2 Shifted Chebyshev Integration Matrices

First, we let $u_j(x)$ to be an approximate solution of the unknown function $v_j(x)$ in (1.1), (1.3), and (1.4). Next, to construct the shifted Chebyshev integration matrices, let M be a positive integer, $u_j(x)$ be a linear combination of the shifted Chebyshev polynomials $T_0^*(x), T_1^*(x), T_2^*(x), \ldots, T_{M-1}^*(x)$ and x_k be grid points generated by the zeros of shifted Chebyshev polynomial T_M^* as defined in (2.2) for all $k \in \{1, 2, 3, \ldots, M\}$, where $0 < x_1 < x_2 < x_3 < \cdots < x_M < 1$. Then, we approximate u_j at node x_k by

$$u_j(x_k) = \sum_{n=0}^{M-1} c_{n_j} T_n^*(x_k), \tag{2.11}$$

where c_{n_j} is unknown coefficients to be considered. For $k \in \{1, 2, 3, \ldots, M\}$, it can be express in the matrix form

$$\begin{bmatrix} u_j(x_1) \\ u_j(x_2) \\ \vdots \\ u_j(x_M) \end{bmatrix} = \begin{bmatrix} T_0^*(x_1) & T_1^*(x_1) & \cdots & T_{M-1}^*(x_1) \\ T_0^*(x_2) & T_1^*(x_2) & \cdots & T_{M-1}^*(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ T_0^*(x_M) & T_1^*(x_M) & \cdots & T_{M-1}^*(x_M) \end{bmatrix} \begin{bmatrix} c_{0_j} \\ c_{1_j} \\ \vdots \\ c_{M-1_j} \end{bmatrix},$$

which is denoted by $\mathbf{u}_j = \mathbf{T}^* \mathbf{c}_j$. Since $\mathbf{T}^*$ is invertible by Lemma 2.1(v), $\mathbf{c}_j = (\mathbf{T}^*)^{-1} \mathbf{u}_j$, where $\mathbf{T}^*$ and $(\mathbf{T}^*)^{-1}$ are defined in (2.9) and (2.10) for all $j \in \{1, 2, 3, \ldots, m\}$.

Now, for $k \in \{1, 2, 3, \ldots, M\}$, we consider the single-layer integration of u_j from 0 to the zero x_k denoted by $U_j^{(1)}(x_k)$, we obtain

$$U_j^{(1)}(x_k) = \int_0^{x_k} u_j(\xi)\, d\xi = \sum_{n=0}^{M-1} c_{n_j} \int_0^{x_k} T_n^*(\xi)\, d\xi = \sum_{n=0}^{M-1} c_{n_j} \overline{T}_n^*(x_k) \tag{2.12}$$

where $\overline{T}_n^*$ is the single-layer integration of shifted Chebyshev polynomial that can explicitly find by (2.5), (2.6), and (2.7) depending on its degree n. After substituting each node x_k into $U_j^{(1)}(x_k)$, it can be written in the matrix equation

$$\begin{bmatrix} U_j^{(1)}(x_1) \\ U_j^{(1)}(x_2) \\ \vdots \\ U_j^{(1)}(x_M) \end{bmatrix} = \begin{bmatrix} \overline{T}_0^*(x_1) & \overline{T}_1^*(x_1) & \cdots & \overline{T}_{M-1}^*(x_1) \\ \overline{T}_0^*(x_2) & \overline{T}_1^*(x_2) & \cdots & \overline{T}_{M-1}^*(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \overline{T}_0^*(x_M) & \overline{T}_1^*(x_M) & \cdots & \overline{T}_{M-1}^*(x_M) \end{bmatrix} \begin{bmatrix} c_{0_j} \\ c_{1_j} \\ \vdots \\ c_{M-1_j} \end{bmatrix}, \tag{2.13}$$

which is denoted by $\mathbf{U}_j^{(1)} = \overline{\mathbf{T}}^* \mathbf{c}_j = \overline{\mathbf{T}}^* (\mathbf{T}^*)^{-1} \mathbf{u}_j := \mathbf{A} \mathbf{u}_j$, where $\mathbf{A} = \overline{\mathbf{T}}^* (\mathbf{T}^*)^{-1}$ is called *the first order shifted Chebyshev integration matrix* for the FIM-SCP. If we defined the matrix $\mathbf{A} := [a_{ki}]_{M\times M}$, then (2.12) can be written another form

$$U_j^{(1)}(x_k) = \int_0^{x_k} u_j(\xi)\,d\xi = \sum_{i=1}^{M} a_{ki}u_j(x_i)$$

for $k \in \{1, 2, 3, \ldots, M\}$ or in the matrix form

$$\begin{bmatrix} U_j^{(1)}(x_1) \\ U_j^{(1)}(x_2) \\ \vdots \\ U_j^{(1)}(x_M) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1M} \\ a_{21} & a_{22} & \cdots & a_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M1} & a_{M2} & \cdots & a_{MM} \end{bmatrix} \begin{bmatrix} u_j(x_1) \\ u_j(x_2) \\ \vdots \\ u_j(x_M) \end{bmatrix}.$$

Next, we consider the double-layer integration of u_j from 0 to x_k, $k \in \{1, 2, 3, \ldots, M\}$. It is denoted by $U_j^{(2)}(x_k)$. Then, we have

$$\begin{aligned} U_j^{(2)}(x_k) &= \int_0^{x_k}\int_0^{\xi_2} u_j(\xi_1)\,d\xi_1 d\xi_2 \\ &= \int_0^{x_k} U_j^{(1)}(\xi_2)\,d\xi_2 \\ &= \sum_{i=1}^{M} a_{ki}\,U_j^{(1)}(x_i) \\ &= \sum_{l=1}^{M}\sum_{i=1}^{M} a_{ki}a_{il}\,u_j(x_l) \end{aligned}$$

for $k \in \{1, 2, 3, \ldots, M\}$ or in the matrix form

$$\begin{bmatrix} U_j^{(2)}(x_1) \\ U_j^{(2)}(x_2) \\ \vdots \\ U_j^{(2)}(x_M) \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{M} a_{1i}a_{i1} & \sum_{i=1}^{M} a_{1i}a_{i2} & \cdots & \sum_{i=1}^{M} a_{1i}a_{iM} \\ \sum_{i=1}^{M} a_{2i}a_{i1} & \sum_{i=1}^{M} a_{2i}a_{i2} & \cdots & \sum_{i=1}^{M} a_{2i}a_{iM} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{M} a_{Mi}a_{i1} & \sum_{i=1}^{M} a_{Mi}a_{i2} & \cdots & \sum_{i=1}^{M} a_{Mi}a_{iM} \end{bmatrix} \begin{bmatrix} u_j(x_1) \\ u_j(x_2) \\ \vdots \\ u_j(x_M) \end{bmatrix},$$

which can be written in the matrix form as $\mathbf{U}_j^{(2)} = \mathbf{A}^2\mathbf{u}_j$. The matrix $\mathbf{A}^2$ is called *the second order shifted Chebyshev integration matrix* for the FIM-SCP.

Similarly, we can construct the m-layer integration of u_j from 0 to x_k, by using the same process of the double-layer integration, that is denoted by $U_j^{(m)}(x_k)$, we have

$$\begin{aligned} U_j^{(m)}(x_k) &= \int_0^{x_k}\int_0^{\xi_m}\dots\int_0^{\xi_2} u_j(\xi_1)\,d\xi_1\dots\xi_{m-1}\xi_m \\ &= \int_0^{x_k} U_j^{(m-1)}(\xi_m)\,d\xi_m \\ &= \sum_{i=1}^{M} a_{ki}\,U_j^{(m-1)}(x_i) \\ &= \sum_{l=1}^{M}\sum_{i=1}^{M} a_{ki}\left[\mathbf{A}^{m-1}\right]_{il} u_j(x_l) \end{aligned}$$

for $k \in \{1,2,3,\dots,M\}$, whose equation can be composed as $\mathbf{U}_j^{(m)} = \mathbf{A}^m\mathbf{u}_j$. The matrix $\mathbf{A}^m$ is called *the m^{th} order shifted Chebyshev integration matrix* for the FIM-SCP.

Next, we can further construct the shifted Chebyshev integration matrix at the upper boundary $x = 1$ in order to benefit for devising a numerical algorithm to solve the system of m linear FIDEs. Let us first consider the single-layer integration of u_j from 0 to 1 denoted by $U_j^{(1)}(1)$. Then, we have

$$\begin{aligned} U_j^{(1)}(1) &= \int_0^1 u_j(\xi)\,d(\xi) \\ &= \sum_{n=0}^{M-1} c_{n_j}\int_0^1 T_n^*(\xi)\,d\xi \\ &= \sum_{n=0}^{M-1} c_{n_j}\overline{T}_n^*(1) \\ &:= \mathbf{b}\mathbf{c}_j \\ &= \mathbf{b}(\mathbf{T}^*)^{-1}\mathbf{u}_j, \end{aligned} \tag{2.14}$$

where $\mathbf{b} = \left[\overline{T}_0^*(1), \overline{T}_1^*(1), \overline{T}_2^*(1), \dots, \overline{T}_{M-1}^*(1)\right]$ for its elements can be computed by (2.8), $(\mathbf{T}^*)^{-1}$ is defined by (2.10) and $\mathbf{u}_j = [u_j(x_1), u_j(x_2), u_j(x_3), \dots, u_j(x_{M-1})]^\top$.

CHAPTER III

SYSTEM OF LINEAR ODES

We note that if the considered system of linear IDEs contains no integral terms, then the system becomes the system of linear ODEs. In this chapter, the numerical algorithm of solving the stiff system of m linear ODEs with the given boundary conditions is constructed. Finally, we implement our numerical algorithm with several numerical examples to demonstrate the accuracy compare with the differential transformation method (DTM) [24] and the Runge–Kutta fourth-order (RK-4) method [1].

3.1 Algorithm for Solving System of Linear ODEs

In this section, we can devise a numerical algorithm for solving the system of m linear ODEs (1.1) with boundary conditions by hiring our proposed FIM-SCP. Let u_j be the approximate solution of v_j in (2.11), then (1.1) becomes

$$\sum_{j=1}^{m} \mathcal{L}_{i,j} u_j(x) = f_i(x), \quad x \in (a,b) \tag{3.1}$$

for all $i \in \{1,2,3,\ldots,m\}$. Then, we apply the idea of FIM-SCP in Chapter 2 to formulate the numerical procedure for solving (3.1) as the following steps:

<u>**Step 1.**</u> We use the linear mapping $\bar{x} = \frac{x-a}{b-a}$ to transform $x \in [a,b]$ into $\bar{x} \in [0,1]$. Let $\hat{k} = \frac{1}{b-a}$. Then, (3.1) for $x \in (a,b)$ becomes

$$\sum_{j=1}^{m} \bar{\mathcal{L}}_{i,j} \bar{u}_j(\bar{x}) = \bar{f}_i(\bar{x}), \quad \bar{x} \in (0,1) \tag{3.2}$$

where $\bar{\mathcal{L}}_{i,j} := \hat{k}^{l_{i,j}} \bar{p}_{i,j}^{l_{i,j}}(\bar{x}) \bar{D}^{l_{i,j}} + \hat{k}^{l_{i,j}-1} \bar{p}_{i,j}^{l_{i,j}-1}(\bar{x}) \bar{D}^{l_{i,j}-1} + \cdots + \hat{k} \bar{p}_{i,j}^{1}(\bar{x}) \bar{D}^{1} + \bar{p}_{i,j}^{0}(\bar{x})$ for $\bar{D}^k := \frac{d^k}{d\bar{x}^k}$, $\bar{p}^k_{l_{i,j}}(\bar{x}) := p^k_{l_{i,j}}((b-a)\bar{x}+a)$, $\bar{u}_j(\bar{x}) := u_j((b-a)\bar{x}+a))$ and $\bar{f}_i(\bar{x}) := f_i((b-a)\bar{x}+a)$.

Step 2. We discretize our domain $[0,1]$ into M nodes which are the grid points x_1, x_2, x_3, $\ldots$, x_M generated by the zeros of the shifted Chebyshev polynomial $T_M^*(x)$ defined in (2.2), where $0 < x_1 < x_2 < x_3 < \cdots < x_M < 1$.

Step 3. Let $h_i = \max_{1\le j\le m} l_{i,j}$ for all $i \in \{1,2,3,\ldots,m\}$, where $l_{i,j}$ is the highest order derivative of $\bar{u}_j$ for i^{th} equation of (3.2). We eliminate all derivatives from (3.2) by taking h_i-layer integral from 0 to $\bar{x}$ on both sides of each i^{th} equation in (3.2) for all $i \in \{1,2,3,\ldots,m\}$. Thus, the i^{th} equation of (3.2) becomes

$$\int_0^{\bar{x}} \cdots \int_0^{\xi_2} \sum_{j=1}^{m} \bar{\mathcal{L}}_{i,j}\bar{u}_j(\xi_1)\,d\xi_1 \ldots d\xi_{h_i} = \int_0^{\bar{x}} \cdots \int_0^{\xi_2} \bar{f}_i(\xi_1)\,d\xi_1 \ldots d\xi_{h_i}. \tag{3.3}$$

We substitute $\bar{x}$ in (3.3) by each zero x_k of the shifted Chebyshev polynomial T_M^* for $k \in \{1,2,3,\ldots,M\}$ and use the technique of integration by parts for each term in (3.3). Then, for $l_{i,j} = h_i$, the left-hand side (LHS) of i^{th} equation in (3.3) becomes

$$\begin{aligned}
&\hat{k}^{h_i}\left[\sum_{\beta=0}^{h_i}(-1)^{\beta}\binom{h_i}{\beta}\int_0^{x_k}\cdots\int_0^{\eta_2}(\bar{p}_{i,j}^{h_i})^{(\beta)}\bar{u}_j\,d\eta_1\ldots d\eta_\beta\right]\\
&+\hat{k}^{h_i-1}\int_0^{x_k}\left[\sum_{\beta=0}^{h_i-1}(-1)^{\beta}\binom{h_i-1}{\beta}\int_0^{\xi_{h_i}}\cdots\int_0^{\eta_2}(\bar{p}_{i,j}^{h_i-1})^{(\beta)}\bar{u}_j\,d\eta_1\ldots d\eta_\beta\right]d\xi_{h_i}\\
&+\hat{k}^{h_i-2}\int_0^{x_k}\int_0^{\xi_{h_i}}\left[\sum_{\beta=0}^{h_i-2}(-1)^{\beta}\binom{h_i-2}{\beta}\int_0^{\xi_{h_i-1}}\cdots\int_0^{\eta_2}(\bar{p}_{i,j}^{h_i-2})^{(\beta)}\bar{u}_j\,d\eta_1\ldots d\eta_\beta\right]d\xi_{h_i-1}d\xi_{h_i}\\
&\quad\vdots\\
&+\int_0^{x_k}\cdots\int_0^{\xi_2}\bar{p}_{i,j}^{0}\bar{u}_j\,d\xi_1\ldots d\xi_{h_i}+\frac{d_{i,1}^{j}x_k^{h_i-1}}{(h_i-1)!}+\frac{d_{i,2}^{j}x_k^{h_i-2}}{(h_i-2)!}+\frac{d_{i,3}^{j}x_k^{h_i-3}}{(h_i-3)!}+\cdots+d_{i,h_i}^{j},
\end{aligned} \tag{3.4}$$

and for $l_{i,j} < h_i$, we have

$$\hat{k}^{l_{i,j}} \int_0^{x_k} \dots \int_0^{\xi_{l_{i,j}+2}} \left[\sum_{\beta=0}^{l_{i,j}} (-1)^{\beta} \binom{l_{i,j}}{\beta} \int_0^{\xi_{l_{i,j}-1}} \dots \int_0^{\eta_2} (\bar{p}_{i,j}^{l_{i,j}})^{(\beta)} \bar{u}_j \, d\eta_1 \dots d\eta_\beta \right] d\xi_{l_{i,j}-1} \dots d\xi_{h_i}$$

$$+ \hat{k}^{l_{i,j}-1} \int_0^{x_k} \dots \int_0^{\xi_{l_{i,j}+1}} \left[\sum_{\beta=0}^{l_{i,j}-1} (-1)^{\beta} \binom{l_{i,j}-1}{\beta} \int_0^{\xi_{l_{i,j}-2}} \dots \int_0^{\eta_2} (\bar{p}_{i,j}^{l_{i,j}-1})^{(\beta)} \bar{u}_j \, d\eta_1 \dots d\eta_\beta \right] d\xi_{l_{i,j}-2} \dots d\xi_{h_i}$$

$$+ \hat{k}^{l_{i,j}-2} \int_0^{x_k} \dots \int_0^{\xi_{l_{i,j}}} \left[\sum_{\beta=0}^{l_{i,j}-2} (-1)^{\beta} \binom{l_{i,j}-2}{\beta} \int_0^{\xi_{l_{i,j}-3}} \dots \int_0^{\eta_2} (\bar{p}_{i,j}^{l_{i,j}-2})^{(\beta)} \bar{u}_j \, d\eta_1 \dots d\eta_\beta \right] d\xi_{l_{i,j}-3} \dots d\xi_{h_i}$$

$$\vdots$$

$$+ \int_0^{x_k} \dots \int_0^{\xi_2} \bar{p}_{i,j}^{0} \bar{u}_j \, d\xi_1 \dots d\xi_{h_i} + \frac{d_{i,1}^{j} x_k^{h_i-1}}{(h_i-1)!} + \frac{d_{i,2}^{j} x_k^{h_i-2}}{(h_i-2)!} + \frac{d_{i,3}^{j} x_k^{h_i-3}}{(h_i-3)!} + \dots + d_{i,h_i}^{j}, \tag{3.5}$$

where $d_{i,1}^j = d_{i,2}^j = d_{i,3}^j = \dots = d_{i,h_i-l_{i,j}}^j = 0$ for $l_{i,j} < h_i$. Here, $d_{i,1}^j, d_{i,2}^j, d_{i,3}^j, \dots, d_{i,h_i}^j$ are any constants emerged in the process of integration of i^{th} equation in (3.2) and $(\bar{p}_{i,j}^{r})^{(\beta)}$ is an β^{th} order derivative of the coefficient function $\bar{p}_{i,j}^{r}(x)$, where $0 \le r \le h_i$, for all $i, j \in \{1, 2, 3, \dots, m\}$.

Step 4. We can transform the equations (3.4) and (3.5) in Step 3 into a matrix form by using the idea described in Chapter 2. Thus, for $h_i = l_{i,j}$, (3.4) can be written in the matrix form as

$$\hat{k}^{h_i} \left[\sum_{\beta=0}^{h_i} (-1)^{\beta} \binom{h_i}{\beta} \mathbf{A}^{\beta} (\mathbf{P}_{i,j}^{h_i})^{(\beta)} \mathbf{u}_j \right]$$

$$+ \hat{k}^{h_i-1} \mathbf{A}^1 \left[\sum_{\beta=0}^{h_i-1} (-1)^{\beta} \binom{h_i-1}{\beta} \mathbf{A}^{\beta} (\mathbf{P}_{i,j}^{h_i-1})^{(\beta)} \mathbf{u}_j \right]$$

$$+ \hat{k}^{h_i-2} \mathbf{A}^2 \left[\sum_{\beta=0}^{h_i-2} (-1)^{\beta} \binom{h_i-2}{\beta} \mathbf{A}^{\beta} (\mathbf{P}_{i,j}^{h_i-2})^{(\beta)} \mathbf{u}_j \right]$$

$$\vdots$$

$$+ \mathbf{A}^{h_i} \mathbf{P}_{i,j}^{0} \mathbf{u}_j + d_{i,1}^{j} \mathbf{x}_{h_i-1} + d_{i,2}^{j} \mathbf{x}_{h_i-2} + d_{i,3}^{j} \mathbf{x}_{h_i-3} + \dots + d_{i,h_i}^{j} \mathbf{x}_0, \tag{3.6}$$

and for $l_{i,j} < h_i$, (3.5) can be written in the matrix form as

$$
\begin{aligned}
&\hat{k}^{l_{i,j}}\mathbf{A}^{h_i-l_{i,j}}\left[\sum_{\beta=0}^{l_{i,j}}(-1)^{\beta}\binom{l_{i,j}}{\beta}\mathbf{A}^{\beta}(\mathbf{P}_{i,j}^{l_{i,j}})^{(\beta)}\mathbf{u}_j\right]\\
&+\hat{k}^{l_{i,j}-1}\mathbf{A}^{h_i-l_{i,j}+1}\left[\sum_{\beta=0}^{l_{i,j}-1}(-1)^{\beta}\binom{l_{i,j}-1}{\beta}\mathbf{A}^{\beta}(\mathbf{P}_{i,j}^{l_{i,j}-1})^{(\beta)}\mathbf{u}_j\right]\\
&+\hat{k}^{l_{i,j}-2}\mathbf{A}^{h_i-l_{i,j}+2}\left[\sum_{\beta=0}^{l_{i,j}-2}(-1)^{\beta}\binom{l_{i,j}-2}{\beta}\mathbf{A}^{\beta}(\mathbf{P}_{i,j}^{l_{i,j}-2})^{(\beta)}\mathbf{u}_j\right]\\
&\qquad\vdots\\
&+\mathbf{A}^{h_i}\mathbf{P}_{i,j}^{0}\mathbf{u}_j+d_{i,1}^{j}\mathbf{x}_{h_i-1}+d_{i,2}^{j}\mathbf{x}_{h_i-2}+d_{i,3}^{j}\mathbf{x}_{h_i-3}+\cdots+d_{i,h_i}^{j}\mathbf{x}_0,
\end{aligned}
\tag{3.7}
$$

where

$d_{i,1}^{j} = d_{i,2}^{j} = d_{i,3}^{j} = \cdots = d_{i,h_i-l_{i,j}}^{j} = 0$ for $l_{i,j} < h_i$,

$(\mathbf{P}_{l_{i,j}}^{k})^{(\beta)} = \operatorname{diag}\left((\bar{p}_{i,j}^{k})^{(\beta)}(x_1), (\bar{p}_{i,j}^{k})^{(\beta)}(x_2), (\bar{p}_{i,j}^{k})^{(\beta)}(x_3), \ldots, (\bar{p}_{i,j}^{k})^{(\beta)}(x_M)\right)$,

$\mathbf{x}_{h_i-l} = \frac{1}{(h_i-l)!}\left[x_1^{h_i-l}, x_2^{h_i-l}, x_3^{h_i-l}, \ldots, x_M^{h_i-l}\right]^{\top}$ for $l \in \{1, 2, 3, \ldots, h_i\}$,

$\mathbf{A} = \overline{\mathbf{T}}^{*}(\mathbf{T}^{*})^{-1}$ as defined in Chapter 2,

$\bar{\mathbf{f}}_i = \left[\bar{f}_i(x_1), \bar{f}_i(x_2), \bar{f}_i(x_3), \ldots, \bar{f}_i(x_M)\right]^{\top}$,

$\mathbf{u}_j = [\bar{u}_j(x_1), \bar{u}_j(x_2), \bar{u}_j(x_3), \ldots, \bar{u}_j(x_M)]^{\top}$.

Simplified the above matrix equations, for $h_i = l_{i,j}$, (3.6) becomes

$$
\begin{aligned}
&\hat{k}^{h_i}\left[\sum_{\beta=0}^{h_i}(-1)^{\beta}\binom{h_i}{\beta}\mathbf{A}^{\beta}(\mathbf{P}_{i,j}^{h_i})^{(\beta)}\mathbf{u}_j\right]\\
&+\hat{k}^{h_i-1}\left[\sum_{\beta=0}^{h_i-1}(-1)^{\beta}\binom{h_i-1}{\beta}\mathbf{A}^{\beta+1}(\mathbf{P}_{i,j}^{h_i-1})^{(\beta)}\mathbf{u}_j\right]\\
&+\hat{k}^{h_i-2}\left[\sum_{\beta=0}^{h_i-2}(-1)^{\beta}\binom{h_i-2}{\beta}\mathbf{A}^{\beta+2}(\mathbf{P}_{i,j}^{h_i-2})^{(\beta)}\mathbf{u}_j\right]\\
&\qquad\vdots\\
&+\mathbf{A}^{h_i}\mathbf{P}_{i,j}^{0}\mathbf{u}_j+d_{i,1}^{j}\mathbf{x}_{h_i-1}+d_{i,2}^{j}\mathbf{x}_{h_i-2}+d_{i,3}^{j}\mathbf{x}_{h_i-3}+\cdots+d_{i,h_i}^{j}\mathbf{x}_0,
\end{aligned}
\tag{3.8}
$$

and for $l_{i,j} < h_i$, (3.7) becomes

$$\begin{aligned}
&\hat{k}^{l_{i,j}}\left[\sum_{\beta=0}^{l_{i,j}}(-1)^{\beta}\binom{l_{i,j}}{\beta}\mathbf{A}^{\beta+(h_i-l_{i,j})}(\mathbf{P}_{i,j}^{l_{i,j}})^{(\beta)}\mathbf{u}_j\right] \\
&+\hat{k}^{l_{i,j}-1}\left[\sum_{\beta=0}^{l_{i,j}-1}(-1)^{\beta}\binom{l_{i,j}-1}{\beta}\mathbf{A}^{\beta+(h_i-l_{i,j}+1)}(\mathbf{P}_{i,j}^{l_{i,j}-1})^{(\beta)}\mathbf{u}_j\right] \\
&+\hat{k}^{l_{i,j}-2}\left[\sum_{\beta=0}^{l_{i,j}-2}(-1)^{\beta}\binom{l_{i,j}-2}{\beta}\mathbf{A}^{\beta+(h_i-l_{i,j}+2)}(\mathbf{P}_{i,j}^{l_{i,j}-2})^{(\beta)}\mathbf{u}_j\right] \\
&\qquad\vdots \\
&+\mathbf{A}^{h_i}\mathbf{P}_{i,j}^{0}\mathbf{u}_j + d_{i,1}^{j}\mathbf{x}_{h_i-1} + d_{i,2}^{j}\mathbf{x}_{h_i-2} + d_{i,3}^{j}\mathbf{x}_{h_i-3} + \cdots + d_{i,h_i}^{j}\mathbf{x}_0.
\end{aligned} \tag{3.9}$$

Next, the right-hand side (RHS) of i^{th} equation in (3.3) can written in the matrix form

$$\mathbf{A}^{h_i}\bar{\mathbf{f}}_i$$

Now, we let

$$\mathbf{K}_{ij} = \sum_{\beta=0}^{l_{i,j}}\left(\hat{k}^{l_{i,j}-\beta}\sum_{k=0}^{l_{i,j}-\beta}(-1)^{k}\binom{l_{i,j}}{\beta}\mathbf{A}^{k+h_i-l_{i,j}+\beta}\left(\mathbf{P}_{i,j}^{l_{i,j}-\beta}\right)^{(k)}\right), \tag{3.10}$$

for all $j \in \{1, 2, 3, \ldots, m\}$. Hence, we can simplify (3.2) in a matrix form

$$\sum_{j=1}^{m}\mathbf{K}_{ij}\mathbf{u}_j + \sum_{k=1}^{h_i}D_{i,k}\mathbf{x}_{h_i-k} = \mathbf{A}^{h_I}\bar{\mathbf{f}}_i, \tag{3.11}$$

where $D_{i,k} = \sum_{j=1}^{m} d_{i,k}^{j}$ for all $k \in \{1, 2, 3, .., h_i\}$ and $i \in \{1, 2, 3, .., m\}$.

Step 5. We write the given boundary conditions which have the number m of conditions at the endpoints $x = 0$ and $x = 1$ into the vector forms by using linear combination (2.11) and Lemma 2.1 (ii). Let $p \in \mathbb{N} \cup \{0\}$ and $i \in \{1, 2, 3, \ldots, m\}$. Then, we obtain

$$\bar{u}_j(0) = \sum_{n=0}^{M-1} c_{n_j} T_n^*(0) = \sum_{n=0}^{M-1} c_{n_j} (-1)^n = \mathbf{t}_{0,l}\mathbf{c}_j = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_j,$$
$$\bar{u}_j(1) = \sum_{n=0}^{M-1} c_{n_j} T_n^*(1) = \sum_{n=0}^{M-1} c_{n_j} (1)^n = \mathbf{t}_{0,r}\mathbf{c}_j = \mathbf{t}_{0,r}(\mathbf{T}^*)^{-1}\mathbf{u}_j,$$

where $\mathbf{t}_{0,l} = \left[1, -1, 1, \ldots, (-1)^{M-1}\right]$ and $\mathbf{t}_{0,r} = \left[1, 1, 1, \ldots, (1)^{M-1}\right]$, and

$$\bar{u}_j^{(p)}(0) = \sum_{n=0}^{M-1} c_{n_j} (T^*)_n^{(p)}(0) = \sum_{n=0}^{M-1} c_{n_j} (-1)^{p+n} \prod_{i=0}^{p-1} \frac{n^2-k^2}{2k+1} = \mathbf{t}_{p,l}\mathbf{c}_j = \mathbf{t}_{p,l}(\mathbf{T}^*)^{-1}\mathbf{u}_j,$$
$$\bar{u}_j^{(p)}(1) = \sum_{n=0}^{M-1} c_{n_j} (T^*)_n^{(p)}(1) = \sum_{n=0}^{M-1} c_{n_j} (1)^{p+n} \prod_{i=0}^{p-1} \frac{n^2-k^2}{2k+1} = \mathbf{t}_{p,r}\mathbf{c}_j = \mathbf{t}_{p,r}(\mathbf{T}^*)^{-1}\mathbf{u}_j,$$

where

$$\mathbf{t}_{p,l} = \begin{bmatrix} (-1)^{p+0}\prod_{k=0}^{p-1}\frac{0^2-k^2}{2k+1} \\ (-1)^{p+1}\prod_{k=0}^{p-1}\frac{1^2-k^2}{2k+1} \\ (-1)^{p+2}\prod_{k=0}^{p-1}\frac{2^2-k^2}{2k+1} \\ \vdots \\ (-1)^{p+M-1}\prod_{k=0}^{p-1}\frac{(M-1)^2-k^2}{2k+1} \end{bmatrix}^\top \quad \text{and} \quad \mathbf{t}_{p,r} = \begin{bmatrix} \prod_{k=0}^{p-1}\frac{0^2-k^2}{2k+1} \\ \prod_{k=0}^{p-1}\frac{1^2-k^2}{2k+1} \\ \prod_{k=0}^{p-1}\frac{2^2-k^2}{2k+1} \\ \vdots \\ \prod_{k=0}^{p-1}\frac{(M-1)^2-k^2}{2k+1} \end{bmatrix}^\top.$$

Note that, for left and right boundary conditions are defined by $\bar{u}_j^{(k)}(0) = \mathbf{t}_{k,l}(\mathbf{T}^*)^{-1}\mathbf{u}_j = b_{k_j}$ and $\bar{u}_j^{(k)}(1) = \mathbf{t}_{k,r}(\mathbf{T}^*)^{-1}\mathbf{u}_j = b_{k_j}$, where $\mathbf{t}_{k,l}$ and $\mathbf{t}_{k,r}$ are the row vector $\mathbf{t}_k$ for $k \in \{0, 1, 2, \ldots, h_i - 1\}$ that their elements are substituted by 0 and 1, respectively.

Let $i, j \in \{1, 2, 3, \ldots, m\}$. We consider the given boundary conditions in terms of $\bar{u}_j^{(p)}(x) = b_{k_j}, x \in \{0, 1\}$ for $p, k \in \{0, 1, 2, \ldots, h_i - 1\}$, where $b_{k_j} \in \mathbb{R}$. Thus, we have

$$\mathbf{t}_0(\mathbf{T}^*)^{-1}\mathbf{u}_j = b_{0_j},$$
$$\mathbf{t}_1(\mathbf{T}^*)^{-1}\mathbf{u}_j = b_{1_j},$$
$$\mathbf{t}_2(\mathbf{T}^*)^{-1}\mathbf{u}_j = b_{2_j},$$
$$\vdots$$
$$\mathbf{t}_{h_i-1}(\mathbf{T}^*)^{-1}\mathbf{u}_j = b_{h_i-1_j}.$$

For $i, j \in \{1, 2, 3, \ldots, m\}$, we can write the above all equations in the matrix form as

$$\mathbf{T}_i(\mathbf{T}^*)^{-1}\mathbf{u}_j = \mathbf{b}_{ij}, \tag{3.12}$$

where $\mathbf{T}_i = [\mathbf{t}_0, \mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_{h_i-1}]^\top$ and $\mathbf{b}_{ij} = \left[b_{0_j}, b_{1_j}, b_{2_j}, \ldots, b_{h_i-1_j}\right]^\top$.

Note that, actually, we need exactly $\sum_{i=0}^{m} h_i$ boundary conditions. In practice, all missing conditions will be replaced by zero.

<u>**Step 6.**</u> We construct a linear system by using the matrix equation (3.11) together with the boundary conditions (3.12). Then, we obtain the linear system in a block matrix form

$$\begin{bmatrix} \mathbf{K}_o & \mathbf{Q} \\ \mathbf{R} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{W} \\ \mathbf{b} \end{bmatrix}, \tag{3.13}$$

where $\mathbf{0}$ is the square zero matrix with size $z := \sum_{i=1}^{m} h_i$,

$$\begin{aligned}
\mathbf{K}_o &= \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} & \cdots & \mathbf{K}_{1m} \\ \mathbf{K}_{21} & \mathbf{K}_{22} & \cdots & \mathbf{K}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{K}_{m1} & \mathbf{K}_{m2} & \cdots & \mathbf{K}_{mm} \end{bmatrix}_{mM \times mM}, \\
\mathbf{Q} &= \begin{bmatrix} \mathbf{x}_{h_1-1} & \cdots & \mathbf{x}_0 & \mathbf{0} & \cdots & \cdots & \cdots & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{x}_{h_2-1} & \cdots & \mathbf{x}_0 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \vdots & \ddots & \cdots & \cdots & \cdots & \mathbf{0} & \ddots & \mathbf{0} & \cdots & \vdots \\ \mathbf{0} & \cdots & \cdots & \cdots & \cdots & \cdots & \mathbf{0} & \mathbf{x}_{h_m-1} & \cdots & \mathbf{x}_0 \end{bmatrix}_{mM \times z}, \\
\mathbf{u} &= [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \ldots, \mathbf{u}_m]^\top, \\
\mathbf{W} &= \left[\mathbf{A}^{h_1}\bar{\mathbf{f}}_1, \mathbf{A}^{h_2}\bar{\mathbf{f}}_2, \mathbf{A}^{h_3}\bar{\mathbf{f}}_3, \ldots, \mathbf{A}^{h_m}\bar{\mathbf{f}}_m\right]^\top, \\
\mathbf{R} &= \left[\mathbf{T}_1(\mathbf{T}^*)^{-1}, \mathbf{T}_2(\mathbf{T}^*)^{-1}, \ldots, \mathbf{T}_M(\mathbf{T}^*)^{-1}\right]^\top_{z \times mM}, \\
\mathbf{b} &= [b_{0_1}, b_{1_1}, \ldots, b_{h_1-1_1}, b_{0_2}, b_{1_2}, \ldots, b_{h_2-1_2}, \ldots, b_{0_m}, b_{1_m}, \ldots, b_{h_m-1_m}]^\top, \\
\mathbf{D} &= [D_{1,1}, D_{1,2}, \ldots, D_{1,h_1}, D_{2,1}, D_{2,2}, \ldots, D_{2,h_2}, D_{m,1}, D_{m,2}, \ldots, D_{m,h_m}]^\top.
\end{aligned}$$

Hence, we can solve the linear system (3.13) to find the approximate solution $\bar{u}_j(\bar{x})$ of the system of m linear ODEs (1.1) for all $j \in \{1, 2, 3, \ldots, m\}$. We assume the $\mathbf{K}_o$ and $\mathbf{R}\mathbf{K}_o^{-1}\mathbf{Q}$ are nonsingular matrices. Thus,

$$\mathbf{u} = \mathbf{K}_o^{-1}\left[\mathbf{W} - \mathbf{Q}\left(\mathbf{R}\mathbf{K}_o^{-1}\mathbf{Q}\right)^{-1}\left(\mathbf{R}\mathbf{K}_o^{-1}\mathbf{W} - \mathbf{b}\right)\right]. \tag{3.14}$$

Finally, we can obtain the approximate solution $u_j(x)$ for $x \in [a, b]$ by using the linear mapping $\bar{x} = \frac{x-a}{b-a}$.

3.2 Numerical Examples of System of Linear ODEs

In this section, we implement numerical examples with MatLab program to find the approximate solutions of some system of m linear ODEs that have been interested in several literature by using our numerical algorithm. For an error of the solutions, we use the absolute error $E = |u_j^*(x) - u_j(x)|$ for all $j \in \{1, 2, 3, \ldots, m\}$, where u_j^* and u_j are respectively the analytical and numerical solutions at each x in the domain. For the first example, we start with a system of linear first order ODEs with constant coefficients.

Example 3.1. Consider the following system of linear first order ODEs over $x \in (0, 1)$

$$u_1'(x) = -u_1(x) + 2u_2(x), \tag{3.15}$$

$$u_2'(x) = 2u_1(x) - u_2(x), \tag{3.16}$$

with the initial conditions $u_1(0) = 3$ and $u_2(0) = 1$. The analytical solutions are $u_1^*(x) = 2e^x + e^{-3x}$ and $u_2^*(x) = 2e^x - e^{-3x}$.

From this problem, we have $f_1(x) = 0$ and $f_2(x) = 0$. By using our numerical procedure described in Section 3.1, we take single-layer integration both sides of (3.15) and (3.16). Then, it can be transformed into a matrix form as

$$\mathbf{u}_1 + \mathbf{A}\mathbf{u}_1 - 2\mathbf{A}\mathbf{u}_2 + D_{1,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_1,$$

$$\mathbf{u}_2 - 2\mathbf{A}\mathbf{u}_1 + \mathbf{A}\mathbf{u}_2 + D_{2,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_2.$$

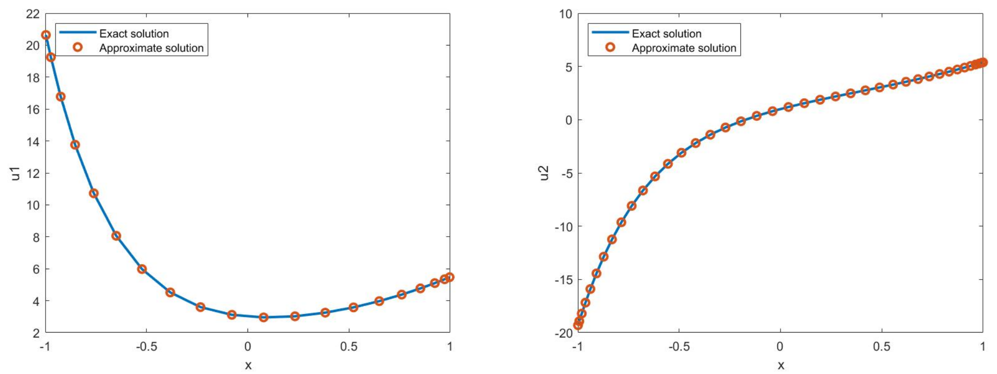
By using the initial conditions, we have $u_1(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 3$ and $u_2(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 1$, where $\mathbf{t}_{0,l} = \left[1, -1, 1, \ldots, (-1)^{M-1}\right]^\top$. Thus, we can construct the linear system in a matrix form as

$$\left[\begin{array}{cc|cc} \mathbf{I}+\mathbf{A} & -2\mathbf{A} & \mathbf{x}_0 & \mathbf{0} \\ -2\mathbf{A} & \mathbf{I}+\mathbf{A} & \mathbf{0} & \mathbf{x}_0 \\ \hline \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{2,1} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}\bar{\mathbf{f}}_1 \\ \mathbf{A}\bar{\mathbf{f}}_2 \\ \hline 3 \\ 1 \end{array}\right]. \tag{3.17}$$

We solve (3.17) to obtain the approximate solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ of (3.15) and (3.16) by taking $M = 15$. By substituting the solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ into (3.14), we can get the approximate solution $u_1(x)$ and $u_2(x)$ for each arbitrary $x \in [0, 1]$. We compare the absolute errors of our approximate solutions $u_1(x)$ and $u_2(x)$ with those obtained by the DTM [24] with $M = 15$ at $x \in \{0.6, 0.7, 0.8, 0.9, 1.0\}$ as shown in Table 3.1. Note that, the absolute errors of our approximate solutions and the solutions from [24] at $x \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$, that computed by MatLab software, provide all zeros for both $u_1(x)$ and $u_2(x)$. Figure 3.1 shows the graphical solutions of our approximate solutions and the exact solutions with $M = 40$. The average run-time is 0.0546 seconds.

Table 3.1: A comparison of absolute errors of $u_1(x)$ for Example 3.1

x_i	DTM [24]	FIM-SCP	DTM [24]	FIM-SCP
0.6	5.0000×10^{-9}	3.0300×10^{-10}	5.0000×10^{-9}	3.0300×10^{-10}
0.7	5.2000×10^{-8}	5.6200×10^{-10}	5.2000×10^{-8}	5.6200×10^{-10}
0.8	3.8600×10^{-7}	7.8000×10^{-10}	3.8600×10^{-7}	7.8000×10^{-10}
0.9	2.2590×10^{-6}	7.3500×10^{-10}	2.2590×10^{-6}	7.3500×10^{-10}
1.0	1.0973×10^{-5}	7.9900×10^{-10}	6.3656×10^{-10}	1.0973×10^{-10}



(a) A graphical solution for $u_1(x)$ **(b)** A graphical solution for $u_2(x)$

Figure 3.1: The graph of the approximate and exact solutions in Example 3.1

The second example is about the stiff system of linear ODEs in which our proposed algorithm also give high accurate results.

Example 3.2. Consider the following stiff system of differential equations over $x \in (0,1)$

$$u_1'(x) = -20u_1(x) - 0.25u_2(x) - 19.75u_3(x), \tag{3.18}$$

$$u_2'(x) = 20u_1(x) - 20.25u_2(x) + 0.25u_3(x), \tag{3.19}$$

$$u_3'(x) = 20u_1(x) - 19.75u_2(x) - 0.25u_3(x), \tag{3.20}$$

with initial conditions $u_1(0) = 1$, $u_2(0) = 0$ and $u_3(0) = -1$. The analytical solutions are

$$\begin{aligned} u_1^*(x) &= \frac{1}{2}\left(e^{-\frac{1}{2}x} + e^{-20x}(\cos(20x) + \sin(20x))\right), \\ u_2^*(x) &= \frac{1}{2}\left(e^{-\frac{1}{2}x} - e^{-20x}(\cos(20x) - \sin(20x))\right), \\ u_3^*(x) &= \frac{1}{2}\left(e^{-\frac{1}{2}x} + e^{-20x}(\cos(20x) - \sin(20x))\right). \end{aligned}$$

From the problem, we have $f_1(x) = 0$, $f_2(x) = 0$ and $f_3(x) = 0$. By using our numerical procedure described in Section 3.1, we take single-layer integration both sides of (3.18), (3.19), and (3.20) and transform it into a matrix form

$$\begin{aligned} \mathbf{u}_1 + 20\mathbf{A}\mathbf{u}_1 + 0.25\mathbf{A}\mathbf{u}_2 + 19.75\mathbf{A}\mathbf{u}_3 + D_{1,1}\mathbf{x}_0 &= \mathbf{A}\bar{\mathbf{f}}_1, \\ -20\mathbf{A}\mathbf{u}_1 + \mathbf{u}_2 + 20.25\mathbf{A}\mathbf{u}_2 - 0.25\mathbf{A}\mathbf{u}_3 + D_{2,1}\mathbf{x}_0 &= \mathbf{A}\bar{\mathbf{f}}_2, \\ -20\mathbf{A}\mathbf{u}_1 + 19.75\mathbf{A}\mathbf{u}_2 + \mathbf{u}_3 + 0.25\mathbf{A}\mathbf{u}_3 + D_{3,1}\mathbf{x}_0 &= \mathbf{A}\bar{\mathbf{f}}_3. \end{aligned}$$

Next, from the given initial conditions can be written as

$$\begin{aligned} u_1(0) &= \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1, \\ u_2(0) &= \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 0, \\ u_3(0) &= \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_3 = -1, \end{aligned}$$

where $\mathbf{t}_{0,l} = \left[1, -1, 1, \dots, (-1)^{M-1}\right]^\top$. Therefore, we can construct the linear system into the matrix form as

$$\left[\begin{array}{ccc|ccc} \mathbf{I}+20\mathbf{A} & 0.25\mathbf{A} & 19.75\mathbf{A} & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} \\ -20\mathbf{A} & \mathbf{I}+20.25\mathbf{A} & -0.25\mathbf{A} & \mathbf{0} & \mathbf{x}_0 & \mathbf{0} \\ -20\mathbf{A} & 19.75\mathbf{A} & \mathbf{I}+0.25\mathbf{A} & \mathbf{0} & \mathbf{0} & \mathbf{x}_0 \\ \hline \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \hline D_{1,1} \\ D_{2,1} \\ D_{3,1} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}\bar{\mathbf{f}}_1 \\ \mathbf{A}\bar{\mathbf{f}}_2 \\ \mathbf{A}\bar{\mathbf{f}}_3 \\ \hline 1 \\ 0 \\ -1 \end{array}\right]. \tag{3.21}$$

We solve (3.21) to obtain the approximate solutions $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_3$ of (3.18), (3.19), and (3.20). Therefore, we can get the approximate solutions $u_1(x)$, $u_2(x)$, and $u_3(x)$ for each arbitrary $x \in [0, 1]$ by substituting the solutions $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_3$ into (3.14). We compare the absolute errors of our approximate solutions with the absolute errors obtained from RK-4 method [25] and DTM [24] by taking $M = 16$ as shown in Tables 3.2, 3.3, and 3.4 corresponding to $u_1(x)$, $u_2(x)$, and $u_3(x)$, respectively. Note that, for $M = 16$ of our FIM-SCP, it corresponds to $N = 16$ in [25] and [24]. Figure 3.2 plots the graphical solutions between our approximate solutions and the analytical solutions with $M = 40$. The average run-time is 0.0562 seconds.

Table 3.2: A comparison of absolute errors of $u_1(x)$ for Example 3.2

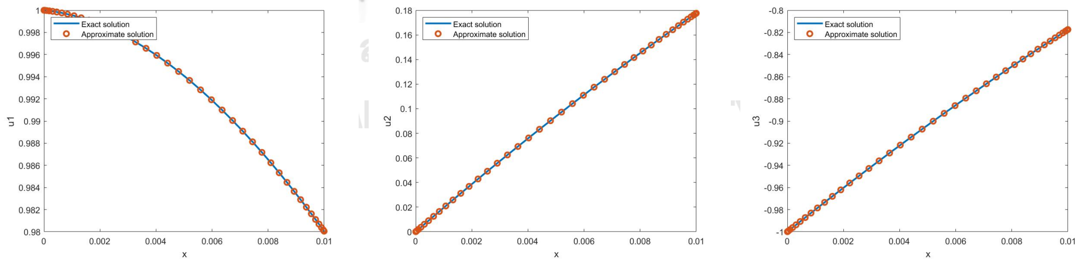
x_i	RK-4 [25]	DTM [24]	FIM-SCP
0.002	1.48800×10^{-11}	2.60000×10^{-13}	6.10622×10^{-15}
0.004	2.96886×10^{-11}	3.32882×10^{-11}	3.55271×10^{-15}
0.006	4.43044×10^{-11}	5.68759×10^{-10}	4.55191×10^{-15}
0.008	5.86098×10^{-11}	4.26088×10^{-9}	1.66534×10^{-15}
0.010	7.25029×10^{-11}	2.03175×10^{-8}	5.5511×10^{-16}

Table 3.3: A comparison of absolute errors of $u_2(x)$ for Example 3.2

x_i	RK-4 [25]	DTM [24]	FIM-SCP
0.002	1.487737×10^{-11}	3.00000×10^{-17}	3.88578×10^{-16}
0.004	2.96886×10^{-11}	5.00000×10^{-17}	1.11022×10^{-16}
0.006	4.43044×10^{-11}	$< 10^{-17}$	7.21645×10^{-16}
0.008	5.86098×10^{-11}	$< 10^{-17}$	5.55111×10^{-17}
0.010	7.25029×10^{-11}	1.00000×10^{-16}	3.05311×10^{-16}

Table 3.4: A comparison of absolute errors of $u_3(x)$ for Example 3.2

x_i	RK-4 [25]	DTM [24]	FIM-SCP
0.002	1.487737×10^{-11}	$< 10^{-17}$	4.55191×10^{-15}
0.004	2.96886×10^{-11}	2.00000×10^{-16}	3.88578×10^{-15}
0.006	4.43044×10^{-11}	$< 10^{-17}$	4.32987×10^{-15}
0.008	5.86098×10^{-11}	$< 10^{-17}$	5.55111×10^{-16}
0.010	7.25029×10^{-11}	2.00000×10^{-16}	3.33067×10^{-16}



(a) A graph of $u_1(x)$ **(b)** A graph of $u_2(x)$ **(c)** A graph of $u_3(x)$

Figure 3.2: The graph of the approximate and exact solutions in Example 3.2

The third example is the stiff system of linear ODEs, given by [26], which demonstrates that our devised method also provides the high accurate results.

Example 3.3. Consider the following stiff system of differential equations

$$u_1'(x) = 998u_1(x) + 1998u_2(x), \tag{3.22}$$

$$u_2'(x) = -999u_1(x) - 1999u_2(x), \tag{3.23}$$

for $x \in (0, 0.001)$ with initial conditions $u_1(0) = 1$ and $u_2(0) = 1$. The analytical solutions are $u_1^*(x) = 4e^{-x} - 3e^{-1000x}$ and $u_2^*(x) = -2e^{-x} + 3e^{-1000x}$.

From the example, we have $f_1(x) = 0$ and $f_2(x) = 0$. By using our numerical procedure described in Section 3.1, we take single-layer integration both sides of (3.22) and (3.23). Then, we transform it into a matrix form as

$$\mathbf{u}_1 - 998\mathbf{A}\mathbf{u}_1 - 1998\mathbf{A}\mathbf{u}_2 + D_{1,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_1,$$

$$999\mathbf{A}\mathbf{u}_1 + \mathbf{u}_2 + 1999\mathbf{A}\mathbf{u}_2 + D_{2,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_2.$$

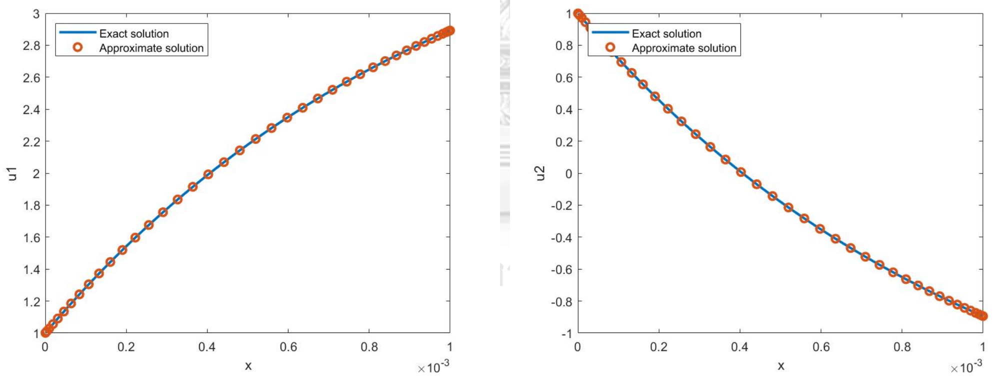
By the given initial conditions, we have $\bar{u}_1(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1$ and $\bar{u}_2(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 1$, where $\mathbf{t}_{0,l}$ is defined as same as Example 3.2. Thus, we can construct the linear system in a matrix form as

$$\left[\begin{array}{cc|cc} \mathbf{I} - 998\mathbf{A} & -1998\mathbf{A} & \mathbf{x}_0 & \mathbf{0} \\ 999\mathbf{A} & \mathbf{I} + 1999\mathbf{A} & \mathbf{0} & \mathbf{x}_0 \\ \hline \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{2,1} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}\bar{\mathbf{f}}_1 \\ \mathbf{A}\bar{\mathbf{f}}_2 \\ \hline 1 \\ 1 \end{array}\right]. \tag{3.24}$$

To obtain the approximate solution $\mathbf{u}_1$ and $\mathbf{u}_2$ of (3.22) and (3.23), we solve (3.24). Hence, we can get the approximate solutions $u_1(x)$ and $u_2(x)$ for each arbitrary $x \in [0, 1]$ by substituting the solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ into (3.14). We compare the absolute error of $u_1(x)$ and $u_2(x)$ with their analytical solutions for $M = 10$ as shown in Tables 3.5. Figure 3.3 shows the graphs of our approximate solutions with $M = 40$. The average run-time is 0.0451 seconds.

Table 3.5: Numerical comparisons of $u_1(x)$ and $u_2(x)$ for Example 3.3

x_i	$u_1(x)$	$u_2(x)$
0.0001	7.8000×10^{-14}	6.9000×10^{-14}
0.0002	1.7420×10^{-12}	1.7390×10^{-12}
0.0003	1.3000×10^{-13}	1.3300×10^{-13}
0.0004	2.6500×10^{-13}	2.6900×10^{-13}
0.0005	1.5340×10^{-12}	1.5390×10^{-12}
0.0006	6.0000×10^{-14}	7.2000×10^{-14}
0.0007	4.8000×10^{-14}	2.8000×10^{-14}
0.0008	1.3660×10^{-12}	1.3800×10^{-12}
0.0009	5.8400×10^{-13}	5.6800×10^{-13}
0.0010	6.0800×10^{-13}	6.1100×10^{-13}



(a) A graphical solution for $u_1(x)$ **(b)** A graphical solution for $u_2(x)$

Figure 3.3: The graph of the approximate and exact solutions in Example 3.3

The last example for this section is a system of linear second order ODEs with variable coefficients.

Example 3.4. Consider the following system of second order ODEs over $x \in (0,1)$

$$u_1''(x) - u_2'(x) + u_3'(x) - e^{-x}u_1(x) + e^{x}u_3(x) = e^{x} - 2e^{-x} + xe^{-x}, \tag{3.25}$$

$$u_2''(x) + u_1'(x) - u_3'(x) - u_1(x) = xe^{-x} - e^{-x}, \tag{3.26}$$

$$u_3''(x) - u_1'(x) + u_2'(x) + e^{x}u_2(x) + u_3(x) = 2e^{-x} - xe^{-x} + x, \tag{3.27}$$

with boundary conditions $u_1(0) = 1$, $u_1(1) = 2.718282$, $u_2(0) = 0$, $u_2(1) = 0.367879$, $u_3(0) = 1$, and $u_3(1) = 0.367879$. The analytical solutions are $u_1^*(x) = e^{x}$, $u_2^*(x) = xe^{-x}$, and $u_3^*(x) = e^{-x}$.

From the example, we have $f_1(x) = e^{x} - 2e^{-x} + xe^{-x}$, $f_2(x) = xe^{-x} - e^{-x}$, $f_3(x) = 2e^{-x} - xe^{-x} + x$, $p_{1,1}^0(x) = -e^{-x}$, $p_{1,3}^0(x) = e^{x}$ and $p_{3,2}^0(x) = e^{x}$. By using our numerical procedure described in Section 3.1, we take double-layer integration both sides of (3.25), (3.26), and (3.27). Then, we transform it into a matrix form as

$$\mathbf{K}_{11}\mathbf{u}_1 + \mathbf{K}_{12}\mathbf{u}_2 + \mathbf{K}_{13}\mathbf{u}_3 + D_{1,1}\mathbf{x}_1 + D_{1,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_1,$$
$$\mathbf{K}_{21}\mathbf{u}_1 + \mathbf{K}_{22}\mathbf{u}_2 + \mathbf{K}_{23}\mathbf{u}_3 + D_{2,1}\mathbf{x}_1 + D_{2,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_2,$$
$$\mathbf{K}_{31}\mathbf{u}_1 + \mathbf{K}_{32}\mathbf{u}_2 + \mathbf{K}_{33}\mathbf{u}_3 + D_{3,1}\mathbf{x}_1 + D_{3,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_3,$$

where

$$\begin{aligned}
&\mathbf{K}_{11} = \mathbf{I} + \mathbf{A}^2\left(\mathbf{P}_{1,1}^0\right)^{(0)}, && \mathbf{K}_{21} = \mathbf{A} - \mathbf{A}^2, && \mathbf{K}_{31} = -\mathbf{A} + \mathbf{A}^2,\\
&\mathbf{K}_{12} = -\mathbf{A}, && \mathbf{K}_{22} = \mathbf{I}, && \mathbf{K}_{32} = \mathbf{A} + \mathbf{A}^2\left(\mathbf{P}_{3,2}^0\right)^{(0)},\\
&\mathbf{K}_{13} = \mathbf{A} + \mathbf{A}^2\left(\mathbf{P}_{1,3}^0\right)^{(0)}, && \mathbf{K}_{23} = -\mathbf{A}, && \mathbf{K}_{33} = \mathbf{I}.
\end{aligned}$$

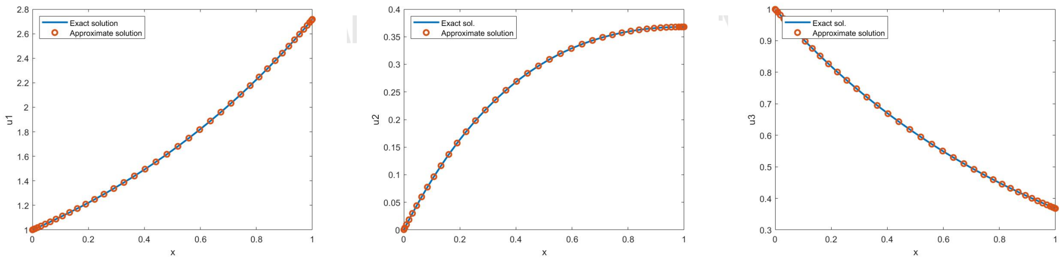
Next, from the given boundary conditions can be written as

$$u_1(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1, \quad u_1(1) = \mathbf{t}_{0,r}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 2.718282,$$
$$u_2(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 0, \quad u_2(1) = \mathbf{t}_{0,r}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 0.367879,$$
$$u_3(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_3 = 1, \quad u_3(1) = \mathbf{t}_{0,r}(\mathbf{T}^*)^{-1}\mathbf{u}_3 = 0.367879,$$

where $\mathbf{t}_{0,r} = \left[1, 1, 1, \ldots, (1)^{M-1}\right]^\top$ and $\mathbf{t}_{0,l}$ is defined as same as Example 3.2. Thus, we can construct the linear system in a matrix form as

$$\left[\begin{array}{ccc|cccccc} \mathbf{K}_{11} & \mathbf{K}_{12} & \mathbf{K}_{13} & \mathbf{x}_1 & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{K}_{21} & \mathbf{K}_{22} & \mathbf{K}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{x}_1 & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{K}_{31} & \mathbf{K}_{32} & \mathbf{K}_{33} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{x}_1 & \mathbf{x}_0 \\ \hline \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{t}_1(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_1(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{t}_1(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \hline D_{1,1} \\ D_{1,2} \\ D_{2,1} \\ D_{2,2} \\ D_{3,1} \\ D_{3,2} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}^2\bar{\mathbf{f}}_1 \\ \mathbf{A}^2\bar{\mathbf{f}}_2 \\ \mathbf{A}^2\bar{\mathbf{f}}_3 \\ \hline 1 \\ 0 \\ 1 \\ 2.718282 \\ 0.367879 \\ 0.367879 \end{array}\right].$$

To obtain the approximate solution $\mathbf{u}_1$, $\mathbf{u}_2$, and and $\mathbf{u}_3$ of (3.25), (3.26), and (3.27), we solve the above equation. Hence, we can get the approximate solutions $u_1(x)$, $u_2(x)$, and $u_3(x)$ for each arbitrary $x \in [0, 1]$ by substituting the solutions $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_3$ into (3.14). We compare the absolute error of $u_1(x)$, $u_2(x)$, and $u_3(x)$ with their analytical solutions for $M = 12$ as shown in Tables 3.6. Figure 3.4 shows the graphs of our approximate solutions with $M = 40$. The average run-time is 0.0598 seconds.



(a) A graph of $u_1(x)$ **(b)** A graph of $u_2(x)$ **(c)** A graph of $u_3(x)$

Figure 3.4: The graph of the approximate and exact solutions in Example 3.4

Table 3.6: Numerical comparisons of $u_1(x)$, $u_2(x)$, and $u_3(x)$ for Example 3.4

x_i	$u_1(x)$	$u_2(x)$	$u_3(x)$
0.0043	3.9968×10^{-15}	2.3999×10^{-15}	1.1102×10^{-15}
0.0381	1.5543×10^{-15}	3.2335×10^{-15}	7.7716×10^{-16}
0.1033	1.3323×10^{-15}	1.7625×10^{-15}	5.5511×10^{-16}
0.1956	2.6645×10^{-15}	3.8858×10^{-15}	9.9920×10^{-16}
0.3087	3.3307×10^{-15}	3.3584×10^{-15}	7.7716×10^{-16}
0.4347	8.4377×10^{-15}	3.1086×10^{-15}	3.3307×10^{-16}
0.5653	6.4393×10^{-15}	3.2196×10^{-15}	5.5511×10^{-16}
0.6913	1.1102×10^{-14}	5.1625×10^{-15}	1.5543×10^{-15}
0.8044	1.7319×10^{-14}	2.7200×10^{-15}	4.9960×10^{-16}
0.8967	1.8652×10^{-14}	2.7756×10^{-15}	5.5511×10^{-16}
0.9619	2.0872×10^{-14}	1.1657×10^{-15}	1.6653×10^{-16}
0.9957	1.7764×10^{-14}	2.6090×10^{-15}	2.1094×10^{-15}

CHAPTER IV

SYSTEMS OF LINEAR IDES

In this chapter, we construct numerical algorithms for solving the system of m linear IDEs which consist of the system of m linear VIDEs (1.3) and the system of m linear FIDEs (1.4) with the given boundary conditions by hiring our proposed FIM-SCP. Finally, we implement our numerical procedures on several numerical examples to demonstrate the efficiency and the accuracy of our method. For (1.3), we compare the absolute errors with Genocchi polynomials method (GPM) [27], single term Walsh series technique (STWS) [28] and bi-orthogonal system (BOS) [29]. For (1.4), we compare absolute error with Tau method (TAU) [16], the collocation method with Bessel polynomials (CM-BP) [17] and the collocation method with Fibonacci polynomials (CM-FP) [18].

4.1 Algorithm for Solving System of linear VIDEs

We first introduce the system of linear m VIDEs with the given boundary conditions which is the problem to be solved by letting $u_j(x)$ be the approximate solution of $v_j(x)$ defined in (2.11), then (1.3) becomes

$$\sum_{j=1}^{m} \mathcal{L}_{i,j} u_j(x) = f_i(x) + \sum_{j=1}^{m} \lambda_{i,j} \int_a^x \mathcal{K}_{i,j}(x,t) u_j(t)\,dt, \quad x \in (a,b) \tag{4.1}$$

with the given boundary conditions $u_j^{(p)}(x_{bd}) = b_i$ for $i, j \in \{1, 2, 3, \ldots, m\}$, where x_{bd} can be the boundary of the interval (a, b), $b_i \in \mathbb{R}$, $p \in \mathbb{N} \cup \{0\}$ and $p \leq m$. We apply the idea of our proposed FIM-SCP described in Chapter 2 to deal with the integration term in (4.1). Then, the numerical procedure for solving (4.1) is formulated. First of all, let us consider each of the integration term in i^{th} equation of (4.1) for $i \in \{1, 2, 3, \ldots, m\}$

which is denoted by

$$J_{i,j}(x) := \int_a^x \mathcal{K}_{i,j}(x,t)u_j(t)\ dt, \quad x \in (a,b) \tag{4.2}$$

for $j \in \{1,2,3,\dots,m\}$. Thus, (4.1) becomes

$$\sum_{j=1}^{m} \mathcal{L}_{i,j}u_j(x) = f_i(x) + \sum_{j=1}^{m} \lambda_{i,j}J_{i,j}(x), \quad x \in (a,b). \tag{4.3}$$

Next, the numerical algorithm for solving systems of linear m VIDEs is devised in the following steps:

Step 1. We use the linear mapping $\bar{x} = \frac{x-a}{b-a}$ to transform $x \in [a,b]$ into $\bar{x} \in [0,1]$. Let $\hat{k} = \frac{1}{b-a}$. Then, (4.1) for $x \in (a,b)$ becomes

$$\sum_{j=1}^{m} \bar{\mathcal{L}}_{i,j}\bar{u}_j(\bar{x}) = \bar{f}_i(\bar{x}) + \frac{1}{\hat{k}}\sum_{j=1}^{m} \lambda_{i,j}\bar{J}_{i,j}(\bar{x}), \quad \bar{x} \in (0,1) \tag{4.4}$$

where $\bar{\mathcal{L}}_{i,j}$, $\bar{D}^k$, $\bar{p}^k_{l_{i,j}}(\bar{x})$, $\bar{u}_j(\bar{x})$, and $\bar{f}_i(\bar{x})$ are defined the same parameters in Step 1 of Section 3.1, $\bar{J}_{i,j}(\bar{x}) = \int_0^{\bar{x}} \bar{\mathcal{K}}_{i,j}(\bar{x},\bar{t})\bar{u}_j(\bar{t})\,d\bar{t}$ and $\bar{\mathcal{K}}_{i,j}(\bar{x},\bar{t}) = \mathcal{K}_{i,j}((b-a)\bar{x}+a,(b-a)\bar{t}+a)$. Henceforth, the problem is considered over $[0,1]$.

Step 2. We discretize our domain $[0,1]$ into M nodes, which are the zeros x_k of shifted Chebyshev polynomial $T^*_M(x)$ defined in (2.2), as described in Step 2 of Section 3.1.

Step 3. We eliminate all derivatives of (4.4) by taking h_i-layer integration from 0 to x_k on both sides of each i^{th} equation in (4.4) and using the technique of integration by parts for all $i \in \{1,2,3,\dots,m\}$, where h_i is defined in Step 3 of Section 3.1 and x_k is defined in (2.2). Thus, for the LHS of i^{th} equation of (4.4), we obtain the integral term similar to the LHS of (3.4) for $l_{i,j} = h_i$ and similar to the LHS of (3.5) for $l_{i,j} < h_i$. Next, the RHS of i^{th} equation in (4.4) becomes

$$\int_0^{x_k} \cdots \int_0^{\xi_2} \bar{f}_i(\xi_1)\,d\xi_1 \dots d\xi_{h_i} + \frac{1}{\hat{k}}\int_0^{x_k} \cdots \int_0^{\xi_2} \sum_{j=1}^{m} \lambda_{i,j}\bar{J}_{i,j}(\xi_1)d\xi_1 \dots d\xi_{h_i}.$$

Step 4. We apply the idea of our proposed FIM-SCP to transform $\bar{J}_{i,j}(x_k)$ for all $k \in \{1,2,3,\ldots,M\}$ into the matrix form. By using the idea of the single-layer integration of u_j from 0 to x_k presented in Chapter 2, we have

$$\begin{aligned}\bar{J}_{i,j}(x_k) &= \int_0^{x_k} \bar{\mathcal{K}}_{i,j}(x_k,\bar{t})\bar{u}_j(\bar{t})d\bar{t} \\ &= \sum_{\beta=1}^{M} a_{k\beta}\bar{\mathcal{K}}_{i,j}(x_k,x_\beta)\bar{u}_j(x_\beta) \\ &= \mathbf{a}_k\bar{K}_{i,j}(x_k)\mathbf{u}_j,\end{aligned}$$

where $\bar{\mathbf{K}}_{i,j}(x_k) = \operatorname{diag}\left(\bar{\mathcal{K}}_{i,j}(x_k,x_1),\bar{\mathcal{K}}_{i,j}(x_k,x_2),\bar{\mathcal{K}}_{i,j}(x_k,x_3),\ldots,\bar{\mathcal{K}}_{i,j}(x_k,x_M)\right)$ and $\mathbf{a}_k = [a_{k1},a_{k2},a_{k3},\ldots,a_{kM}]$. Therefore, we obtain the matrix equation

$$\mathbf{J}_{i,j} = \mathbf{A}'\bar{\mathbf{K}}'_{i,j}\mathbf{u}_j, \tag{4.5}$$

where $\mathbf{J}_{i,j} = \left[\bar{J}_{i,j}(x_1),\bar{J}_{i,j}(x_2),\bar{J}_{i,j}(x_3),\ldots,\bar{J}_{i,j}(x_M)\right]^\top$. $\mathbf{A}'$ and $\bar{\mathbf{K}}'_{i,j}$ are $M \times M^2$ and $M^2 \times M$ matrices, respectively, which can be written by the block matrices as follows:

$$\mathbf{A}' = \begin{bmatrix} \mathbf{a}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{a}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{a}_M \end{bmatrix} \quad \text{and} \quad \bar{\mathbf{K}}'_{i,j} = \begin{bmatrix} \bar{\mathbf{K}}_{i,j}(x_1) \\ \bar{\mathbf{K}}_{i,j}(x_2) \\ \vdots \\ \bar{\mathbf{K}}_{i,j}(x_M) \end{bmatrix}.$$

Note that $\mathbf{A} := [a_{ki}]_{M\times M}$ is the first order shifted Chebyshev integration matrix which is defined in Chapter 2.

Step 5. We transform the LHS of (4.4) presented in Step 3 together with the RHS of (4.4) presented in Steps 3 and 4. Then, it can be simplified into a matrix form. Thus, we obtain the matrix form of the LHS of the i^{th} equation in (4.4) similar to the LHS of (3.8) for $l_{i,j} = h_i$ and the matrix form of the LHS of the i^{th} equation in (4.4) similar to the LHS of (3.9) for $l_{i,j} < h_i$. Next, we change the RHS of i^{th} equation in (4.4) into a matrix form by using (4.5). Then it can be written as

$$\mathbf{A}^{h_i}\bar{\mathbf{f}}_i + \frac{1}{\hat{k}}\mathbf{A}^{h_i}\sum_{j=1}^{m}\lambda_{i,j}\mathbf{J}_{i,j},$$

where $\bar{\mathbf{f}}_i = \left[\bar{f}_i(x_1), \bar{f}_i(x_2), \bar{f}_i(x_3), \dots, \bar{f}_i(x_M)\right]^\top$. Hence, we can simplify (4.4) into the following matrix equation

$$\sum_{j=1}^{m}\mathbf{K}_{ij}\mathbf{u}_j + \sum_{k=1}^{h_i} D_{i,k}\mathbf{x}_{h_i-k} = \mathbf{A}^{h_i}\bar{\mathbf{f}}_i + \frac{1}{\hat{k}}\mathbf{A}^{h_i}\sum_{j=1}^{m}\lambda_{i,j}\mathbf{J}_{i,j}, \tag{4.6}$$

where $\mathbf{K}_{ij}$ and $D_{i,k}$ for all $k \in \{1, 2, 3, \dots, m\}$ and $i \in \{1, 2, 3, \dots, m\}$ are defined the same parameters in Step 4 of Section 3.1. Let us define $\mathbf{H}_{ij} := \frac{1}{\hat{k}}\lambda_{i,j}\mathbf{A}^{h_i}\mathbf{A}'\bar{\mathbf{K}}'_{i,j}$. Then, for all $i \in \{1, 2, 3, \dots, m\}$, (4.6) can be simplified in the form as

$$\sum_{j=1}^{m}(\mathbf{K}_{ij} - \mathbf{H}_{ij})\mathbf{u}_j + \sum_{k=1}^{h_i} D_{i,k}\mathbf{x}_{h_i-k} = \mathbf{A}^{h_i}\bar{\mathbf{f}}_i, \tag{4.7}$$

Step 6. We can obtain the boundary conditions as same as (3.12) described in Step 5 of Section 3.1. After that, we use it and (4.7) to construct the linear system. Then, we obtain the linear system in a block matrix form

$$\begin{bmatrix} \mathbf{K}_v & \mathbf{Q} \\ \mathbf{R} & \mathbf{0} \end{bmatrix}\begin{bmatrix} \mathbf{u} \\ \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{W} \\ \mathbf{b} \end{bmatrix}, \tag{4.8}$$

where $\mathbf{W}$, $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{D}$, $\mathbf{0}$, $\mathbf{u}$ and $\mathbf{b}$ are defined the same in Step 6 of Section 3.1 and

$$\mathbf{K}_v = \begin{bmatrix} \mathbf{K}_{11} - \mathbf{H}_{11} & \mathbf{K}_{12} - \mathbf{H}_{12} & \cdots & \mathbf{K}_{1m} - \mathbf{H}_{1m} \\ \mathbf{K}_{21} - \mathbf{H}_{21} & \mathbf{K}_{22} - \mathbf{H}_{22} & \cdots & \mathbf{K}_{2m} - \mathbf{H}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{K}_{m1} - \mathbf{H}_{m1} & \mathbf{K}_{m2} - \mathbf{H}_{m2} & \cdots & \mathbf{K}_{mm} - \mathbf{H}_{mm} \end{bmatrix}_{mM \times mM}.$$

Hence, we can solve the linear system (4.8) to find the approximate solution $\bar{u}_j(\bar{x})$ of the system (1.3). We assume that $\mathbf{K}_v$ and $\mathbf{R}\mathbf{K}_v^{-1}\mathbf{Q}$ are nonsingular matrices. Thus,

$$\mathbf{u} = \mathbf{K}_v^{-1}\left[\mathbf{W} - \mathbf{Q}\left(\mathbf{R}\mathbf{K}_v^{-1}\mathbf{Q}\right)^{-1}\left(\mathbf{R}\mathbf{K}_v^{-1}\mathbf{W} - \mathbf{b}\right)\right]. \tag{4.9}$$

Finally, we can obtain $u_j(x)$ for $x \in [a, b]$ by using the linear mapping $\bar{x} = \frac{x-a}{b-a}$.

4.2 Numerical Examples of System of Linear VIDEs

In this section, we apply our proposed numerical algorithm to find the approximate solutions of some system of m linear VIDEs. We implement numerical examples with MatLab program base on our numerical algorithm to show the efficiency and effectiveness of our numerical algorithm. For an error of the solutions, we use the absolute error E which defined by $E = |u_j^*(x) - u_j(x)|$ for all $j \in \{1, 2, 3, \ldots, m\}$, where u_j^* and u_j are respectively the analytical solution and the numerical solution at each x in the domain. We start with the first example which is a system of linear first order VIDEs with constant coefficients, constant kernel functions and polynomial forcing terms.

Example 4.1. Consider the following system of linear first order VIDEs over $x \in (0, 1)$

$$u_1'(x) + u_2(x) = 1 + x + x^2 - \int_0^x (u_1(t) + u_2(t))\, dt, \tag{4.10}$$

$$u_2'(x) - u_2(x) = -1 - x - \int_0^x (u_1(t) - u_2(t))\, dt \tag{4.11}$$

subject to the initial conditions $u_1(0) = 1$ and $u_2(0) = -1$. The analytical solutions are $u_1^*(x) = x + e^x$ and $u_2^*(x) = x - e^x$.

In the example, we have $f_1(x) = 1+x+x^2$, $f_2(x) = -1-x$, $\mathcal{K}_{1,1}(x,t) = \mathcal{K}_{1,2}(x,t) = \mathcal{K}_{2,1}(x,t) = \mathcal{K}_{2,2}(x,t) = 1$, $\lambda_{1,1} = \lambda_{1,2} = \lambda_{2,1} = -1$ and $\lambda_{2,2} = 1$. By using our numerical procedure described in Section 4.1, we take one-layer integration both sides of (4.10) and (4.11). Then, we can transform it into the matrix forms:

$$\mathbf{I}\mathbf{u}_1 + \mathbf{A}\mathbf{u}_2 + D_{1,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_1 + \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{1,1}'\mathbf{u}_1 + \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{1,2}'\mathbf{u}_2,$$

$$-\mathbf{A}\mathbf{u}_1 + \mathbf{I}\mathbf{u}_2 + D_{2,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_2 + \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{2,1}'\mathbf{u}_1 - \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{2,2}'\mathbf{u}_2$$

or its simplified form:

$$(\mathbf{K}_{11} - \mathbf{H}_{11})\mathbf{u}_1 + (\mathbf{K}_{12} - \mathbf{H}_{12})\mathbf{u}_2 + D_{1,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_1,$$

$$(\mathbf{K}_{21} - \mathbf{H}_{21})\mathbf{u}_1 + (\mathbf{K}_{22} - \mathbf{H}_{22})\mathbf{u}_2 + D_{2,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_2,$$

where

$$\mathbf{K}_{11}=\mathbf{I},\quad \mathbf{H}_{11}=\mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}'_{1,1},\quad \mathbf{K}_{12}=\mathbf{A},\quad \mathbf{H}_{12}=\mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}'_{1,2},$$
$$\mathbf{K}_{21}=-\mathbf{A},\quad \mathbf{H}_{21}=\mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}'_{2,1},\quad \mathbf{K}_{2,2}=\mathbf{I},\quad \mathbf{H}_{22}=-\mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}'_{2,2}.$$

The given initial conditions can be written in the matrix forms: $u_1(0)=\mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1=1$ and $u_2(0)=\mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2=-1$, where $\mathbf{t}_{0,l}=\left[1,-1,1,\ldots,(-1)^{M-1}\right]$. Thus, we can construct the linear system in the matrix form:

$$\left[\begin{array}{cc|cc} \mathbf{K}_{11}-\mathbf{H}_{11} & \mathbf{K}_{12}-\mathbf{H}_{12} & \mathbf{x}_0 & \mathbf{0} \\ \mathbf{K}_{21}-\mathbf{H}_{21} & \mathbf{K}_{22}-\mathbf{H}_{22} & \mathbf{0} & \mathbf{x}_0 \\ \hline \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} \end{array}\right]\left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{2,1} \end{array}\right]=\left[\begin{array}{c} \mathbf{A}\bar{\mathbf{f}}_1 \\ \mathbf{A}\bar{\mathbf{f}}_2 \\ \hline 1 \\ -1 \end{array}\right]. \tag{4.12}$$

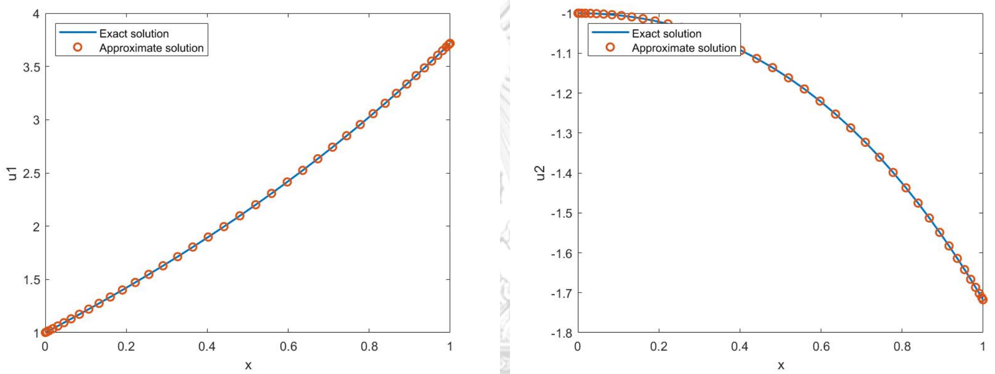
We obtain the approximate solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ of (4.10) and (4.11). After that, by substituting the solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ into (4.9), we can get the approximate solutions $u_1(x)$ and $u_2(x)$ for each arbitrary $x\in[0,1]$. A comparison of the absolute error between the numerical solutions $u_1(x)$ and $u_2(x)$ obtained by our proposed method and the other methods such as the GPM [27] and the BOS [29], with their exact solutions by using $M=8$ as shown in Tables 4.1 and 4.2. With $M=8$, our method corresponds to $N=8$ in [27] and $h=4$, $j=33$ in [29]. Figure 4.1 shows the graphs of the approximate and exact solutions with $M=40$. The average run-time is 0.0437 seconds.

Table 4.1: A comparison of absolute errors of $u_1(x)$ for Example 4.1

x_i	GPM [27]	BOS [29]	FIM-SCP
0.2	1.19266×10^{-8}	4.94774×10^{-8}	9.05715×10^{-10}
0.4	1.31366×10^{-8}	2.72109×10^{-7}	1.79678×10^{-9}
0.6	1.21589×10^{-8}	8.98239×10^{-7}	1.75629×10^{-9}
0.8	1.57033×10^{-8}	3.11105×10^{-7}	1.38063×10^{-9}
1.0	2.57296×10^{-8}	1.50285×10^{-5}	6.36561×10^{-10}

Table 4.2: A comparison of absolute errors of $u_2(x)$ for Example 4.1

x_i	GPM [27]	BOS [29]	FIM-SCP
0.2	7.56814×10^{-9}	3.47816×10^{-6}	4.55619×10^{-10}
0.4	6.17369×10^{-9}	1.51051×10^{-5}	1.01455×10^{-9}
0.6	3.71515×10^{-9}	3.71146×10^{-5}	7.71357×10^{-10}
0.8	2.14741×10^{-8}	7.24787×10^{-5}	3.26418×10^{-10}
1.0	1.95063×10^{-8}	1.24516×10^{-4}	3.58086×10^{-10}



(a) A Graphical solution for $u_1(x)$ **(b)** A Graphical solution for $u_2(x)$

Figure 4.1: The graph of the approximate and exact solutions in Example 4.1

The second example is a system of linear second order VIDEs with variable coefficients, polynomial forcing terms and kernel functions are in term of functions depending on variables x and t.

Example 4.2. Consider the following system of linear second order VIDEs over $x \in (0, 1)$

$$u_1''(x) + (-3x^2 - 6x + 7)u_1(x) - 2x^2(x+1)u_2(x) = x^4 - x^3 - 2x^2 - 6 + \int_0^x (t^3 - x^3)u_1(t)\,dt + \int_0^x x^2(t^2 - x^2)u_2(t)\,dt, \tag{4.13}$$

$$u_2''(x) + 2(x-1)u_1(x) + (2x^4 + 2x^3 + 2x^2 - 1)u_2(x) = x^4 + 3x^3 - 2 + \int_0^x (x^2 - t^2)u_1(t)\,dt - \int_0^x x^2(t^2 + x^2)u_2(t))\,dt \tag{4.14}$$

subject to the initial conditions $u_1(0) = 1$, $u_2(0) = 1$, $u_1'(0) = 1$ and $u_2'(0) = -1$. The analytical solutions are $u_1^*(x) = e^x$ and $u_2^*(x) = e^{-x}$.

From the problem, we have $m = 2$, $f_1(x) = x^4 - x^3 - 2x^2 - 6$, $f_2(x) = x^4 + 3x^3 - 2$, $p_{1,1}^0 = -3x^2 - 6x + 7$, $p_{1,2}^0 = -2x^2(x+1)$, $p_{2,1}^0 = 2(x-1)$, $p_{2,2}^0 = 2x^4 + 2x^3 + 2x^2 - 1$, $\mathcal{K}_{1,1}(x,t) = t^3 - x^3$, $\mathcal{K}_{1,2}(x,t) = x^2(t^2 - x^2)$, $\mathcal{K}_{2,1}(x,t) = x^2 - t^2$, $\mathcal{K}_{2,2}(x,t) = -x^2(t^2 + x^2)$ and $\lambda_{1,1} = \lambda_{1,2} = \lambda_{2,1} = \lambda_{2,2} = 1$.

By using our numerical procedure described in Section 4.1, we take double-layer integration both sides of (4.13) and (4.14), respectively. Then, we can transform them into the matrix forms to obtain

$$\begin{aligned}
&\left(\mathbf{I} + \mathbf{A}^2(\mathbf{P}_{1,1}^0)^{(0)}\right)\mathbf{u}_1 + \mathbf{A}^2(\mathbf{P}_{1,2}^0)^{(0)}\mathbf{u}_2 + D_{1,1}\mathbf{x}_1 + D_{1,2}\mathbf{x}_0 \\
&\qquad\qquad\qquad = \mathbf{A}^2\bar{\mathbf{f}}_1 + \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{1,1}'\mathbf{u}_1 + \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{1,2}'\mathbf{u}_2, \\
&\mathbf{A}^2(\mathbf{P}_{2,1}^0)^{(0)}\mathbf{u}_1 + \left(\mathbf{I} + \mathbf{A}^2(\mathbf{P}_{2,2}^0)^{(0)}\right)\mathbf{u}_2 + D_{2,1}\mathbf{x}_1 + D_{2,2}\mathbf{x}_0 \\
&\qquad\qquad\qquad = \mathbf{A}^2\bar{\mathbf{f}}_1 + \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{2,1}'\mathbf{u}_1 - \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{2,2}'\mathbf{u}_2.
\end{aligned}$$

We rearranged the above equations into the simplified matrix forms:

$$\begin{aligned}
(\mathbf{K}_{11} - \mathbf{H}_{11})\mathbf{u}_1 + (\mathbf{K}_{12} - \mathbf{H}_{12})\mathbf{u}_2 + D_{1,1}\mathbf{x}_1 + D_{1,2}\mathbf{x}_0 &= \mathbf{A}\bar{\mathbf{f}}_1, \\
(\mathbf{K}_{21} - \mathbf{H}_{21})\mathbf{u}_1 + (\mathbf{K}_{22} - \mathbf{H}_{22})\mathbf{u}_2 + D_{2,1}\mathbf{x}_1 + D_{2,2}\mathbf{x}_0 &= \mathbf{A}\bar{\mathbf{f}}_2,
\end{aligned}$$

where

$$\begin{aligned}
\mathbf{K}_{11} &= \mathbf{I} + \mathbf{A}^2(\mathbf{P}_{1,1}^0)^{(0)}, & \mathbf{H}_{11} &= \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{1,1}', \\
\mathbf{K}_{\mathbf{12}} &= \mathbf{A}^2(\mathbf{P}_{1,2}^0)^{(0)}, & \mathbf{H}_{12} &= \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{1,2}', \\
\mathbf{K}_{\mathbf{21}} &= \mathbf{A}^2(\mathbf{P}_{2,1}^0)^{(0)}, & \mathbf{H}_{21} &= \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{2,1}', \\
\mathbf{K}_{22} &= \mathbf{I} + \mathbf{A}^2(\mathbf{P}_{2,2}^0)^{(0)}, & \mathbf{H}_{22} &= -\mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}_{2,2}'.
\end{aligned}$$

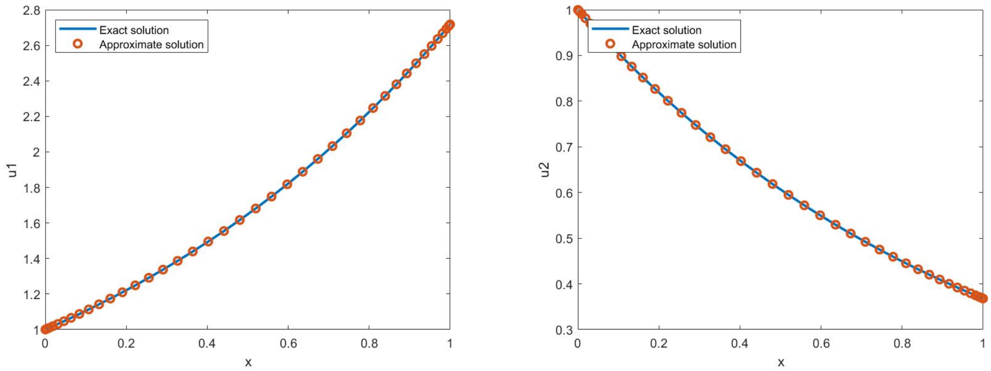
The given initial conditions, we get $u_1(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1$, $u_2(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 1$, $u_1'(0) = \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1$ and $u_2'(0) = \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = -1$, where $\mathbf{t}_{0,l}$ are defined in Example 4.1 and $\mathbf{t}_{1,l} = \left[0, 1, -4, \ldots, (-1)^M(M-1)^2\right]$. Hence, we can construct the linear system in the matrix form:

$$
\left[\begin{array}{cc|cccc}
\mathbf{K}_{11}-\mathbf{H}_{11} & \mathbf{K}_{12}-\mathbf{H}_{12} & \mathbf{x}_1 & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} \\
\mathbf{K}_{21}-\mathbf{H}_{21} & \mathbf{K}_{22}-\mathbf{H}_{22} & \mathbf{0} & \mathbf{0} & \mathbf{x}_1 & \mathbf{x}_0 \\
\hline
\mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & 0 & 0 & 0 & 0 \\
\mathbf{t}_{1,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & 0 & 0 & 0 & 0 \\
\mathbf{0} & \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & 0 & 0 & 0 & 0 \\
\mathbf{0} & \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1} & 0 & 0 & 0 & 0
\end{array}\right]
\left[\begin{array}{c}
\mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{1,2} \\ D_{2,1} \\ D_{2,2}
\end{array}\right]
=
\left[\begin{array}{c}
\mathbf{A}^2\bar{\mathbf{f}}_1 \\ \mathbf{A}^2\bar{\mathbf{f}}_2 \\ \hline 1 \\ 1 \\ 1 \\ -1
\end{array}\right]. \tag{4.15}
$$

Hence, we solve (4.15) to obtain the approximate solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ with $M = 8$. After that by substituting the solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ into (4.9), we can get the approximate solutions $u_1(x)$ and $u_2(x)$ for each arbitrary $x \in [0, 1]$. We compare the absolute errors which are given by our numerical algorithm with the STWS [28] by using $M = 8$ as shown in Table 4.3 together with the graphs between our approximate solutions and the exact solutions with $M = 40$ depicted in Figure 4.2. With $M = 8$, our FIM-SCP corresponds to $m = 200$ by [28]. The average run-time is 0.0880 seconds.

Table 4.3: A comparison of absolute errors of $u_1(x)$ and $u_2(x)$ for Example 4.2

x_i	$u_1(x_i)$		$u_2(x_i)$	
	STWS [28]	FIM-SCP	STWS [28]	FIM-SCP
0.1	3.25×10^{-7}	5.11×10^{-10}	1.64×10^{-7}	2.23×10^{-10}
0.2	8.59×10^{-7}	1.49×10^{-10}	2.25×10^{-7}	2.66×10^{-10}
0.3	1.58×10^{-6}	1.68×10^{-9}	1.59×10^{-7}	9.29×10^{-10}
0.4	2.46×10^{-6}	4.52×10^{-10}	5.95×10^{-8}	4.79×10^{-10}
0.5	3.50×10^{-6}	8.52×10^{-10}	4.60×10^{-7}	2.64×10^{-11}
0.6	4.70×10^{-6}	5.03×10^{-10}	1.06×10^{-6}	5.35×10^{-10}
0.7	6.06×10^{-6}	1.82×10^{-9}	1.87×10^{-6}	9.62×10^{-10}
0.8	7.57×10^{-6}	1.31×10^{-10}	2.85×10^{-6}	3.15×10^{-10}
0.9	9.26×10^{-6}	2.74×10^{-10}	3.87×10^{-6}	3.37×10^{-10}
1.0	1.11×10^{-5}	4.44×10^{-15}	4.69×10^{-6}	2.39×10^{-15}



(a) A graphical solution for $u_1(x)$ **(b)** A graphical solution for $u_2(x)$

Figure 4.2: The graph of the approximate and exact solutions in Example 4.2

The next example is a system of linear second order VIDEs with variable coefficients, constant kernel functions and the forcing terms of trigonometry and exponential functions.

Example 4.3. Consider the following system of linear second order VIDEs over $x \in (0, 1)$

$$u_1''(x) + 2xu_1'(x) - u_1(x) = 2 + x - e^x + 2xe^x - \cos(x) + \int_0^x (u_1(t) - u_2(t))\,dt, \quad (4.16)$$

$$u_2''(x) + u_2'(x) - 2xu_2(x) = -3x - e^x - (1 + 2x)\sin(x) + 2\cos(x) + \int_0^x (u_1(t) + u_2(t))\,dt \quad (4.17)$$

with initial conditions $u_1(0) = u_2(0) = u_1'(0) = u_2'(0) = 1$. The analytical solutions are $u_1^*(x) = e^x$ and $u_2^*(x) = 1 + \sin(x)$.

From the example, we know that $m = 2$, $p^1_{1,1} = 2x$, $p^0_{2,2} = -2x$, $f_1(x) = 2+x-e^x+2xe^x-\cos(x)$, $f_2(x) = -3x-e^x-(1+2x)\sin(x)+2\cos(x)$, $\lambda_{1,1} = \lambda_{1,2} = \lambda_{2,1} = \lambda_{2,2} = 1$.

By using our numerical procedure described in Section 4.1, we take double-layer integration both sides of (4.16) and (4.17), respectively. The problem can be transformed and simplified into the matrix forms as

$$(\mathbf{K}_{11} - \mathbf{H}_{11})\mathbf{u}_1 + (-\mathbf{H}_{12})\mathbf{u}_2 + D_{1,1}\mathbf{x}_1 + D_{1,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_1,$$

$$(-\mathbf{H}_{21})\mathbf{u}_1 + (\mathbf{K}_{22} - \mathbf{H}_{22})\mathbf{u}_2 + D_{2,1}\mathbf{x}_1 + D_{2,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_2,$$

where

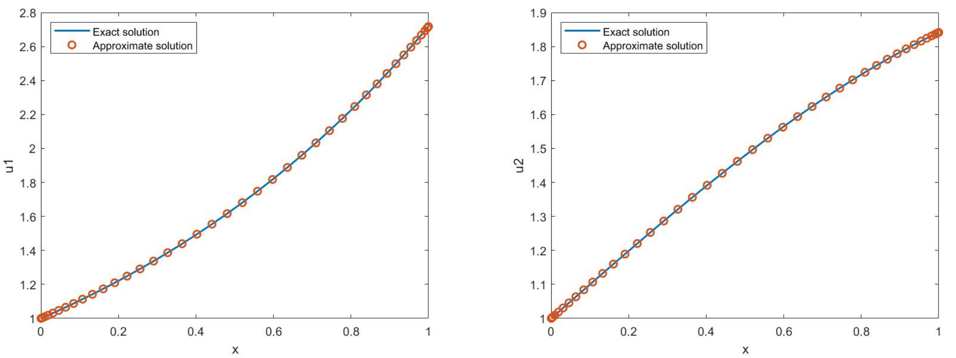
$$\mathbf{K}_{11} = \mathbf{I} + \mathbf{A}(\mathbf{P}^1_{1,1})^{(0)} - \mathbf{A}^2(\mathbf{P}^1_{1,1})^{(1)} + \mathbf{A}^2(\mathbf{P}^0_{1,1})^{(0)},\ \mathbf{K}_{22} = \mathbf{I} + \mathbf{A} + \mathbf{A}^2(\mathbf{P}^0_{2,2})^{(0)},$$

$$\mathbf{H}_{11} = \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}'_{1,1},\ \mathbf{H}_{12} = -\mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}'_{1,2},\ \mathbf{H}_{21} = \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}'_{2,1} \text{ and } \mathbf{H}_{22} = \mathbf{A}^2\mathbf{A}'\bar{\mathbf{K}}'_{2,2}.$$

The given initial conditions can be written in a matrix form as $u_1(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1$, $u_2(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 1$, $u_1'(0) = \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1$ and $u_2'(0) = \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 1$, where $\mathbf{t}_{0,l}$ and $\mathbf{t}_{1,l}$ is defined in Examples 4.1 and 4.2, respectively. Hence, we can construct the linear system in a matrix form as follows

$$\left[\begin{array}{cc|cccc} \mathbf{K}_{11}-\mathbf{H}_{11} & -\mathbf{H}_{12} & \mathbf{x}_1 & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} \\ -\mathbf{H}_{21} & \mathbf{K}_{22}-\mathbf{H}_{22} & \mathbf{0} & \mathbf{0} & \mathbf{x}_1 & \mathbf{x}_0 \\ \hline \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{t}_1(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_1(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{1,2} \\ D_{2,1} \\ D_{2,2} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}^2\bar{\mathbf{f}}_1 \\ \mathbf{A}^2\bar{\mathbf{f}}_2 \\ \hline 1 \\ 1 \\ 1 \\ 1 \end{array}\right]. \qquad (4.18)$$

Hence, we solve (4.18) with $M = 8$ to get $\mathbf{u}_1$ and $\mathbf{u}_2$ of (4.16) and (4.17). To find the approximate solutions $u_1(x)$ and $u_2(x)$ for each arbitrary $x \in [0, 1]$, we substitute $\mathbf{u}_1$ and $\mathbf{u}_2$ into (4.9). Then, we compare our absolute errors with those given by [27] and [28] by taking $M = 8$ as shown in Tables 4.4 and 4.5. Finally, the approximate and exact solutions with $M = 40$ is shown in Figure 4.3. The average run-time is 0.0503 seconds.



(a) A graphical solution for $u_1(x)$ **(b)** A graphical solution for $u_2(x)$

Figure 4.3: The graph of the approximate and exact solutions in Example 4.3

Table 4.4: A comparison of absolute errors of $u_1(x)$ for Example 4.3

x_i	GPM [27]	STWS [28]	FIM-SCP
0.1	9.10139×10^{-10}	2.28×10^{-10}	8.13771×10^{-10}
0.2	1.85461×10^{-9}	4.89×10^{-7}	5.67609×10^{-10}
0.3	3.29800×10^{-9}	7.74×10^{-7}	2.35358×10^{-9}
0.4	1.07693×10^{-8}	1.08×10^{-6}	1.29180×10^{-9}
0.5	2.40393×10^{-8}	1.38×10^{-6}	6.50428×10^{-11}
0.6	3.27914×10^{-8}	1.69×10^{-6}	1.12844×10^{-9}
0.7	2.40101×10^{-8}	2.00×10^{-6}	2.36337×10^{-9}
0.8	5.52469×10^{-9}	2.29×10^{-6}	6.91728×10^{-10}
0.9	4.26837×10^{-8}	2.56×10^{-6}	7.30513×10^{-10}
1.0	6.83253×10^{-8}	2.81×10^{-6}	8.88178×10^{-16}

Table 4.5: A comparison of absolute errors of $u_2(x)$ for Example 4.3

x_i	GPM [27]	STWS [28]	FIM-SCP
0.1	2.22861×10^{-10}	1.79×10^{-7}	1.92190×10^{-10}
0.2	4.35240×10^{-10}	3.09×10^{-7}	1.04064×10^{-10}
0.3	8.04688×10^{-10}	3.99×10^{-7}	6.12261×10^{-10}
0.4	2.97754×10^{-9}	4.58×10^{-7}	2.95272×10^{-10}
0.5	7.13762×10^{-9}	4.91×10^{-7}	1.30359×10^{-10}
0.6	1.02531×10^{-8}	5.03×10^{-7}	2.05141×10^{-10}
0.7	7.59375×10^{-9}	4.96×10^{-7}	6.03731×10^{-10}
0.8	3.32624×10^{-9}	4.72×10^{-7}	1.45898×10^{-10}
0.9	1.93192×10^{-8}	4.31×10^{-7}	1.29673×10^{-10}
1.0	3.35382×10^{-8}	3.70×10^{-7}	4.44089×10^{-16}

The last example for this section is a system of linear first order VIDEs with constant coefficients, variable kernel functions and the forcing terms of polynomials.

Example 4.4. Consider the following system of linear first order VIDEs over $x \in (0,1)$

$$u_1'(x) = 1 + x - \frac{1}{3}x^3 + \int_0^x (x-t)u_1(t)dt + \int_0^x (x-t)u_2(t)dt, \quad (4.19)$$
$$u_2'(x) = 1 - x - \frac{1}{12}x^4 + \int_0^x (x-t)u_1(t)dt - \int_0^x (x-t)u_2(t)dt, \quad (4.20)$$

with boundary conditions $u_1(0) = 1$ and $u_1(1) = 1.5$ The analytical solutions are $u_1^*(x) = x + \frac{1}{2}x^2$ and $u_2^*(x) = x - \frac{1}{2}x^2$.

From the example, we know that $m = 2$, $f_1(x) = 1+x-\frac{1}{3}x^3$, $f_2(x) = 1-x-\frac{1}{12}x^4$, $K_{1,1}(x,t) = K_{1,2}(x,t) = K_{2,1}(x,t) = x-t$, $K_{2,2}(x,t) = -(x-t)$, and $\lambda_{1,1} = \lambda_{1,2} = \lambda_{2,1} = \lambda_{2,2} = 1$.

By using our numerical procedure described in Section 4.1, we take single-layer integration both sides of (4.19) and (4.20), respectively. The problem can be transformed and simplified into the matrix forms

$$(\mathbf{K}_{11} - \mathbf{H}_{11})\mathbf{u}_1 + (-\mathbf{H}_{12})\mathbf{u}_2 + D_{1,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_1,$$
$$(-\mathbf{H}_{21})\mathbf{u}_1 + (\mathbf{K}_{22} - \mathbf{H}_{22})\mathbf{u}_2 + D_{2,1}\mathbf{x}_0 = \mathbf{A}\bar{\mathbf{f}}_2,$$

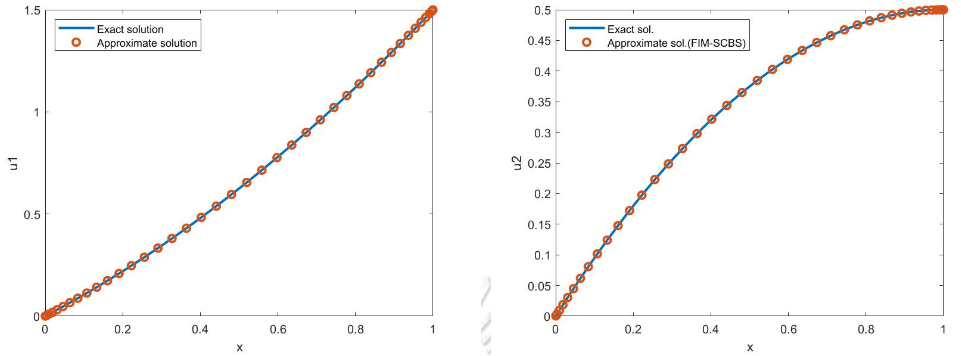
where

$$\mathbf{K}_{11} = \mathbf{I},\ \mathbf{K}_{22} = \mathbf{I},\ \mathbf{H}_{11} = \mathbf{A}\mathbf{A}'\bar{\mathbf{K}}_{1,1}',\ \mathbf{H}_{12} = \mathbf{A}\mathbf{A}'\bar{\mathbf{K}}_{1,2}',\ \mathbf{H}_{21} = \mathbf{A}\mathbf{A}'\bar{\mathbf{K}}_{2,1}',\ \mathbf{H}_{22} = -\mathbf{A}\mathbf{A}'\bar{\mathbf{K}}_{2,2}'.$$

The given boundary conditions can be written in the matrix forms as follow $u_1(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1$, and $u_1(1) = \mathbf{t}_{0,r}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1.5$, where $\mathbf{t}_{0,l}$ and $\mathbf{t}_{0,r}$ is defined in Examples 4.1 and 4.2, respectively. Hence, we can construct the linear system in a matrix form as follows

$$\left[\begin{array}{cc|cc} \mathbf{K}_{11} - \mathbf{H}_{11} & -\mathbf{H}_{12} & \mathbf{x}_0 & \mathbf{0} \\ -\mathbf{H}_{21} & \mathbf{K}_{22} - \mathbf{H}_{22} & \mathbf{0} & \mathbf{x}_0 \\ \hline \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{2,1} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}\bar{\mathbf{f}}_1 \\ \mathbf{A}\bar{\mathbf{f}}_2 \\ \hline 1 \\ 1.5 \end{array}\right]. \quad (4.21)$$

Hence, we compute (4.21) with $M = 10$ to get $\mathbf{u}_1$ and $\mathbf{u}_2$ of (4.19) and (4.20). To find the approximate solutions $u_1(x)$ and $u_2(x)$ for each arbitrary $x \in [0, 1]$, we substitute $\mathbf{u}_1$ and $\mathbf{u}_2$ into (4.9).



(a) A graphical solution for $u_1(x)$ **(b)** A graphical solution for $u_2(x)$

Figure 4.4: The graph of the approximate and exact solutions in Example 4.4

Then, we compare our absolute errors with the analytical solutions by taking $M = 10$ as shown in Tables 4.6. Finally, the approximate and exact solutions with $M = 40$ is shown in Figure 4.4. The average run-time is 0.0889 seconds.

Table 4.6: Numerical comparisons of $u_1(x)$ and $u_2(x)$ for Example 4.4

x_i	$u_1(x)$	$u_2(x)$
0.0062	5.3949×10^{-16}	2.0144×10^{-14}
0.0545	6.5919×10^{-16}	2.1233×10^{-14}
0.1464	5.2736×10^{-16}	1.9651×10^{-14}
0.2730	6.1062×10^{-16}	1.8874×10^{-14}
0.4218	2.2204×10^{-16}	2.0872×10^{-14}
0.5782	1.1102×10^{-16}	1.9040×10^{-14}
0.7269	8.8818×10^{-16}	1.7097×10^{-14}
0.8536	8.8818×10^{-16}	1.7652×10^{-14}
0.9455	2.4425×10^{-15}	1.9706×10^{-14}
0.9938	2.2204×10^{-15}	1.8541×10^{-14}

4.3 Algorithm for Solving System of linear FIDEs

In this section, we can devise a numerical algorithm for solving a system of linear m FIDEs with the given boundary conditions which is the problem to solved by letting u_j to be the approximate solution of v_j as defined in (2.11), then (1.4) becomes

$$\sum_{j=1}^{m} \mathcal{L}_{i,j} u_j(x) = f_i(x) + \sum_{j=1}^{m} \lambda_{i,j} \int_a^b \mathcal{K}_{i,j}(x,t) u_j(t)\, dt, \quad x \in (a,b) \tag{4.22}$$

with the given boundary conditions $u_j^{(p)}(x_bd) = b_i$ for $i \in \{1,2,3,\ldots,m\}$, where x_{bd} can be the boundary of the interval (a,b), $b_i \in \mathbb{R}$, $p \in \mathbb{N} \cup \{0\}$ and $p \leq m$. Then, we apply the idea of Chapter 2 to formulate the procedure for solving (4.22). Similarly to the system of m linear VIDEs, let us consider each of the integration term in i^{th} equation of (4.22) for $i \in \{1,2,3,\ldots,m\}$ which is denoted by

$$G_{i,j}(x) := \int_a^b \mathcal{K}_{i,j}(x,t) u_j(t)\, dt, \tag{4.23}$$

for $j \in \{1,2,3,\ldots,m\}$. Thus, for all $i \in \{1,2,3,\ldots,m\}$, (4.22) becomes

$$\sum_{j=1}^{m} \mathcal{L}_{i,j} u_j(x) = f_i(x) + \sum_{j=1}^{m} \lambda_{i,j} G_{i,j}(x), \quad x \in (a,b). \tag{4.24}$$

We construct the numerical procedure for finding approximate solutions of the system of m linear FIDEs. Steps 1 to 3 of the procedure for solving the system of linear VIDEs as described in Section 4.1 can be used to construct an algorithm for solving the system of m linear FIDEs. The numerical algorithm is devised by the following steps:

Step 1. We use the linear mapping $\bar{x} = \frac{x-a}{b-a}$ to transform $x \in [a,b]$ into $\bar{x} \in [0,1]$. Let $\hat{k} = \frac{1}{b-a}$. Then, (4.24) for $x \in (a,b)$ becomes

$$\sum_{j=1}^{m} \bar{\mathcal{L}}_{i,j} \bar{u}_j(\bar{x}) = \bar{f}_i(\bar{x}) + \frac{1}{\hat{k}} \sum_{j=1}^{m} \lambda_{i,j} \bar{G}_{i,j}(\bar{x}), \quad \bar{x} \in (0,1) \tag{4.25}$$

where $\bar{\mathcal{L}}_{i,j}$, $\bar{u}_j(\bar{x})$ and $\bar{f}_i(\bar{x})$ are defined in Step 1 of Section 3.1, $\bar{G}_{i,j}(\bar{x}) = \int_0^1 \bar{\mathcal{K}}_{i,j}(\bar{x},\bar{t}) \bar{u}_j(\bar{t})\, d\bar{t}$

and $\bar{\mathcal{K}}_{i,j}(\bar{x},\bar{t}) = \mathcal{K}_{i,j}((b-a)\bar{x}+a,(b-a)\bar{t}+a)$.

Step 2. We mesh our domain $[0,1]$ into M nodes as described in Step 2 of Section 3.1.

Step 3. We eliminate all derivatives from (4.25) by taking h_i-layer integration from 0 to x_k on both sides of each i^{th} equation in (4.25) and using the technique of integration by parts for all $i \in \{1,2,3,\dots,m\}$, where h_i is defined in Step 3 of Section 3.1 and x_k is the zeros of the shifted Chebyshev polynomials described in (2.2). Thus, for the LHS of i^{th} equation of (4.25), we obtain the integral term similar to the LHS of (3.4) for $l_{i,j} = h_i$ and similar to the LHS of (3.5) for $l_{i,j} < h_i$. Next, the RHS of i^{th} equation in (4.25) becomes

$$\int_0^{x_k} \cdots \int_0^{\xi_2} \bar{f}_i(\xi_1)\, d\xi_1 \dots d\xi_{h_i} + \frac{1}{\hat{k}} \int_0^{x_k} \cdots \int_0^{\xi_2} \sum_{j=1}^{m} \lambda_{i,j} \bar{G}_{i,j}(\xi_1) d\xi_1 \dots d\xi_{h_i}.$$

Step 4. We apply the idea of the single-layer integration of u_j from 0 to 1 described by (2.14) in Section 2.2 to transform $\bar{G}_{i,j}(x_k)$ for all $k \in \{1,2,3,\dots,m\}$ into the matrix form. Thus, we can get

$$\bar{G}_{i,j}(x_k) = \int_0^1 \bar{\mathcal{K}}_{i,j}(x_k,\bar{t})\bar{u}_j(\bar{t})d\bar{t} = \mathbf{b}(\mathbf{T}^*)^{-1}\bar{\mathbf{K}}_{i,j}(x_k)\mathbf{u}_j := \mathbf{B}\bar{\mathbf{K}}_{i,j}(x_k)\mathbf{u}_j,$$

where $\mathbf{B} = \mathbf{b}(\mathbf{T}^*)^{-1}$, $\mathbf{b} = \left[\overline{T}_0^*(1), \overline{T}_1^*(1), \overline{T}_2^*(1), \dots, \overline{T}_{M-1}^*(1)\right]$ for each entry can be found in (2.8), $\bar{\mathbf{K}}_{i,j}(x_k) = \operatorname{diag}(\bar{\mathcal{K}}_{i,j}(x_k,x_1), \bar{\mathcal{K}}_{i,j}(x_k,x_2), \bar{\mathcal{K}}_{i,j}(x_k,x_3), \dots, \bar{\mathcal{K}}_{i,j}(x_k,x_M))$ and $\mathbf{u}_j = [\bar{u}_j(x_1), \bar{u}_j(x_2), \bar{u}_j(x_3), \dots, \bar{u}_j(x_M)]^\top$.

Next, we vary each x_k for $k \in \{1,2,3,\dots,M\}$ to transform $\bar{G}_{i,j}(x_k)$ into the following matrix equation as

$$\begin{bmatrix} \bar{G}_{i,j}(x_1) \\ \bar{G}_{i,j}(x_2) \\ \vdots \\ \bar{G}_{i,j}(x_M) \end{bmatrix}_{1\times M} = \begin{bmatrix} \mathbf{B} & 0 & \cdots & 0 \\ 0 & \mathbf{B} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{B} \end{bmatrix}_{M\times M^2} \begin{bmatrix} \bar{\mathbf{K}}_{i,j}(x_1) \\ \bar{\mathbf{K}}_{i,j}(x_2) \\ \vdots \\ \bar{\mathbf{K}}_{i,j}(x_M) \end{bmatrix}_{M^2\times M} \begin{bmatrix} u_j(x_1) \\ u_j(x_2) \\ \vdots \\ u_j(x_M) \end{bmatrix}_{M\times 1},$$

that is denoted to the simplified form:

$$\bar{\mathbf{G}}_{i,j} = \mathbf{B}'\bar{\mathbf{K}}'_{i,j}\mathbf{u}_j. \tag{4.26}$$

Step 5. We transform the LHS of (4.25) presented in Step 3 together with the RHS of (4.25) presented in Steps 3 and 4 and simplify it into a matrix form by using the idea of FIM-SCP described in Chapter 2. Thus, we obtain the matrix form of the LHS of the i^{th} equation in (4.25) similar to the LHS of (3.8) for $l_{i,j} = h_i$ and the matrix form of the LHS of the i^{th} equation in (4.25) similar to the LHS of (3.9) for $l_{i,j} < h_i$.

Next, we change the RHS of i^{th} equation in (4.25) into the matrix form by applying (4.26) similar to the idea described in Step 5 of Section 4.1. Then, it can be written as

$$\mathbf{A}^{h_i}\bar{\mathbf{f}}_i + \frac{1}{\hat{k}}\mathbf{A}^{h_i}\sum_{j=1}^{m}\lambda_{i,j}\bar{\mathbf{G}}_{i,j} = \mathbf{A}^{h_i}\bar{\mathbf{f}}_i + \frac{1}{\hat{k}}\mathbf{A}^{h_i}\lambda_{i,j}\sum_{j=1}^{m}\mathbf{B}'\bar{\mathbf{K}}'_{i,j}\mathbf{u}_j,$$

where $\bar{\mathbf{f}}_i = \left[\bar{f}_i(x_1), \bar{f}_i(x_2), \bar{f}_i(x_3), \ldots, \bar{f}_i(x_M)\right]^\top$. Hence, we can simplify (4.25) in the following matrix equation

$$\sum_{j=1}^{m}\mathbf{K}_{ij}\mathbf{u}_j + \sum_{k=1}^{h_i}D_{i,k}\mathbf{x}_{h_i-k} = \mathbf{A}^{h_i}\bar{\mathbf{f}}_i + \frac{1}{\hat{k}}\mathbf{A}^{h_i}\lambda_{i,j}\sum_{j=1}^{m}\mathbf{B}'\bar{\mathbf{K}}'_{i,j}\mathbf{u}_j, \tag{4.27}$$

where $\mathbf{K}_{ij}$ and $D_{i,k}$ for all $k \in \{1, 2, 3, \ldots, m\}$ and $i \in \{1, 2, 3, \ldots, m\}$ are defined in Step 4 of Section 3.1. Let us define $\mathbf{H}'_{ij} := \frac{1}{\hat{k}}\lambda_{i,j}\mathbf{A}^{h_i}\mathbf{B}'\bar{\mathbf{K}}'_{i,j}$. Consequently, (4.27) can be simplified in the form as

$$\sum_{j=1}^{m}(\mathbf{K}_{ij} - \mathbf{H}'_{ij})\mathbf{u}_j + \sum_{k=1}^{h_i}D_{i,k}\mathbf{x}_{h_i-k} = \mathbf{A}^{h_i}\bar{\mathbf{f}}_i, \tag{4.28}$$

for all $i \in \{1, 2, 3, \ldots, m\}$.

Step 6. We can obtain the boundary conditions as same as (3.12) presented in Step 5 of Section 3.1. After that, we use it and (4.28) to construct the linear system. Then, we obtain the linear system in a block matrix form

$$\begin{bmatrix} \mathbf{K}_f & \mathbf{Q} \\ \mathbf{R} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{W} \\ \mathbf{b} \end{bmatrix}, \tag{4.29}$$

where $\mathbf{W}$, $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{D}$, $\mathbf{0}$, $\mathbf{u}$ and $\mathbf{b}$ are defined the same in Step 6 of Section 3.1 and

$$\mathbf{K}_f = \begin{bmatrix} \mathbf{K}_{11} - \mathbf{H}'_{11} & \mathbf{K}_{12} - \mathbf{H}'_{12} & \cdots & \mathbf{K}_{1m} - \mathbf{H}'_{1m} \\ \mathbf{K}_{21} - \mathbf{H}'_{21} & \mathbf{K}_{22} - \mathbf{H}'_{22} & \cdots & \mathbf{K}_{2m} - \mathbf{H}'_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{K}_{m1} - \mathbf{H}'_{m1} & \mathbf{K}_{m2} - \mathbf{H}'_{m2} & \cdots & \mathbf{K}_{mm} - \mathbf{H}'_{mm} \end{bmatrix}_{mM \times mM}.$$

Hence, we can solve the linear system (4.29) to find the approximate solutions $\bar{u}_j(\bar{x})$ of the system of linear m FIDEs (1.4) for all $j \in \{1, 2, 3, \ldots, m\}$. We assume that $\mathbf{K}_f$ and $\mathbf{R}\mathbf{K}_f^{-1}\mathbf{Q}$ are nonsingular matrices. Thus,

$$\mathbf{u} = \mathbf{K}_f^{-1} \left[\mathbf{W} - \mathbf{Q} \left(\mathbf{R}\mathbf{K}_f^{-1}\mathbf{Q} \right)^{-1} \left(\mathbf{R}\mathbf{K}_f^{-1}\mathbf{W} - \mathbf{b} \right) \right]. \tag{4.30}$$

Finally, we can obtain the approximate solutions $u_j(x)$ for $x \in [a, b]$ by using the linear mapping $\bar{x} = \frac{x-a}{b-a}$.

4.4 Numerical Examples of System of Linear FIDEs

In the section, we implement numerical examples to find the approximate solutions of some system of m linear FIDEs by using our proposed method with Matlab program. We compare our results with the analytical solution to show the efficiency of our numerical algorithm. For an error of the solutions, we use the absolute error E which defined by $E = |u_j^*(x) - u_j(x)|$ for all $j \in \{1, 2, 3, \ldots, m\}$, where u_j^* and u_j are respectively the analytical and numerical solution at each x in the domain.

We start with the first example which is a system of linear second order FIDEs with constant coefficients, kernel functions and the forcing terms are in terms of trigonometry and exponential functions.

Example 4.5. Consider the following system of linear second order FIDEs for $x \in (0, \pi)$

$$u_1''(x) + u_2'(x) = 2(e^x - \sin(x)) - \int_0^\pi e^x(u_1(t) - u_2(t))\,dt, \tag{4.31}$$

$$2u_1'(x) + u_2''(x) = (1 + \frac{\pi}{2})\cos(x) - \frac{\pi}{2}\sin(x) - \int_0^\pi \cos(x+t)(u_1(t) + u_2(t))\,dt \tag{4.32}$$

with the boundary conditions $u_1(0) + u_1'(0) = 1$, $u_1(\pi) + u_1'(\pi) = -1$, $u_2(0) + u_2'(0) = 1$ and $u_2(\pi) + u_2'(\pi) = -1$. The exact solutions are $u_1^*(x) = \sin(x)$ and $u_2^*(x) = \cos(x)$.

From the problem, we have $f_1(x) = 2(e^x - \sin(x))$, $f_2(x) = (1+\frac{\pi}{2})\cos(x) - \frac{\pi}{2}\sin(x)$, $\mathcal{K}_{1,1}(x,t) = -e^x$, $\mathcal{K}_{1,2}(x,t) = -e^x$, $\mathcal{K}_{2,1}(x,t) = -\cos(x+t)$, $\mathcal{K}_{2,2}(x,t) = -\cos(x+t)$ and $\lambda_{1,1} = \lambda_{1,2} = \lambda_{2,1} = \lambda_{2,2} = 1$. First, we transform $x \in [0, \pi]$ into $\bar{x} \in [0, 1]$ by using $\bar{x} = \frac{x}{\pi}$. Let $\hat{k} = \frac{1}{\pi}$. Then, we obtain

$$\frac{1}{\pi^2}u_1''(\bar{x}) + \frac{1}{\pi}u_2'(\bar{x}) = \bar{f}_1(\bar{x}) + \pi\int_0^1 \bar{\mathcal{K}}_{1,1}(\bar{x},\bar{t})u_1(\bar{t}) + \pi\int_0^1 \bar{\mathcal{K}}_{1,2}(\bar{x},\bar{t})u_2(\bar{t})\,dt,$$
$$\frac{2}{\pi}u_1'(\bar{x}) + \frac{1}{\pi^2}u_2''(\bar{x}) = \bar{f}_2(\bar{x}) + \pi\int_0^1 \bar{\mathcal{K}}_{2,1}(\bar{x},\bar{t})u_1(\bar{t}) + \pi\int_0^1 \bar{\mathcal{K}}_{2,2}(\bar{x},\bar{t})u_2(\bar{t})\,dt,$$

where $\bar{f}_1(\bar{x}) = 2(e^{\pi\bar{x}} - \sin(\pi\bar{x}))$, $\bar{f}_2(\bar{x}) = (1 + \frac{\pi}{2})\cos(\pi\bar{x}) - \frac{\pi}{2}\sin(\pi\bar{x})$, $\bar{\mathcal{K}}_{1,1}(\bar{x},\bar{t}) = -e^{\pi\bar{x}}$, $\bar{\mathcal{K}}_{1,2}(\bar{x},\bar{t}) = -e^{\pi\bar{x}}$, $\bar{\mathcal{K}}_{2,1}(\bar{x},\bar{t}) = -\cos(\pi\bar{x}+\pi\bar{t})$ and $\bar{\mathcal{K}}_{2,2}(\bar{x},\bar{t}) = -\cos(\pi\bar{x}+\pi\bar{t})$. The exact solutions become $u_1^*(\bar{x}) = \sin(\pi\bar{x})$ and $u_2^*(\bar{x}) = \cos(\pi\bar{x})$.

Next, we take double-layer integration both sides of (4.31) and (4.32) and transform its into the matrix form by using our numerical procedure described in Section 4.3. Then, we rearrange its into a simplified matrix form

$$(\mathbf{K}_{11} - \mathbf{H}'_{11})\mathbf{u}_1 + (\mathbf{K}_{12} - \mathbf{H}'_{12})\mathbf{u}_2 + D_{1,1}\mathbf{x}_1 + D_{1,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_1$$
$$(\mathbf{K}_{21} - \mathbf{H}'_{21})\mathbf{u}_1 + (\mathbf{K}_{22} - \mathbf{H}'_{22})\mathbf{u}_2 + D_{2,1}\mathbf{x}_1 + D_{2,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_2,$$

where

$$\mathbf{K}_{11} = \tfrac{1}{\pi^2}\mathbf{I}, \quad \mathbf{H}'_{11} = \pi\mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}'_{1,1}, \quad \mathbf{K}_{12} = \tfrac{1}{\pi^2}\mathbf{A}, \quad \mathbf{H}'_{12} = \pi\mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}'_{2,1},$$
$$\mathbf{K}_{21} = \tfrac{2}{\pi}\mathbf{A}, \quad \mathbf{H}'_{21} = \pi\mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}'_{2,1}, \quad \mathbf{K}_{22} = \tfrac{1}{\pi^2}\mathbf{I}, \quad \mathbf{H}'_{22} = \pi\mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}'_{2,2}.$$

The given boundary conditions can be written to the matrix forms:

$$u_1(0) + u_1'(0) = (\mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} + \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1})\mathbf{u}_1 = 1,$$
$$u_1(\pi) + u_1'(\pi) = (\mathbf{t}_{0,r}(\mathbf{T}^*)^{-1} + \mathbf{t}_{1,r}(\mathbf{T}^*)^{-1})\mathbf{u}_1 = -1,$$
$$u_2(0) + u_2'(0) = (\mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} + \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1})\mathbf{u}_2 = 1,$$
$$u_2(\pi) + u_2'(\pi) = (\mathbf{t}_{0,r}(\mathbf{T}^*)^{-1} + \mathbf{t}_{1,r}(\mathbf{T}^*)^{-1})\mathbf{u}_2 = -1,$$

where

$$\mathbf{t}_{0,l} = \left[1, -1, 1, \ldots, (-1)^{M-1}\right], \quad \mathbf{t}_{1,l} = \left[0, 1, -4, \ldots, (-1)^{M}(M-1)^2\right],$$
$$\mathbf{t}_{0,r} = \left[1, 1, 1, \ldots, 1^{M-1}\right] \text{ and } \quad \mathbf{t}_{1,r} = \left[0, 1, 4, \ldots, (M-1)^2\right].$$

Therefore, we construct a linear system in a matrix form as follows

$$\left[\begin{array}{cc|cccc} \mathbf{K}_{11}-\mathbf{H}'_{11} & \mathbf{K}_{12}-\mathbf{H}'_{12} & \mathbf{x}_1 & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{K}_{21}-\mathbf{H}'_{21} & \mathbf{K}_{22}-\mathbf{H}'_{22} & \mathbf{0} & \mathbf{0} & \mathbf{x}_1 & \mathbf{x}_0 \\ \hline (\mathbf{t}_{0,l}+\mathbf{t}_{1,l})(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ (\mathbf{t}_{0,r}+\mathbf{t}_{1,r})(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (\mathbf{t}_{0,l}+\mathbf{t}_{1,l})(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (\mathbf{t}_{0,l}+\mathbf{t}_{1,l})(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{1,2} \\ D_{2,1} \\ D_{2,2} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}^2\bar{\mathbf{f}}_1 \\ \mathbf{A}^2\bar{\mathbf{f}}_2 \\ \hline 1 \\ -1 \\ 1 \\ -1 \end{array}\right].$$

After solving the above matrix equation, we can obtain the approximate solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ of (4.31) and (4.32) and take these equations to (4.30), then we get the approximate solutions $u_1(x)$ and $u_2(x)$ for each arbitrary $x \in [0, 1]$. A comparison of the absolute errors of our proposed method with the TAU [16] by using $M = 5$, $M = 10$ and $M = 15$ as shown in Tables 4.7 - 4.12. Figure 4.5 shows the graphical solutions of our approximate solutions with the exact solutions. The average run-time is 0.0572 seconds.

Table 4.7: A comparison of absolute errors of $u_1(x)$ for Example 4.5 ($M = 5$)

x_i	TAU [16]	FIM-SCP
0	4.106683×10^{-3}	8.326673×10^{-17}
$(1/5)\pi$	3.394788×10^{-3}	1.647847×10^{-3}
$(2/5)\pi$	1.828708×10^{-4}	3.441410×10^{-4}
$(3/5)\pi$	3.080457×10^{-3}	3.786036×10^{-4}
$(4/5)\pi$	1.122245×10^{-2}	3.093337×10^{-3}
π	9.431156×10^{-3}	2.473648×10^{-16}

Table 4.8: A comparison of absolute errors of $u_2(x)$ for Example 4.5 ($M = 5$)

x_i	TAU [16]	FIM-SCP
0	1.885015×10^{-2}	4.440892×10^{-16}
$(1/5)\pi$	7.915380×10^{-3}	5.529300×10^{-3}
$(2/5)\pi$	3.003007×10^{-3}	7.344982×10^{-4}
$(3/5)\pi$	3.215101×10^{-3}	3.313541×10^{-3}
$(4/5)\pi$	1.402093×10^{-3}	4.408611×10^{-3}
π	1.051111×10^{-3}	1.554312×10^{-15}

Table 4.9: A comparison of absolute errors of $u_1(x)$ for Example 4.5 ($M = 10$)

x_i	TAU [16]	FIM-SCP
0	4.420342×10^{-8}	9.992010×10^{-16}
$(1/5)\pi$	1.247323×10^{-8}	5.209195×10^{-8}
$(2/5)\pi$	2.136281×10^{-8}	1.458277×10^{-8}
$(3/5)\pi$	5.186232×10^{-8}	1.848885×10^{-8}
$(4/5)\pi$	6.725784×10^{-8}	1.426471×10^{-8}
π	5.403331×10^{-8}	3.147822×10^{-15}

Table 4.10: A comparison of absolute errors of $u_2(x)$ for Example 4.5 ($M = 10$)

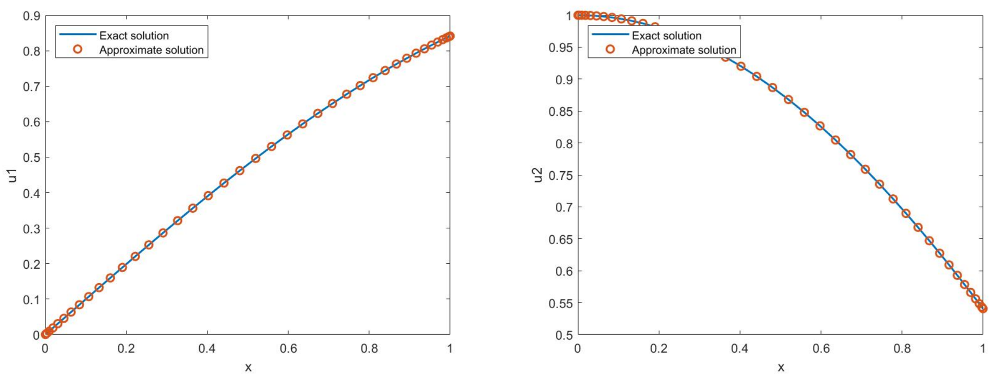
x_i	TAU [16]	FIM-SCP
0	4.701099×10^{-8}	1.110220×10^{-16}
$(1/5)\pi$	2.261104×10^{-8}	3.243422×10^{-8}
$(2/5)\pi$	3.873452×10^{-8}	2.639139×10^{-8}
$(3/5)\pi$	4.240854×10^{-9}	5.453175×10^{-9}
$(4/5)\pi$	3.007024×10^{-8}	1.499293×10^{-8}
π	2.175167×10^{-8}	9.992007×10^{-16}

Table 4.11: A comparison of absolute errors of $u_1(x)$ for Example 4.5 ($M = 15$)

x_i	TAU [16]	FIM-SCP
0	1.304721×10^{-9}	2.775558×10^{-16}
$(1/5)\pi$	5.264506×10^{-9}	2.788880×10^{-13}
$(2/5)\pi$	4.701600×10^{-9}	2.966516×10^{-13}
$(3/5)\pi$	1.430526×10^{-9}	3.603784×10^{-13}
$(4/5)\pi$	1.865835×10^{-8}	4.463097×10^{-13}
π	1.405449×10^{-5}	1.931448×10^{-15}

Table 4.12: A comparison of absolute errors of $u_2(x)$ for Example 4.5 ($M = 15$)

x_i	TAU [16]	FIM-SCP
0	6.028561×10^{-9}	4.440892×10^{-16}
$(1/5)\pi$	3.606308×10^{-9}	8.104628×10^{-13}
$(2/5)\pi$	5.473832×10^{-9}	6.925571×10^{-13}
$(3/5)\pi$	7.694197×10^{-9}	1.331713×10^{-13}
$(4/5)\pi$	8.618511×10^{-9}	3.380629×10^{-13}
π	6.447411×10^{-9}	8.88178×10^{-16}



(a) A graphical solution for $u_1(x)$ **(b)** A graphical solution for $u_2(x)$

Figure 4.5: The graph of the approximate and exact solutions in Example 4.5

The second example is a system of linear second order FIDEs with constant coefficients, polynomial forcing terms and kernel functions are in terms of functions depending on variables x and t.

Example 4.6. Consider the following system of linear second order FIDEs over $x \in (0,1)$

$$u_1''(x) + {u_2}'(x) = 3x^2 + \frac{3}{10}x + 8 - \int_0^1 (2xt)u_1(t)\ dt + \int_0^1 (6xt)u_2(t)\ dt, \tag{4.33}$$

$$u_1'(x) + u_2''(x) = 21x + \frac{4}{5} - \int_0^1 3(2x+t^2)u_1(t)\ dt + \int_0^1 6(2x+t^2)u_2(t)\ dt \tag{4.34}$$

with the boundary conditions $u_1(0)+u_1'(0) = 1$, $u_1(1)+u_1'(1) = 10$, $u_2(0)+u_2'(0) = 1$ and $u_2(1)+u_2'(1) = 7$. The analytical solutions are $u_1^*(x) = 3x^2+1$ and $u_2^*(x) = x^3+2x-1$.

In this example, we have $f_1(x) = 3x^2 + \frac{3}{10}x + 8$, $f_2(x) = 21x + \frac{4}{5}$, $\mathcal{K}_{1,1}(x,t) = -2xt$, $\mathcal{K}_{1,2}(x,t) = 6xt$, $\mathcal{K}_{2,1}(x,t) = -3(2x+t^2)$, $\mathcal{K}_{2,2}(x,t) = 6(2x+t^2)$ and $\lambda_{1,1} = \lambda_{1,2} = \lambda_{2,1} = \lambda_{2,2} = 1$.

By using our numerical procedure described in Section 4.3, we take double-layer integration both sides of (4.33) and (4.34). The problem can be transformed and simplified into the matrix form as

$$(\mathbf{K}_{11} - \mathbf{H}_{11}')\mathbf{u}_1 + (\mathbf{K}_{12} - \mathbf{H}_{12}')\mathbf{u}_2 + D_{1,1}\mathbf{x}_1 + D_{1,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_1,$$

$$(\mathbf{K}_{21} - \mathbf{H}_{21}')\mathbf{u}_1 + (\mathbf{K}_{22} - \mathbf{H}_{22}')\mathbf{u}_2 + D_{2,1}\mathbf{x}_1 + D_{2,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_2,$$

where

$$\mathbf{K}_{11} = \mathbf{I}, \quad \mathbf{H}_{11}' = \mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}_{1,1}', \quad \mathbf{K}_{12} = \mathbf{A}, \quad \mathbf{H}_{12}' = \mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}_{2,1}',$$

$$\mathbf{K}_{21} = \mathbf{A}, \quad \mathbf{H}_{21}' = \mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}_{2,1}', \quad \mathbf{K}_{22} = \mathbf{I}, \quad \mathbf{H}_{22}' = \mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}_{2,2}'.$$

The given boundary conditions can written in a matrix form as

$$
\begin{aligned}
u_1(0) + u_1'(0) &= (\mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} + \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1})\mathbf{u}_1 = 1,\\
u_1(\pi) + u_1'(\pi) &= (\mathbf{t}_{0,r}(\mathbf{T}^*)^{-1} + \mathbf{t}_{1,r}(\mathbf{T}^*)^{-1})\mathbf{u}_1 = 10,\\
u_2(0) + u_2'(0) &= (\mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} + \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1})\mathbf{u}_2 = 1,\\
u_2(\pi) + u_2'(\pi) &= (\mathbf{t}_{0,r}(\mathbf{T}^*)^{-1} + \mathbf{t}_{1,r}(\mathbf{T}^*)^{-1})\mathbf{u}_2 = 7,
\end{aligned}
$$

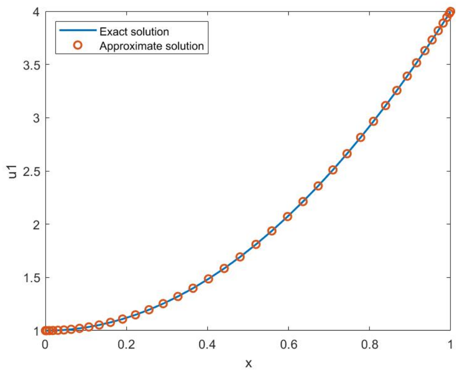
where $\mathbf{t}_{0,l}$, $\mathbf{t}_{0,r}$, $\mathbf{t}_{1,l}$ and $\mathbf{t}_{1,r}$ are defined in Example 4.5. Thus, we can construct the linear system in a matrix form

$$
\left[\begin{array}{cc|cccc}
\mathbf{K}_{11} - \mathbf{H}'_{11} & \mathbf{K}_{12} - \mathbf{H}'_{12} & \mathbf{x}_1 & \mathbf{x}_0 & \mathbf{0} & \mathbf{0}\\
\mathbf{K}_{21} - \mathbf{H}'_{21} & \mathbf{K}_{22} - \mathbf{H}'_{22} & \mathbf{0} & \mathbf{0} & \mathbf{x}_1 & \mathbf{x}_0\\
\hline
(\mathbf{t}_{0,l} + \mathbf{t}_{1,l})(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
(\mathbf{t}_{0,r} + \mathbf{t}_{1,r})(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & (\mathbf{t}_{0,l} + \mathbf{t}_{1,l})(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & (\mathbf{t}_{0,l} + \mathbf{t}_{1,l})(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}
\end{array}\right]
\left[\begin{array}{c}
\mathbf{u}_1\\ \mathbf{u}_2\\ \hline D_{1,1}\\ D_{1,2}\\ D_{2,1}\\ D_{2,2}
\end{array}\right]
=
\left[\begin{array}{c}
\mathbf{A}^2\bar{\mathbf{f}}_1\\ \mathbf{A}^2\bar{\mathbf{f}}_2\\ \hline 1\\ 10\\ 1\\ 7
\end{array}\right].
$$

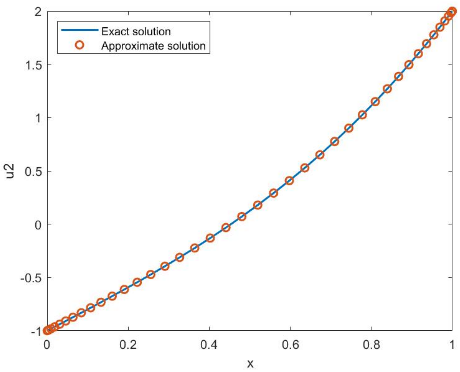
We solve the above matrix equation to obtain the approximate solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ of (4.33) and (4.34) and take these equations to (4.30) in order to obtain the approximate solutions $u_1(x)$ and $u_2(x)$ for each arbitrary $x \in [0, 1]$. We compare the absolute errors of our approximate results $u_1(x)$ and $u_2(x)$ with the analytical solutions by using $M = 10$ as demonstrated in Tables 4.13. The graphs of our approximate solutions with $M = 40$ are shown in Figure 4.6. The average run-time is 0.0554 seconds.

Table 4.13: A comparison of absolute errors of $u_1(x)$ and $u_2(x)$ for Example 4.6

x_i	$u_1(x)$	$u_2(x)$
0.006156	8.881785×10^{-15}	2.775557×10^{-15}
0.054497	3.330669×10^{-15}	9.547918×10^{-15}
0.146447	3.108624×10^{-15}	9.769963×10^{-15}
0.273005	4.662937×10^{-15}	5.995204×10^{-15}
0.421783	1.554312×10^{-15}	4.996004×10^{-15}
0.578217	4.440892×10^{-16}	9.103829×10^{-16}
0.726995	1.332268×10^{-15}	5.107026×10^{-15}
0.853553	3.996803×10^{-15}	5.773160×10^{-15}
0.945503	2.664535×10^{-15}	5.107026×10^{-15}
0.993844	5.773160×10^{-15}	1.998401×10^{-15}



(a) A graphical solution for $u_1(x)$



(b) A graphical solution for $u_2(x)$

Figure 4.6: The graph of the approximate and exact solutions in Example 4.6

The last example is a system of linear second order FIDEs with variable coefficients, the forcing terms and kernel functions are in terms of trigonometry.

Example 4.7. Consider the following system of linear second order FIDEs on $x \in (0,1)$

$$u_1''(x) - xu_2'(x) - u_1(x) = (x-2)\sin(x) + \int_0^1 (x\cos(t)u_1(t) - x\sin(t)u_2(t))\,dt, \quad (4.35)$$

$$u_2''(x) - 2xu_1'(x) + u_2(x) = -2x\cos(x) + \sin(x)\cos(t)\int_0^1 (u_1(t) - u_2(t))\,dt \quad (4.36)$$

with the initial conditions $u_1(0) = 0$, $u_1'(0) = 1$, $u_2(0) = 1$ and $u_2'(0) = 1$. The exact solutions are $u_1^*(x) = \sin(x)$ and $u_2^*(x) = \cos(x)$.

From this example, we have $p_{1,2}^1(x) = -x$, $p_{2,1}^1(x) = -2x$, $f_1(x) = (x-2)\sin(x)$, $f_2(x) = -2x\cos(x)$, $\mathcal{K}_{1,1}(x,t) = x\cos(t)$, $\mathcal{K}_{1,2}(x,t) = -x\sin(t)$, $\mathcal{K}_{2,1}(x,t) = \sin(x)\cos(t)$, $\mathcal{K}_{2,2}(x,t) = -\sin(x)\cos(t)$ and $\lambda_{1,1} = \lambda_{1,2} = \lambda_{2,1} = \lambda_{2,2} = 1$.

Taking double-layer integration both sides of (4.35) and (4.36) by using our numerical procedure described in Section 4.3. Then, the problem can be transformed and simplified into the matrix forms

$$(\mathbf{K}_{11} - \mathbf{H}_{11}')\mathbf{u}_1 + (\mathbf{K}_{12} - \mathbf{H}_{12}')\mathbf{u}_2 + D_{1,1}\mathbf{x}_1 + D_{1,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_1,$$
$$(\mathbf{K}_{21} - \mathbf{H}_{21}')\mathbf{u}_1 + (\mathbf{K}_{22} - \mathbf{H}_{22}')\mathbf{u}_2 + D_{2,1}\mathbf{x}_1 + D_{2,2}\mathbf{x}_0 = \mathbf{A}^2\bar{\mathbf{f}}_2,$$

where

$$\begin{aligned}
\mathbf{K}_{11} &= \mathbf{I} - \mathbf{A}^2, & \mathbf{H}_{11}' &= \mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}_{1,1}',\\
\mathbf{K}_{12} &= \mathbf{A}\left(\mathbf{P}_{1,2}^1\right)^{(0)} - \mathbf{A}^2\left(\mathbf{P}_{1,2}^1\right)^{(1)}, & \mathbf{H}_{12}' &= \mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}_{2,1}',\\
\mathbf{K}_{21} &= \mathbf{A}\left(\mathbf{P}_{2,1}^1\right)^{(0)} - \mathbf{A}^2\left(\mathbf{P}_{2,1}^1\right)^{(1)}, & \mathbf{H}_{21}' &= \mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}_{2,1}',\\
\mathbf{K}_{22} &= \mathbf{I} + \mathbf{A}^2, & \mathbf{H}_{22}' &= \mathbf{A}^2\mathbf{B}'\bar{\mathbf{K}}_{2,2}'.
\end{aligned}$$

By using the boundary conditions, $u_1(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 0$, $u_1'(0) = \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1}\mathbf{u}_1 = 1$, $u_2(0) = \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 1$ and $u_2'(0) = \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1}\mathbf{u}_2 = 1$, where $\mathbf{t}_{0,l}$ and $\mathbf{t}_{1,l}$ are defined in Example 4.5. Thus, we can construct a linear system in a matrix form

$$\left[\begin{array}{cc|cccc} \mathbf{K}_{11}-\mathbf{H}'_{11} & \mathbf{K}_{12}-\mathbf{H}'_{12} & \mathbf{x}_1 & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{K}_{21}-\mathbf{H}'_{21} & \mathbf{K}_{22}-\mathbf{H}'_{22} & \mathbf{0} & \mathbf{0} & \mathbf{x}_1 & \mathbf{x}_0 \\ \hline \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{t}_{1,r}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{1,2} \\ D_{2,1} \\ D_{2,2} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}^2\bar{\mathbf{f}}_1 \\ \mathbf{A}^2\bar{\mathbf{f}}_2 \\ \hline 0 \\ 1 \\ 1 \\ 1 \end{array}\right]. \tag{4.37}$$

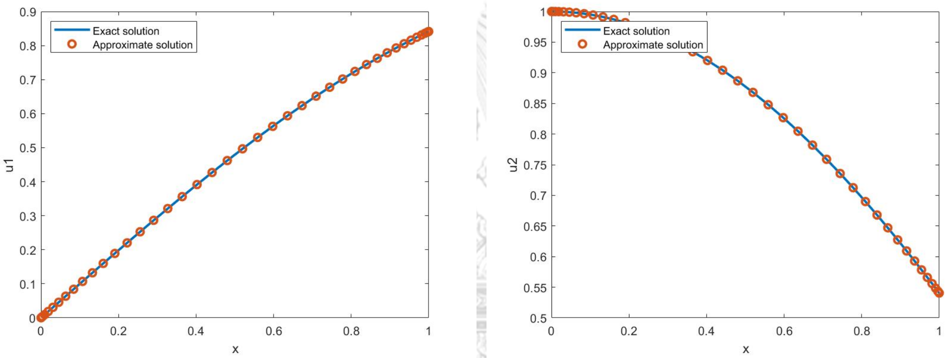
We compute (4.37) to obtain the approximate solutions $\mathbf{u}_1$ and $\mathbf{u}_2$ of (4.33) and (4.34) and take these equation (4.30) to get the approximate solutions $u_1(x)$ and $u_2(x)$ for each arbitrary $x \in [0,1]$. The comparison of the average absolute errors of our approximate solutions $u_1(x)$ and $u_2(x)$ with [17] and [18] by using $M \in \{3,7,9,10,11,12\}$ as demonstrated in Tables 4.14 and 4.15, respectively. Figure 4.7 show the graphs of our approximate solutions with $M = 40$. The average run-time is 0.0574 seconds.

Table 4.14: A comparison of average absolute errors of $u_1(x)$ for Example 4.7

M	CM-BP [17]	CM-FP [18]	FIM-SCP
3	5.0207×10^{-3}	5.0207×10^{-3}	2.8326×10^{-3}
7	5.0207×10^{-7}	5.0207×10^{-7}	1.3485×10^{-10}
9	3.9722×10^{-9}	3.9722×10^{-9}	1.1567×10^{-11}
10	2.6596×10^{-10}	2.6596×10^{-10}	2.3278×10^{-13}
11	2.4875×10^{-11}	2.4875×10^{-11}	7.3000×10^{-15}
12	1.2126×10^{-12}	1.2126×10^{-12}	2.0921×10^{-15}

Table 4.15: A comparison of average absolute errors of $u_2(x)$ for Example 4.7

M	CM-BP [17]	CM-FP [18]	FIM-SCP
3	1.3565×10^{-2}	1.3565×10^{-2}	1.4316×10^{-3}
7	6.3006×10^{-7}	6.3006×10^{-7}	9.0883×10^{-9}
9	4.2348×10^{-9}	4.2348×10^{-9}	7.8634×10^{-13}
10	2.9397×10^{-10}	2.9397×10^{-10}	4.7902×10^{-13}
11	2.5629×10^{-11}	2.5629×10^{-11}	3.0279×10^{-15}
12	1.5526×10^{-12}	1.5526×10^{-12}	2.0262×10^{-15}



(a) A graphical solution for $u_1(x)$ **(b)** A graphical solution for $u_2(x)$

Figure 4.7: The graph of the approximate and exact solutions in Example 4.7

CHAPTER V

CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

In this thesis, we devise the numerical algorithms based on the idea of [22] with slightly modify by using the shifted Chebyshev polynomials for finding the approximate solutions to the systems of linear ODEs, VIDEs and FIDEs problems. We utilize the zeros of shifted Chebyshev polynomials of a certain degree to be the computational nodes and construct the shifted Chebyshev integration matrices for these devised algorithms.

Several numerical examples illustrate the performance of our numerical algorithms and the accuracy of our approximate solutions comparing with some other numerical methods in literatures. In Section 3.2, for Example 3.1, our method provides a better accuracy than other methods in terms of the absolute errors at the same number of nodal points and under the same conditions which can be seen in Table 3.1. For Example 3.2 which is the stiff system of linear ODEs, our method gives a good result compare to other methods for every computational grid point in terms of the absolute errors at the same number of nodal points and under the same conditions which can be seen in Tables 3.2 - 3.4. For Example 3.3 which is the stiff system of linear ODEs and Example 3.4 which is the system of linear ODEs with the boundary conditions, our method also gives the high accuracy compare to the analytical solutions in terms of the average absolute errors as shown in Tables 3.5 and Table 3.6, respectively. We also plot the graphical solutions at the number of nodes $M = 40$ as shown in Figures 3.1 - 3.4.

In Section 4.2, our method provides a higher accuracy than other methods in terms of the absolute errors at the same number of nodal points and under the same conditions for every computational grid point which can be seen in Tables 4.1 - 4.5. For Example 4.4

which is the system of linear VIDEs with the boundary conditions, our method also gives the high accuracy compare to the analytical solutions in terms of the average absolute errors as shown in Tables 4.6. We further show the graphical solutions at $M = 40$ as shown in Figures 4.1 - 4.4.

In Section 4.4, for Example 4.5 and 4.7, our method provides a higher accuracy than other methods in terms of the absolute errors at the same number of nodal points and under the same conditions for every computational grid point which can be seen in Tables 4.7 - 4.12 and Tables 4.14 - 4.15. For Example 4.6, our method provides the high accuracy compare to the analytical solutions in terms of the average absolute errors as shown in Tables 4.13. We finally show the graphical solutions at $M = 40$ as shown in Figures 4.5 - 4.7.

For $M \in \{3, 5, 7, 9, 11, 13, 15\}$, Tables 5.1 - 5.11 demonstrate the average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 3.1 - 3.4 and Example 4.1 - 4.7, respectively.

Table 5.1: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 3.1

M	$u_1(x)$	$u_2(x)$
3	1.707608	1.979499
5	9.453146×10^{-2}	9.628591×10^{-2}
7	3.914707×10^{-3}	3.926651×10^{-3}
9	1.027767×10^{-4}	1.028068×10^{-4}
11	1.818047×10^{-6}	1.818131×10^{-6}
13	2.441199×10^{-8}	2.441209×10^{-8}
15	2.418820×10^{-10}	2.419012×10^{-10}

Table 5.2: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 3.2

M	$u_1(x)$	$u_2(x)$	$u_3(x)$
3	6.597985×10^{-5}	1.361354×10^{-5}	1.361418×10^{-5}
5	3.338292×10^{-9}	1.665486×10^{-8}	1.665485×10^{-8}
7	1.986126×10^{-12}	3.975261×10^{-13}	3.956041×10^{-13}
9	2.146431×10^{-15}	3.335367×10^{-15}	1.048544×10^{-15}
11	9.780056×10^{-15}	1.275455×10^{-15}	9.911264×10^{-15}
13	1.298107×10^{-15}	2.101167×10^{-15}	1.639714×10^{-15}
15	7.786364×10^{-15}	8.681857×10^{-16}	6.550316×10^{-15}

Table 5.3: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 3.3

M	$u_1(x)$	$u_2(x)$
3	6.071904×10^{-3}	6.071904×10^{-3}
5	1.902571×10^{-5}	1.902571×10^{-3}
7	2.843494×10^{-8}	2.843493×10^{-8}
9	2.471849×10^{-11}	2.471813×10^{-11}
11	2.860338×10^{-14}	1.602254×10^{-14}
13	1.848094×10^{-14}	5.642922×10^{-15}
15	2.016165×10^{-14}	4.637031×10^{-15}

Table 5.4: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 3.4

M	$u_1(x)$	$u_2(x)$	$u_3(x)$
3	6.653674×10^{-3}	6.826086×10^{-3}	1.956917×10^{-3}
5	1.866672×10^{-5}	2.959495×10^{-5}	7.409434×10^{-6}
7	2.879149×10^{-8}	6.521237×10^{-8}	1.373126×10^{-8}
9	2.708902×10^{-11}	7.116367×10^{-11}	1.465876×10^{-11}
11	1.735985×10^{-14}	4.902896×10^{-14}	1.024938×10^{-14}
13	1.540648×10^{-14}	5.073666×10^{-15}	3.322129×10^{-15}
15	9.947598×10^{-15}	2.643949×10^{-15}	4.718448×10^{-15}

Table 5.5: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 4.1

M	$u_1(x)$	$u_2(x)$
3	1.260764×10^{-2}	7.729789×10^{-3}
5	3.648782×10^{-5}	2.139675×10^{-5}
7	5.317952×10^{-8}	3.083810×10^{-8}
9	4.577092×10^{-11}	2.642817×10^{-11}
11	2.488918×10^{-14}	1.388788×10^{-14}
13	7.105427×10^{-15}	2.100884×10^{-15}
15	5.536312×10^{-15}	2.827368×10^{-15}

Table 5.6: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 4.2

M	$u_1(x)$	$u_2(x)$
3	5.233532×10^{-3}	2.580935×10^{-3}
5	1.667650×10^{-5}	6.588292×10^{-6}
7	2.533888×10^{-8}	1.035765×10^{-8}
9	2.198742×10^{-11}	8.618883×10^{-12}
11	7.145799×10^{-15}	6.651245×10^{-15}
13	1.187085×10^{-15}	2.895120×10^{-15}
15	8.822572×10^{-15}	3.204844×10^{-15}

Table 5.7: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 4.3

M	$u_1(x)$	$u_2(x)$
3	5.099841×10^{-3}	3.017017×10^{-3}
5	2.075038×10^{-5}	8.926379×10^{-6}
7	2.756037×10^{-8}	1.355109×10^{-8}
9	2.408834×10^{-11}	1.160289×10^{-11}
11	8.599182×10^{-15}	9.891078×10^{-15}
13	1.463786×10^{-14}	1.144384×10^{-14}
15	9.118632×10^{-15}	8.319271×10^{-15}

Table 5.8: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 4.4

M	$u_1(x)$	$u_2(x)$
3	1.208510×10^{-5}	1.850717×10^{-3}
5	3.393119×10^{-16}	3.749084×10^{-15}
7	1.485419×10^{-15}	8.242662×10^{-15}
9	1.367733×10^{-15}	2.314179×10^{-14}
11	2.573778×10^{-15}	2.956821×10^{-14}
13	5.623039×10^{-15}	5.832751×10^{-14}
15	2.703104×10^{-15}	3.857025×10^{-14}

Table 5.9: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 4.5

M	$u_1(x)$	$u_2(x)$
3	2.832573×10^{-3}	1.431580×10^{-3}
5	8.881220×10^{-6}	7.167711×10^{-6}
7	1.348496×10^{-8}	9.088295×10^{-9}
9	1.156664×10^{-11}	7.863352×10^{-12}
11	7.299953×10^{-15}	3.027881×10^{-15}
13	4.423745×10^{-15}	2.997602×10^{-15}
15	2.332162×10^{-15}	2.301862×10^{-15}

Table 5.10: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 4.6

M	$u_1(x)$	$u_2(x)$
3	2.627196×10^{-3}	1.897008×10^{-2}
5	3.330669×10^{-15}	2.525757×10^{-15}
7	5.551115×10^{-15}	8.556647×10^{-15}
9	5.625129×10^{-15}	3.938208×10^{-15}
11	8.619368×10^{-15}	6.762268×10^{-15}
13	2.252899×10^{-14}	1.173634×10^{-14}
15	1.178317×10^{-14}	7.329322×10^{-15}

Table 5.11: Average absolute errors of $u_1(x)$ and $u_2(x)$ for Example 4.7

M	$u_1(x)$	$u_2(x)$
3	2.832573×10^{-3}	1.431580×10^{-3}
5	8.881220×10^{-6}	7.167711×10^{-6}
7	1.348496×10^{-8}	9.088295×10^{-9}
9	1.156664×10^{-11}	7.863352×10^{-12}
11	7.299953×10^{-15}	3.027881×10^{-15}
13	4.423745×10^{-15}	2.997602×10^{-15}
15	2.332162×10^{-15}	2.301862×10^{-15}

5.2 Future work

The plan of our future works for improving our results and extend the scope of the research for our proposed method based on shifted Chebyshev polynomials are the followings

1. To extend our proposed algorithm for solving the system of linear FIDEs with Neumann and mixed boundary conditions.

2. To improve our devised method for the system of nonlinear IDEs.

3. To extend the scope of our domains for solving the system of linear IDEs in the other domains such as circle and polygons by using our presented method.

4. To find the theoretical accuracy of our presented method.

REFERENCES

[1] W. M. Abd-Elhameed, Y. H. Youssri, and E. H. Doha, "A novel operational matrix method based on shifted Legendre polynomials for solving second-order boundary value problems involving singular, singularly perturbed and Bratu-type equations," *Mathematical Sciences*, vol. 9, pp. 93–102, 2015.

[2] O. Coulaud, D. Funaro, and O. Kavian, "Laguerre spectral approximation of elliptic problems in exterior domains," *Computer Methods in Applied Mechanics and Engineering*, vol. 80, no. 1–3, pp. 451–458, 1990.

[3] M. Erfanian and H. Zeidabadi, "Using of Bernstein spectral Galerkin method for solving of weakly singular Volterra–Fredholm integral equations," *Mathematical Sciences*, vol. 12, pp. 103–109, 2018.

[4] N. Kurt and M. Çevik, "Polynomial solution of the single degree of freedom system by Taylor matrix method," *Mechanics Research Communications*, vol. 35, no. 8, pp. 530–536, 2008.

[5] N. Kurt and M. Sezer, "Polynomial solution of high-order linear Fredholm integro-differential equations with constant coefficients," *Journal of the Franklin Institute*, vol. 345, no. 8, pp. 839–850, 2008.

[6] C. Oguz and M. Sezer, "Chebyshev collocation method for a class of mixed functional integro-differential equations," *Applied Mathematics and Computation*, vol. 259, pp. 943–954, 2015.

[7] N. Bellomo, B. Firmani, and L. Guerri, "Bifurcation analysis for a nonlinear system of integro-differential equations modelling tumor-immune cells competition," *Applied Mathematics Letters*, vol. 12, pp. 39–44, 1999.

[8] T. L. Bo, L. Xie, and X. L. Zheng, "Numerical approach to wind ripple in desert," *International Journal of Nonlinear Sciences and Numerical Simulation*, vol. 8, no. 2,

pp. 223–228, 2007.

[9] F. Z. Sun, M. Gao, and S. H. Lei, “The fractal dimension of the fractal model of drop-wise condensation and its experimental study,” *International Journal of Nonlinear Sciences and Numerical Simulation*, vol. 8, no. 2, pp. 211–222, 2007.

[10] H. Wang, H. M. Fu, and H. F. Zhang, “A practical thermodynamic method to calculate the best glass-forming composition for bulk metallic glasses,” *International Journal of Nonlinear Sciences and Numerical Simulation*, vol. 8, no. 2, pp. 171–178, 2007.

[11] A. M. Wazwaz, “The existence of noise terms for systems of inhomogeneous differential and integral equations,” *Applied Mathematics and Computation*, vol. 146, no. 1, pp. 81–92, 2003.

[12] L. Xu, J. H. He, and Y. Liu, “Electrospun nanoporous spheres with Chinese drug,” *International Journal of Nonlinear Sciences and Numerical Simulation*, vol. 8, no. 2, pp. 199–202, 2007.

[13] J. Zhao and R. M. Corless, “Compact finite difference method has been used for integro-differential equations,” *Applied Mathematics and Computation*, vol. 177, no. 1, pp. 271–288, 2006.

[14] H. Brunner, “Implicitly linear collocation method for nonlinear Volterra equations,” *Applied Numerical Mathematics*, vol. 9, no. 3–5, pp. 235–247, 1992.

[15] B. Sepehrian and M. Razzaghi, “Single-term Walsh series method for the Volterra integro-differential equations,” *Engineering Analysis with Boundary Elements*, vol. 28, no. 11, pp. 1315–1319, 2004.

[16] J. Pour-Mahmoud, M. Y. Rahimi-Ardabili, and S. Shahmorad, “Numerical solution of the system of Fredholm integro-differential equations by the Tau method,” *Applied Mathematics and Computation*, vol. 168, no. 1, pp. 465–478, 2005.

[17] Y. Şuayip, S. Niyazi, and S. Mehmet, “Numerical solutions of systems of linear Fredholm integro-differential equations with Bessel polynomial bases,” *Computers*

and Mathematics with Applications, vol. 61, no. 10, pp. 3079–3096, 2011.

[18] M. Farshid and F. H. Seyede, "Solving systems of linear Fredholm integro-differential equations with Fibonacci polynomials," *Ain Shams Engineering Journal*, vol. 5, no. 1, pp. 271–283, 2014.

[19] P. H. Wen, Y. C. Hon, M. Li, and T. Korakianitis, "Finite integration method for partial differential equations," *Applied Mathematical Modelling*, vol. 37, no. 24, pp. 10092–10106, 2013.

[20] M. Li, C. S. Chen, Y. C. Hon, and P. H. Wen, "Finite integration method for solving multi-dimensional partial differential equations," *Applied Mathematical Modelling*, vol. 39, no. 7, pp. 4979–4994, 2015.

[21] M. Li, Z. L. Tian, Y. C. Hon, C. S. Chen, and P. H. Wen, "Improved finite integration method for partial differential equations," *Engineering Analysis with Boundary Elements*, vol. 64, pp. 230–236, 2016.

[22] R. Boonklurb, A. Duangpan, and T. Treeyaprasert, "Modified finite integration method using Chebyshev polynomial for solving linear differential equations," *Journal of Numerical Analysis, Industrial and Applied Mathematics*, vol. 12, no. 3–4, pp. 1–19, 2018.

[23] J. Boyd, *Chebyshev and Fourier Spectral Methods.* University of Michigan: DOVER Publishing, 2 ed., 2000.

[24] C. H. C. Hussin and A. Kilicman, "Solving system of linear differential equations by using differential transformation method," *AIP Conference Proceedings*, vol. 1522, pp. 245–150, 2013.

[25] I. H. A.-H. Hassan, "Application to differential transformation method for solving systems of differential equations," *Applied Mathematical Modelling*, vol. 32, no. 12, pp. 2552–2559, 2018.

[26] M. M. Stabrowski, "An efficient algorithm for solving stiff ordinary differential equations," *Simulation Practice and Theory*, vol. 5, no. 4, pp. 333–344, 1997.

[27] J. R. Loh and C. Phang, "A new numerical scheme for solving system of Volterra integro-differential equation," *Alexandria Engineering Journal*, vol. 57, no. 2, pp. 1117–1124, 2018.

[28] R. C. G. Sekar and K. Murugesan, "System of linear second order Volterra integro-differential equations using single term Walsh series technique," *Applied Mathematics and Computation*, vol. 273, pp. 484–492, 2016.

[29] M. Berenguer, A. Garralda-Guillem, and M. R. Galán, "An approximation method for solving systems of Volterra integro-differential equations," *Applied Numerical Mathematics*, vol. 67, pp. 126–135, 2013.

APPENDICES

In this thesis, we implement our propose numerical algorithms with MatLab software to calculate the approximate solutions of each example in this research. In this appendix, we would like to present some examples of the code which the linear systems are solved by the Gaussian elimination method.

APPENDIX A : Example of MatLab code for solving the stiff system of ODEs.

Example A. We consider Example 3.2

$$\begin{aligned} u_1'(x) &= -20u_1(x) - 0.25u_2(x) - 19.75u_3(x), \\ u_2'(x) &= 20u_1(x) - 20.25u_2(x) + 0.25u_3(x), \\ u_3'(x) &= 20u_1(x) - 19.75u_2(x) - 0.25u_3(x), \end{aligned}$$

with initial conditions $u_1(0) = 1$, $u_2(0) = 0$ and $u_3(0) = -1$. The analytical solutions are

$$\begin{aligned} u_1(x) &= \frac{1}{2}\left(e^{-\frac{1}{2}x} + e^{-20x}(\cos(20x) + \sin(20x))\right), \\ u_2(x) &= \frac{1}{2}\left(e^{-\frac{1}{2}x} - e^{-20x}(\cos(20x) - \sin(20x))\right), \\ u_3(x) &= \frac{1}{2}\left(e^{-\frac{1}{2}x} + e^{-20x}(\cos(20x) - \sin(20x))\right). \end{aligned}$$

Thus, we can construct the linear system in a matrix form as follows

$$\left[\begin{array}{ccc|ccc} \mathbf{I} + 20\mathbf{A} & 0.25\mathbf{A} & 19.75\mathbf{A} & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} \\ -20\mathbf{A} & \mathbf{I} + 20.25\mathbf{A} & -0.25\mathbf{A} & \mathbf{0} & \mathbf{x}_0 & \mathbf{0} \\ -20\mathbf{A} & 19.75\mathbf{A} & \mathbf{I} + 0.25\mathbf{A} & \mathbf{0} & \mathbf{0} & \mathbf{x}_0 \\ \hline \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \hline D_{1,1} \\ D_{2,1} \\ D_{3,1} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}\bar{\mathbf{f}}_1 \\ \mathbf{A}\bar{\mathbf{f}}_2 \\ \mathbf{A}\bar{\mathbf{f}}_3 \\ \hline 1 \\ 0 \\ -1 \end{array}\right].$$

```
%% Input parameters----------------------------------------------
m = 1;                   % The higher order derivative
M = 16;                  % The number of nodal points
```

```
a = 0;                      % The left boundary
b = 0.01;                   % The right boundary
fx1 = 0;                    % The forcing term f_1(x)
fx2 = 0;                    % The forcing term f_2(x)
%% Analytical solutions------------------------------------------
ex1 = @(x) (1/2)*(exp(-x./2)+exp(-20*x).*(cos(20*x)+sin(20*x)));
ex2 = @(x) (1/2)*(exp(-x./2)-exp(-20*x).*(cos(20*x)-sin(20*x)));
ex3 = @(x) -(1/2)*(exp(-x./2)+exp(-20*x).*(cos(20*x)-sin(20*x)));
%% Compute xbar in [0,1]------------------------------------------
xbar = flip(((0.01)*cos((2*(1:M)'-1)/(2*M)*pi)+0.01)/2);
%% Integration matrix A-------------------------------------------
%----- Construct matrix T* -----
T(:,1) = ones(M,1);
T(:,2) = (2*xbar-0.01)/(0.01);
for n = 2:M
    T(:,n+1) = 2*(2*xbar-0.01)/(0.01).*T(:,n)-T(:,n-1);
end
%----- Construct matrix (T*)bar -----
Tbar(:,1) = xbar;
Tbar(:,2) = (xbar).*(xbar-0.01)/(0.01);
for n = 2:M-1
    Tbar(:,n+1) = (0.01)/4*(T(:,n+2)/(n+1)-T(:,n)/(n-1)-2*(-1)^n/(n^2-1));
end
Tinv = 1/M*diag([1 2*ones(1,M-1)])*T(:,1:M)';
A = Tbar*Tinv;
%% Boundary conditions--------------------------------------------
t0l = (-1).^(0:M-1);
r1 = [t0l*Tinv zeros(1,M) zeros(1,M)];
r2 = [zeros(1,M) t0l*Tinv zeros(1,M)];
```

```
r3 = [zeros(1,M) zeros(1,M) t01*Tinv];
%% Construct linear system-----------------------------------------
%----- Construct matrix K_ij -----
K_11 = eye(M)+20*A;
K_12 = 0.25*A;
K_13 = 19.75*A;
K_21 = -20*A;
K_22 = eye(M)+20.25*A;
K_23 = -0.25*A;
K_31 = -20*A;
K_32 = 19.75*A;
K_33 = eye(M)+0.25*A;
%----- Constuct matrix equation -----
K_o=[K_11 K_12 K_13; K_21 K_22 K_23; K_31 K_32 K_33]; % Matrix K_o
Q = [ ones(M,1) zeros(M,2); zeros(M,1) ones(M,1) zeros(M,1);
    zeros(M,2) ones(M,1)];                               % Matrix Q
R = [r1; r2; r3];                                        % Matrix R
M0 = [0 0 0; 0 0 0; 0 0 0];                              % Matrix 0
W = [zeros(M,1);zeros(M,1); zeros(M,1)];                 % Matrix W
b = [ex1(0); ex2(0); ex3(0)];                            % Matrix b
Z = [K_o Q; R M0];                          % The LHS Of linear system
B = [W; b];                                 % The RHS Of linear system
%% Solve u---------------------------------------------------------
u = pinv(Z)*B;                              % Numerical Solutions
e1 = ex1(xbar);                             % Analytical solution u1
e2 = ex2(xbar);                             % Analytical solution u2
e3 = ex3(xbar);                             % Analytical solution u3
E1 = mean(abs(u(1:M)-e1))           % Average absolute error of u1
E2 = mean(abs(u(M+1:2*M)-e2))       % Average absolute error of u2
E3 = mean(abs(u(2*M+1:3*M)-e3))     % Average absolute error of u3
```

```
[xbar u(1:M) e1 abs(u(1:M)-e1)];
[xbar u(1:M) e2 abs(u(M+1:2*M)-e2)];
[xbar u(1:M) e3 abs(u(2*M+1:3*M)-e3)];
%% Compute u for arbitary x---------------------------------------
x1 = [0.000 0.002 0.004 0.006 0.008 0.010]';
T1 = @(n,x1) cos(n*acos((2*x1-0.01)/0.01));
for j=1: length(x1)
    for i=0:M-1
        T1x(1,i+1)= T1(i,x1(j));
    end
        ur1(j) = T1x*Tinv*u(1:M);          % u1 for arbitary x
        ur2(j) = T1x*Tinv*u(M+1:2*M);      % u2 for arbitary x
        ur3(j) = T1x*Tinv*u(2*M+1:3*M);    % u3 for arbitary x
        er1(j) = abs(ur1(j)-ex1(x1(j)));   % Absolute error of u1
        er2(j) = abs(ur2(j)-ex2(x1(j)));   % Absolute error of u2
        er3(j) = abs(ur3(j)-ex3(x1(j)));   % Absolute error of u3
end
[x1 er1' er2' er3'];
%% Plot our numerical & analytical solutions----------------------
p1 = plot(xbar,e1,'red');
hold on
p2 = plot(xbar,u(1:M),'bo');
figure
p3 = plot(xbar,e2,'red')
hold on
p4 = plot(xbar,u(M+1:2*M),'bo');
figure
p5 = plot(xbar,e3,'red')
hold on
p6 = plot(xbar,u(2*M+1:3*M),'bo');
```

APPENDIX B : Example of MatLab code for solving the system of linear VIDEs.

Example B. We consider Example 4.2

$$
\begin{aligned}
u_1''(x) + (-3x^2 - 6x + 7)u_1(x) - 2x^2(x+1)u_2(x) &= x^4 - x^3 - 2x^2 - 6 \\
&\quad + \int_0^x (t^3 - x^3)u_1(t)\,dt + \int_0^x x^2(t^2 - x^2)u_2(t)\,dt, \\
u_2''(x) + 2(x-1)u_1(x) + (2x^4 + 2x^3 + 2x^2 - 1)u_2(x) &= x^4 + 3x^3 - 2 \\
&\quad + \int_0^x (x^2 - t^2)u_1(t)\,dt - \int_0^x x^2(t^2 + x^2)u_2(t))\,dt
\end{aligned}
$$

subject to the initial conditions $u_1(0) = 1$, $u_2(0) = 1$, $u_1'(0) = 1$ and $u_2'(0) = -1$. The analytical solutions arre $u_1(x) = e^x$ and $u_2(x) = e^{-x}$. We can construct the linear system in a matrix form as follows

$$
\left[\begin{array}{cc|cccc}
\mathbf{K}_{11} - \mathbf{H}_{11} & \mathbf{K}_{12} - \mathbf{H}_{12} & \mathbf{x}_1 & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} \\
\mathbf{K}_{21} - \mathbf{H}_{21} & \mathbf{K}_{22} - \mathbf{H}_{22} & \mathbf{0} & \mathbf{0} & \mathbf{x}_1 & \mathbf{x}_0 \\
\hline
\mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{t}_1(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{t}_0(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{t}_1(\mathbf{T}^*)^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}
\end{array}\right]
\left[\begin{array}{c}
\mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{1,2} \\ D_{2,1} \\ D_{2,2}
\end{array}\right]
=
\left[\begin{array}{c}
\mathbf{A}^2\bar{\mathbf{f}}_1 \\ \mathbf{A}^2\bar{\mathbf{f}}_2 \\ \hline 1 \\ 1 \\ 1 \\ -1
\end{array}\right].
$$


```
%% Input parameters-----------------------------------------------
m = 2;                                 % The higher order derivative
M = 8;                                 % The number of nodal points
a = 0;                                 % The left boundary
b = 1;                                 % The right boundary
lam = 1;                               % Value of lamma_{i,j}
f1 = @(x) x.^4-x.^3-2*x.^2-6;          % The forcing term f_1(x)
f2 = @(x) x.^4+3*x.^2-2;               % The forcing term f_2(x)
Kxt11 =@(x,t) t.^3-x.^3;               % The kernel function K_11(x,t)
Kxt12 =@(x,t) (x.^2).*(t.^2-x.^2);     % The kernel function K_12(x,t)
```

```
Kxt21 =@(x,t) x.^2-t.^2;            % The kernel function K_21(x,t)
Kxt22 =@(x,t) (x.^2).*(t.^2+x.^2); % The kernel function K_22(x,t)
%% Analytical solutions-----------------------------------------
ex1 = @(x) exp(x);                  % Analytical solution u1(x)
ex2 = @(x) exp(-x);                 % Analytical solution u2(x)
bl1 = ex1(a);                       % Value of u1(0)
br1 = ex1(b);                       % Value of u1(1)
bl2 = ex2(a);                       % Value of u2(0)
br2 = ex2(b);                       % Value of u2(1)
%% Compute xbar & tbar in [0,1]---------------------------------
xbar = flip((cos((2*(1:M)'-1)/(2*M)*pi)+1)/2);
tbar = flip((cos((2*(1:M)'-1)/(2*M)*pi)+1)/2);
%% Integration matrix A-----------------------------------------
%----- Construct matrix T* -----
T(:,1) = ones(M,1);
T(:,2) = (2*xbar-1);
for n = 2:M
    T(:,n+1) = 2*(2*xbar-1).*T(:,n)-T(:,n-1);
end
%----- Construct matrix (T*)bar -----
Tbar(:,1) = xbar;
Tbar(:,2) = (xbar).*(xbar-1);
for n = 2:M-1
    Tbar(:,n+1) = 1/4*(T(:,n+2)/(n+1)-T(:,n)/(n-1)-2*(-1)^n/(n
        ^2-1));
end
Tinv = 1/M*diag([1 2*ones(1,M-1)])*T(:,1:M)';
A = Tbar*Tinv;
%% Construct matrix A'*(Kbar)'_ij-------------------------------
for i = 1:M
```

```
    for j = 1:M
        K11(i,j) = Kxt11(xbar(i),tbar(j));
        K12(i,j) = Kxt12(xbar(i),tbar(j));
        K21(i,j) = Kxt21(xbar(i),tbar(j));
        K22(i,j) = Kxt22(xbar(i),tbar(j));
    end
end
for k = 1:M
        H11(k,:) = A(k,:)*diag(K11(k,:));
        H12(k,:) = A(k,:)*diag(K12(k,:));
        H21(k,:) = A(k,:)*diag(K21(k,:));
        H22(k,:) = A(k,:)*diag(K22(k,:));
end
%% Boundary conditions---------------------------------------------
tl = (-1).^(0:M-1);
tr = (1).^(0:M-1);
r1 = [tl*Tinv zeros(1,M)];
r2 = [tr*Tinv zeros(1,M)];
r3 = [zeros(1,M) tl*Tinv];
r4 = [zeros(1,M) tr*Tinv];
%% Construct linear system-----------------------------------------
%----- Construct matrix P_ij -----
P_11 = diag(3*xbar.^2-6*xbar+7);
P_12 = diag((2*xbar.^2).*(xbar+1));
P_21 = diag(2*(xbar-1));
P_22 = diag(2*xbar.^4+2*xbar.^3+2*xbar.^2-1);
%----- Construct matrix K_ij -----
K_11 = eye(M)-A^2*P_11;
K_12 = -A^2*P_12;
K_21 = A^2*P_21;
```

```
K_22 = eye(M)+A^2*P_22;
%----- Construct matrix H_ij -----
H_11 = lam*A^2*H11;
H_12 = lam*A^2*H12;
H_21 = lam*A^2*H21;
H_22 = -lam*A^2*H22;
%----- Constuct matrix equation -----
K_v =[K_11-H_11 K_12-H_12; K_21-H_21 K_22-H_22];      % Matrix K_v
Q=[xbar ones(M,1) zeros(M,2);zeros(M,2) xbar ones(M,1)];% Matrix Q
R = [r1; r2; r3; r4];                                  % Matrix R
M0 = [0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0];             % Matrix 0
W = [A^2*f1(xbar); A^2*f2(xbar)];                      % Matrix W
b = [bl1; br1; bl2; br2];                              % Matrix b
Z = [K_v Q; R M0];                   % The LHS Of linear system
B = [W; b];                          % The RHS Of linear system
%% Solve u---------------------------------------------------------
u  = pinv(Z)*B;                      % Numerical Solutions
e1 = ex1(xbar);                      % Analytical solution u1
e2 = ex2(xbar);                      % Analytical solution u2
E1 = mean(abs(e1-u(1:M)))            % Average absolute error of u1
E2 = mean(abs(e2-u(M+1:2*M)))        % Average absolute error of u2
[xbar ex1(xbar) u(1:M) abs(e1-u(1:M))];
[xbar ex2(xbar) u(M+1:2*M) abs(e2-u(M+1:2*M))];
%% Compute u for arbitary x---------------------------------------
x1 = [0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0]';
T1 = @(n,x1) cos(n*acos((2*x1-1.1)/0.9));
for j=1: length(x1)
    for i=0:M-1
        T1x(1,i+1)= T1(i,x1(j));
    end
```

```
    ur1(j) = T1x*Tinv*u(1:M);           % u1 for arbitary x
    ur2(j) = T1x*Tinv*u(M+1:2*M);       % u2 for arbitary x
    er1(j) = abs(ur1(j)-ex1(x1(j)));    % Absolute error of u1
    er2(j) = abs(ur2(j)-ex2(x1(j)));    % Absolute error of u2
end
[x1 er1' er2'];
%% Plot our numerical & analytical solutions----------------------
p1=plot(xbar,e1,'red'); hold on;
p2=plot(xbar,u(1:M),'bo');
figure
p3=plot(xbar,e2,'red'); hold on;
p4=plot(xbar,u(M+1:2*M),'bo');
```

APPENDIX C : Example of MatLab code for solving the system of linear FIDEs.

Example C. We consider Example 4.7

$$u_1''(x) - xu_2'(x) - u_1(x) = (x-2)\sin(x) + \int_0^1 (x\cos(t)u_1(t) - x\sin(t)u_2(t))\,dt,$$

$$u_2''(x) - 2xu_1'(x) + u_2(x) = -2x\cos(x) + \sin(x)\cos(t)\int_0^1 (u_1(t) - u_2(t))\,dt,$$

subject to the initial conditions $u_1(0) = 0$, $u_1'(0) = 1$, $u_2(0) = 1$ and $u_2'(0) = 1$. The exact solutions are $u_1(x) = \sin(x)$ and $u_2(x) = \cos(x)$. Thus, we can construct a linear system in a matrix form as follows

$$\left[\begin{array}{cc|cccc} \mathbf{K}_{11} - \mathbf{H}_{11}' & \mathbf{K}_{12} - \mathbf{H}_{12}' & \mathbf{x}_1 & \mathbf{x}_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{K}_{21} - \mathbf{H}_{21}' & \mathbf{K}_{22} - \mathbf{H}_{22}' & \mathbf{0} & \mathbf{0} & \mathbf{x}_1 & \mathbf{x}_0 \\ \hline \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & \mathbf{0} & 0 & 0 & 0 & 0 \\ \mathbf{t}_{1,r}(\mathbf{T}^*)^{-1} & \mathbf{0} & 0 & 0 & 0 & 0 \\ \mathbf{0} & \mathbf{t}_{0,l}(\mathbf{T}^*)^{-1} & 0 & 0 & 0 & 0 \\ \mathbf{0} & \mathbf{t}_{1,l}(\mathbf{T}^*)^{-1} & 0 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \hline D_{1,1} \\ D_{1,2} \\ D_{2,1} \\ D_{2,2} \end{array}\right] = \left[\begin{array}{c} \mathbf{A}^2\bar{\mathbf{f}}_1 \\ \mathbf{A}^2\bar{\mathbf{f}}_2 \\ \hline 0 \\ 1 \\ 1 \\ 1 \end{array}\right].$$

```
%% Input parameters-----------------------------------------------
m = 2;                             % The higher order derivative
M = 12;                            % The number of nodal points
a = 0;                             % The left boundary
b = 1;                             % The right boundary
lam = 1;                           % Value of lamma_{i,j}
f1  = @(x) (x-2).*sin(x);          % The forcing term f_1(x)
f2  = @(x) -(2*x).*cos(x);         % The forcing term f_2(x)
Kxt11 = @(x,t) x.*cos(t);          % The kernel function K_11(x,t)
Kxt12 = @(x,t) -x.*sin(t);         % The kernel function K_12(x,t)
Kxt21 = @(x,t) sin(x).*cos(t);     % The kernel function K_21(x,t)
Kxt22 = @(x,t) -sin(x).*sin(t);    % The kernel function K_22(x,t)
%% Analytical solutions-------------------------------------------
ex1 = @(x) sin(x);                 % Analytical solution u1(x)
ex2 = @(x) cos(x);                 % Analytical solution u2(x)
bl1  = ex1(a);                     % Value of u1(0)
br1  = ex1(b);                     % Value of u1(1)
bl2  = ex2(a);                     % Value of u2(0)
br2  = ex2(b);                     % Value of u2(1)
%% Compute xbar & tbar in [0,1]-----------------------------------
xbar = flip((cos((2*(1:M)'-1)/(2*M)*pi)+1)/2);
tbar = flip((cos((2*(1:M)'-1)/(2*M)*pi)+1)/2);
%% Integration matrix A-------------------------------------------
%----- construct matrix T* -----
T(:,1) = ones(M,1);
T(:,2) = (2*xbar-1);
for n = 2:M
    T(:,n+1) = 2*(2*xbar-1).*T(:,n)-T(:,n-1);
end
%----- construct matrix (T*)bar -----
```

```
Tbar(:,1) = xbar;
Tbar(:,2) = (xbar).*(xbar-1);
for n = 2:M-1
    Tbar(:,n+1) = 1/4*(T(:,n+2)/(n+1)-T(:,n)/(n-1)-2*(-1)^n/(n^2-1));
end
Tinv = 1/M*diag([1 2*ones(1,M-1)])*T(:,1:M)';
A = Tbar*Tinv;
%% Construct matrix B'*(Kbar)'_ij-----------------------------------
%----- Construct matrix B' -----
Z(1,1) = 1;
Z(1,2) = 0;
for j = 2:M-1
     if mod(j,2)== 0;
         Z(1,j+1)=(1/(1-j^2));
     else
         Z(1,j+1)=0;
     end
 end
B = Z*Tinv;                                    % Matrix B'
%----- Compute B'*(Kbar)'_ij -----
for i = 1:M
    for j = 1:M
        K11(i,j) = Kxt11(xbar(i),xbar(j));
        K12(i,j) = Kxt12(xbar(i),xbar(j));
        K21(i,j) = Kxt21(xbar(i),xbar(j));
        K22(i,j) = Kxt22(xbar(i),xbar(j));
    end
end
for k = 1:M
```

```
        H11(k,:) = B*diag(K11(k,:));
        H12(k,:) = B*diag(K12(k,:));
        H21(k,:) = B*diag(K21(k,:));
        H22(k,:) = B*diag(K22(k,:));
end
%% Boundary conditions--------------------------------------------
tl = (-1).^(0:M-1);
tr = (1).^(0:M-1);
r1 = [tl*Tinv zeros(1,M)];
r2 = [tr*Tinv zeros(1,M)];
r3 = [zeros(1,M) tl*Tinv];
r4 = [zeros(1,M) tr*Tinv];
%% Construct linear system----------------------------------------
%----- Construct matrix P_ij -----
P_120 = diag(xbar);                         % Matrix P_12^(0)
P_121 = eye(M);                             % Matrix P_12^(1)
P_210 = diag(2*xbar);                       % Matrix P_21^(0)
P_211 = eye(M);                             % Matrix P_21^(1)
%----- Construct matrix K_ij -----
K_11 = eye(M)-A^2;
K_12 = -A*P_120+A^2*P_121;
K_21 = -A*P_210+2*A^2*P_211;
K_22 = eye(M)+A^2;
%----- Construct matrix H'_ij -----
H_11 = lam*A^2*H11;                         % Matrix H'_11
H_12 = lam*A^2*H12;                         % Matrix H'_12
H_21 = lam*A^2*H21;                         % Matrix H'_21
H_22 = lam*A^2*H22;                         % Matrix H'_22
%----- Constuct matrix equation -----
K_f = [K_11-H_11 K_12-H_12; K_21-H_21 K_22-H_22];     % Matrix K_f
```

```
Q=[xbar ones(M,1) zeros(M,2);zeros(M,2) xbar ones(M,1)];% Matrix Q
R = [r1; r2; r3; r4];                                  % Matrix R
M0 = [0 0 0 0; 0 0 0 0; 0 0 0 0; 0 0 0 0];             % Matrix 0
W = [A^2*f1(xbar); A^2*f2(xbar)];                      % Matrix W
b = [bl1; br1; bl2; br2];                              % Matrix b
Z = [K_f Q; R M0];                 % The LHS Of linear system
B = [W; b];                        % The RHS Of linear system
%% Solve u-------------------------------------------------------
u  = pinv(Z)*B;                    % Numerical Solutions
e1 = ex1(xbar);                    % Analytical solution u1
e2 = ex2(xbar);                    % Analytical solution u2
E1 = mean(abs(e1-u(1:M)))          % Average absolute error of u1
E2 = mean(abs(e2-u(M+1:2*M)))      % Average absolute error of u2
[xbar ex1(xbar) u(1:M) abs(e1-u(1:M))];
[xbar ex2(xbar) u(M+1:2*M) abs(e2-u(M+1:2*M))];
format long
%% Plot our numerical & analytical solutions----------------------
p1=plot(xbar,ex1(xbar),'red');
hold on
p2=plot(xbar,u(1:M),'bo');
figure
p3=plot(xbar,ex2(xbar),'red');
hold on
p4=plot(xbar,u(M+1:2*M),'bo');
```

BIOGRAPHY

Name	Miss Matinee Juytai
Date of Birth	19 October 1995
Place of Birth	Phrae, Thailand
Education	B.Sc. (Mathematics), Chulalongkorn University, 2014
Publications	The 49th National and International Graduate Research Conference at Nakhon Si Thammarat Rajabhat University