Fall 11-2021

Gradient Preserved And Artificial Neural Network Method For Solving Heat Conduction Equations In Double Layered Structures

Aniruddha Bora

GRADIENT PRESERVED METHOD AND ARTIFICIAL NEURAL NETWORK METHOD FOR SOLVING HEAT CONDUCTION EQUATIONS IN DOUBLE LAYERED STRUCTURES

by

Aniruddha Bora, B.Sc., M.Sc.

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

COLLEGE OF ENGINEERING AND SCIENCE
LOUISIANA TECH UNIVERSITY

November 2021

LOUISIANA TECH UNIVERSITY

GRADUATE SCHOOL

September 24, 2021
Date of dissertation defense

We hereby recommend that the dissertation prepared by

Aniruddha Bora, B.Sc. , MSc.

entitled **GRADIENT PRESERVED AND ARTIFICIAL NEURAL NETWORK METHOD FOR SOLVING HEAT CONDUCTION EQUATIONS IN DOUBLE LAYERED STRUCTURES**

be accepted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Computational Analysis & Modeling

Weizhong Dai

Dr. Weizhong Dai
Supervisor of Dissertation Research

Weizhong Dai

Dr. Weizhong Dai
Head of Computational Analysis & Modeling

Doctoral Committee Members:
Dr. Thomas Bishop
Dr. Pradeep Chowriappa
Dr. Songming Hou
Dr. John Shaw

Approved:
Hisham Hegab

Hisham Hegab
Dean of Engineering & Science

Approved:
Ramu Ramachandran

Ramu Ramachandran
Dean of the Graduate School

GS Form 13a
(01/20)

ABSTRACT

Layered structures have appeared in many engineering systems such as biological tissues, micro-electronic devices, thin films, thermal coating, metal oxide semiconductors, and DNA origami. In particular, the multi-layered metal thin films, gold-coated metal mirrors for example, are often used in high-powered infrared-laser systems to avoid thermal damage at the front surface of a single layer film caused by the high-power laser energy. With the development of new materials, functionally graded materials are becoming of more paramount importance than materials having uniform structures. For instance, in semiconductor engineering, structures can be synthesized from different polymers, which result in various values of conductivity. Analyzing heat transfer in layered structure is crucial for the optimization of thermal processing of such multi-layered materials.

There are many numerical methods dealing with heat conduction in layered structures such as the Immersed Interface Method, the Matched Interface Method, and the Boundary Method. However, development of higher-order accurate stable finite difference schemes using three grid points across the interface between layers for variable coefficient case is mathematically challenging. Having three grid points ensures that the finite difference scheme leads to a tridiagonal matrix that can be solved easily using the Thomas Algorithm. But extension of such methods to higher dimensions is very tedious. Recently there have been some solution to such complex systems

with the use of neural networks, that can be easily extended to higher dimensions. For the above purposes, in this dissertation, we first develop a gradient preserved method for solving heat conduction equations with variable coefficients in double layers. To this end, higher-order compact finite difference schemes based on three grid points are developed. The first-order spatial derivative is preserved across the interface. Unconditional stability and convergence with $O(\tau^2 + h^4)$ are analyzed using the discrete energy method, where τ and h are the time step and grid size, respectively. Numerical error and convergence rates are tested in an example. We then present an artificial neural network (ANN) method for solving the parabolic two-step heat conduction equations in double-layered thin films exposed to ultrashort-pulsed lasers. Convergence of the ANN solution to the analytical solution is theoretically analyzed using the energy method. Finally, both developed methods are applied for predicting electron and lattice temperature of a solid thin film padding on a chromium film exposed to the ultrashort-pulsed lasers. Compared with the existing results, both methods provide accurate solutions that are promising.

APPROVAL FOR SCHOLARLY DISSEMINATION

The author grants to the Prescott Memorial Library of Louisiana Tech University the right to reproduce, by appropriate methods, upon request, any or all portions of this dissertation. It is understood that "proper request" consists of the agreement, on the part of the requesting party, that said reproduction is for his personal use and that subsequent reproduction will not occur without written approval of the author of this dissertation. Further, any portions of the dissertation used in books, papers, and other works must be appropriately referenced to this dissertation. Finally, the author of this dissertation reserves the right to publish freely, in the literature, at any time, any or all portions of this dissertation.

Author ______________________________

Date ______________________________

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENTS

The degree of Doctor of Philosophy is a journey. I am most thankful to my advisor and mentor, Dr. Weizhong Dai. He helped me to be a more mature person in life, both mathematically and socially. He made me understand the true ideas behind mathematics and gave me academic direction. No matter how much I thank him, it will never be enough. Not only he is my academic advisor/mentor but, he also acted like my family and never let me feel I was away from home.

Next, I would like to thank my advisory committee members: Dr. Thomas Bishop, Dr. John Shaw, Dr. Pradeep Chowriappa and Dr. Songming Hou for their valuable input related to the research of this dissertation and their recommendations. Also, I would like to thank Dr. Joshua Wilson and Dr. Yun Yan, who provided me with much valuable advice related to research and programming and for all the long discussions we had during our meetings. They all are like family to me.

I am very grateful towards the College of Engineering and Science, Louisiana Tech University, especially Dr. Collin Wick and Dr. B. Ramu Ramachandran for providing me with assistantship and other scholarships that helped me complete my PhD.

I am also very thankful to the International Student Association for their assistance in so many scenarios, especially Mr. Daniel Erickson, Mr. Jay Ligon and

Miss Van Tran. I am very thankful to Dr. Paula Brown, who gave me her valuable time for the dissertation English checks.

I would like to especially thank Dr. Ranjan Kumar Mohanty, without whose guidance, support, advice and recommendation, I would never have come to the USA. Because of him I came to Louisiana Tech University in the first place. Also, I would like to thank my mentors and guides, Hemanta Saikia Sir, Dr. Sinam Iboton Singh, Dr. Biswajit Banerjee and Dr. Bamdev Dey, who supported and guided me in various important aspects of my life.

Also, I would like to thank my parents, Mr. Arindam Bora (Father) and Mrs. Anjana Bora (Mother), for their motivation and support, without whom I would have not been where I am today. Also, my father is the reason I got introduced and motivated towards science. It was his scientific temperament that made me go for research and academics. I would like to thank my brother, Arunangshu Bora, for his exchange of ideas, which in many ways gave me a new direction of thinking. I would like to thank my fiance, Miss Zahra Farazpay, for her kindness and support throughout. Also, I would like to thank Mrs. Mahbube Dalili and Mrs. Ezzat Alijaniya as well as my friends Dr. Anik Karan, Shashank Reddy, Harsha Manoj Dannina, Narendra Kumar, Ahmed Humayun and Supratim Bhatacharjee for their valuable discussions and insights into many endless topics of academic/social interests and also for their financial assistance. I would also like to thank my uncle and aunty, Mr. Kalpesh Daya and Mrs. Rekha Daya, who helped and supported me both morally and financially a lot during this journey. And finally, I would like to thank Mukondo borta, Bhaiti khura, Pinku khura and Dudu khura, who helped me a lot during the visa process.

Lastly, I would like to thank all the people that came in touch with me in every aspect of my life and motivated me one way or the other.

CHAPTER 1

INTRODUCTION

1.1 General Overview

Structures having layers are used very commonly in engineering systems and applications. Some examples where researchers use layered structures are thermometric power conversion, thermal coating, metal oxide semiconductors, biological tissues, micro-electronic devices, thin films, reactor walls, and thermal processing of DNA origami nano structures, etc [1-8]. When dealing with multilayered structure and thermal processing, lasers, especially ultrashort-pulsed lasers, are important tools. They have wide applications in biology, chemistry, medicine, physics, and optical technology due to their high efficiency, high power density, minimal collateral material damage, lower ablation thresholds, high precision production ability, and high-precision control of heating times and locations in thermal processing of materials [9]. This technology in particular has been extensively used in thermal processing of materials, such as structural monitoring of thin metal films and laser processing in thin-film deposition [9]. During ultrashort pulsed laser heating, especially when involving metals, the thermal conductivity varies with time because of its dependence in electron and lattice temperature. Also, with the development of new materials nowadays, functionally graded materials are becoming more important than materials having uniform structures [10, 11, 12]. For example, in semiconductor engineering, structures can be synthesized from different polymers. This results in variable conductivity. Therefore, thermal analysis in layered structures is crucial for the design and operation of devices and the optimization of thermal processing of materials.

There are many numerical methods dealing with the thermal analysis of layered structures; the Peskin Immersed Boundary Method [13-18], the Immersed Interface Method [19-31], the Ghost Fluid Method [32-34], the Matched Interface and Boundary Method [35-42], the Immersed Finite Element method [43-49], the Petrov-Galerkin Finite Element Method [50-53], and body-fitting approaches [54, 55], as well as the summation-by-parts operator with simultaneous approximation terms for time-dependent problems [56-60]. However, these methods across the interface usually provide only a second-order truncation error when using three-grid points. This reduces the accuracy of the overall numerical solution even if the higher-order Compact Finite Difference method is used at other points. If not using three grid points the resulting matrix from finite difference scheme becomes very complex and time consuming to solve, whereas having three grid points makes the matrix tridiagonal which can be easily solved using the Thomas Algorithm. Coming up with three point grid scheme, is mathematically very challenging specially for the interface. Dai and his collaborators recently [61] have developed the Gradient Preserved Method (GPM). This is a higher-order accurate finite difference method that uses three grid points across the interface between layers by preserving the first-order derivative, u_x, in the interfacial condition and/or the boundary condition. By coupling it with the three grid points in space and fourth-order accurate Padé scheme [62] at interior points, an accurate, stable, and convergent scheme has been obtained for solving heat conduction equations with constant coefficients in double layers. But in many cases such as functionally gradient materials, and also when dealing with ultrashort pulsed lasers, the thermal conductivity is not constant. To incorporate variable thermal conductivity into the finite difference scheme, while maintaining three grid points is mathematically very challenging, even in one spatial dimension. With the advancement of computation power, new directions for solving problems such as data driven scientific computing, artificial intelligence, machine, and deep learning techniques have emerged [64-72]. Dr. Karniadakis and his colleagues [73] recently have introduced the idea of Physics Informed Neural Nets (PINN) for solving physics-based problems. The PINN method consists of a fully connected deep neural network whose output is considered as the solution of the equation based on the equation that we want to solve. The loss function consists of the equation along with the initial and boundary conditions, which are used to optimize the weights and biases in the neural network solution. The iteration continues until the loss function attains a small enough value to obtain an accurate neural network solution. This method although is not tedious as the finite difference method, is much slower in computation. However the computation speed can be greatly increased with the implementation of GPU and parallel computation. Also, this method can be easily extended to higher dimension with just some minor changes which is not the case for finite difference methods. But one of the issues in general for neural networks is its failure to capture high shock values or sharp discontinuities. Such high shock values are common during thermal processing of materials. For example, heating with an ultrashort-pulsed laser involves high-rate heat flow from electrons to lattices within picoseconds. When heated by photons (lasers), the free electrons that are confined within skin depth primarily absorb the laser energy and get excited. Within a few picoseconds, electrons shoot up to several hundreds or thousands of degrees without the metal lattices getting disturbed. A major portion of the thermal electron energy afterwards is transferred to the lattices; meanwhile another part of the energy diffuses to the electrons that are in the deeper region of the material. Since the pulse duration is so short, the laser is turned off before thermal equilibrium between electrons and lattices is reached. In this time interval, this stage is often called non-equilibrium heating due to the large difference of temperatures in electrons and lattices [74]. This kind of problem requires an accurate numerical method, and using a neural network method is challenging.

1.2 Research Objective and Organization

The objective of this dissertation is to propose two computational techniques for solving heat conduction with variable coefficients in double layers. For this purpose, the first method in this dissertation extends the GPM to the variable coefficient case (even temperature-dependent coefficients). Developing a higher-order accurate and stable finite difference scheme using three grid points across the interface between layers for the variable coefficient case is much more mathematically challenging than that for the constant coefficient case. We aim at obtaining a stable and convergent Compact Finite Difference scheme for solving the heat conduction equation having variable coefficients in double layers. We then apply it to the parabolic two-temperature heat equations for predicting the electron and lattice temperature in double-layered thin film exposed to ultrashort-pulsed lasers. For the second method, we aim to create a neural network method based on Physics Informed Neural Network (PINN), that is able to capture high shock values efficiently. We then apply it to solving the parabolic two-temperature heat conduction equations in double-layered thin film exposed to ultrashort-pulsed lasers. Finally, we compare our computational results with existing references.

The organization of the rest of dissertation is as follows. **Chapter 2** provides a background review related to this research. **Chapter 3** proposes the Gradient Preserve Method for heat conduction equations with variable coefficients in double layers. **Chapter 4** proposes the Neural Network Method for solving the parabolic two-temperature heat conduction equations in double-layered thin film exposed to ultrashort-pulsed lasers. **Chapter 5** tests both the GPM and the Neural Network method for thermal analysis in a gold layer padding on a chromium layer exposed to ultrashort-pulsed lasers, and compares with existing references. **Chapter 6** concludes the dissertation and discusses some directions for future research.

CHAPTER 2

BACKGROUND REVIEW

This chapter provides the related research background for the research in this dissertation. Also, it discusses the previous work done related to this dissertation.

2.1 Heat Conduction Equations

A heat conduction equation in mathematics and physics represents a particular partial differential equation that deals with the flow of heat. Joseph Fourier in 1822 gave the first theory of heat equation[75]. It is also known as Fourier's law, and it states that: " the heat flux q, resulting from thermal conduction, is proportional to the magnitude of the temperature gradient and opposite to it in sign" [76]. If k is the proportionality constant then mathematically,

$$q(\vec{X}, t) = -k\nabla T(\vec{X}, t). \tag{2.1}$$

The SI unit of q is Wm^{-2} (Watt per meter square). The constant k is known as the thermal conductivity with SI unit $Wm^{-1}K^{-1}$ (Watt per metre Kelvin). $\vec{X}$ is the spatial vector, t is time, and ∇T is the temperature gradient with SI unit Km^{-1} (Kelvin per metre). This law of Fourier has other equivalent forms. The discrete analogue form is Newton's law of cooling; the electrical analogue is Ohm's law, and the chemical analogue is Fick's laws of diffusion.

2.1.1 Heat conduction equation with constant coefficients

The thermal conductivity k is usually considered as a constant (which is always the case). The Fourier law from Eq. (2.1) in one dimension can be written as:

$$\frac{Rate\ of\ heat\ transfer}{area} = -k\frac{\partial T}{\partial x}. \tag{2.2}$$

To obtain the heat equation for a material with constant thermal conductivity k, in the form of a rod with uniform cross section [77, 78], we let the density of the material be ρ, specific heat be c, and cross section area be A. Consider a very small arbitrary element of the rod of length Δx and assume that the temperature throughout the element is $T(x,t)$. Then, the heat energy needed in this small segment in order to raise the temperature $T(x,t)$ degree can be calculated as:

$$heat\ energy\ of\ segment = c\rho A\Delta x T(x,t). \tag{2.3}$$

By the conservation of energy, we have:

$$\begin{aligned}&change\ of\ heat\ energy\ of\ segment\ in\ time\ \Delta t\\ &= heat\ flow\ in\ from\ left\ boundary - heat\ flow\ out\ from\ right\ boundary.\end{aligned}$$

Therefore, from Eq. (2.2) and (2.3) we have:

$$c\rho A\Delta x T(x,t+\Delta t) = \Delta t A\left(k\frac{\partial T(x,t)}{\partial x} - k\frac{\partial T(x+\Delta x,t)}{\partial x}\right), \tag{2.4a}$$

$$\frac{T(x,t+\Delta t)-T(x,t)}{\Delta t} = \frac{k}{c\rho}\left(\frac{\frac{\partial T}{\partial x}|_{x+\Delta x} - \frac{\partial T}{\partial x}|_{x}}{\Delta x}\right). \tag{2.4b}$$

Taking the limit Δt, $\Delta x \to 0$, gives the following equation:

$$\frac{\partial T(x,t)}{\partial t} = K\frac{\partial^2 T(x,t)}{\partial x^2}, \tag{2.5}$$

where $K = k/(c\rho)$ is called the thermal diffusivity. This equation is known as the heat conduction equation with constant coefficient thermal conductivity. To obtain the temperature $T(x,t)$, we need more information. This information includes:

1. initial condition (the initial temperature information $T(x,0)$ of the material),
2. boundary condition (the temperature condition on the boundaries of the material).

There are three types of boundary conditions: (a) Dirichlet boundary condition, (b) Neumann boundary condition, and (c) Robin boundary condition. The Dirichlet boundary condition specifies the temperature at the boundary for $t \geq 0$. The Neumann boundary condition gives the heat flux information on the boundary in the form of spatial derivative $(\partial T/\partial x)$. The Robin boundary condition gives us information in the form of an equation combining both the temperature T and its spatial derivative $(\partial T/\partial x)$. Thus, the one dimensional heat conduction problem can be written as:

$$\frac{\partial T}{\partial t} = K\frac{\partial^2 T}{\partial x^2}, \quad 0 \leq x \leq L, \;\; t > 0, \tag{2.6}$$

with initial condition

$$T(x,0) = \alpha, \quad 0 \leq x \leq L, \tag{2.7}$$

and one of the boundary conditions (Dirichlet, Neumann, or Robin)

$$\mathit{Dirichlet}: \qquad T(0,t) = \beta_1, T(L,t) = \beta_2, \;\; t > 0 \tag{2.8a}$$

$$\mathit{Neumann}: \qquad \frac{\partial T(0,t)}{\partial x} = \gamma_1, \frac{\partial T(L,t)}{\partial x} = \gamma_2, \;\; t > 0 \tag{2.8b}$$

$$\mathit{Robin}: \qquad aT(0,t) + b\frac{\partial T(0,t)}{\partial x} = \eta_1, aT(L,t) + b\frac{\partial T(L,t)}{\partial x} = \eta_2, \;\; t > 0 \tag{2.8c}$$

where a and b are constants.

The heat conduction equation with constant thermal conductivity can be extended to n-dimensions $(x_1, x_2, ..., x_n)$ as :

$$\frac{\partial T}{\partial t} = K \sum_{n=1}^{n} \frac{\partial^2 T}{\partial x_i^2}. \tag{2.9}$$

If there is a source term involved in the heat conduction equation, such as an external source of heat that is being used to heat the material, Eq. (2.9) becomes

$$\frac{\partial T}{\partial t} = K \sum_{i=1}^{n} \frac{\partial^2 T}{\partial x_i^2} + F(\mathbf{X}, t), \tag{2.10}$$

where $\mathbf{X} = x_1, x_2, ..., x_n$. Here, $F(\mathbf{X}, t)$ is the source term.

2.1.2 Heat conduction equation with variable coefficients

The case discussed above has a fixed thermal conductivity k. In most cases the thermal conductivity may not be constant because it may depend on a lot of factors such as the density of material, non uniformity of material, ambient temperature, moisture of material, etc. Nowadays with the advancement of technology in manufacturing industry, Functionally Graded Materials (FGMs) have become of paramount importance. FGMs are characterized by spatially variable microstructures, are heterogeneous materials that have spatially variable macroscopic properties to enhance the material or its structural performance [79]. The concept of modern man-made FGMs was proposed first by material scientists in 1984 in Japan as a means of developing thermal protection materials.

To obtain the temperature in such variable thermal conductivity structures, the heat equation with variable thermal conductivity in one dimension can be changed to:

$$\frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k(x)\frac{\partial T}{\partial x}\right) + F(x,t), \tag{2.11}$$

where $k(x)$ is the spatially varying thermal conductivity. The initial conditions and the boundary conditions are developed similarly as discussed for the constant thermal conductivity case.

For n-dimension form, Eq. (2.11) can be extended as:

$$\frac{\partial T}{\partial t} = \sum_{i=1}^{n} \frac{\partial}{\partial x_i}\left(k(\mathbf{X})\frac{\partial T}{\partial x_i}\right) + F(\mathbf{X},t), \tag{2.12}$$

where $\mathbf{X} = x_1, x_2, ..., x_n$.

2.1.3 Heat conduction equation in double layers

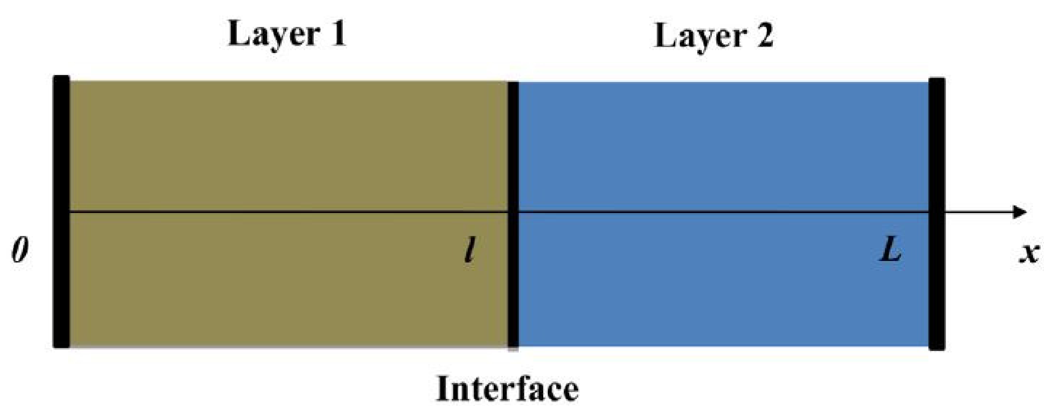


Figure 2.1: Double-layered structure.

A Heat conduction problem in a single layer can be solved by having the heat conduction equation with the initial and boundary conditions. But when it comes to double layers apart from the initial and boundary conditions, we need additional information on the interface between layers to solve the problem. The interfacial conditions are mainly based on heat and mass conservation principles. For perfectly thermal contact cases, it requires that:

1. the same temperature at the area of contact be maintained between the two layers,
2. the heat flux at the surface of the first material must be the same as the heat flux at the surface of the second material, because the heat flux flows only from one surface to the other.

Mathematically, these two conditions give the following interficial conditions:

$$T(x_{l+}, t) = T(x_{l-}, t), \qquad t \geq 0, \tag{2.13a}$$

$$k_2 \frac{\partial T(x_{l+}, t)}{\partial x} = k_1 \frac{\partial T(x_{l-}, t)}{\partial x}, \qquad t \geq 0, \tag{2.13b}$$

where k_1 and k_2 are the thermal conductivity for layer 1 and layer 2, respectively, as shown in Figure **??**. Here x_{l-} and x_{l+} represent the locations in the interface in layer 1 and layer 2, respectively. For the variable coefficient case, k_1 and k_2 in Eq, (2.13b) are replaced with variable functions $k_1(x)$ and $k_2(x)$.

2.1.4 Parabolic two-temperature heat conduction equations in double layers

Laser heating of materials is most commonly given by the heat conduction model [79]:

$$C \frac{\partial T}{\partial t} = \nabla(k \nabla T) + S, \tag{2.14}$$

where C is the volumetric heat capacity, k is the thermal conductivity, and S is the laser heat source. This model has two underlying assumptions:

1. radiation energy is instantaneously converted into lattice energy

2. energy transfer in solids is a diffusion process.

The above equation is called the Parabolic One Step (POS) model. For ultrashort-pulsed laser heating, the above basic assumptions may not be accurate. The wave type of propagation in heat has been suggested in [80-98], which gives us the Hyperbolic One-Step radiating heating model (HOS):

$$C\frac{\partial T}{\partial t} = -\nabla Q + S \tag{2.15a}$$

$$\tau\frac{\partial Q}{\partial t} + k\nabla T + Q = 0, \tag{2.15b}$$

where Q is the heat flux and τ is the relaxation time of free electrons in metal. In 1974, S. I. Anisimov, B. L. Kapeliovich and T. L. Perelman suggested that the conversion of radiation energy into internal energy is not instantaneous [91]. It involves two energy-deposition steps:

1. radiation heating of free electrons
2. the subsequent energy redistribution between electrons and the metal lattice.

They proposed a Parabolic Two-Temperature radiation heating model (PTTM) given by:

$$C_e(T_e)\frac{\partial T_e}{\partial t} = \nabla(k\nabla T_e) - G(T_e - T_l) + S, \tag{2.16a}$$

$$C_l(T_l)\frac{\partial T_l}{\partial t} = G(T_e - T_l), \tag{2.16b}$$

where T_e and T_l are the electron and lattice temperature, C_e and C_l are electron and lattice heat capacity respectively, G is the electron lattice coupling factor, and S is the laser source term. Qui and Tien in 1994 [100] proposed the hyperbolic two-temperature model (HTTM):

$$C_e(T_e)\frac{\partial T_e}{\partial t} = -\nabla Q - G(T_e - T_l) + S, \tag{2.17a}$$

$$C_l(T_l)\frac{\partial T_l}{\partial t} = G(T_e - T_l), \tag{2.17b}$$

$$\tau\frac{\partial Q}{\partial t} + \frac{T_e}{T_l}k\nabla T_e + Q = 0. \tag{2.17c}$$

This model removes the assumption of instantaneous radiation deposition as well as diffusive energy transport in the POS model.

Energy transfer mechanisms during laser heating are determined by the electron-lattice thermalization time and the electron relaxation time. It can be seen from Figures 3. and 4. in [80] that the thermalization time very weakly depends on temperature, and especially at low temperatures and the electron relaxation time is extremely sensitive to the lattice temperature. At high temperatures, the thermalization time is longer than the relaxation time, which indicates that the effect of two-temperature non-equilibrium heating is stronger than that of hyperbolic transport. The situation is reversed at lower temperatures. For slow heating processes, the POS model is applicable, whereas for the fast heating processes the PTTM model is applicable at relatively high temperatures, and for low-temperature and fast heating processes the HOS model is applicable. In certain low-temperature and fast heating regimes, the HTTM model must be used. For more details, we refer to [79]. Keeping this idea in mind, Qui and Tien in 1994 [79], gave the parabolic two temperature model as simplified from the HTTM model:

$$C_e\frac{\partial T_e}{\partial t} = \frac{\partial}{\partial x}\left(k\frac{T_e}{T_l}\frac{\partial T_e}{\partial x}\right) - G(T_e - T_l) + S, \tag{2.18a}$$

$$\frac{\partial T_l}{\partial t} = G(T_e - T_l), \tag{2.18b}$$

where the electron heat capacity is proportional to the electron temperature, and the thermal conductivity is modified by the ratio of the electron temperature and the lattice temperature.

2.2 Numerical Methods

The main idea behind numerical methods or simulations is discretization. For this purpose, a continuous problem has to be changed into a discrete problem, based on which the computational domain will change into a mesh or grid that has multiple cells or elements. This discrete problem will be in the form of algebraic equations. Discretization of the domain and the equations are co-related.

2.2.1 Compact finite difference method

Let us consider an example where the derivative of a function at a point x is given:

$$f'(x) = \lim_{h \longrightarrow 0} \frac{f(x+h) - f(x)}{h}. \tag{2.19}$$

Therefore, the difference quotient $(f(x+h) - f(x))/h$ is an approximation of the derivative $f'(x)$. This approximation gets better as h gets smaller. To know how far this estimation is from the actual value, we use the Taylor theorem:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}f''(\eta). \quad \eta \in [x, x+h] \tag{2.20}$$

Rearranging the above equation gives us:

$$\frac{f(x+h) - f(x)}{h} - f'(x) = \frac{h}{2}f''(\eta). \tag{2.21}$$

Eq. (2.21) tells us that error is proportional to h^1. Therefore, $(f(x+h)-f(x))/h$ gives us first order approximation of $f'(x)$. If $h=\Delta x$ is a finite positive number then $(f(x+\Delta x)-f(x))/\Delta x$ is known as the first-order or $O(\Delta x)$ forward approximation of $f'(x)$. Similarly, if $h=-\Delta x$, which gives $(f(x)-f(x-\Delta x))/\Delta x$ is the first-order backward approximation of $f'(x)$. In addition, $f'(x)$ can be approximated by the symmetric difference quotient $(f(x+\Delta x/2)-f(x-\Delta x/2))/(\Delta x)$, expanding which we get:

$$\begin{aligned}\frac{f(x+\Delta x/2)-f(x-\Delta x/2)}{\Delta x} &= \frac{1}{\Delta x}(f(x)+\frac{\Delta x}{2}f'(x)+\frac{\Delta x^2}{8}f''(\eta)+\frac{\Delta x^3}{48}f'''(\eta) \\ &\quad - f(x)+\frac{\Delta x}{2}f'(x)-\frac{\Delta x^2}{8}f''(\eta) \\ &\quad + \frac{\Delta x^3}{48}f'''(\eta)), \qquad (2.22a) \\ &= f'(x)+\frac{\Delta x^2}{24}f'''(\eta), \qquad (2.22b)\end{aligned}$$

where η is a number in the interval $[x-\Delta x/2, x+\Delta x/2]$. Here, one may see that the approximation is of second order or $O(\Delta x^2)$ as the error is directly proportional to Δx^2. This error is called the truncation error. Therefore, different ways or schemes can be used to approximate the same derivative, leading to different orders of accuracy. This kind of descretization technique to solve the partial differential equation is known as the finite difference method. Here are some of the commonly used second and fourth order finite difference formulas for approximating first and second-order derivatives, $O\left(\Delta x^2\right)$ centered difference approximations:

$$f'(x) : \frac{f(x+\Delta x)-f(x-\Delta x)}{2\Delta x}, \qquad (2.23a)$$

$$f''(x) : \frac{f(x+\Delta x) - 2f(x) + f(x-\Delta x)}{\Delta x^2}; \tag{2.23b}$$

$O(\Delta x^2)$ forward difference approximations:

$$f'(x) : \frac{-3f(x) + 4f(x+\Delta x) - f(x+2\Delta x)}{2\Delta x}, \tag{2.24a}$$

$$f''(x) : \frac{2f(x) - 5f(x+\Delta x) + 4f(x+2\Delta x) - f(x+3\Delta x)}{\Delta x^3}; \tag{2.24b}$$

$O(\Delta x^2)$ backward difference approximations:

$$f'(x) : \frac{3f(x) - 4f(x-\Delta x) + f(x-2\Delta x)}{2\Delta x}, \tag{2.25a}$$

$$f''(x) : \frac{2f(x) - 5f(x-\Delta x) + 4f(x-2\Delta x) - f(x-3\Delta x)}{\Delta x^3}; \tag{2.25b}$$

$O(\Delta x^4)$ centered difference approximations:

$$f'(x) : \frac{-f(x+2\Delta x) + 8f(x+\Delta x) - 8f(x-\Delta x) + f(x-2\Delta x)}{12\Delta x}, \tag{2.26a}$$

$$f''(x) : \frac{-f(x+2\Delta x) + 16f(x+\Delta x) - 30f(x) + 16f(x-\Delta x) - f(x-2\Delta x)}{12\Delta x^2}. \tag{2.26b}$$

In general, we may obtain O(Δx^n) accuracy using the formula as

$$f'(x)_i \cong \sum_{i=-n}^{n} \alpha_i f(x)_{i-n} + O(\Delta x^n), \tag{2.27}$$

where $x_{i\pm1} = x \pm \Delta x$. However, we may need to know at least $2n+1$ values of $f(x)$. This creates a lot of difficulties when solving the partial differential equation near boundaries because the function values outside the boundary are unknowns. In 1992, Lele proposed a new method called the Compact Finite Difference Method to overcome this problem[62]. This method uses as few grid-points as possible to obtain the higher order possible. For example, let us consider the 1-D heat equation:

$$\frac{\partial u}{\partial t} = k\frac{\partial^2 u}{\partial x^2} + F(x,t), \qquad 0 \le x \le b, \quad 0 \le t \le T \tag{2.28}$$

with initial and boundary conditions

$$u(x,0) = u_0(x), \qquad u_x(0,t) = u_x(b,t) = 0, \tag{2.29}$$

where $u(x,t)$ is the temperature at location x and time t. Now let us discretize the domain in the x-direction into N points with equal grid size Δx and in the t-direction with time increment Δt into M points such that:

$$\Delta x = \frac{b-0}{N}, \qquad \Delta t = \frac{T-0}{M}, \tag{2.30}$$

Figure 2.2 shows the general structure of discretized mesh used in the finite difference method.

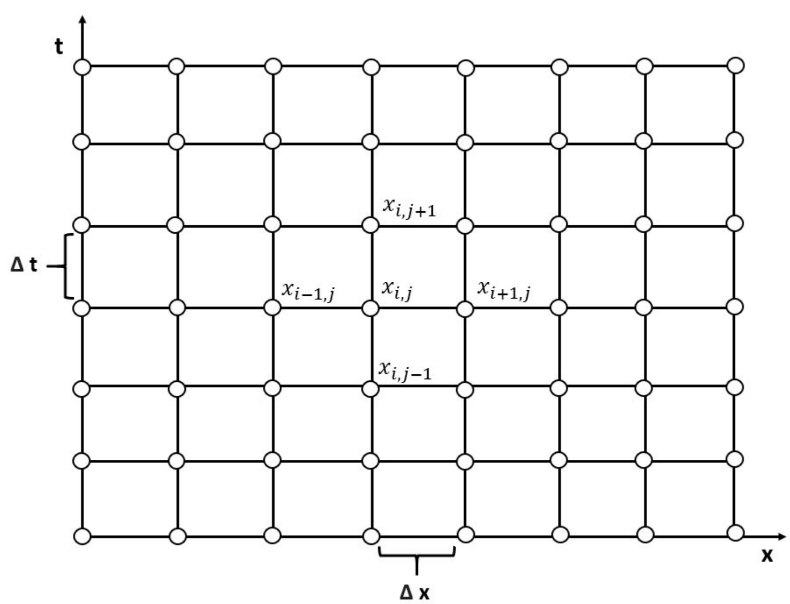


Figure 2.2: General structure of a mesh for finite difference method

We use the Compact Finite Difference formula as given in [63],

$$\frac{1}{6}(5\delta_t u_0^{n+\frac{1}{2}} + \delta_t u_1^{n+\frac{1}{2}}) = k\frac{2}{(\Delta x)^2}(u_1^{n+\frac{1}{2}} - u_0^{n+\frac{1}{2}}) + \frac{1}{6}(5F_0^{n+\frac{1}{2}} + F_1^{n+\frac{1}{2}})$$

$$+ \frac{\Delta x}{6}(F_x)_0^{n+\frac{1}{2}} + O(\Delta x^3 + \Delta t^2), \tag{2.31a}$$

$$\frac{1}{12}(\delta_t u_{i-1}^{n+\frac{1}{2}} + 10\delta_t u_i^{n+\frac{1}{2}} + \delta_t u_{i+1}^{n+\frac{1}{2}}) = \delta_x^2 u_i + \frac{1}{12}(F_{i-1}^{n+\frac{1}{2}} + 10F_i^{n+\frac{1}{2}} + F_{i+1}^{n+\frac{1}{2}})$$

$$+ O(\Delta x^4 + \Delta t^2), \quad 1 \le i \le N-1 \tag{2.31b}$$

$$\frac{1}{6}(\delta_t u_{N-1}^{n+\frac{1}{2}} + 5\delta_t u_N^{n+\frac{1}{2}}) = -k\frac{2}{(\Delta x)^2}(u_N^{n+\frac{1}{2}} - u_{N-1}^{n+\frac{1}{2}}) + \frac{1}{6}(5F_{N-1}^{n+\frac{1}{2}} + F_N^{n+\frac{1}{2}})$$

$$+ \frac{\Delta x}{6}(F_x)_{N-1}^{n+\frac{1}{2}} + O(\Delta x^3 + \Delta t^2), \tag{2.31c}$$

where

$$\delta_t u_i^{n+\frac{1}{2}} = \frac{1}{\Delta t}(u_i^{n+1} - u_i^n), \quad u_i^{n+\frac{1}{2}} = 1/2(u_i^{n+1} + u_i^n), \quad 0 \le i \le N, \tag{2.32}$$

$$\delta_x^2 u_i^n = \frac{1}{\Delta x^2}(u_{i+1}^n - 2u_i^n + u_{i+1}^n), \quad 0 \le i \le N. \tag{2.33}$$

Now this system forms a tridiagonal system for solving u_i^{n+1} $(i = 0, 1, \ldots, N)$ as

$$\begin{bmatrix} b_0 & -c_0 & 0 & 0 & 0 & 0 \\ -a_1 & b_1 & -c_1 & 0 & 0 & 0 \\ 0 & -a_2 & b_2 & -c_2 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & -a_{N-1} & b_{N-1} & -c_{N-1} \\ 0 & 0 & 0 & 0 & -a_N & b_N \end{bmatrix} \begin{bmatrix} u_0^{n+1} \\ u_1^{n+1} \\ u_2^{n+1} \\ \vdots \\ u_{N-1}^{n+1} \\ u_N^{n+1} \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_{N-1} \\ d_N \end{bmatrix},$$

where

$$b_0 = \frac{5}{6} + k\frac{\Delta t}{\Delta x^2}, \quad c_0 = -\frac{1}{6} + k\frac{\Delta t}{\Delta x^2}, \tag{2.34a}$$

$$d_0 = [\frac{5}{6} - k\frac{\Delta t}{\Delta x^2}]u_0^n + [\frac{1}{6} + k\frac{\Delta t}{\Delta x^2}]u_1^n + \frac{\Delta t}{12}[5(F_0^{n+1} + F_0^n)$$

$$+ (F_1^{n+1} + F_1^n)] + \frac{\Delta x \Delta t}{12}((F_x)_0^{n+1} + (F_x)_0^n), \tag{2.34b}$$

$$a_i = -\frac{1}{12} + \frac{\Delta t}{\Delta x^2}, \quad b_i = \frac{5}{6} + k\frac{\Delta t}{\Delta x^2}, \quad c_i = -\frac{1}{12} + k\frac{\Delta t}{\Delta x^2}, \tag{2.35a}$$

$$d_i = [\frac{1}{12} + \frac{\Delta t}{\Delta x^2}]u_{i-1}^n + [\frac{5}{6} - k\frac{\Delta t}{\Delta x^2}]u_i^n + [\frac{1}{12} + \frac{\Delta t}{\Delta x^2}]u_{i+1}^n$$

$$+ \frac{\Delta t}{24}[(F_{i-1}^{n+1} + F_{i-1}^n) + 10(F_i^{n+1} + F_i^n) + (F_{i+1}^{n+1} + F_{i+1}^n)], \tag{2.35b}$$

$$a_N = -\frac{1}{6} + k\frac{\Delta t}{\Delta x^2}, \quad b_N = \frac{5}{6} + k\frac{\Delta t}{\Delta x^2}, \tag{2.36a}$$

$$d_N = [\frac{1}{6} + k\frac{\Delta t}{\Delta x^2}]u_{N-1}^n + [\frac{5}{6} - k\frac{\Delta t}{\Delta x^2}]u_N^n + \frac{\Delta t}{12}[5(F_N^{n+1} + F_N^n)$$

$$+ (F_N^{n+1} + F_N^n)] + \frac{\Delta x \Delta t}{12}((F_x)_N^{n+1} + (F_x)_N^n). \tag{2.36b}$$

This tridiagonal matrix along with the information in Eq. (2.39) can be easily solved using the Thomas Algorithm. The above described method is a Compact Finite Difference Method and has a trunction error of $O(\Delta x^4 + \Delta t^2)$. It is compact because it uses just information from three grid points at a single time level for the interior points and two grid points at a single time level at the boundary.

2.2.2 Numerical methods for interface

Obtaining an accurate solution for interfacial problems is challenging. There has been a lot of research done in addressing the interfacial problems. In 1977, the

Immersed Boundary Method (IBM) was proposed by Peskin to simulate the blood flow in the heart [10-12]. The interfacial problem here was introduced from the singular source at the time-varying boundary. The main idea in IBM is the use of a discrete delta function to distribute a singular source to nearby grid points [14]. This method is first order accurate in the interface. Due to the simplicity, efficiency and robustness of the IBM, it has been used for many applications [104-106]. Later, Peskin and his co-worker proposed higher order versions of IBM [9-10]. A globally fourth-order scheme for problems with singular sources was developed by Tornberg and Engquist at the interface [105]. They obtained the scheme by using some sophisticated discrete delta functions with a narrow support.

For solving elliptic equations with interface problems, LeVeque and Li proposed the Immersed Interface Method (IIM) in 1994 [23]. IIM uses the Taylor Series Expansion Technique near the grid points in the interface. It then utilizes the interface condition to determine the weights for these points. The local truncation error at the interface is of first order, but the overall accuracy of IIM is second-order. In the following years many methods have been improved the original IIM. Some of them are the multi-grid method [20], the discrete maximum principle [24], a fast iterative algorithim for problems with piecewise constant coefficient [26]. Moving interfaces have also been solved using IIM with the level set approach [20, 21]. The IIM has a lot of applications [106-108]. IIM was explained in detail by Li and Ito [?].

The Ghost Fluid Method (GFM) was proposed by Merriman and his colleagues in 1994 [32]. Due to its simplicity, GFM has been widely used even though it is a first order method. In this method, the interface jump conditions are captured implicitly by assuming the interface extends to the other layer.

The Matched Interface and Boundary (MIB) method was developed by Wei and his colleagues in 2006 [35, 39]. It was mainly developed for the purpose of simulating electromagnetic wave scattering and propagation. It was formulated based on the Tensor Product Derivative Matching Method [110]. The MIB can be viewed as the generalization of the IBM, IIM and GFM techniques for solving elliptical interface problems. In this method, fictitious values are extended to both sides of the interface. These values are extrapolated numerically by enforcing the boundary condition, and the number of these values are determined by the order of the Central Finite Difference Method. The MIB Method can also be used without fictitious values based on interpolation formulation as described in [39].

By generalizing MIB, GFM and IIM techniques, Pan and his colleagues developed an interpolation matched Interface and Boundary (IMIB) Method [109]. This method is of second-order accuracy. All methods discussed above have been extended to deal with time dependent interface problems. Methods to solve time dependent interface problems by using summation-by-parts operators with simultaneous approximation were described in [110, 111]. These methods can be utilized to derive higher order spatial discretization that are stable, as described in [114-116].

Sun and Dai provided a Compact Finite Difference scheme for solving heat conduction in a double-layered film with the Neumann boundary condition [115]. This method has fourth order accuracy in space and is conditionally stable. A fourth-order compact finite difference method for solving the 1-D Pennes bioheat transfer was given by Dai and his collaborators in 2004 [116]. This method solves the heat equation in a triple-layered skin structure. Dai and his collaborators have proposed several finite difference schemes for solving the heat conduction equations in nano-scale [119-123]. All of these method are second order accurate in space. Lately, Dai *et. al* presented a finite difference scheme for solving the fractional parabolic two step heat equation for nano-scale heat conduction [122]. This method is fourth-order accurate in space.

Recently, Dai and his collaborators [61] have developed the Gradient Preserved Method (GPM). By preserving the first-order derivative, u_x, in the interfacial condition and/or the boundary condition, they obtained a higher-order accurate finite difference method. This method uses only three grid points across the interface between layers. They coupled the three point grid in space with the fourth-order accurate Padé scheme [62] at interior points to obtain an accurate, stable and convergent scheme for heat equation with constant coefficients in double layers. We use this idea of preserving the derivative at the interface and the boundary to derive our compact finite difference scheme for the variable coefficient case.

2.3 Energy Method

2.3.1 Energy estimate method for the heat conduction equations

Consider the simple heat equation,

$$u_t = u_{xx}, \qquad 0 \le x \le 1, \quad t \ge 0, \tag{2.37a}$$

$$u(0,t) = u(1,t) = 0, \qquad t \ge 0, \tag{2.37b}$$

$$u(x,0) = g(x), \qquad 0 \le x \le 1. \tag{2.37c}$$

Then the term

$$E(t) = \int_0^1 (u(x,t))^2 dx, \tag{2.38}$$

defines the energy of the solution. To see how the energy evolves with time, we take the derivative of $E(t)$ w.r.t. t,

$$E'(t) = \frac{d}{dt}\int_0^1 |u(x,t)|^2 dx. \tag{2.39}$$

We interchange the order of integration and derivative assuming $u(x,t)$ is smooth,

$$E'(t) = \int_0^1 \frac{\partial}{\partial t}(u(x,t))^2 dx, \tag{2.40a}$$

$$= 2\int_0^1 u(x,t)u_t(x,t)dx \tag{2.40b}$$

$$= 2\int_0^1 u(x,t)u_{xx}(x,t)dx \tag{2.40c}$$

$$= 2[u(x,t)u_x(x,t)]_0^1 - 2\int_0^1 (u_x(x,t))^2 dx \tag{2.40d}$$

$$= -2\int_0^1 (u_x(x,t))^2 dx \qquad \leq 0. \tag{2.40e}$$

This shows that $E(t)$ is a non-increasing function, i.e.

$$E(t) \leq E(0) = \int_0^1 g^2(x)dx. \tag{2.41}$$

This energy estimate could help us to prove uniqueness and stability.

Uniqueness of the solution

Assume that Eq. (2.37) has two solutions $u_1(x,t)$ and $u_2(x,t)$. Let $r(x,t) = u_1(x,t) - u_2(x,t)$. Then, $r(x,t)$ satisfies

$$r_t(x,t) = r_{xx}(x,t), \qquad 0 \leq x \leq 1, \quad t \geq 0, \tag{2.42a}$$

$$r(0,t) = r(1,t) = 0, \qquad t \geq 0, \tag{2.42b}$$

$$r(x,0) = 0, \qquad\qquad 0 \leq x \leq 1. \tag{2.42c}$$

Since both $u_1(x,t)$ and $u_2(x,t)$ satisfy Eq. (2.37), we obtain from Eq. (2.41) that

$$\int_0^1 (r(x,t))^2 dx \leq \int_0^1 (r(x,0))^2 dx = 0, \tag{2.43}$$

implying that $r(x,t) = 0$, and hence $u_1(x,t) = u_2(x,t)$. This indicates that Eq. (2.37) has a unique solution.

Stability of the solution

Assume the Eq. (2.37) has two solutions $u_1(x,t)$ and $u_2(x,t)$ with initial conditions $g_1(x)$ and $g_2(x)$, where $g_1(x) = g_2(x) + \epsilon$ and the same boundary condition. Let $r(x,t) = u_1(x,t) - u_2(x,t)$. Using a similar argument as above, we obtain

$$\int_0^1 (r(x,t))^2 dx \leq \int_0^1 (r(x,0))^2 dx = \int_0^1 \epsilon^2 dx, \tag{2.44a}$$

implying that

$$\int_0^1 (u_1(x,t) - u_2(x,t))^2 dx \leq \int_0^1 \epsilon^2 dx. \tag{2.44b}$$

Thus, if ϵ is small, the difference between $u_1(x,t)$ and $u_2(x,t)$ will be also small. This indicates that a small perturbation in the initial conditions leads to small perturbation in the solution. The inequality in Eq. (2.44b) can be referred to as the stability estimate.

2.3.2 Discrete energy method

We list some fundamental lemmas, which are usually used in the discrete energy method for proving the stability and convergence of finite difference schemes.

Lemma 2.3.2.1 [123] (Cauchy-Schwarz inequality). If a, $b \in \mathbb{R}$, then it holds that

$$2ab \leq \varepsilon a^2 + \frac{1}{\varepsilon} b^2, \tag{2.45}$$

where ε is a small positive number.

Lemma 2.3.2.2 [124] (Gronwall's Lemma). If $\phi(t) \geq 0$, $\psi(t) \geq 0$, $\phi(t) \leq K + L \int_{t_0}^{t} \psi(s)\phi(s)ds$ on $t_0 \leq t \leq t_1$, then we have

$$\phi(t) \leq K e^{L \int_{t_0}^{t_1} \psi(s)ds}. \tag{2.46}$$

The discrete version of this inequality is given by

$$E^n \leq K + L\Delta t \sum_{k=1}^{n-1} E^k \implies E^n \leq K e^{Ln\Delta t}. \tag{2.47}$$

Lemma 2.3.2.3 [125] If U_i and V_i are mesh functions, then

$$-h \sum_{i=1}^{m-1} (\partial_x^2 U_i) V_i = h \sum_{i=1}^{m} (\partial_x U_{i-\frac{1}{2}})(\partial_x V_{i-\frac{1}{2}}) + (D_+ U_0) V_0 + (D_- U_m) V_m, \tag{2.48}$$

where $(D_\pm U_i) = \partial_x U_{i \pm \frac{1}{2}}$.

Proof. The proof can be seen in [125]. However, for convenience, we give a proof here. It can be seen that

$$\begin{aligned}
-h \sum_{i=1}^{m-1} (\partial_x^2 U_i) V_i &= -\sum_{i=1}^{m-1} (\partial_x U_{i+\frac{1}{2}} - \partial_x U_{i-\frac{1}{2}}) V_i \\
&= \sum_{i=1}^{m-1} (\partial_x U_{i-\frac{1}{2}}) V_i - \sum_{i=2}^{m} (\partial_x U_{i-\frac{1}{2}}) V_{i-1} \\
&= \sum_{i=1}^{m-1} (\partial_x U_{i-\frac{1}{2}})(V_i - V_{i-1}) + (\partial_x U_{\frac{1}{2}}) V_0 - (\partial_x U_{m-\frac{1}{2}}) V_m \\
&= h \sum_{i=1}^{m} (\partial_x U_{i-\frac{1}{2}})(\partial_x V_{i-\frac{1}{2}}) + (D_+ U_0) V_0 + (D_- U_m) V_m.
\end{aligned} \tag{2.49}$$

If $V_0 = V_m = 0$, $U_i = V_i$, then

$$-h\sum_{i=1}^{m-1}(\partial_x^2 U_i)U_i = |U|_1^2. \tag{2.50}$$

Lemma 2.3.2.4[125] If U_i is a mesh function, then

$$\frac{4}{h^2} sin\left(\frac{\pi h}{2(b-a)}\right)||U||^2 \leq |U|_1^2 \leq \frac{4}{h^2}||U||^2, \qquad a \leq x \leq b. \tag{2.51}$$

Proof. It can be seen that

$$\begin{aligned} |U|_1^2 = h\sum_{i=1}^{m}(\partial_x U_{i-\frac{1}{2}})^2 &\leq \frac{1}{h^2}h\sum_{i=1}^{m}(U_i - U_{i-1})^2, \\ &\leq \frac{2}{h^2}h\sum_{i=1}^{m}(U_i^2 + U_{i-1}^2) \\ &= \frac{4}{h^2}h(\frac{1}{2}U_0^2 + \sum_{i=1}^{m-1}U_i^2 + \frac{1}{2}U_m^2) \leq \frac{4}{h^2}||U||^2. \end{aligned} \tag{2.52}$$

Lemma 2.3.2.4[125] If V_i is a mesh function with $V_0 = V_m = 0$, we have,

$$h\sum_{i=1}^{m-1}(-\partial_x^2 V_i)V_i = |V|_1^2, ||V||_\infty \leq \frac{\sqrt{b-a}}{2}|V|_1, ||V|| \leq |V|_1. \tag{2.62}$$

Proof. We have,

$$V_i = \sum_{j=1}^{i}(V_j - V_{j-1}) = h\sum_{j=1}^{i}\partial_x V_{j-\frac{1}{2}}, \tag{2.63a}$$

$$V_i = \sum_{j=i+1}^{m}(V_j - V_{j-1}) = h\sum_{j=i+1}^{m}\partial_x V_{j-\frac{1}{2}}. \tag{2.63b}$$

Now using Lemma 2.3.2.1, we have

$$V_i^2 \leq (h\sum_{j=1}^{i}1^2)[h\sum_{j=1}^{i}(\partial_x V_{i-\frac{1}{2}})^2] = (x_i - a)h\sum_{j=1}^{i}(\partial_x V_{i-\frac{1}{2}})^2, \tag{2.64a}$$

$$V_i^2 \leq (h\sum_{j=i+1}^{m}1^2)[h\sum_{j=i+1}^{m}(\partial_x V_{i-\frac{1}{2}})^2] = (b - x_i)h\sum_{j=i+1}^{m}(\partial_x V_{i-\frac{1}{2}})^2. \tag{2.64b}$$

Multiplying $(b - x_i)$ with Eq. (2.64a) and $(x_i - a)$ with Eq. (2.64b) respectively and adding them together, we obtain

$$(b-a)V_i^2 \leq (x_i - a)(b - x_i)|V|_1^2 \leq \frac{(b-a)^2}{4}|V|_1^2, \tag{2.65a}$$

implying that

$$|V_i| \leq \frac{\sqrt{b-a}}{2}|V|_1, \qquad i = 1, 2, ..., m-1. \tag{2.65b}$$

Hence,

$$||V||_\infty \leq \frac{\sqrt{b-a}}{2}|V|_1. \tag{2.65c}$$

Furthermore,

$$(b-a)||V||^2 \leq h\sum_{i=1}^{m-1}(x_i - a)(b - x_i)|V|_1^2 = h^3\sum_{i=1}^{m-1} i(m-i)|V|_1^2, \tag{2.66a}$$

$$= h^3(m\sum_{i=1}^{m-1} i - \sum_{i=1}^{m-1} i^2)|V|_1^2 = \frac{1}{6}m(m^3-1)h^3|V|_1^2, \tag{2.66b}$$

$$\leq \frac{1}{6}(mh)^3|V|_1^2. \tag{2.66c}$$

Thus, we obtain

$$||V|| \leq \frac{b-a}{\sqrt{6}}|V|_1. \tag{2.66d}$$

Lemma 2.3.2.5[125] For mesh function V_i we have:

$$||V||_\infty^2 \leq \varepsilon|V|_1^2 + \frac{1}{4\varepsilon}||V||^2, \qquad V_0 = V_m = 0. \tag{2.67a}$$

$$||V||_\infty^2 \leq \varepsilon|V|_1^2 + (\frac{1}{\varepsilon} + \frac{1}{b-a})||V||^2, \quad a \leq x \leq b. \tag{2.67b}$$

Proof. It can be seen that

$$V_i^2 = \sum_{j=1}^{i-1}(V_{j+1}^2 - V_j^2) = 2h\sum_{j=0}^{i-1} V_{j+\frac{1}{2}}\partial_x V_{j+\frac{1}{2}}, \tag{2.68a}$$

$$V_i^2 = \sum_{j=i}^{m-1}(V_{j+1}^2 - V_j^2) = -2h\sum_{j=i}^{m-1} V_{j+\frac{1}{2}}\partial_x V_{j+\frac{1}{2}}. \tag{2.68b}$$

Now using the Cauchy Schwarz inequality, we obtain

$$V_i^2 \le h\sum_{j=0}^{m-1}|V_{j+\frac{1}{2}}||\partial_x V_{j+\frac{1}{2}}| \le \varepsilon h\sum_{j=0}^{m-1}(\partial_x V_{j+\frac{1}{2}})^2 + \frac{1}{4\varepsilon}h\sum_{j=0}^{m-1}(V_{j+\frac{1}{2}})^2,$$

$$\le \varepsilon|V|_1^2 + \frac{1}{4\varepsilon}||V||^2. \tag{2.69}$$

Similarly we can prove Eq. (2.67b).

As the stability estimate of the analytical solution was found in Eq. (2.44b), one may derive the stability estimate of the numerical solution. Since the numerical solution is in discrete form, so the energy of the solution often is called the discrete energy, such as

$$E^p = h\sum_{i=1}^{N-1}(U_i^p)^2, \tag{2.70}$$

where U_i^p is the numerical solution at time t_p and location x_i, and h is the grid size. To show the discrete energy estimate of the numerical solution, we consider the simple discretization of Eq. (2.37a) at (x_i, t_p) as an example:

$$\frac{U_i^{p+1} - U_i^p}{\tau} = \frac{U_{i-1}^p - 2U_i^p + U_{i+1}^p}{h^2}, \tag{2.71a}$$

$$U_i^{p+1} = U_i^p + \frac{\tau}{h^2}(U_{i-1}^p - 2U_i^p + U_{i+1}^p), \tag{2.71b}$$

with the boundary condition

$$U(0,t) = U(1,t) = 0, \qquad t \ge 0, \tag{2.71c}$$

where τ is the time increment and h is the space increment. It can be seen that

$$E^{p+1} - E^p = h\sum_{i=1}^{N-1}\left[(U_i^{p+1})^2 - (U_i^p)^2\right]$$

$$= h\sum_{i=1}^{N-1}(U_i^{p+1} + U_i^p)(U_i^{p+1} - U_i^p). \tag{2.72}$$

Substituting Eq. (2.71b) in Eq. (2.72), we have

$$E^{p+1} - E^p = \frac{\tau}{h}\sum_{i=1}^{N-1}(U_i^{p+1} + U_i^p)(U_{i-1}^p - 2U_i^p + U_{i+1}^p)$$

$$= \frac{\tau}{h}[\sum_{i=1}^{N-1} U_i^p(U_{i-1}^p - 2U_i^p + U_{i+1}^p) - 2\sum_{i=1}^{N-1} U_i^{p+1}U_i^p$$

$$+ \sum_{i=1}^{N-1} U_i^{p+1}(U_{i-1}^p + U_{i+1}^p)]. \tag{2.73}$$

Using the summation by parts we obtain

$$\sum_{i=1}^{N-1} U_i^p(U_{i-1}^p - 2U_i^p + U_{i+1}^p) = -\sum_{i=1}^{N-1}(U_{i+1}^p - U_i^p)^2. \tag{2.74}$$

Now, using the time stepping scheme in Eq. (2.71b), we can obtain

$$-2\sum_{i=1}^{N-1} U_i^{p+1}U_i^p = -2\sum_{i=1}^{N-1}(U_i^p)^2 + 2\frac{\tau}{h^2}\sum_{i=1}^{N-1}(U_{i+1}^p - U_i^p)^2, \tag{2.75}$$

and using the Cauchy Schwarz inequality, we have:

$$\sum_{i=1}^{N-1} U_i^{p+1}(U_{i-1}^p + U_{i+1}^p)) \leq \sum_{i=1}^{N-1}((U_i^{p+1})^2 + (U_i^p)^2), \tag{2.77a}$$

$$= \frac{1}{h}(E^{p+1} - E^p). \tag{2.77b}$$

Therefore by using the Eq. (2.74a-2.77b):

$$E^{p+1} - E^p \leq \frac{\tau}{h}(-1 + 2\frac{\tau}{h^2})(\sum_{i=1}^{N-1}(U_{i+1}^p - U_i^p)^2) + \frac{\tau}{h^2}(E^{p+1} - E^p). \tag{2.78}$$

Now, if

$$-1 + 2\frac{\tau}{h^2} \leq 0, \qquad \frac{\tau}{h^2} \leq \frac{1}{2}, \tag{2.79}$$

then from Eq.(2.78)

$$(1 - \frac{\tau}{h^2})(E^{p+1} - E^0) \leq 0 \tag{2.80a}$$

$$\implies (E^{p+1} - E^0) \leq 0, \tag{2.80b}$$

provided the condition in Eq. (2.79) holds. Hence, Eq. (2.80b) gives us the stability estimate of the scheme in Eq. (2.24c) for the Eq. (2.23a). Also, this stability estimate tells us that in order for the numerical scheme described in Eq. (2.24c) to be stable, the step size in time and space direction should satisfy Eq. (2.24). Thus the scheme shown in Eq. (2.72b) is conditionally stable. The above example has been taken from [102].

This analysis of the numerical method is known as the energy method or the Discrete Energy Method.

2.4 Thomas Algorithm

The Thomas Algorithm, also known as the tridiagonal matrix algorithm, is a simplified form of Gaussian elimination, which is used to solve the tridiagonal linear system of equations. Note that in a tridiagonal linear system, the equation number j involves unknowns with numbers $j-1$, j and $j+1$, which means that the matrix of the system has non-zero elements only on the diagonal and in the positions to immediate off diagonal lines. Consider a tridiagonal linear system to be solved as shown below [101] :

$$\begin{bmatrix} b_0 & -c_0 & 0 & 0 & 0 & 0 \\ -a_1 & b_1 & -c_1 & 0 & 0 & 0 \\ 0 & -a_2 & b_2 & -c_2 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & -a_{N-1} & b_{N-1} & -c_{N-1} \\ 0 & 0 & 0 & 0 & -a_N & b_N \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_{N-1} \\ d_N \end{bmatrix}.$$

This system can be written in the equation form:

$$-a_i u_{j-1} + b_i u_i - c_i u_{i+1} = d_i, \qquad i = 1, 2, \cdots, N-1, \tag{2.81a}$$

$$u_0 = 0, \quad u_N = 0. \tag{2.81b}$$

The coefficients a_i, b_i, c_i and d_i are known and assumed to satisfy the condition:

$$|b_i| > |a_i| + |c_i|, \qquad i = 1, 2, \cdots, N-1, \tag{2.82a}$$

$$|b_0| > |c_0|, \qquad |b_N| > |a_N|. \tag{2.82b}$$

These conditions ensure that the matrix is diagonally dominant, which guarantees that the system has a unique solution. The Thomas Algorithm reduces the system to upper triangular form. It does so by eliminating the term u_{j-1} in each of the equations. Assume that the first n equations of Eq. (2.81) have been reduced to

$$u_i - \alpha_i u_{i+1} = g_i, \qquad i = 1, 2, \cdots, n, \tag{2.83}$$

and the next equation that needs to be reduced is

$$-a_{n+1} u_n + b_{n+1} u_{n+1} - c_{n+1} u_{n+2} = d_{n+1}. \tag{2.84}$$

Eliminating u_n from Eqs. (2.83-2.84), we obtain

$$u_{n+1} - \frac{c_{n+1}}{b_{n+1} - a_{n+1}\alpha_n} u_{n+2} = \frac{d_{n+1} + a_{n+1}g_n}{b_{n+1} - a_{n+1}\alpha_n}. \tag{2.85}$$

Comparing Eq. (2.83) with Eq. (2.85), we have the recurrence relation:

$$\alpha_i = \frac{c_i}{b_i - a_i\alpha_{i-1}}, \quad g_i = \frac{d_i + a_i g_{i-1}}{b_i - a_i\alpha_{i-1}}, \quad i = 1, 2, \cdots, N-1, \tag{2.86a}$$

$$\alpha_0 = \frac{c_0}{b_0}, \quad g_0 = \frac{d_0}{b_0}. \tag{2.86b}$$

On the other hand, the linear system reduces to

$$\begin{bmatrix} 1 & \alpha_0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \alpha_1 & 0 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 1 & \alpha_{N-1} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{N-1} \\ g_N \end{bmatrix}.$$

Using back substitution, the above system can be solved easily by the following recurrence:

$$u_N = g_N, \quad u_i = g_i - \alpha_i u_{i+1}. \quad i = N-1, N-2, \cdots, 1. \tag{2.87}$$

2.5 Neural Network Method

A neural network consists of layers of interconnected nodes as shown in Figure 2.3. Each node is called a perceptron. Each perceptron feeds the data produced by a multiple linear combination into an activation function. An activation function is the output produced by the node, when a set of input is given to it. The activation function can be linear or nonlinear. The most commonly used activation functions are sigmoid or logistic function, hyperbolic tangent function ($tanh$), Rectified Linear Unit (ReLU). One of the major properties of neural networks is that it can adapt to changing input, so it generates the best possible result without needing to redesign the criteria for output. In general, a perceptron consists of three main parts: (i) input layer, (ii) hidden layers and (iii) output layer as shown in Figure 2.4. The input layer is the beginning of the neural network. This feeds in the input data to the hidden layers of the neural network for further processing. In the hidden layers, weights and biases are applied to the inputs, and they are passed through the activation function as the output. The number of hidden layers can vary from one to many depending on the data to be processed. The output layer produces the final result. Whenever there are multiple hidden layers associated with a neural network, it is called the Deep Neural Network (DNN). Figure 2.3 shows a picture of a fully connected DNN with 3-hidden layers. It is called fully connected because each node is connected to every other node in the previous layer.

It can be seen from Figure 2.4 that the perceptron performs two functions. One is to multiply each input by weights, sum them up and then add a constant (bias) to it. The other is to pass the previously calculated value through an activation function G. This is now the output for the perceptron becomes an input for the next perceptron, and the process continues in the same way till the output layer is reached.

As such, for the neural net in Figure 2.3, we have

$$y_i^{(1)} = G[\sum_{i=1}^{3}(W_i^{(1)}x_i + b_i^{(1)})], \tag{2.88a}$$

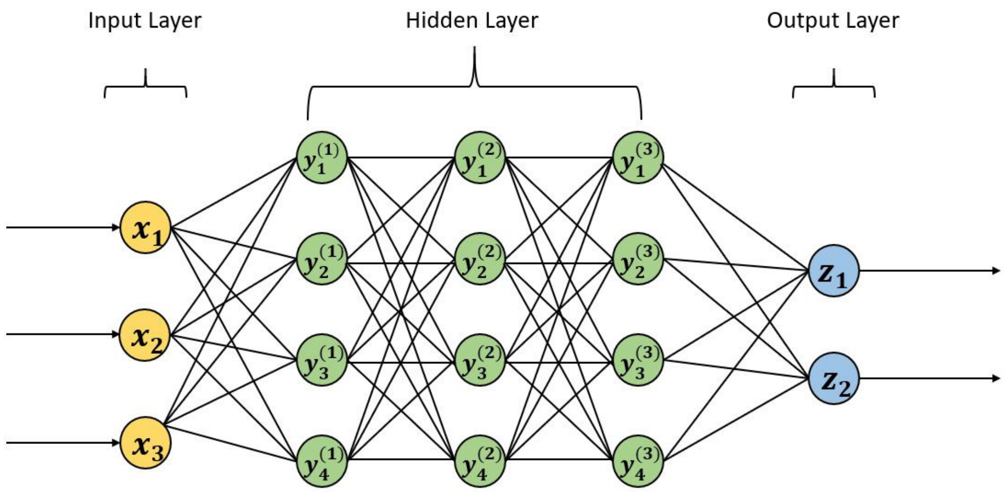


Figure 2.3: General structure of a fully connected neural network.

$$y_i^{(2)} = G[\sum_{i=1}^{4}(W_i^{(2)} y_i^{(1)} + b_i^{(2)})], \tag{2.88b}$$

$$y_i^{(3)} = G[\sum_{i=1}^{4}(W_i^{(3)} y_i^{(2)} + b_i^{(3)})], \tag{2.88c}$$

and

$$z_1 = \sum_{i=1}^{4} W_i y_i^{(3)} + b_1, \tag{2.88d}$$

$$z_2 = \sum_{i=1}^{4} \tilde{W}_i y_i^{(3)} + b_2, \tag{2.88e}$$

where $W_i^{(j)}$, $b_i^{(j)}$, W_i, $\tilde{W}_i$, b_1 and b_2 are the weights and biases involved in the neural network for $i, j = 1, 2, 3$.

The first idea of how a neuron works was given by Warren McCulloch, a neurophysiologist, and Walter Pitts, a mathematician, on their publication in 1943 [126, 127]. They modeled a simple neural network using electrical circuits. In 1949,

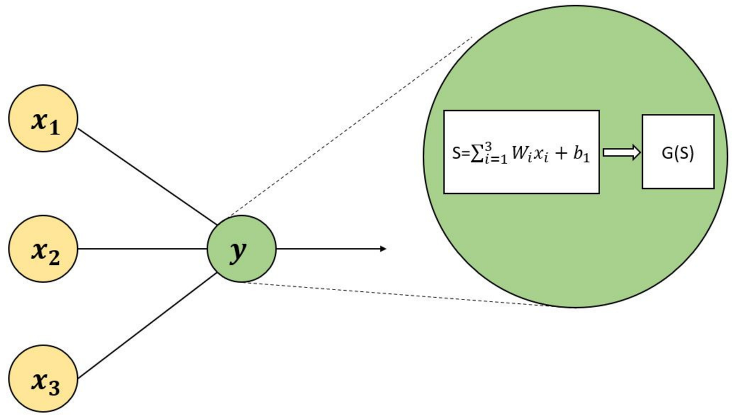


Figure 2.4: General operation in a perceptron.

D. O. Hebb pointed out the fact that neural pathways are strengthened each time they are used in his book [128]. In the 1950's, the first step towards simulating a hypothetical neural network was made by Nathanial Rochester from the IBM research laboratories.

Models called ADALINE and MADELINE were developed in 1959 by Bernard Widrow and Marcian Hoff in Stanford University. ADALINE recognized binary patterns. It could predict the next bit by reading streaming bits from a phone line. The first neural network applied to a real world problem was MADELINE. It uses an adaptive filter that eliminates echoes on phone lines. The air traffic control systems still use it commercially. A learning procedure was developed by Widrow and Hoff in 1962 [127]. They gave a rule for weight change as: Weight Change = (Pre-Weight line value) * (Error / (Number of Inputs)). The procedure examines the value and adjusts the weights (i.e. 0 or 1) according to this rule.

The first multilayered network was developed in 1975 by Fukushima [129]. In 1982, John Hopfield from Caltech, California presented a paper to the National

Academy of Sciences that proposed to create more useful machines by using bidirectional lines [130, 131]. In the same year, a hybrid network that used multiple layers was given by Reilly and his colleagues [132]. In 1986, David Rumelhart and his collaborators came up with ideas similar to what it is now called back propagation networks [133]. Throughout the network, it distributed pattern recognition errors. Hybrid networks have only two layers, whereas back-propagation networks may have many layers.

Up to day, neural networks have been used in many applications. Methods like data driven scientific computing have become more and more popular. Machine and deep learning techniques have been playing a major role in it [64, 65, 66, 67, 68, 69, 70, 71, 73, 72]. Karniadakis et al. recently have introduced the Physics Informed Neural Nets (PINN) [73] for solving partial difference equations. The main idea in PINN is to use the output of a deep neural network and treat it as the solution of the partial differential equation. Also, it exploits the use of auto derivative for differentiating the neural network output and uses it in the loss function. As such, the loss function is composed of the differential equation itself, the initial condition, and the boundary condition. The neural network is trained till it achieves a certain tolerance/ minimum number of iterations. The neural network method in the dissertation takes the idea from this method.

Karniadakis et al. also presented the DeepONet, which is based on the universal approximation theorem. This technique can learn non-linear operators for identifying differential equations [164]. DeepONet is a deep neural network that encodes the discrete input function space (branch net) and another deep neural net that encodes the domain of the output functions (trunk net). Karniadakis et al. showed that their technique could learn various explicit/implicit operators such as integrals, fractional laplacians, and operators for deterministic and stochastic differential equations.

Various PINN methods have been applied/developed for solving partial differential equations as well as inverse problems since 2019 [134-163]. Recently, more methods have been developed for solving similar kinds of problems. Li et al. came up with the Graph Kernel Network for solving partial differential equations in their paper [165, 166]. This method generalizes neural networks in order to learn mappings between infinite-dimensional spaces (operators). The same group also developed the multipole graph neural operator method for solving partial differential equations. Their method was developed based on the classical multipole methods proposing a novel multi-level graph neural network framework for solving partial differential equations with linear complexity. They also created the Fourier Neural Operator for the parametric partial differential equation, where the network could map from functions to functions [167].

2.6 Summary

In this chapter, we have reviewed the required background for understanding the research of this dissertation. We have reviewed the heat conduction equations that we plan to solve in this dissertation. Also, we have summarized the previous work that has been done, both in the field of numerical methods and data driven scientific computing methods that lay the foundation of our research.

CHAPTER 3

GRADIENT PRESERVED METHOD FOR VARIABLE COEFFICIENT CASE

3.1 Mathematical Equation for Double-Layered Structure

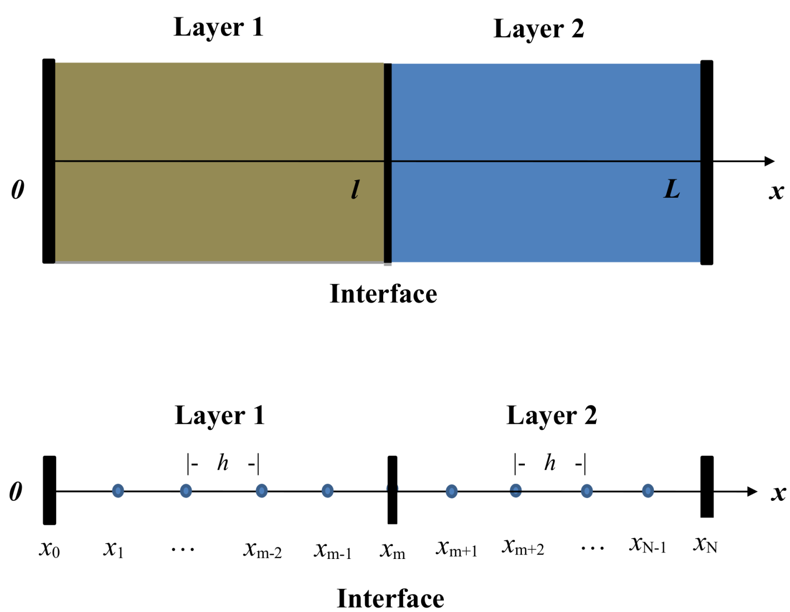


Figure 3.1: Double-layered structure (above) and mesh for numerical schemes (below).

In this dissertation, we consider a heat conduction problem with variable coefficients in double layers, as shown in 3.1. The mathematical equations are given as

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial}{\partial x}(k_1(x)\frac{\partial u(x,t)}{\partial x}) + F_1(x,t), \quad 0 \le x \le l, \quad t > 0 \tag{3.1a}$$

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial}{\partial x}(k_2(x)\frac{\partial u(x,t)}{\partial x}) + F_2(x,t), \quad l \le x \le L, \quad t > 0 \tag{3.1b}$$

where initial and boundary conditions are given as

$$u(x,0) = \phi_1(x), \quad 0 \le x \le l; \quad u(x,0) = \phi_2(x), \quad l \le x \le L, \tag{3.2}$$

$$u_x(0,t) = \alpha(t), \qquad u_x(L,t) = \beta(t), \qquad t \ge 0, \tag{3.3}$$

and the interfacial condition is assumed to be

$$u(x_{l+},t) = u(x_{l-},t), \quad k_2(x_{l+})u_x(x_{l+},t) = k_1(x_{l-})u_x(x_{l-},t), \ t \ge 0. \tag{3.4}$$

Here, $k_1(x)$ and $k_2(x)$ represent the spatially varying functions for the first and second layers, respectively.

3.2 Higher-Order Compact Finite Difference Scheme

To develop an accurate finite difference scheme, we initially design an equidistant mesh as shown in Figure 3.1, where $x_j = jh$, $t_n = n\tau$, $0 \le j \le N$, $n \ge 0$, and $x_m = mh$ is the interfacial grid point. Here, h and τ are the grid size and time step, respectively. We use the same grid size h for double layers to simplify our derivations. However, it can be easily generalized to the case of using different grid sizes for different layers. We denote u_j^n as the analytical solution $u(x_j,t_n)$ and U_j^n as the numerical approximation of u_j^n. We further define the following finite difference operators as

$$\nabla_x u_j^n = \frac{u_{j+1}^n - u_j^n}{h}, \quad \bar{u}_j^{n+\frac{1}{2}} = \frac{u_j^{n+1} + u_j^n}{2}, \quad \delta_t u_j^{n+\frac{1}{2}} = \frac{u_j^{n+1} - u_j^n}{\tau}.$$

We now establish some important lemmas related to our finite difference scheme.

Lemma 3.2.1. Assume that the analytical solution $u(x,t)$ in Eqs. (3.1)-(3.4) is smooth in $[0,L] \times [0,\infty)$. Then it holds at x_0

$$c_1\delta_t u_0^{n+\frac{1}{2}} + c_2\delta_t u_1^{n+\frac{1}{2}} = \frac{1}{h}[(k_1)_{1/2}\frac{u_1^{n+\frac{1}{2}} - u_0^{n+\frac{1}{2}}}{h} - c_3(k_1)_0(u_x)_0^{n+\frac{1}{2}}]$$

$$+ [-\frac{h}{12} + \frac{h^2}{24}\frac{(k_{1x})_0}{(k_1)_0}]\alpha'(t_{n+\frac{1}{2}}) + f_0^{n+\frac{1}{2}} + O(\tau^2 + h^3), \qquad (3.5)$$

where

$$c_1 = \frac{5}{12} - \frac{h}{12}\frac{(k_{1x})_0}{(k_1)_0} + \frac{h^2}{48}\frac{[4(k_{1x})_0]^2 - 3(k_1)_0(k_{1xx})_0}{[(k_1)_0]^2}, \qquad (3.6a)$$

$$c_2 = \frac{1}{12}, \qquad (3.6b)$$

$$c_3 = 1 + \frac{h^2}{24(k_1)_0}[\frac{2[(k_{1x})_0]^2}{(k_1)_0} - (k_{1xxx})_0]$$

$$+ \frac{h^3}{48(k_1)_0}[\frac{5(k_1)_0(k_{1x})_0(k_{1xx})_0 - 4[(k_{1x})_0]^3}{[(k_1)_0]^2} - (k_{1xxx})_0], \qquad (3.6c)$$

$$f_0^{n+\frac{1}{2}} = [\frac{1}{2} - \frac{h}{12}\frac{(k_{1x})_0}{(k_1)_0} - \frac{h^2}{48}\frac{4[(k_{1x})_0]^2 - 3(k_1)_0(k_{1xx})_0}{[(k_1)_0]^2}](F_1)_0^{n+\frac{1}{2}}$$

$$+ [\frac{h}{6} - \frac{h^2}{24}\frac{(k_{1x})_0}{(k_1)_0}](F_{1x})_0^{n+\frac{1}{2}} + \frac{h^2}{24}(F_{1xx})_0^{n+\frac{1}{2}}. \qquad (3.6d)$$

Here, k_{1x} denotes $\partial k_1/\partial x$, and $(k_1)_0$ denotes $k_1(x_0)$, and so on for others. It can be seen that if $k_1(x)$ is a constant. Thus, Eq. (3.5) reduces to

$$\frac{1}{12}\delta_t u_0^{n+\frac{1}{2}} + \frac{5}{12}\delta_t u_1^{n+\frac{1}{2}} = \frac{k_1}{h}[\frac{u_1^{n+\frac{1}{2}} - u_0^{n+\frac{1}{2}}}{h} - (u_x)_0^{n+\frac{1}{2}}] + \frac{1}{2}(F_1)_0^{n+\frac{1}{2}}$$

$$+ \frac{h}{6}(F_{1x})_0^{n+\frac{1}{2}} + \frac{h^2}{24}(F_{1xx})_0^{n+\frac{1}{2}} + O(\tau^2 + h^3). \qquad (3.7)$$

Proof. We apply the Taylor series expansions at $x_{1/2}$ and then at x_0. This gives

$$\frac{1}{h}(k_1)_{1/2}\frac{u_1^{n+\frac{1}{2}}-u_0^{n+\frac{1}{2}}}{h} = \frac{1}{h}(k_1u_x)|_0^{n+\frac{1}{2}} + \frac{1}{2}\frac{\partial}{\partial x}(k_1u_x)|_0^{n+\frac{1}{2}} + \frac{h}{8}\frac{\partial^2}{\partial x^2}(k_1u_x)|_0^{n+\frac{1}{2}}$$
$$+ \frac{h^2}{48}\frac{\partial^3}{\partial x^3}(k_1u_x)|_0^{n+\frac{1}{2}} + \frac{h}{24}(k_1u_{xxx})|_0^{n+\frac{1}{2}} + \frac{h^2}{48}\frac{\partial}{\partial x}(k_1u_{xxx})|_0^{n+\frac{1}{2}}$$
$$+ O(h^3). \tag{3.8}$$

From Eq. (1a), one may see

$$u_t = k_{1x}u_x + k_1u_{xx} + F_1, \tag{3.9a}$$

$$u_{tx} = k_{1xx}u_x + 2k_{1x}u_{xx} + k_1u_{xxx} + F_{1x}$$
$$= k_{1xx}u_x + 2\frac{k_{1x}}{k_1}(u_t - k_{1x}u_x - F_1) + k_1u_{xxx} + F_{1x}. \tag{3.9b}$$

Solving for k_1u_{xxx} from Eq. (3.9) gives

$$k_1u_{xxx} = u_{tx} - 2\frac{k_{1x}}{k_1}u_t + [2\frac{(k_{1x})^2}{k_1} - k_{1xx}]u_x + 2\frac{k_{1x}}{k_1}F_1 - F_{1x}, \tag{3.10a}$$

$$\frac{\partial}{\partial x}(k_1u_{xxx}) = u_{txx} - 2\frac{k_{1x}}{k_1}u_{tx} + \frac{4(k_{1x})^2 - 3k_1k_{1xx}}{(k_1)^2}u_t$$
$$+ [\frac{5k_1k_{1x}k_{1xx} - 4(k_{1x})^3}{(k_1)^2} - k_{1xxx}]u_x + \frac{3k_1k_{1xx} - 4(k_{1x})^2}{(k_1)^2}F_1$$
$$+ 2\frac{k_{1x}}{k_1}F_{1x} - F_{1xx}. \tag{3.10b}$$

Thus, substituting Eq. (3.10) at $(x_0, t_{n+1/2})$ into Eq. (3.8) and then Eq. (3.8) into the right-hand-side (RHS) of Eq. (3.5), we obtain

$$RHS = A_2(u_x)_0^{n+\frac{1}{2}} + A_1(u_t)_0^{n+\frac{1}{2}} + [\frac{h}{6} - \frac{h^2}{24}\frac{(k_{1x})_0}{(k_1)_0}](u_{tx})_0^{n+\frac{1}{2}} + \frac{h^2}{24}(u_{txx})_0^{n+\frac{1}{2}}$$
$$+ B_1 + [-\frac{h}{12} + \frac{h^2}{24}\frac{(k_{1x})_0}{(k_1)_0}]\alpha'(t_{n+\frac{1}{2}}) + f_0^{n+\frac{1}{2}} + O(\tau^2 + h^3), \tag{3.11}$$

where

$$A_1 = \frac{1}{2} - \frac{h}{12}\frac{(k_{1x})_0}{(k_1)_0} + \frac{h^2}{48}\frac{4[(k_{1x})_0]^2 - 3(k_1)_0(k_{1xx})_0}{[(k_1)_0]^2}, \tag{3.12a}$$

$$A_2 = \frac{1}{h}(1-c_3)(k_1)_0 + \frac{h^2}{48}[\frac{5(k_1)_0(k_{1x})_0(k_{1xx})_0 - 4[(k_1)_0]^3}{[(k_1)_0]^2} - (k_{1xxx})_0]$$

$$+ \frac{h}{24}[2\frac{[(k_{1x})_0]^2}{(k_1)_0} - (k_{1xx})_0]. \tag{3.12b}$$

$$B_2 = -\frac{1}{2}(F_1)_0^{n+\frac{1}{2}} + \frac{h}{24}[2\frac{(k_{1x})_0}{(k_1)_0}(F_1)_0^{n+\frac{1}{2}} - 4(F_{1x})_0^{n+\frac{1}{2}}]$$

$$+ \frac{h^2}{48}[\frac{3(k_1)_0(k_{1xx})_0 - 4[(k_{1x})_0]^2}{[(k_1)_0]^2}(F_1)_0^{n+\frac{1}{2}} + 2\frac{(k_{1x})_0}{(k_1)_0}(F_{1x})_0^{n+\frac{1}{2}}]$$

$$- \frac{h^2}{24}(F_{1xx})_0^{n+\frac{1}{2}} \tag{3.12c}$$

Based on Eq. (3), we obtain $(u_{tx})_0^{n+1/2} = \alpha'(t_{n+1/2})$ and then substitute it and Eq. (6d) into Eq. (11). This gives

$$RHS = A_2(u_x)_0^{n+\frac{1}{2}} + A_1(u_t)_0^{n+\frac{1}{2}} + \frac{h}{12}(u_{tx})_0^{n+\frac{1}{2}} + \frac{h^2}{24}(u_{txx})_0^{n+\frac{1}{2}} + O(\tau^2 + h^3). \tag{3.13}$$

On the other hand, we apply the Taylor series expansion at $(x_0, t_{n+1/2})$ to the left-hand-side (LHS) of Eq. (3.5). This gives

$$LHS = (c_1 + c_2)(u_t)_0^{n+\frac{1}{2}} + c_2 h(u_{tx})_0^{n+\frac{1}{2}} + c_2\frac{h^2}{2}(u_{txx})_0^{n+\frac{1}{2}} + O(\tau^2 + h^3). \tag{3.14}$$

Matching Eqs. (3.13) and (3.14), we obtain

$$A_2 = 0, \quad c_1 + c_2 = A_1, \quad c_2 = \frac{1}{12}, \tag{3.15}$$

which gives c_1, c_2, c_3 as listed in Eqs. (3.6a)-(3.6c), and hence complete the proof.

Using a similar argument, one may obtain the following lemma for a scheme at x_N.

Lemma 3.2.2. Assume that the analytical solution $u(x,t)$ in Eqs. (3.1)-(3.4) is smooth in $[0,L]\times[0,\infty)$. Then it holds at x_N

$$\tilde{c}_1\delta_t u_N^{n+\frac{1}{2}} + \tilde{c}_2\delta_t u_{N-1}^{n+\frac{1}{2}} = \frac{1}{h}[\tilde{c}_3(k_2)_N(u_x)_N^{n+\frac{1}{2}} - (k_2)_{N-\frac{1}{2}}\frac{u_N^{n+\frac{1}{2}} - u_{N-1}^{n+\frac{1}{2}}}{h}]$$

$$+[\frac{h}{12}+\frac{h^2}{24}\frac{(k_{2x})_N}{(k_2)_N}]\beta'(t_{n+\frac{1}{2}}) + f_N^{n+\frac{1}{2}} + O(\tau^2+h^3), \qquad (3.16)$$

where

$$\tilde{c}_1 = \frac{5}{12} + \frac{h}{12}\frac{(k_{2x})_N}{(k_2)_N} + \frac{h^2}{48}\frac{4[(k_{2x})_N]^2 - 3(k_2)_N(k_{2xx})_N}{[(k_2)_N]^2}, \qquad (3.17a)$$

$$\tilde{c}_2 = \frac{1}{12}, \qquad (3.17b)$$

$$\tilde{c}_3 = 1 + \frac{h^2}{24(k_2)_N}[\frac{2[(k_{2x})_N]^2}{(k_2)_N} - (k_{2xxx})_N]$$

$$- \frac{h^3}{48(k_2)_N}[\frac{5(k_2)_N(k_{2x})_N(k_{2xx})_N - 4[(k_{2x})_N]^3}{[(k_2)_N]^2} - (k_{2xxx})_N], \qquad (3.17c)$$

$$f_N^{n+\frac{1}{2}} = [\frac{1}{2} + \frac{h}{12}\frac{(k_{2x})_N}{(k_2)_N} + \frac{h^2}{48}\frac{4[(k_{2x})_N]^2 - 3(k_2)_N(k_{2xx})_N}{[(k_2)_N]^2}](F_2)_N^{n+\frac{1}{2}}$$

$$-[\frac{h}{6} + \frac{h^2}{24}\frac{(k_{2x})_N}{(k_2)_N}](F_{2x})_N^{n+\frac{1}{2}} + \frac{h^2}{24}(F_{2xx})_N^{n+\frac{1}{2}}. \qquad (3.17d)$$

Here, k_{2x} denotes $\partial k_2/\partial x$, and so on for others. It can be seen that if $k_2(x)$ is a constant, Eq. (3.16) reduces to

$$\frac{5}{12}\delta_t u_N^{n+\frac{1}{2}} + \frac{1}{12}\delta_t u_{N-1}^{n+\frac{1}{2}} = \frac{k_2}{h}[(u_x)_N^{n+\frac{1}{2}} - \frac{u_N^{n+\frac{1}{2}} - u_{N-1}^{n+\frac{1}{2}}}{h}] + \frac{1}{2}(F_2)_N^{n+\frac{1}{2}} - \frac{h}{6}(F_{2x})_N^{n+\frac{1}{2}}$$

$$+ \frac{h^2}{24}(F_{2xx})_N^{n+\frac{1}{2}} + O(\tau^2+h^3). \qquad (3.18)$$

Lemma 3.2.3. Assume that the analytical solution $u(x,t)$ in Eqs. (3.1)-(3.4) is smooth in $[0,L]\times[0,\infty)$. Then it holds at the interior points x_j, $j=1,2,\ldots,m-1$,

$$(c_4)_j\delta_t u_{j-1}^{n+\frac{1}{2}} + (c_5)_j\delta_t u_j^{n+\frac{1}{2}} + (c_6)_j\delta_t u_{j+1}^{n+\frac{1}{2}} = \frac{1}{h^2}[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}](u_{j+1}^{n+\frac{1}{2}} - u_j^{n+\frac{1}{2}})$$

$$- \frac{1}{h^2}[(k_1)_{j-\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j-\frac{1}{2}}](u_j^{n+\frac{1}{2}} - u_{j-1}^{n+\frac{1}{2}})$$

$$+ f_j^{n+\frac{1}{2}} + O(\tau^2 + h^4), \tag{3.19}$$

where

$$(c_4)_j = \frac{1}{12} + \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}}, \quad (c_5)_j = \frac{5}{6}, \quad (c_6)_j = \frac{1}{12} - \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}}, \tag{3.20a}$$

$$(D_1)_{j+\frac{1}{2}} = 2\frac{[(k_{1x})_{j+\frac{1}{2}}]^2}{(k_1)_{j+\frac{1}{2}}} - (k_{1xx})_{j+\frac{1}{2}}, \tag{3.20b}$$

$$f_j^{n+\frac{1}{2}} = [1 - \frac{h^2}{12}\frac{(k_1)_j(k_{1xx})_j - (k_1)_j^2}{(k_1)_j^2}](F_1)_j^{n+\frac{1}{2}} - \frac{h^2}{12}\frac{(k_{1x})_j}{(k_1)_j}(F_{1x})_j^{n+\frac{1}{2}}$$

$$+ \frac{h^2}{12}(F_{1xx})_j^{n+\frac{1}{2}}. \tag{3.20c}$$

It can be seen that if $k_1(x)$ is constant, then Eq. (3.19) reduces to

$$\frac{1}{12}\delta_t u_{j-1}^{n+\frac{1}{2}} + \frac{10}{12}\delta_t u_j^{n+\frac{1}{2}} + \frac{1}{12}\delta_t u_{j+1}^{n+\frac{1}{2}} = \frac{k_1}{h^2}[u_{j-1}^{n+\frac{1}{2}} - 2u_j^{n+\frac{1}{2}} + u_{j+1}^{n+\frac{1}{2}}] + (F_1)_j^{n+\frac{1}{2}}$$

$$+ \frac{h^2}{12}(F_{1xx})_j^{n+\frac{1}{2}} + O(\tau^2 + h^4). \tag{3.21}$$

Proof. Using the Taylor expansions at $x_{j+1/2}$and $x_{j-1/2}$, respectively, and then at x_j, we obtain

$$\frac{1}{h^2}[(k_1)_{j+\frac{1}{2}}(u_{j+1}^{n+\frac{1}{2}} - u_j^{n+\frac{1}{2}}) - (k_1)_{j-\frac{1}{2}}(u_j^{n+\frac{1}{2}} - u_{j-1}^{n+\frac{1}{2}})]$$

$$= \frac{(k_1)_{j+\frac{1}{2}}}{h^2}[h(u_x)_{j+\frac{1}{2}}^{n+\frac{1}{2}} + \frac{h^3}{24}(u_{xxx})_{j+\frac{1}{2}}^{n+\frac{1}{2}} + \frac{h^5}{1920}(u_{x^5})_{j+\frac{1}{2}}^{n+\frac{1}{2}}]$$

$$- \frac{(k_1)_{j-\frac{1}{2}}}{h^2}[h(u_x)_{j-\frac{1}{2}}^{n+\frac{1}{2}} + \frac{h^3}{24}(u_{xxx})_{j-\frac{1}{2}}^{n+\frac{1}{2}} + \frac{h^5}{1920}(u_{x^5})_{j-\frac{1}{2}}^{n+\frac{1}{2}}] + O(h^4)$$

$$= \frac{\partial}{\partial x}(k_1 u_x)|_j^{n+\frac{1}{2}} + \frac{h^2}{24}\frac{\partial^3}{\partial x^3}(k_1 u_x)|_j^{n+\frac{1}{2}} + \frac{h^2}{24}\frac{\partial}{\partial x}(k_1 u_{xxx})|_j^{n+\frac{1}{2}} + O(h^4). \tag{3.22}$$

Substituting Eq. (3.1a) and Eq. (3.10b) at $(x_j, t_{n+1/2})$ into Eq. (3.22), and then using the second-order central difference approximation, we obtain

$$\frac{1}{h^2}[(k_1)_{j+\frac{1}{2}}(u_{j+1}^{n+\frac{1}{2}} - u_j^{n+\frac{1}{2}}) - (k_1)_{j-\frac{1}{2}}(u_j^{n+\frac{1}{2}} - u_{j-1}^{n+\frac{1}{2}})]$$

$$= (u_t)_j^{n+\frac{1}{2}} + \frac{h^2}{12}(u_{txx})_j^{n+\frac{1}{2}} - \frac{h^2}{12}\frac{\partial}{\partial x}(\frac{k_{1x}}{k_1}u_t)|_j^{n+\frac{1}{2}} + \frac{h^2}{24}\frac{\partial}{\partial x}[(2\frac{(k_{1x})^2}{k_1} - k_{1xx})u_x]|_j^{n+\frac{1}{2}}$$

$$+ [-1 + \frac{h^2}{12}\frac{(k_1)_j(k_{1xx})_j - (k_1)_j^2}{(k_1)_j^2}](F_1)_j^{n+\frac{1}{2}} + \frac{h^2}{12}\frac{(k_{1x})_j}{(k_1)_j}(F_{1x})_j^{n+\frac{1}{2}} - \frac{h^2}{12}(F_{1xx})_j^{n+\frac{1}{2}}$$

$$+ O(h^4)$$

$$= (u_t)_j^{n+\frac{1}{2}} + \frac{h^2}{12}(u_{txx})_j^{n+\frac{1}{2}} - \frac{h}{24}[\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}}(u_t)_{j+1}^{n+\frac{1}{2}} - \frac{(k_{1x})_{j-1}}{(k_1)_{j-1}}(u_t)_{j-1}^{n+\frac{1}{2}}]$$

$$+ \frac{h^2}{24}\{[2\frac{[(k_{1x})_{j+\frac{1}{2}}]^2}{(k_1)_{j+\frac{1}{2}}} - (k_{1xx})_{j+\frac{1}{2}}]\frac{u_{j+1}^{n+\frac{1}{2}} - u_j^{n+\frac{1}{2}}}{h^2}\}$$

$$+ \frac{h^2}{24}\{-[2\frac{[(k_{1x})_{j-\frac{1}{2}}]^2}{(k_1)_{j-\frac{1}{2}}} - (k_{1xx})_{j-\frac{1}{2}}]\frac{u_j^{n+\frac{1}{2}} - u_{j-1}^{n+\frac{1}{2}}}{h^2}\}$$

$$+ [-1 + \frac{h^2}{12}\frac{(k_1)_j(k_{1xx})_j - (k_1)_j^2}{(k_1)_j^2}](F_1)_j^{n+\frac{1}{2}} + \frac{h^2}{12}\frac{(k_{1x})_j}{(k_1)_j}(F_{1x})_j^{n+\frac{1}{2}}$$

$$- \frac{h^2}{12}(F_{1xx})_j^{n+\frac{1}{2}} + O(h^4). \tag{3.23}$$

Thus, substituting Eq. (3.23) into the RHS of Eq. (3.19) gives

$$RHS = (u_t)_j^{n+\frac{1}{2}} + \frac{h^2}{12}(u_{txx})_j^{n+\frac{1}{2}} - \frac{h}{24}[\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}}(u_t)_{j+1}^{n+\frac{1}{2}} - \frac{(k_{1x})_{j-1}}{(k_1)_{j-1}}(u_t)_{j-1}^{n+\frac{1}{2}}]$$

$$+ O(\tau^2 + h^4). \tag{3.24}$$

Now we take the term $-h/24[(k_{1x})_{j+1}/(k_1)_{j+1}](u_t)_{j+1}^{n+1/2}+h/24[(k_{1x})_{j-1}/(k_1)_{j-1}](u_t)_{j-1}^{n+1/2}]$ in Eq. (3.24) to the LHS of Eq. (3.19), apply the Taylor expansions at $t_{n+1/2}$ and then at x_j for the LHS. This gives

$$\begin{aligned}
LHS &= [(c_4)_j - \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}}](u_t)_{j-1}^{n+\frac{1}{2}} + (c_5)_j(u_t)_j^{n+\frac{1}{2}} \\
&\quad + [(c_6)_j + \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}}](u_t)_{j+1}^{n+\frac{1}{2}} + O(\tau^2) \\
&= [(c_4)_j - \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}} + (c_5)_j + (c_6)_j + \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}}](u_t)_j^{n+\frac{1}{2}} \\
&\quad + h[(c_6)_j + \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}} - (c_4)_j + \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}}](u_{tx})_j^{n+\frac{1}{2}} \\
&\quad + \frac{h^2}{2}[(c_6)_j + \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}} + (c_4)_j - \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}}](u_{txx})_j^{n+\frac{1}{2}} \\
&\quad + \frac{h^3}{6}[(c_6)_j + \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}} - (c_4)_j + \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}}](u_{txxx})_j^{n+\frac{1}{2}} \\
&\quad + O(\tau^2 + h^4).
\end{aligned} \tag{3.25}$$

On the other hand, Eq. (3.24) reduces to

$$RHS = (u_t)_j^{n+\frac{1}{2}} + \frac{h^2}{12}(u_{txx})_j^{n+\frac{1}{2}} + O(\tau^2 + h^4). \tag{3.26}$$

Matching Eqs. (3.25) and (3.26), we obtain

$$(c_4)_j - \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}} + (c_5)_j + (c_6)_j + \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}} = 1, \tag{3.27a}$$

$$(c_6)_j + \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}} - (c_4)_j + \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}} = 0, \tag{3.27b}$$

$$(c_6)_j + \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}} + (c_4)_j - \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}} = \frac{1}{6}, \tag{3.27c}$$

which gives $(c_4)_j, (c_5)_j, (c_6)_j$ as listed in Eq. (3.20a), and hence complete the proof.

Using a similar argument, one may obtain the following lemma for a scheme at the interior points x_j, $j = m+1, \ldots, N-1$.

Lemma 3.2.4. Assume that the analytical solution $u(x,t)$ in Eqs. (3.1)-(3.4) is smooth in $[0, L] \times [0, \infty)$. Then it holds at the interior points x_j, $j = m+1, m+2, \ldots, N-1$,

$$\begin{aligned}(\tilde{c}_4)_j \delta_t u_{j-1}^{n+\frac{1}{2}} + (\tilde{c}_5)_j \delta_t u_j^{n+\frac{1}{2}} + (\tilde{c}_6)_j \delta_t u_{j+1}^{n+\frac{1}{2}} &= \frac{1}{h^2}[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}](u_{j+1}^{n+\frac{1}{2}} - u_j^{n+\frac{1}{2}}) \\ &- \frac{1}{h^2}[(k_2)_{j-\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j-\frac{1}{2}}](u_j^{n+\frac{1}{2}} - u_{j-1}^{n+\frac{1}{2}}) \\ &+ \tilde{f}_j^{n+\frac{1}{2}} + O(\tau^2 + h^4), \end{aligned} \tag{3.28}$$

where

$$(\tilde{c}_4)_j = \frac{1}{12} + \frac{h}{24}\frac{(k_{2x})_{j-1}}{(k_2)_{j-1}}, \quad (\tilde{c}_5)_j = \frac{5}{6}, \quad (\tilde{c}_6)_j = \frac{1}{12} - \frac{h}{24}\frac{(k_{2x})_{j+1}}{(k_2)_{j+1}}, \tag{3.29a}$$

$$(D_2)_{j+\frac{1}{2}} = 2\frac{[(k_{2x})_{j+\frac{1}{2}}]^2}{(k_2)_{j+\frac{1}{2}}} - (k_{2xx})_{j+\frac{1}{2}}, \tag{3.29b}$$

$$\begin{aligned}\tilde{f}_j^{n+\frac{1}{2}} &= [1 - \frac{h^2}{12}\frac{(k_2)_j(k_{2xx})_j - (k_2)_j^2}{(k_2)_j^2}](F_2)_j^{n+\frac{1}{2}} - \frac{h^2}{12}\frac{(k_{2x})_j}{(k_2)_j}(F_{2x})_j^{n+\frac{1}{2}} \\ &+ \frac{h^2}{12}(F_{2xx})_j^{n+\frac{1}{2}}. \end{aligned} \tag{3.29c}$$

It can be seen that if $k_2(x)$ is constant, then Eq. (3.28) reduces to

$$\begin{aligned}\frac{1}{12}\delta_t u_{j-1}^{n+\frac{1}{2}} + \frac{10}{12}\delta_t u_j^{n+\frac{1}{2}} + \frac{1}{12}\delta_t u_{j+1}^{n+\frac{1}{2}} &= \frac{k_2}{h^2}[u_{j-1}^{n+\frac{1}{2}} - 2u_j^{n+\frac{1}{2}} + u_{j+1}^{n+\frac{1}{2}}] + (F_2)_j^{n+\frac{1}{2}} \\ &+ \frac{h^2}{12}(F_{2xx})_j^{n+\frac{1}{2}} + O(\tau^2 + h^4). \end{aligned} \tag{3.30}$$

We now derive a scheme at the interfacial grid point $(x_{m-}, t_{n+1/2})$.

Lemma 3.2.5. Assume that the analytical solution $u(x,t)$ in Eqs. (3.1)-(3.4) is smooth in $[0,L]\times[0,\infty)$. Then it holds at x_{m-}

$$c_7\delta_t u_{m-1}^{n+\frac{1}{2}} + c_8\delta_t u_m^{n+\frac{1}{2}} = \frac{1}{h}[c_9(k_1)_m(u_x)_{m-}^{n+\frac{1}{2}} - \frac{1}{h}(k_1)_{m-\frac{1}{2}}(u_m^{n+\frac{1}{2}} - u_{m-1}^{n+\frac{1}{2}})]$$

$$- T_{m-}^{n+\frac{1}{2}} + O(\tau^2+h^3), \tag{3.31}$$

where

$$c_7 = \frac{1}{6} + \frac{h}{24}\frac{(k_{1x})_m}{(k_1)_m}, \tag{3.32a}$$

$$c_8 = \frac{1}{3} + \frac{h}{24}\frac{(k_{1x})_m}{(k_1)_m} + \frac{h^2}{48}\frac{4[(k_{1x})_m]^2 - 3(k_1)_m(k_{1xx})_m}{[(k_1)_m]^2}, \tag{3.32b}$$

$$c_9 = 1 + \frac{h^2}{24(k_1)_m}[\frac{2[(k_{1x})_m]^2}{(k_1)_m} - (k_{1xx})_m]$$

$$- \frac{h^3}{48(k_1)_m}[\frac{5(k_1)_m(k_{1x})_m(k_{1xx})_m - 4[(k_{1x})_m]^3}{[(k_1)_m]^2} - (k_{1xxx})_m], \tag{3.32c}$$

$$T_{m-}^{n+\frac{1}{2}} = z_1(u_{txx})_{m-}^{n+\frac{1}{2}} - f_{m-}^{n+\frac{1}{2}}, \tag{3.32d}$$

with

$$f_{m-}^{n+\frac{1}{2}} = [\frac{1}{2} + \frac{h}{12}\frac{(k_{1x})_m}{(k_1)_m} - \frac{h^2}{48}\frac{3(k_1)_m(k_{1xx})_m - 4[(k_{1x})_m]^2}{[(k_1)_m]^2}](F_1)_m^{n+\frac{1}{2}}$$

$$- [\frac{h}{6} + \frac{h^2}{24}\frac{(k_{1x})_m}{(k_1)_m}](F_{1x})_m^{n+\frac{1}{2}} + \frac{h^2}{24}(F_{1xx})_m^{n+\frac{1}{2}}, \tag{3.32e}$$

$$z_1 = -\frac{h^2}{24} - \frac{h^3}{48}\frac{(k_{1x})_m}{(k_1)_m}. \tag{3.32f}$$

If $k_1(x)$ is a constant, then Eq. (3.31) reduces to

$$\frac{1}{6}\delta_t u_{m-1}^{n+\frac{1}{2}} + \frac{1}{3}\delta_t u_m^{n+\frac{1}{2}} = \frac{k_1}{h}[(u_x)_{m-}^{n+\frac{1}{2}} - \frac{1}{h}(u_m^{n+\frac{1}{2}} - u_{m-1}^{n+\frac{1}{2}})] + \frac{h}{24}(u_{txx})_{m-}^{n+\frac{1}{2}} + \frac{1}{2}(F_1)_m^{n+\frac{1}{2}}$$

$$- \frac{h}{6}(F_{1x})_m^{n+\frac{1}{2}} + \frac{h^2}{24}(F_{1xx})_m^{n+\frac{1}{2}} + O(\tau^2+h^3). \tag{3.33}$$

Proof. Using the Taylor series expansion at x_{m-} similar to that for the RHS of Eq. (3.11) in Lemma 3.2.1, we have the RHS of Eq. (3.31) as

$$
\begin{aligned}
RHS = {} & P_2(u_x)_{m-}^{n+\frac{1}{2}} + P_1(u_t)_{m-}^{n+\frac{1}{2}} + [-\frac{h}{6} - \frac{h^2}{24}\frac{(k_{1x})_m}{(k_1)_m}](u_{tx})_{m-}^{n+\frac{1}{2}} + \frac{h^2}{24}(u_{txx})_{m-}^{n+\frac{1}{2}} \\
& - \frac{1}{2}(F_1)_m^{n+\frac{1}{2}} - \frac{h}{24}[2\frac{(k_{1x})_m}{(k_1)_m}(F_1)_m^{n+\frac{1}{2}} - 4(F_{1x})_m^{n+\frac{1}{2}}] \\
& + \frac{h^2}{48}\{\frac{3(k_1)_m(k_{1xx})_m - 4[(k_{1x})_m]^2}{[(k_1)_m]^2}(F_1)_m^{n+\frac{1}{2}} + 2\frac{(k_{1x})_m}{(k_1)_m}(F_{1x})_m^{n+\frac{1}{2}} - 2(F_{1xx})_m^{n+\frac{1}{2}}\} \\
& + [\frac{h^2}{24} + \frac{h^3}{48}\frac{(k_{1x})_m}{(k_1)_m}](u_{txx})_{m-}^{n+\frac{1}{2}} + f_{m-}^{n+\frac{1}{2}} + O(\tau^2 + h^3) \\
= {} & P_2(u_x)_{m-}^{n+\frac{1}{2}} + P_1(u_t)_{m-}^{n+\frac{1}{2}} + [-\frac{h}{6} - \frac{h^2}{24}\frac{(k_{1x})_m}{(k_1)_m}](u_{tx})_{m-}^{n+\frac{1}{2}} \\
& + [\frac{h^2}{12} + \frac{h^3}{48}\frac{(k_{1x})_m}{(k_1)_m}](u_{txx})_{m-}^{n+\frac{1}{2}} + O(\tau^2 + h^3), \qquad (3.34)
\end{aligned}
$$

where

$$
P_1 = \frac{1}{2} + \frac{h}{12}\frac{(k_{1x})_m}{(k_1)_m} + \frac{h^2}{48}\frac{4[(k_{1x})_m]^2 - 3(k_1)_m(k_{1xx})_m}{[(k_1)_m]^2}, \qquad (3.35a)
$$

$$
\begin{aligned}
P_2 = {} & \frac{1}{h}(c_9 - 1)(k_1)_m + \frac{h^2}{48}[\frac{5(k_1)_m(k_{1x})_m(k_{1xx})_m - 4[(k_1)_m]^3}{[(k_1)_m]^2} - (k_{1xxx})_m] \\
& - \frac{h}{24}[2\frac{[(k_{1x})_m]^2}{(k_1)_m} - (k_{1xx})_m]. \qquad (3.35b)
\end{aligned}
$$

Expanding the LHS of Eq. (3.31) at $(x_m, t_{n+1/2})$ gives

$$
LHS = (c_7 + c_8)(u_t)_{m-}^{n+\frac{1}{2}} - c_7 h(u_{tx})_{m-}^{n+\frac{1}{2}} + c_7\frac{h^2}{2}(u_{txx})_{m-}^{n+\frac{1}{2}} + O(\tau^2 + h^3). \qquad (3.36)
$$

Matching Eqs. (3.34) and (3.36) gives $P_2 = 0, c_7 + c_8 = P_1, c_7 = 1/6 + h/24[(k_{1x})_m/(k_1)_m]$, from which we obtain c_7, c_8, c_9 as listed in Eqs. (3.31a)-(3.31c), and hence the proof is completed.

Using a similar argument, one may obtain the following lemma for a scheme at x_m from the RHS of interface.

Lemma 3.2.6. Assume that the analytical solution $u(x,t)$ in Eqs. (3.1)-(3.4) is smooth in $[0,L]\times[0,\infty)$. Then it holds at x_{m+}

$$\tilde{c}_8\delta_t u_m^{n+\frac{1}{2}} + \tilde{c}_7\delta_t u_{m+1}^{n+\frac{1}{2}} = \frac{1}{h}[(k_2)_{m+\frac{1}{2}}\frac{u_{m+1}^{n+\frac{1}{2}} - u_m^{n+\frac{1}{2}}}{h} - \tilde{c}_9(k_2)_m(u_x)_{m+}^{n+\frac{1}{2}}]$$
$$- T_{m+}^{n+\frac{1}{2}} + O(\tau^2 + h^3), \tag{3.37}$$

where

$$\tilde{c}_7 = \frac{1}{6} - \frac{h}{24}\frac{(k_{2x})_m}{(k_2)_m}, \tag{3.38a}$$

$$\tilde{c}_8 = \frac{1}{3} - \frac{h}{24}\frac{(k_{2x})_m}{(k_2)_m} + \frac{h^2}{48}\frac{4[(k_{2x})_m]^2 - 3(k_2)_m(k_{2xx})_m}{[(k_2)_m]^2}, \tag{3.38b}$$

$$\tilde{c}_9 = 1 + \frac{h^2}{24(k_2)_m}[\frac{2[(k_{2x})_m]^2}{(k_2)_m} - (k_{2xx})_m]$$
$$+ \frac{h^3}{48(k_2)_m}[\frac{5(k_2)_m(k_{2x})_m(k_{2xx})_m - 4[(k_{2x})_m]^3}{[(k_2)_m]^2} - (k_{2xxx})_m], \tag{3.38c}$$

$$T_{m+}^{n+\frac{1}{2}} = z_2(u_{txx})_{m+}^{n+\frac{1}{2}} - f_{m+}^{n+\frac{1}{2}}, \tag{3.38d}$$

with

$$f_{m+}^{n+\frac{1}{2}} = [\frac{1}{2} - \frac{h}{12}\frac{(k_{2x})_m}{(k_2)_m} - \frac{h^2}{48}\frac{3(k_2)_m(k_{2xx})_m - 4[(k_{2x})_m]^2}{[(k_2)_m]^2}](F_2)_m^{n+\frac{1}{2}}$$
$$+ [\frac{h}{6} - \frac{h^2}{24}\frac{(k_{2x})_m}{(k_2)_m}](F_{2x})_m^{n+\frac{1}{2}} + \frac{h^2}{24}(F_{2xx})_m^{n+\frac{1}{2}}, \tag{3.38e}$$

$$z_2 = -\frac{h^2}{24} + \frac{h^3}{48}\frac{(k_{2x})_m}{(k_2)_m}. \tag{3.38f}$$

If $k_2(x)$ is a constant, then Eq. (3.37) reduces to

$$\frac{1}{3}\delta_t u_m^{n+\frac{1}{2}} + \frac{1}{6}\delta_t u_{m+1}^{n+\frac{1}{2}} = \frac{k_2}{h}\Big[\frac{u_{m+1}^{n+\frac{1}{2}} - u_m^{n+\frac{1}{2}}}{h} - (u_x)_{m+}^{n+\frac{1}{2}}\Big] + \frac{h}{24}(u_{txx})_{m+}^{n+\frac{1}{2}}$$
$$+ \frac{1}{2}(F_2)_m^{n+\frac{1}{2}} + \frac{h}{6}(F_{2x})_m^{n+\frac{1}{2}} + \frac{h^2}{24}(F_{2xx})_m^{n+\frac{1}{2}}$$
$$+ O(\tau^2 + h^3). \tag{3.39}$$

Based on the above lemmas 3.2.5-3.2.6, we now are able to derive a finite difference scheme at the interfacial grid point $(x_m, t_{n+1/2})$ using only three grid points.

Lemma 3.2.7. Assume that the analytical solution $u(x,t)$ in Eqs. (3.1)-(3.4) is smooth in $[0, L] \times [0, \infty)$. Then it holds at $(x_m, t_{n+1/2})$

$$c_{10}\delta_t u_{m-1}^{n+\frac{1}{2}} + c_{11}\delta_t u_m^{n+\frac{1}{2}} + c_{12}\delta_t u_{m+1}^{n+\frac{1}{2}}$$
$$= c_0(k_2)_{m+\frac{1}{2}}\frac{u_{m+1}^{n+\frac{1}{2}} - u_m^{n+\frac{1}{2}}}{h^2} - (k_1)_{m-\frac{1}{2}}\frac{u_m^{n+\frac{1}{2}} - u_{m-1}^{n+\frac{1}{2}}}{h^2} - T_m^{n+\frac{1}{2}} + O(\tau^2 + h^3), \tag{3.40}$$

where

$$c_0 = c_9/\tilde{c}_9, \qquad c_{10} = c_7 + z_1 a + c_0 z_2 \tilde{c}, \tag{3.41a}$$

$$c_{11} = c_8 + c_0\tilde{c}_8 + z_1 b + c_0 z_2 \tilde{b}, \quad c_{12} = c_0\tilde{c}_7 + z_1 c + c_0 z_2 \tilde{a}, \tag{3.41b}$$

$$T_m^{n+\frac{1}{2}} = -f_{m-}^{n+\frac{1}{2}} - c_0 f_{m+}^{n+\frac{1}{2}} + z_1 S_{m-}^{n+\frac{1}{2}} + c_0 z_2 S_{m+}^{n+\frac{1}{2}}. \tag{3.41c}$$

Here, $c_7, \tilde{c}_7, c_8, \tilde{c}_8, c_9, \tilde{c}_9, z_1, z_2, f_{m-}^{n+1/2}, f_{m+}^{n+1/2}$ are given in lemmas 3.2.5-3.2.6, and

$$a = \frac{2}{h^2} - \frac{c(k_1)_m}{(k_2)_m}, \quad b = -(a + c), \tag{3.42a}$$

$$c = \left[\frac{h^2(k_1)_m}{(k_2)_m} + \frac{h^3}{4(k_2)_m}[(k_{1x})_m - \frac{(k_{2x})_m(k_1)_m}{(k_2)_m}]\right]^{-1}, \tag{3.42b}$$

$$\tilde{a} = \frac{2}{h^2} - \frac{\tilde{c}(k_2)_m}{(k_1)_m}, \quad \tilde{b} = -(\tilde{a} + \tilde{c}), \tag{3.42c}$$

$$\tilde{c} = \left[\frac{h^2(k_2)_m}{(k_1)_m} + \frac{h^3}{4(k_1)_m}[(k_{2x})_m - \frac{(k_{1x})_m(k_2)_m}{(k_1)_m}]\right]^{-1}, \tag{3.42d}$$

$$S_{m-}^{n+\frac{1}{2}} = \frac{c}{2(k_2)_m}[(F_{2t})_m^{n+\frac{1}{2}} - (F_{1t})_m^{n+\frac{1}{2}}], \tag{3.42e}$$

$$S_{m+}^{n+\frac{1}{2}} = \frac{\tilde{c}}{2(k_1)_m}[(F_{1t})_m^{n+\frac{1}{2}} - (F_{2t})_m^{n+\frac{1}{2}}]. \tag{3.42f}$$

Proof. We multiply Eq. (3.37) by c_0 and then add with Eq. (3.31). This gives

$$\begin{aligned} & c_7\delta_t u_{m-1}^{n+\frac{1}{2}} + (c_8 + c_0\tilde{c}_8)\delta_t u_m^{n+\frac{1}{2}} + c_0\tilde{c}_7\delta_t u_{m+1}^{n+\frac{1}{2}} \\ & = \frac{1}{h^2}[c_0(k_2)_{m+\frac{1}{2}}(u_{m+1}^{n+\frac{1}{2}} - u_m^{n+\frac{1}{2}}) - (k_1)_{m-\frac{1}{2}}(u_m^{n+\frac{1}{2}} - u_{m-1}^{n+\frac{1}{2}})] \\ & - (T_{m-}^{n+\frac{1}{2}} + c_0 T_{m+}^{n+\frac{1}{2}}) + O(\tau^2 + h^3), \end{aligned} \tag{3.43}$$

where

$$T_{m-}^{n+\frac{1}{2}} = z_1(u_{txx})_{m-}^{n+\frac{1}{2}} - f_{m-}^{n+\frac{1}{2}}, \quad T_{m+}^{n+\frac{1}{2}} = z_2(u_{txx})_{m+}^{n+\frac{1}{2}} - f_{m+}^{n+\frac{1}{2}}. \tag{3.44}$$

We now discretize $(u_{txx})_{m-}^{n+1/2}$ as

$$(u_{txx})_{m-}^{n+\frac{1}{2}} = a\delta_t u_{m-1}^{n+\frac{1}{2}} + b\delta_t u_m^{n+\frac{1}{2}} + c\delta_t u_{m+1}^{n+\frac{1}{2}} - g_{m-}^{n+\frac{1}{2}}, \tag{3.45}$$

where $\delta_t u_m^{n+1/2} = (u_m^{n+1} - u_m^n)/\tau$, and a, b, c and $g_{m-}^{n+1/2}$ are constants to be determined. Using the Taylor series expansion at $t_{n+1/2}$, we obtain the RHS of Eq. (3.45) as

$$RHS = a(u_t)_{m-1}^{n+\frac{1}{2}} + b(u_t)_{m-}^{n+\frac{1}{2}} + c(u_t)_{m+1}^{n+\frac{1}{2}} - g_{m-}^{n+\frac{1}{2}} + O(\tau^2). \tag{3.46}$$

Using the Taylor series expansion at x_m, we obtain

$$(u_t)_{m+1}^{n+\frac{1}{2}} = (u_t)_{m+}^{n+\frac{1}{2}} + h(u_{tx})_{m+}^{n+\frac{1}{2}} + \frac{h^2}{2}(u_{txx})_{m+}^{n+\frac{1}{2}} + O(h^3), \tag{3.47a}$$

$$(u_t)_{m-1}^{n+\frac{1}{2}} = (u_t)_{m-}^{n+\frac{1}{2}} - h(u_{tx})_{m-}^{n+\frac{1}{2}} + \frac{h^2}{2}(u_{txx})_{m-}^{n+\frac{1}{2}} + O(h^3). \tag{3.47b}$$

From the interfacial condition in Eq. (3.4), we have $(u_x)_{m+}^{n+1/2} = (k_1)_m/(k_2)_m(u_x)_{m-}^{n+1/2}$ and $(u_{tt})_{m+}^{n+1/2} = (u_{tt})_{m-}^{n+1/2}$. Furthermore, from Eq. (3.1b) we obtain

$$(u_t)_{m+}^{n+\frac{1}{2}} = (k_{2x})_m(u_x)_{m+}^{n+\frac{1}{2}} + (k_2)_m(u_{xx})_{m+}^{n+\frac{1}{2}} + (F_2)_m^{n+\frac{1}{2}}, \tag{3.48}$$

and hence

$$\begin{aligned}(u_{xx})_{m+}^{n+\frac{1}{2}} &= \frac{1}{(k_2)_m}[(u_t)_{m+}^{n+\frac{1}{2}} - (k_{2x})_m(u_x)_{m+}^{n+\frac{1}{2}} - (F_2)_m^{n+\frac{1}{2}}] \\ &= \frac{1}{(k_2)_m}[(u_t)_{m-}^{n+\frac{1}{2}} - (k_{2x})_m\frac{(k_1)_m}{(k_2)_m}(u_x)_{m-}^{n+\frac{1}{2}} - (F_2)_m^{n+\frac{1}{2}}],\end{aligned} \tag{3.49a}$$

$$\begin{aligned}(u_{txx})_{m+}^{n+\frac{1}{2}} &= \frac{1}{(k_2)_m}[(u_{tt})_{m-}^{n+\frac{1}{2}} - (k_{2x})_m\frac{(k_1)_m}{(k_2)_m}(u_{tx})_{m-}^{n+\frac{1}{2}} - (F_{2t})_m^{n+\frac{1}{2}}] \\ &= \frac{1}{(k_2)_m}[(k_{1x})_m(u_{tx})_{m-}^{n+\frac{1}{2}} + (k_1)_m(u_{txx})_{m-}^{n+\frac{1}{2}} + (F_{1t})_m^{n+\frac{1}{2}}] \\ &\quad - (k_{2x})_m\frac{(k_1)_m}{[(k_2)_m]^2}(u_{tx})_{m-}^{n+\frac{1}{2}} - \frac{1}{(k_2)_m}(F_{2t})_m^{n+\frac{1}{2}}.\end{aligned} \tag{3.49b}$$

Substituting them into Eq. (3.47a) and then Eq. (3.47) into Eq. (3.46), we have from Eq. (3.45)

$$\begin{aligned}(u_{txx})_{m-}^{n+\frac{1}{2}} &= (a+b+c)(u_t)_{m-}^{n+\frac{1}{2}} \\ &+ \left(-ah + \frac{ch(k_1)_m}{(k_2)_m} + \frac{ch^2}{2(k_2)_m}[(k_{1x})_m - (k_{2x})_m\frac{(k_1)_m}{(k_2)_m}]\right)(u_{tx})_{m-}^{n+\frac{1}{2}} \\ &+ [\frac{ah^2}{2} + \frac{ch^2(k_1)_m}{2(k_2)_m}](u_{txx})_{m-}^{n+\frac{1}{2}} - S_{m-}^{n+\frac{1}{2}} - g_{m-}^{n+\frac{1}{2}} + O(\tau^2 + h^3),\end{aligned} \tag{3.50}$$

where $S_{m-}^{n+1/2}$ is given in Eq. (3.42e). Matching both sides of Eq. (3.50) indicates

$$a + b + c = 0, \tag{3.51a}$$

$$-ah + \frac{ch(k_1)_m}{(k_2)_m} + \frac{ch^2}{2(k_2)_m}[(k_{1x})_m - (k_{2x})_m\frac{(k_1)_m}{(k_2)_m}] = 0, \tag{3.51b}$$

$$\frac{ah^2}{2} + \frac{ch^2(k_1)_m}{2(k_2)_m} = 1, \tag{3.51c}$$

$$-S_{m-}^{n+\frac{1}{2}} - g_{m-}^{n+\frac{1}{2}} + O(\tau^2 + h^3) = 0, \tag{3.51d}$$

which gives a, b and c as listed in Eqs. (3.42a)-(3.42b), and $g_{m-}^{n+1/2} = -S_{m-}^{n+1/2}+O(\tau^2+h)$ (note that a, c are hidden in $O(h^3)$). Similarly, we can express $(u_{txx})_{m+}^{n+1/2}$ as

$$(u_{txx})_{m+}^{n+\frac{1}{2}} = \tilde{c}\delta_t u_{m-1}^{n+\frac{1}{2}} + \tilde{b}\delta_t u_m^{n+\frac{1}{2}} + \tilde{a}\delta_t u_{m+1}^{n+\frac{1}{2}} - g_{m+}^{n+\frac{1}{2}}, \tag{3.52}$$

where coefficients $\tilde{a}, \tilde{b}$, $\tilde{c}$ are listed in Eqs. (3.42c)-(3.42d), and $g_{m+}^{n+1/2} = -S_{m+}^{n+1/2} + O(\tau^2 + h)$. Substituting Eqs. (3.45) and (3.52) into Eq. (3.44) and then substituting Eq. (3.44) into Eq. (3.43), we obtain

$$\begin{aligned}
&(c_7 + z_1 a + c_0 z_2 \tilde{c})\delta_t u_{m-1}^{n+\frac{1}{2}} + (c_8 + c_0\tilde{c}_8 + z_1 b + c_0 z_2 \tilde{b})\delta_t u_m^{n+\frac{1}{2}} + (c_0\tilde{c}_7 + z_1 c + c_0 z_2 \tilde{a})\delta_t u_{m+1}^{n+\frac{1}{2}} \\
&= \frac{1}{h^2}[c_0(k_2)_{m+\frac{1}{2}}(u_{m+1}^{n+\frac{1}{2}} - u_m^{n+\frac{1}{2}}) - (k_1)_{m-\frac{1}{2}}(u_m^{n+\frac{1}{2}} - u_{m-1}^{n+\frac{1}{2}})] \\
&\quad + f_{m-}^{n+\frac{1}{2}} + c_0 f_{m+}^{n+\frac{1}{2}} - z_1 S_{m-}^{n+\frac{1}{2}} - c_0 z_2 S_{m+}^{n+\frac{1}{2}} + O(\tau^2 + h^3),
\end{aligned} \tag{3.53}$$

which gives Eq. (3.40) and coefficients as listed in Eq. (3.41), and hence complete the proof.

It should be pointed out that one may easily extend the above lemma for the discontinuous interface case, such as $u(x_{l+}, t) - u(x_{l-}, t) = \varphi_1(t)$, $k_2(x_l)u_x(x_{l+}, t) - k_1(x_l)u_x(x_{l-}, t) = \varphi_2(t)$. Based on the above lemmas 3.2.1-3.2.7, we first replace $u_j^{n+1/2}$ with

$$u_j^{n+\frac{1}{2}} = \frac{1}{2}(u_j^{n+1} + u_j^n) + O(\tau^2), \tag{3.54}$$

and then drop out the truncation errors, as well as use the notation U_j^n, an approximation of u_j^n, and the notations

$$\bar{U}_j^{n+\frac{1}{2}} = \frac{U_j^{n+1} + U_j^n}{2}, \quad \delta_t U_j^{n+\frac{1}{2}} = \frac{U_j^{n+1} - U_j^n}{\tau}. \tag{3.55}$$

Thus, an accurate compact finite difference scheme for solving Eqs. (3.1)-(3.4) is obtained as

$$C_1 \delta_t U_0^{n+\frac{1}{2}} + C_2 \delta_t U_1^{n+\frac{1}{2}} = \frac{1}{h}[(k_1)_{1/2} \frac{\bar{U}_1^{n+\frac{1}{2}} - \bar{U}_0^{n+\frac{1}{2}}}{h} - C_3 (k_1)_0 \alpha(t_{n+\frac{1}{2}})]$$

$$+ [-\frac{h}{12} + \frac{h^2}{24} \frac{(k_{1x})_0}{(k_1)_0}] \alpha'(t_{n+\frac{1}{2}}) + f_0^{n+\frac{1}{2}}; \tag{3.56a}$$

$$(C_4)_j \delta_t U_{j-1}^{n+\frac{1}{2}} + (C_5)_j \delta_t U_j^{n+\frac{1}{2}} + (C_6)_j \delta_t U_{j+1}^{n+\frac{1}{2}} = \frac{1}{h^2}[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}](\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}})$$

$$- \frac{1}{h^2}[(k_1)_{j-\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j-\frac{1}{2}}](\bar{U}_j^{n+\frac{1}{2}} - \bar{U}_{j-1}^{n+\frac{1}{2}})$$

$$+ f_j^{n+\frac{1}{2}}, 1 \leq j \leq m-1; \tag{3.56b}$$

$$C_7 \delta_t U_{m-1}^{n+\frac{1}{2}} + C_8 \delta_t U_m^{n+\frac{1}{2}} + C_9 \delta_t U_{m+1}^{n+\frac{1}{2}} = \frac{1}{h^2}[C_0 (k_2)_{m+\frac{1}{2}} (\bar{U}_{m+1}^{n+\frac{1}{2}} - \bar{U}_m^{n+\frac{1}{2}})$$

$$- (k_1)_{m-\frac{1}{2}} (\bar{U}_m^{n+\frac{1}{2}} - \bar{U}_{m-1}^{n+\frac{1}{2}})] - T_m^{n+\frac{1}{2}}; \tag{3.56c}$$

$$(\tilde{C}_4)_j \delta_t U_{j-1}^{n+\frac{1}{2}} + (\tilde{C}_5)_j \delta_t U_j^{n+\frac{1}{2}} + (\tilde{C}_6)_j \delta_t U_{j+1}^{n+\frac{1}{2}} = \frac{1}{h^2}[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}](\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}})$$

$$- \frac{1}{h^2}[(k_2)_{j-\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j-\frac{1}{2}}](\bar{U}_j^{n+\frac{1}{2}} - \bar{U}_{j-1}^{n+\frac{1}{2}})$$

$$+ \tilde{f}_j^{n+\frac{1}{2}}, m+1 \leq j \leq N-1; \tag{3.56d}$$

$$\tilde{C}_1 \delta_t U_N^{n+\frac{1}{2}} + \tilde{C}_2 \delta_t U_{N-1}^{n+\frac{1}{2}} = \frac{1}{h}[\tilde{C}_3 (k_2)_N \beta(t_{n+\frac{1}{2}}) - (k_2)_{N-\frac{1}{2}} \frac{\bar{U}_N^{n+\frac{1}{2}} - \bar{U}_{N-1}^{n+\frac{1}{2}}}{h}]$$

$$+ [\frac{h}{12} + \frac{h^2}{24} \frac{(k_{2x})_N}{(k_2)_N}] \beta'(t_{n+\frac{1}{2}}) + f_N^{n+\frac{1}{2}}; \tag{3.56e}$$

$$U_j^0 = \phi_1(x_j), \quad j = 0, 1, \cdots, m-1, m; \;\; U_j^0 = \phi_2(x_j), \quad j = m+1, \cdots, N, \tag{3.56e}$$

where coefficients $C_1 = c_1$, $C_2 = c_2$, $C_3 = c_3$, $(C_4)_j = (c_4)_j$, $(C_5)_j = (c_5)_j$, $(C_6)_j = (c_6)_j$, $C_7 = c_{10}$, $C_8 = c_{11}$, $C_9 = c_{12}$, $C_0 = c_0$, $\tilde{C}_1 = \tilde{c}_1$, $\tilde{C}_2 = \tilde{c}_2$, $\tilde{C}_3 = \tilde{c}_3$, $(\tilde{C}_4)_j = (\tilde{c}_4)_j$, $(\tilde{C}_5)_j = (\tilde{c}_5)_j$ and $(\tilde{C}_6) = (\tilde{c}_6)_j$ which are based on Lemmas 3.2.1-3.2.7, and also $f_0^{n+1/2}$,$f_j^{n+1/2}$, $\tilde{f}_j^{n+1/2}$, $f_N^{n+1/2}$, $T_m^{n+1/2}$, D_1 and D_2 are given in Lemmas 3.2.1-3.2.7.

It can be seen that the truncation error for the above scheme is $O(\tau^2 + h^3)$ at x_0, x_m and x_N, and $O(\tau^2 + h^4)$ at interior point x_j. Furthermore, Eqs. (3.56a)-(3.56e) consist of a tridiagonal linear system for solving U_j^{n+1}, $j = 0, \cdots, N$, which can be obtained using the Thomas algorithm. It should be pointed out that we have never discretized u_x in the interfacial and boundary conditions in our derivations.

3.3 Stability and Convergence

A Priori Estimate:

To analyze the stability and convergence of the present scheme, we first obtain a priori estimate. A priori estimate is an estimate for the size of a solution or its derivatives of a partial differential equation. We obtain a priori estimate for the following finite difference scheme as

$$C_1 \delta_t U_0^{n+\frac{1}{2}} + C_2 \delta_t U_1^{n+\frac{1}{2}} = \frac{1}{h}(k_1)_{1/2} \frac{\bar{U}_1^{n+\frac{1}{2}} - \bar{U}_0^{n+\frac{1}{2}}}{h} + g_0^{n+\frac{1}{2}}; \tag{3.57a}$$

$$\begin{aligned}(C_4)_j \delta_t U_{j-1}^{n+\frac{1}{2}} + (C_5)_j \delta_t U_j^{n+\frac{1}{2}} + (C_6)_j \delta_t U_{j+1}^{n+\frac{1}{2}} &= \frac{1}{h^2}[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}](\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}}) \\ &- \frac{1}{h^2}[(k_1)_{j-\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j-\frac{1}{2}}](\bar{U}_j^{n+\frac{1}{2}} - \bar{U}_{j-1}^{n+\frac{1}{2}}) \\ &+ g_j^{n+\frac{1}{2}}, 1 \le j \le m-1; \end{aligned} \tag{3.57b}$$

$$C_7\delta_t U_{m-1}^{n+\frac{1}{2}} + C_8\delta_t U_m^{n+\frac{1}{2}} + C_9\delta_t U_{m+1}^{n+\frac{1}{2}} = \frac{1}{h^2}[C_0(k_2)_{m+\frac{1}{2}}(\bar{U}_{m+1}^{n+\frac{1}{2}} - \bar{U}_m^{n+\frac{1}{2}})$$

$$- (k_1)_{m-\frac{1}{2}}(\bar{U}_m^{n+\frac{1}{2}} - \bar{U}_{m-1}^{n+\frac{1}{2}})] + g_m^{n+\frac{1}{2}}; \qquad (3.57c)$$

$$(\tilde{C}_4)_j\delta_t U_{j-1}^{n+\frac{1}{2}} + (\tilde{C}_5)_j\delta_t U_j^{n+\frac{1}{2}} + (\tilde{C}_6)_j\delta_t U_{j+1}^{n+\frac{1}{2}} = \frac{1}{h^2}[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}](\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}})$$

$$- \frac{1}{h^2}[(k_2)_{j-\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j-\frac{1}{2}}](\bar{U}_j^{n+\frac{1}{2}} - \bar{U}_{j-1}^{n+\frac{1}{2}})$$

$$+ g_j^{n+\frac{1}{2}}, m+1 \leq j \leq N-1; \qquad (3.57d)$$

$$\tilde{C}_1\delta_t U_N^{n+\frac{1}{2}} + \tilde{C}_2\delta_t U_{N-1}^{n+\frac{1}{2}} = -\frac{1}{h}(k_2)_{N-\frac{1}{2}}\frac{\bar{U}_N^{n+\frac{1}{2}} - \bar{U}_{N-1}^{n+\frac{1}{2}}}{h} + g_N^{n+\frac{1}{2}}, \qquad (3.57e)$$

where $g_j^{n+1/2}$, $0 \leq j \leq N$, is a mesh function and the coefficients are given in Eq. (3.56).

Lemma 3.3.1 (Gronwall's inequality). If $\{y_n\}, \{f_n\}$ and $\{g_n\}$ are non-negative sequences and satisfy

$$y_n \leq f_n + \sum_{k=0}^{n-1} g_k y_k, \quad n \geq 0, \qquad (3.58)$$

then it holds

$$y_n \leq f_n + \sum_{k=0}^{n-1} f_k g_k \exp(\sum_{j=k}^{n-1} g_j), \quad n \geq 0. \qquad (3.59)$$

Theorem 3.3.1. The finite difference scheme in Eq. (3.57) satisfies

$$E^n \leq \exp(CT)[E^0 + T \max_{0\leq k\leq n-1} G^{k+\frac{1}{2}}], \qquad (3.60)$$

where $0 < n\tau \leq T$ and

$$E^n = h\sum_{j=0}^{m-1}\frac{1}{2}\tilde{c}_9[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}](\nabla_x U_j^n)^2$$

$$+ h\sum_{j=m}^{N-1}\frac{1}{2}c_9[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}](\nabla_x U_j^n)^2, \tag{3.61a}$$

$$G^{k+\frac{1}{2}} = \frac{\tilde{c}_9}{4\varepsilon}h\sum_{j=0}^{m}(g_j^{k+\frac{1}{2}})^2 + \frac{c_9}{4\varepsilon}h\sum_{j=m+1}^{N}(g_j^{k+\frac{1}{2}})^2. \tag{3.61b}$$

Here, C and ε are positive constants, c_9 and $\tilde{c}_9$ are given in Lemmas 3.2.5 and 3.2.6.

Proof. We now multiply Eqs. (3.57a)-(3.57e) by $h\tilde{c}_9\delta_t U_0^{n+1/2}$, $h\tilde{c}_9\delta_t U_j^{n+1/2}$, $h\tilde{c}_9\delta_t U_m^{n+1/2}$, $hc_9\delta_t U_j^{n+1/2}$, and $hc_9\delta_t U_N^{n+1/2}$, respectively, and then add them together. This gives an equation where the LHS is

$$LHS = h\tilde{c}_9[C_1(\delta_t U_0^{n+\frac{1}{2}})^2 + C_2(\delta_t U_1^{n+\frac{1}{2}})(\delta_t U_0^{n+\frac{1}{2}})]$$

$$+ h\sum_{j=1}^{m-1}\tilde{c}_9\left[(C_4)_j(\delta_t U_{j-1}^{n+\frac{1}{2}})(\delta_t U_j^{n+\frac{1}{2}}) + (C_5)_j(\delta_t U_j^{n+\frac{1}{2}})^2 + (C_6)_j(\delta_t U_{j+1}^{n+\frac{1}{2}})(\delta_t U_j^{n+\frac{1}{2}})\right]$$

$$+ h\tilde{c}_9\left[C_7\delta_t(U_{m-1}^{n+\frac{1}{2}})(\delta_t U_m^{n+\frac{1}{2}}) + C_8(\delta_t U_m^{n+\frac{1}{2}})^2 + C_9(\delta_t U_{m+1}^{n+\frac{1}{2}})(\delta_t U_m^{n+\frac{1}{2}})\right]$$

$$+ h\sum_{j=m+1}^{N-1}c_9\left[(\tilde{C}_4)_j(\delta_t U_{j-1}^{n+\frac{1}{2}})(\delta_t U_j^{n+\frac{1}{2}}) + (\tilde{C}_5)_j(\delta_t U_j^{n+\frac{1}{2}})^2 + (\tilde{C}_6)_j(\delta_t U_{j+1}^{n+\frac{1}{2}})(\delta_t U_j^{n+\frac{1}{2}})\right]$$

$$+ hc_9[\tilde{C}_1(\delta_t U_N^{n+\frac{1}{2}})^2 + \tilde{C}_2(\delta_t U_{N-1}^{n+\frac{1}{2}})(\delta_t U_N^{n+\frac{1}{2}})], \tag{3.62}$$

and the RHS is

$$
\begin{aligned}
RHS = & \frac{1}{h}\tilde{c}_9 (k_1)_{\frac{1}{2}} (\bar{U}_1^{n+\frac{1}{2}} - \bar{U}_0^{n+\frac{1}{2}})(\delta_t U_0^{n+\frac{1}{2}}) \\
& + \frac{1}{h}\sum_{j=1}^{m-1} \tilde{c}_9 \left[(k_1)_{j+\frac{1}{2}} (\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}}) - (k_1)_{j-\frac{1}{2}} (\bar{U}_j^{n+\frac{1}{2}} - \bar{U}_{j-1}^{n+\frac{1}{2}}) \right] (\delta_t U_j^{n+\frac{1}{2}}) \\
& - \frac{h}{24}\sum_{j=1}^{m-1} \tilde{c}_9 \left[(D_1)_{j+\frac{1}{2}} (\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}}) - (D_1)_{j-\frac{1}{2}} (\bar{U}_j^{n+\frac{1}{2}} - \bar{U}_{j-1}^{n+\frac{1}{2}}) \right] (\delta_t U_j^{n+\frac{1}{2}}) \\
& + \frac{1}{h} [c_9 (k_2)_{m+\frac{1}{2}} (\bar{U}_{m+1}^{n+\frac{1}{2}} - \bar{U}_m^{n+\frac{1}{2}}) - \tilde{c}_9 (k_1)_{m-\frac{1}{2}} (\bar{U}_m^{n+\frac{1}{2}} - \bar{U}_{m-1}^{n+\frac{1}{2}})] (\delta_t U_m^{n+\frac{1}{2}}) \\
& + \frac{1}{h}\sum_{j=m+1}^{N-1} c_9 \left[(k_2)_{j+\frac{1}{2}} (\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}}) - (k_2)_{j-\frac{1}{2}} (\bar{U}_j^{n+\frac{1}{2}} - \bar{U}_{j-1}^{n+\frac{1}{2}}) \right] (\delta_t U_j^{n+\frac{1}{2}}) \\
& - \frac{h}{24}\sum_{j=m+1}^{N-1} c_9 \left[(D_2)_{j+\frac{1}{2}} (\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}}) - (D_2)_{j-\frac{1}{2}} (\bar{U}_j^{n+\frac{1}{2}} - \bar{U}_{j-1}^{n+\frac{1}{2}}) \right] (\delta_t U_j^{n+\frac{1}{2}}) \\
& - \frac{1}{h} c_9 (k_2)_{N-\frac{1}{2}} (\bar{U}_N^{n+\frac{1}{2}} - \bar{U}_{N-1}^{n+\frac{1}{2}})(\delta_t U_N^{n+\frac{1}{2}}) + h\tilde{c}_9 \sum_{j=0}^{m} g_j^{n+\frac{1}{2}} (\delta_t U_j^{n+\frac{1}{2}}) \\
& + hc_9 \sum_{j=m+1}^{N} g_j^{n+\frac{1}{2}} (\delta_t U_j^{n+\frac{1}{2}}).
\end{aligned} \tag{3.63}
$$

Using the Young inequality, we have

$$
(\delta_t U_j^{n+\frac{1}{2}})(\delta_t U_{j+1}^{n+\frac{1}{2}}) \geq -\frac{1}{2}[(\delta_t U_j^{n+\frac{1}{2}})^2 + (\delta_t U_{j+1}^{n+\frac{1}{2}})^2], \tag{3.64}
$$

and so on for other similar terms. Thus, Eq. (3.62) becomes

$$
\begin{aligned}
LHS \geq & A_0 + h\sum_{j=1}^{m-1} \tilde{c}_9 \left[\frac{-(C_4)_j}{2} (\delta_t U_{j-1}^{n+\frac{1}{2}})^2 + \alpha_j (\delta_t U_j^{n+\frac{1}{2}})^2 - \frac{(C_6)_j}{2} (\delta_t U_{j+1}^{n+\frac{1}{2}})^2 \right] \\
& + A_m + h\sum_{j=m+1}^{N-1} c_9 \left[\frac{-(\tilde{C}_4)_j}{2} (\delta_t U_{j-1}^{n+\frac{1}{2}})^2 + \tilde{\alpha}_j (\delta_t U_j^{n+\frac{1}{2}})^2 - \frac{(\tilde{C}_6)_j}{2} (\delta_t U_{j+1}^{n+\frac{1}{2}})^2 \right] \\
& + A_N,
\end{aligned} \tag{3.65a}
$$

where,

$$\alpha_j = (C_5)_j - \frac{(C_4)_j}{2} - \frac{(C_6)_j}{2} \tag{3.65b}$$

$$\tilde{\alpha}_j = [(\tilde{C}_5)_j - \frac{(\tilde{C}_4)_j}{2} - \frac{(\tilde{C}_6)_j}{2}] \tag{3.65c}$$

$$A_0 = h\tilde{c}_9(C_1 - \frac{C_2}{2})(\delta_t U_0^{n+\frac{1}{2}})^2 - \frac{h\tilde{c}_9 C_2}{2}(\delta_t U_1^{n+\frac{1}{2}})^2 \tag{3.65d}$$

$$A_m = -\frac{\tilde{c}_9 C_7}{2}(\delta_t U_{m-1}^{n+\frac{1}{2}})^2 + h\tilde{c}_9(C_8 - \frac{C_7}{2} - \frac{C_9}{2})(\delta_t U_m^{n+\frac{1}{2}})^2 - \frac{h\tilde{c}_9 C_9}{2}(\delta_t U_{m+1}^{n+\frac{1}{2}})^2 \tag{3.65e}$$

$$A_N = -\frac{hc_9\tilde{C}_2}{2}(\delta_t U_{N-1}^{n+\frac{1}{2}})^2 + hc_9(\tilde{C}_1 - \frac{\tilde{C}_2}{2})(\delta_t U_N^{n+\frac{1}{2}})^2 \tag{3.65f}$$

which can be further simplified to

$$\begin{aligned} LHS \geq\ & h\tilde{c}_9\eta_0(\delta_t U_0^{n+\frac{1}{2}})^2 + h\tilde{c}_9\eta_1(\delta_t U_1^{n+\frac{1}{2}})^2 + h\sum_{j=1}^{m-1}\tilde{c}_9\eta_j(\delta_t U_j^{n+\frac{1}{2}})^2 \\ & + h\tilde{c}_9\gamma_1(\delta_t U_{m-1}^{n+\frac{1}{2}})^2 + h\tilde{c}_9\gamma_2(\delta_t U_m^{n+\frac{1}{2}})^2 + h\gamma_3(\delta_t U_{m+1}^{n+\frac{1}{2}})^2 \\ & + h\sum_{j=m+1}^{N-1} c_9\tilde{\eta}_j(\delta_t U_j^{n+\frac{1}{2}})^2 + hc_9\tilde{\eta}_{N-1}(\delta_t U_{N-1}^{n+\frac{1}{2}})^2 + hc_9\tilde{\eta}_N(\delta_t U_N^{n+\frac{1}{2}})^2. \end{aligned} \tag{3.66a}$$

where,

$$\eta_0 = C_1 - \frac{1}{2}C_2 - \frac{1}{2}(C_4)_1 \tag{3.66b}$$

$$\eta_1 = (C_5)_1 - \frac{(C_4)_1}{2} - \frac{(C_4)_2}{2} - \frac{(C_6)_1}{2} - \frac{C_2}{2} \tag{3.66c}$$

$$\eta_j = \alpha_j - \frac{(C_4)_{j+1}}{2} - \frac{(C_6)_{j-1}}{2} \tag{3.66d}$$

$$\gamma_1 = (C_5)_{m-1} - \frac{1}{2}(C_4)_{m-1} - \frac{1}{2}(C_6)_{m-2} - \frac{1}{2}(C_6)_{m-1} - \frac{1}{2}C_7 \tag{3.66e}$$

$$\gamma_2 = C_8 - \frac{1}{2}C_7 - \frac{1}{2}C_9 - \frac{1}{2}(C_6)_{m-1} - \frac{1}{2}\frac{c_9}{\tilde{c}_9}(\tilde{C}_4)_{m+1} \tag{3.66f}$$

$$\gamma_3 = c_9[(\tilde{C}_5)_{m+1} - \frac{1}{2}(\tilde{C}_4)_{m+1} - \frac{1}{2}(\tilde{C}_4)_{m+2} - \frac{1}{2}(\tilde{C}_6)_{m+1}] - \frac{1}{2}\tilde{c}_9 C_9 \tag{3.66g}$$

$$\tilde{\eta}_j = \left[\tilde{\alpha}_j - \frac{1}{2}(\tilde{C}_4)_{j+1} - \frac{1}{2}(\tilde{C}_6)_{j-1}\right] \tag{3.66h}$$

$$\tilde{\eta}_{N-1} = \left[(\tilde{C}_5)_{N-1} - \frac{1}{2}(\tilde{C}_4)_{N-1} - \frac{1}{2}(\tilde{C}_6)_{N-2} - \frac{1}{2}(\tilde{C}_6)_{N-1} - \frac{1}{2}\tilde{C}_2\right] \tag{3.66i}$$

$$\tilde{\eta}_N = \tilde{C}_1 - \frac{1}{2}\tilde{C}_2 - \frac{1}{2}(\tilde{C}_6)_{N-1} \tag{3.66j}$$

After some detailed algebraic computations, these coefficients in Eq. (3.66a) satisfy

$$\eta_0 = \frac{1}{3} - \frac{5h}{48}\frac{(k_{1x})_0}{(k_1)_0} + \frac{h^2}{48}[\frac{4(k_{1x})_0 - 3(k_1)_0(k_{1xx})_0}{[(k_{1x})_0]^2}], \tag{3.67a}$$

$$\eta_2 = \frac{2}{3} - \frac{h}{48}[\frac{(k_{1x})_0}{(k_1)_0} + \frac{(k_{1x})_1}{(k_1)_1} - \frac{(k_{1x})_2}{(k_1)_2}], \tag{3.67b}$$

$$\eta_j = \frac{2}{3} + \frac{h}{48}[\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}} - \frac{(k_{1x})_{j-1}}{(k_1)_{j-1}}], \tag{3.67c}$$

$$\begin{aligned}\gamma_1 &= \frac{31}{48} + \frac{1}{48}\frac{(k_1)_m}{(k_2)_m} - \frac{h}{96}[\frac{(k_{1x})_m}{(k_1)_m} + \frac{(k_1)_m(k_{2x})_m}{[(k_2)_m]^2}] \\ &+ \frac{h}{48}[\frac{(k_{1x})_m}{(k_1)_m} + \frac{(k_{1x})_{m-1}}{(k_1)_{m-1}} - \frac{(k_{1x})_{m-2}}{(k_1)_{m-2}}] + O(h^2),\end{aligned} \tag{3.67d}$$

$$\begin{aligned}\gamma_2 &= \frac{13}{24} + \frac{1}{16}[\frac{(k_1)_m}{(k_2)_m} + \frac{(k_2)_m}{(k_1)_m}] + \frac{7h}{96}[\frac{(k_{1x})_m}{(k_1)_m} - \frac{(k_{2x})_m}{(k_2)_m}] \\ &+ \frac{h}{32}\frac{(k_{1x})_m(k_2)_m}{[(k_1)_m]^2} - \frac{h}{32}\frac{(k_{2x})_m(k_1)_m}{[(k_2)_m]^2} + O(h^2),\end{aligned} \tag{3.67e}$$

$$\begin{aligned}\gamma_3 &= \frac{31}{48} + \frac{1}{48}\frac{(k_2)_m}{(k_1)_m} + \frac{h}{96}[\frac{(k_2)_m(k_{1x})_m}{[(k_1)_m]^2} + \frac{(k_{2x})_m}{(k_2)_m}] \\ &- \frac{h}{48}[\frac{(k_{2x})_m}{(k_2)_m} + \frac{(k_{2x})_{m+1}}{(k_2)_{m+1}} - \frac{(k_{2x})_{m+2}}{(k_2)_{m+2}}] + O(h^2),\end{aligned} \tag{3.67f}$$

$$\tilde{\eta}_j = \frac{2}{3} + \frac{h}{48}[\frac{(k_{2x})_{j+1}}{(k_2)_{j+1}} - \frac{(k_{2x})_{j-1}}{(k_2)_{j-1}}], \tag{3.67g}$$

$$\tilde{\eta}_{N-1} = \frac{2}{3} + \frac{h}{48}[\frac{(k_{2x})_N}{(k_2)_N} + \frac{(k_{2x})_{N-1}}{(k_2)_{N-1}} - \frac{(k_{2x})_{N-2}}{(k_2)_{N-2}}], \tag{3.67h}$$

$$\tilde{\eta}_N = \frac{1}{3} + \frac{5h}{48}\frac{(k_{2x})_N}{(k_2)_N} + \frac{h^2}{48}[\frac{4(k_{2x})_N - 3(k_2)_N(k_{2xx})_N}{[(k_{2x})_N]^2}], \tag{3.67i}$$

implying that they are all positive if h is small. Hence, when h is small enough, we may obtain

$$LHS \geq K\left[h(\delta_t U_0^{n+\frac{1}{2}})^2 + h\sum_{j=1}^{N-1}(\delta_t U_j^{n+\frac{1}{2}})^2 + h(\delta_t U_N^{n+\frac{1}{2}})^2\right], \tag{3.68}$$

where K is a positive constant.

On the other hand, we rewrite Eq. (3.63) by shifting the indices for couple of the summations and then re-grouping the summations. This gives

$$\begin{aligned}
RHS = {} & \tilde{c}_9\frac{h^2}{24}(D_1)_{\frac{1}{2}}\frac{\bar{U}_1^{n+\frac{1}{2}} - \bar{U}_0^{n+\frac{1}{2}}}{h}(\delta_t U_0^{n+\frac{1}{2}}) \\
& - h\sum_{j=0}^{m-1}\tilde{c}_9[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}]\frac{\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}}}{h}\frac{1}{h}\delta_t(U_{j+1}^{n+\frac{1}{2}} - U_j^{n+\frac{1}{2}}) \\
& - \tilde{c}_9\frac{h^2}{24}(D_1)_{m-\frac{1}{2}}\frac{\bar{U}_m^{n+\frac{1}{2}} - \bar{U}_{m-1}^{n+\frac{1}{2}}}{h}(\delta_t U_m^{n+\frac{1}{2}}) \\
& + c_9\frac{h^2}{24}(D_2)_{m+\frac{1}{2}}\frac{\bar{U}_{m+1}^{n+\frac{1}{2}} - \bar{U}_m^{n+\frac{1}{2}}}{h}(\delta_t U_m^{n+\frac{1}{2}}) \\
& - h\sum_{j=m}^{N-1}c_9[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}]\frac{\bar{U}_{j+1}^{n+\frac{1}{2}} - \bar{U}_j^{n+\frac{1}{2}}}{h}\frac{1}{h}\delta_t(U_{j+1}^{n+\frac{1}{2}} - U_j^{n+\frac{1}{2}}) \\
& - c_9\frac{h^2}{24}(D_2)_{N-\frac{1}{2}}\frac{\bar{U}_N^{n+\frac{1}{2}} - \bar{U}_{N-1}^{n+\frac{1}{2}}}{h}(\delta_t U_N^{n+\frac{1}{2}}) \\
& + h\tilde{c}_9\sum_{j=0}^{m}g_j^{n+\frac{1}{2}}(\delta_t U_j^{n+\frac{1}{2}}) + hc_9\sum_{j=m+1}^{N}g_j^{n+\frac{1}{2}}(\delta_t U_j^{n+\frac{1}{2}}).
\end{aligned} \tag{3.69}$$

Using the notation $\nabla_x \bar{U}_j^{n+1/2} = 1/h(\bar{U}_{j+1}^{n+1/2} - \bar{U}_j^{n+1/2})$, Eq. (3.69) can be simplified to

$$\begin{aligned}
RHS = & -h\sum_{j=0}^{m-1}\tilde{c}_9[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}](\nabla_x \bar{U}_j^{n+\frac{1}{2}})(\delta_t \nabla_x U_j^{n+\frac{1}{2}}) \\
& - h\sum_{j=m}^{N-1}c_9[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}](\nabla_x \bar{U}_j^{n+\frac{1}{2}})(\delta_t \nabla_x U_j^{n+\frac{1}{2}}) \\
& + \frac{h^2}{24}\tilde{c}_9(D_1)_{\frac{1}{2}}(\nabla_x \bar{U}_0^{n+\frac{1}{2}})(\delta_t U_0^{n+\frac{1}{2}}) - \frac{h^2}{24}\tilde{c}_9(D_1)_{m-\frac{1}{2}}(\nabla_x \bar{U}_{m-1}^{n+\frac{1}{2}})(\delta_t U_m^{n+\frac{1}{2}}) \\
& + \frac{h^2}{24}c_9(D_2)_{m+\frac{1}{2}}(\nabla_x \bar{U}_m^{n+\frac{1}{2}})(\delta_t U_m^{n+\frac{1}{2}}) - \frac{h^2}{24}c_9(D_2)_{N-\frac{1}{2}}(\nabla_x \bar{U}_{N-1}^{n+\frac{1}{2}})(\delta_t U_N^{n+\frac{1}{2}}) \\
& + h\tilde{c}_9\sum_{j=0}^{m} g_j^{n+\frac{1}{2}}\delta_t U_j^{n+\frac{1}{2}} + hc_9\sum_{j=m+1}^{N} g_j^{n+\frac{1}{2}}\delta_t U_j^{n+\frac{1}{2}}. \qquad (3.70)
\end{aligned}$$

Note that

$$(\nabla_x \bar{U}_j^{n+\frac{1}{2}})(\delta_t \nabla_x U_j^{n+\frac{1}{2}}) = \frac{1}{2\tau}[(\nabla_x U_j^{n+1})^2 - (\nabla_x U_j^{n})^2], \qquad (3.71)$$

Eq. (3.70) becomes

$$\begin{aligned}
RHS = & -\frac{1}{2\tau}h\sum_{j=0}^{m-1}\tilde{c}_9[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}][(\nabla_x U_j^{n+1})^2 - (\nabla_x U_j^{n})^2] \\
& - \frac{1}{2\tau}h\sum_{j=m}^{N-1}c_9[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}][(\nabla_x U_j^{n+1})^2 - (\nabla_x U_j^{n})^2] \\
& + \frac{h^2}{24}\tilde{c}_9(D_1)_{\frac{1}{2}}(\nabla_x \bar{U}_0^{n+\frac{1}{2}})(\delta_t U_0^{n+\frac{1}{2}}) - \frac{h^2}{24}\tilde{c}_9(D_1)_{m-\frac{1}{2}}(\nabla_x \bar{U}_{m-1}^{n+\frac{1}{2}})(\delta_t U_m^{n+\frac{1}{2}}) \\
& + \frac{h^2}{24}c_9(D_2)_{m+\frac{1}{2}}(\nabla_x \bar{U}_m^{n+\frac{1}{2}})(\delta_t U_m^{n+\frac{1}{2}}) - \frac{h^2}{24}c_9(D_2)_{N-\frac{1}{2}}(\nabla_x \bar{U}_{N-1}^{n+\frac{1}{2}})(\delta_t U_N^{n+\frac{1}{2}}) \\
& + h\tilde{c}_9\sum_{j=0}^{m} g_j^{n+\frac{1}{2}}\delta_t U_j^{n+\frac{1}{2}} + hc_9\sum_{j=m+1}^{N} g_j^{n+\frac{1}{2}}\delta_t U_j^{n+\frac{1}{2}}. \qquad (3.72)
\end{aligned}$$

By the Cauchy-Schwarz inequality, we have

$$\begin{aligned}
\nabla_x \bar{U}_i^{n+\frac{1}{2}} \delta_t U_j^{n+\frac{1}{2}} &\leq \frac{1}{4\varepsilon} (\nabla_x \bar{U}_i^{n+\frac{1}{2}})^2 + \varepsilon (\delta_t U_j^{n+\frac{1}{2}})^2 \\
&\leq \frac{1}{8\varepsilon} [(\nabla_x U_i^{n+1})^2 + (\nabla_x U_i^n)^2] + \varepsilon (\delta_t U_j^{n+\frac{1}{2}})^2, \qquad (3.73a)
\end{aligned}$$

$$g_j^{n+\frac{1}{2}} \delta_t U_j^{n+\frac{1}{2}} \leq \frac{1}{4\varepsilon} (g_j^{n+\frac{1}{2}})^2 + \varepsilon (\delta_t U_j^{n+\frac{1}{2}})^2, \qquad (3.73b)$$

and hence if h is small enough, then Eq. (3.72) can be further written as

$$\begin{aligned}
RHS \leq &- h \sum_{j=0}^{m-1} \frac{\tilde{c}_9}{2\tau} [(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24} (D_1)_{j+\frac{1}{2}}] [(\nabla_x U_j^{n+1})^2 - (\nabla_x U_j^n)^2] \\
&- h \sum_{j=m}^{N-1} \frac{c_9}{2\tau} [(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24} (D_2)_{j+\frac{1}{2}}] [(\nabla_x U_j^{n+1})^2 - (\nabla_x U_j^n)^2] \\
&+ \frac{h^2}{192\varepsilon} \tilde{c}_9 \left| (D_1)_{\frac{1}{2}} \right| [\nabla_x (U_0^{n+1})^2 + \nabla_x (U_0^n)^2] + \frac{h^2 \varepsilon}{24} \tilde{c}_9 (\delta_t U_0^{n+\frac{1}{2}})^2 \\
&+ \frac{h^2}{192\varepsilon} \tilde{c}_9 \left| (D_1)_{m-\frac{1}{2}} \right| [\nabla_x (U_{m-1}^{n+1})^2 + \nabla_x (U_{m-1}^n)^2] + \frac{h^2 \varepsilon}{24} \tilde{c}_9 (\delta_t U_m^{n+\frac{1}{2}})^2 \\
&+ \frac{h^2}{192\varepsilon} c_9 \left| (D_2)_{m+\frac{1}{2}} \right| [\nabla_x (U_m^{n+1})^2 + \nabla_x (U_m^n)^2] + \frac{h^2 \varepsilon}{24} c_9 (\delta_t U_m^{n+\frac{1}{2}})^2 \\
&+ \frac{h^2}{192\varepsilon} c_9 \left| (D_2)_{N-\frac{1}{2}} \right| [\nabla_x (U_{N-1}^{n+1})^2 + \nabla_x (U_{N-1}^n)^2] + \frac{h^2 \varepsilon}{24} c_9 (\delta_t U_N^{n+\frac{1}{2}})^2 \\
&+ h\varepsilon \tilde{c}_9 \sum_{j=0}^{m} (\delta_t U_j^{n+\frac{1}{2}})^2 + h\varepsilon c_9 \sum_{j=m+1}^{N} (\delta_t U_j^{n+\frac{1}{2}})^2 \\
&+ \tilde{c}_9 \frac{1}{4\varepsilon} h \sum_{j=0}^{m} (g_j^{n+\frac{1}{2}})^2 + c_9 \frac{1}{4\varepsilon} h \sum_{j=m+1}^{N} (g_j^{n+\frac{1}{2}})^2. \qquad (3.74)
\end{aligned}$$

Choosing small ε and a constant $\varepsilon_0 \geq \max(h\varepsilon c_9/24, h\varepsilon\tilde{c}_9/24, \varepsilon c_9, \varepsilon \tilde{c}_9)$ such that

$$K - \varepsilon_0 \geq 0,$$

we obtain from Eqs. (3.68) and (3.74) that

$$(K-\varepsilon_0)h[(\delta_t U_0^{n+\frac{1}{2}})^2+\sum_{j=1}^{N-1}(\delta_t U_j^{n+\frac{1}{2}})^2+(\delta_t U_N^{n+\frac{1}{2}})^2]+h\sum_{j=0}^{m-1}\frac{\tilde{c}_9}{2\tau}[(k_1)_{j+\frac{1}{2}}-\frac{h^2}{24}(D_1)_{j+\frac{1}{2}}]$$

$$[(\nabla_x U_j^{n+1})^2-(\nabla_x U_j^n)^2]+h\sum_{j=m}^{N-1}\frac{c_9}{2\tau}[(k_2)_{j+\frac{1}{2}}-\frac{h^2}{24}(D_2)_{j+\frac{1}{2}}][(\nabla_x U_j^{n+1})^2-(\nabla_x U_j^n)^2]$$

$$\leq \frac{h^2}{192\varepsilon}\tilde{c}_9(\left|(D_1)_{\frac{1}{2}}\right|[\nabla_x(U_0^{n+1})^2+\nabla_x(U_0^n)^2]+\left|(D_1)_{m-\frac{1}{2}}\right|[\nabla_x(U_{m-1}^{n+1})^2+\nabla_x(U_{m-1}^n)^2])$$

$$+\frac{h^2}{192\varepsilon}c_9(\left|(D_2)_{m+\frac{1}{2}}\right|[\nabla_x(U_m^{n+1})^2+\nabla_x(U_m^{n+1})^2]+\left|(D_2)_{N-\frac{1}{2}}\right|[\nabla_x(U_{N-1}^{n+1})^2+$$

$$\nabla_x(U_{N-1}^n)^2])+\tilde{c}_9\frac{1}{4\varepsilon}h\sum_{j=0}^{m}(g_j^{n+\frac{1}{2}})^2+c_9\frac{1}{4\varepsilon}h\sum_{j=m+1}^{N}(g_j^{n+\frac{1}{2}})^2. \quad (3.75)$$

Using the notation in Eq. (3.61) and noticing

$$\frac{\tilde{c}_9}{192\varepsilon}\left|(D_1)_{j+\frac{1}{2}}\right|=\frac{\tilde{c}_9}{192\varepsilon}\frac{\left|(D_1)_{j+\frac{1}{2}}\right|}{(k_1)_{j+\frac{1}{2}}}(k_1)_{j+\frac{1}{2}}\leq A(k_1)_{j+\frac{1}{2}}, \quad (3.76)$$

where $A=\max_j[\tilde{c}_9\left|(D_1)_{j+1/2}\right|(k_1)_{j+1/2},c_9\left|(D_2)_{j+1/2}\right|(k_2)_{j+1/2}]/(192\varepsilon)$, Eq. (3.75) can be simplified to

$$E^{n+1}-E^n\leq Ah\tau(E^{n+1}+E^n)+2\tau G^{n+\frac{1}{2}}, \quad (3.77)$$

implying that

$$(1-Ah\tau)(E^{n+1}-E^n)\leq 2Ah\tau E^n+2\tau G^{n+\frac{1}{2}}. \quad (3.78)$$

If τ is sufficiently small such that $1-Ah\tau\geq 1/2$, then we have

$$E^{n+1}-E^n\leq 4A\tau E^n+4\tau G^{n+\frac{1}{2}}, \quad (3.79)$$

implying that

$$E^n\leq E^0+4A\tau\sum_{k=0}^{n-1}E^k+4\tau\sum_{k=0}^{n-1}G^{k+\frac{1}{2}}. \quad (3.80)$$

By the Gronwall inequality, we obtain the priori estimate in Eq. (3.60).

Using the above priori estimate, we can straightforwardly obtain the following theorem.

Theorem 3.3.2. Assume that $(U_1)_j^n$ and $(U_2)_j^n$ are two numerical solutions obtained based on the scheme in Eq. (3.56a)-Eq. (3.56e) with same boundary and interfacial conditions, but different initial conditions and source terms $F_1^{(1)}(x,t)$, $F_2^{(1)}(x,t)$ and $F_1^{(2)}(x,t)$, $F_2^{(2)}(x,t)$. Let $U_j^n = (U_2)_j^n - (U_1)_j^n$, and $F_1(x,t) = F_1^{(2)}(x,t) - F_1^{(1)}(x,t)$, $F_2(x,t) = F_2^{(2)}(x,t) - F_2^{(1)}(x,t)$. Then it holds

$$E^n \leq \exp(CT)[E^0 + T \max_{0\leq k\leq n-1} G^{k+\frac{1}{2}}], \tag{3.81}$$

where $0 \leq n\tau \leq T$ and

$$\begin{aligned} E^n = {} & h\sum_{j=0}^{m-1} \frac{1}{2}\tilde{c}_9[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}](\nabla_x U_j^n)^2 \\ & + h\sum_{j=m}^{N-1} \frac{1}{2}c_9[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}](\nabla_x U_j^n)^2, \end{aligned} \tag{3.82a}$$

$$G^{k+\frac{1}{2}} = \frac{1}{4\varepsilon}\tilde{c}_9 h\sum_{j=0}^{m}[(g)_j^{k+\frac{1}{2}}]^2 + \frac{1}{4\varepsilon}c_9 h\sum_{j=m+1}^{N}[(g)_j^{k+\frac{1}{2}}]^2. \tag{3.82b}$$

Here, $(g)_j^{k+1/2}$ is a mesh function related only to the source terms. Eq. (81) indicates that the scheme is unconditionally stable (i.e., no restriction on the mesh ratio).

We now analyze the convergence of the present scheme. For simplicity, we first define the inner products and norms for mesh functions as

$$(u^n, v^n) = h\sum_{j=0}^{N} u_j^n v_j^n, \quad ||u^n||^2 = h\sum_{j=0}^{N}(u_j^n)^2, \quad ||u^n||_\infty = \max_{0\leq j\leq N}|u_j^n|. \tag{3.83}$$

Lemma 3.3.2 [115]. For any mesh function $v_j, j = 0, \cdots, N$, and any positive constant ε, it holds

$$||v||_\infty^2 \leq \varepsilon||\nabla_x v||^2 + (\frac{1}{\varepsilon} + \frac{1}{L})||v||^2. \tag{3.84}$$

Theorem 3.3. Assume that $u(x_j, t_n)$ is the analytical solution of Eqs. (3.1)-(3.4) and U_j^n is the numerical solution obtained based on the scheme in Eq. (3.56a)-Eq. (3.56e), respectively. Let $e_j^n = u(x_j, t_n) - U_j^n$. Then it holds

$$||e^n||_\infty \le K(\tau^2 + h^4), \tag{3.85}$$

where K is a constant.

Proof. It can be seen that e_j^n satisfies

$$C_1 \delta_t e_0^{n+\frac{1}{2}} + C_2 \delta_t e_1^{n+\frac{1}{2}} = \frac{1}{h}(k_1)_{1/2} \frac{\bar{e}_1^{n+\frac{1}{2}} - \bar{e}_0^{n+\frac{1}{2}}}{h} + r_0^{n+\frac{1}{2}}; \tag{3.86a}$$

$$\begin{aligned}(C_4)_j \delta_t e_{j-1}^{n+\frac{1}{2}} + (C_5)_j \delta_t e_j^{n+\frac{1}{2}} + (C_6)_j \delta_t e_{j+1}^{n+\frac{1}{2}} &= \frac{1}{h^2}[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}](\bar{e}_{j+1}^{n+\frac{1}{2}} - \bar{e}_j^{n+\frac{1}{2}}) \\ &- \frac{1}{h^2}[(k_1)_{j-\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j-\frac{1}{2}}](\bar{e}_j^{n+\frac{1}{2}} - \bar{e}_{j-1}^{n+\frac{1}{2}}) \\ &+ r_j^{n+\frac{1}{2}}, 1 \le j \le m-1;\end{aligned} \tag{3.86b}$$

$$\begin{aligned}C_7 \delta_t e_{m-1}^{n+\frac{1}{2}} + C_8 \delta_t e_m^{n+\frac{1}{2}} + C_9 \delta_t e_{m+1}^{n+\frac{1}{2}} &= \frac{1}{h^2}[C_0 (k_2)_{m+\frac{1}{2}}(\bar{e}_{m+1}^{n+\frac{1}{2}} - \bar{e}_m^{n+\frac{1}{2}}) \\ &- (k_1)_{m-\frac{1}{2}}(\bar{e}_m^{n+\frac{1}{2}} - \bar{e}_{m-1}^{n+\frac{1}{2}})] \\ &+ r_m^{n+\frac{1}{2}};\end{aligned} \tag{3.86c}$$

$$\begin{aligned}(\tilde{C}_4)_j \delta_t e_{j-1}^{n+\frac{1}{2}} + (\tilde{C}_5)_j \delta_t e_j^{n+\frac{1}{2}} + (\tilde{C}_6)_j \delta_t e_{j+1}^{n+\frac{1}{2}} &= \frac{1}{h^2}[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}](\bar{e}_{j+1}^{n+\frac{1}{2}} - \bar{e}_j^{n+\frac{1}{2}}) \\ &- \frac{1}{h^2}[(k_2)_{j-\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j-\frac{1}{2}}](\bar{e}_j^{n+\frac{1}{2}} - \bar{e}_{j-1}^{n+\frac{1}{2}}) \\ &+ r_j^{n+\frac{1}{2}}, m+1 \le j \le N-1;\end{aligned} \tag{3.86d}$$

$$\tilde{C}_1\delta_t e_N^{n+\frac{1}{2}} + \tilde{C}_2\delta_t e_{N-1}^{n+\frac{1}{2}} = -\frac{1}{h}(k_2)_{N-\frac{1}{2}}\frac{\bar{e}_N^{n+\frac{1}{2}} - \bar{e}_{N-1}^{n+\frac{1}{2}}}{h} + r_N^{n+\frac{1}{2}}, \tag{3.86e}$$

$$e_j^0 = 0, \qquad j = 0, 1, \cdots, N, \tag{3.86f}$$

where $r_j^{n+1/2}$ is $O(\tau^2 + h^3)$ at x_0, x_m, x_N, and is $O(\tau^2 + h^4)$ at x_j, $j = 1, \cdots, m-1$, $m+1, \cdots, N-1$. From now on, for simplicity we denote K as a positive constant without specifying. At different places, it may have a different value.

We multiply Eqs. (3.86a)-(3.86e) by $h\tilde{c}_9\delta_t e_0^{n+1/2}$, $h\tilde{c}_9\delta_t e_j^{n+1/2}$, $h\tilde{c}_9\delta_t e_m^{n+1/2}$, $hc_9\delta_t e_j^{n+1/2}$, and $hc_9\delta_t e_N^{n+1/2}$, respectively, where $\tilde{c}_9$ and c_9 are given in Lemmas 3.2.5. and 3.2.6, and then add the results together. Proceeding in the same way by which we obtained Eq. (3.75) and then replacing n by p and summing up to k, we obtain when h is small enough that

$$\begin{aligned}
& K\sum_{p=0}^{k}||\delta_t e^{p+\frac{1}{2}}||^2 + h\sum_{j=0}^{m-1}\frac{\tilde{c}_9}{2\tau}[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}](\nabla_x e_j^{k+1})^2 \\
& + h\sum_{j=m}^{N-1}\frac{c_9}{2\tau}[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}](\nabla_x e_j^{k+1})^2 \\
\leq \quad & \frac{h^2}{192\varepsilon}\sum_{p=0}^{k}\tilde{c}_9(\left|(D_1)_{\frac{1}{2}}\right|[(\nabla_x e_0^{p+1})^2 + (\nabla_x e_0^{p})^2] + \left|(D_1)_{m-\frac{1}{2}}\right|[(\nabla_x e_{m-1}^{p+1})^2 \\
& + (\nabla_x e_{m-1}^{p})^2]) + \frac{h^2}{192\varepsilon}\sum_{p=0}^{k}c_9(\left|(D_2)_{m+\frac{1}{2}}\right|[\nabla_x(e_m^{p+1})^2 + \nabla_x(e_m^{p+1})^2] + \left|(D_2)_{N-\frac{1}{2}}\right| \\
& [\nabla_x(e_{N-1}^{p+1})^2 + \nabla_x(e_{N-1}^{p})^2]) + R,
\end{aligned} \tag{3.87a}$$

where

$$R = \tilde{c}_9 h e_0^{k+1} \frac{r_0^{k+\frac{1}{2}}}{\tau} - \sum_{p=1}^{k} \tilde{c}_9 h e_0^p \frac{r_0^{p+\frac{1}{2}} - r_0^{p-\frac{1}{2}}}{\tau} - \tilde{c}_9 h e_0^0 \frac{r_0^{\frac{1}{2}}}{\tau} + \tilde{c}_9 h e_m^{k+1} \frac{r_m^{k+\frac{1}{2}}}{\tau}$$

$$- \sum_{p=1}^{k} \tilde{c}_9 h e_m^p \frac{r_m^{p+\frac{1}{2}} - r_m^{p-\frac{1}{2}}}{\tau} - \tilde{c}_9 h e_m^0 \frac{r_m^{\frac{1}{2}}}{\tau} + c_9 h e_N^{k+1} \frac{r_N^{k+\frac{1}{2}}}{\tau}$$

$$- \sum_{p=1}^{k} c_9 h e_N^p \frac{r_m^{p+\frac{1}{2}} - r_m^{p-\frac{1}{2}}}{\tau} - c_9 h e_N^0 \frac{r_N^{\frac{1}{2}}}{\tau} + \sum_{p=0}^{k} \frac{1}{4\varepsilon} \tilde{c}_9 h \sum_{j=1}^{m-1} (r_j^{p+\frac{1}{2}})^2$$

$$+ \sum_{p=0}^{k} \frac{1}{4\varepsilon} c_9 h \sum_{j=m+1}^{N-1} (r_j^{p+\frac{1}{2}})^2 \tag{3.87b}$$

Here, we have used the fact that

$$\sum_{p=0}^{k} (\delta_t e_0^{p+\frac{1}{2}}) r_0^{p+\frac{1}{2}} = e_0^{k+1} \frac{r_0^{k+\frac{1}{2}}}{\tau} - \sum_{p=1}^{k} e_0^p \frac{r_0^{p+\frac{1}{2}} - r_0^{p-\frac{1}{2}}}{\tau} - e_0^0 \frac{r_0^{\frac{1}{2}}}{\tau}. \tag{3.88}$$

Using a similar procedure as before, one may obtain

$$\begin{aligned}
& h \sum_{j=0}^{m-1} \frac{1}{2} \tilde{c}_9 [(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24} (D_1)_{j+\frac{1}{2}}] (\nabla_x e_j^{k+1})^2 \\
& + h \sum_{j=m}^{N-1} \frac{1}{2} c_9 [(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24} (D_2)_{j+\frac{1}{2}}] (\nabla_x e_j^{k+1})^2 \\
& + \tau K \sum_{p=0}^{k} ||\delta_t e^{p+\frac{1}{2}}||^2 - \frac{\tilde{c}_9 \tau h^2}{192\varepsilon} \left[\left| (D_1)_{\frac{1}{2}} \right| (\nabla_x e_0^{k+1})^2 - \left| (D_1)_{m-\frac{1}{2}} \right| (\nabla_x e_{m-1}^{p+1})^2 \right] \\
& - \frac{c_9 \tau h^2}{192\varepsilon} \left[\left| (D_2)_{m+\frac{1}{2}} \right| (\nabla_x e_m^{k+1})^2 - \left| (D_2)_{N-\frac{1}{2}} \right| (\nabla_x e_{N-1}^{k+1})^2 \right] \\
\leq \quad & \frac{2\tau h^2}{192\varepsilon} \sum_{p=0}^{k} [\tilde{c}_9 \left| (D_1)_{\frac{1}{2}} \right| (\nabla_x e_0^p)^2 + \tilde{c}_9 \left| (D_1)_{m-\frac{1}{2}} \right| (\nabla_x e_{m-1}^p)^2 + c_9 \left| (D_2)_{m+\frac{1}{2}} \right| (\nabla_x e_m^p)^2 \\
& + c_9 \left| (D_2)_{N-\frac{1}{2}} \right| (\nabla_x e_{N-1}^p)^2] + \tilde{R},
\end{aligned} \tag{3.89a}$$

where

$$\tilde{R} = \tilde{c}_9 h[e_0^{k+1} r_0^{k+\frac{1}{2}} - \sum_{p=1}^{k} e_0^p (r_0^{p+\frac{1}{2}} - r_0^{p-\frac{1}{2}})] + \tilde{c}_9 h[e_m^{k+1} r_m^{k+\frac{1}{2}} - \sum_{p=1}^{k} e_m^p (r_m^{p+\frac{1}{2}} - r_m^{p-\frac{1}{2}})]$$

$$+ c_9 h[e_N^{k+1} r_N^{k+\frac{1}{2}} - \sum_{p=1}^{k} e_N^p (r_N^{p+\frac{1}{2}} - r_N^{p-\frac{1}{2}})] + \sum_{p=0}^{k} \frac{1}{4\varepsilon} \tilde{c}_9 \tau h \sum_{j=1}^{m-1} (r_j^{p+\frac{1}{2}})^2$$

$$+ \sum_{p=0}^{k} \frac{1}{4\varepsilon} c_9 \tau h \sum_{j=m+1}^{N-1} (r_j^{p+\frac{1}{2}})^2 \tag{3.89b}$$

We choose h small enough such that

$$(k_1)_{j+1/2} - h^2 (D_1)_{j+1/2}/24 \geq K, \tag{3.90a}$$

$$-h\tilde{c}_9 \left|(D_1)_{\frac{1}{2}}\right| /(192\varepsilon) \geq -K \tag{3.90b}$$

and, as well as for other terms. As such, we obtain

$$h \sum_{j=0}^{m-1} \frac{1}{2} \tilde{c}_9 [(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24} (D_1)_{j+\frac{1}{2}}] (\nabla_x e_j^{k+1})^2 + h \sum_{j=m}^{N-1} \frac{1}{2} c_9 [(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24} (D_2)_{j+\frac{1}{2}}]$$

$$(\nabla_x e_j^{k+1})^2 - \frac{\tau h^2}{192\varepsilon} [\tilde{c}_9 \left|(D_1)_{\frac{1}{2}}\right| (\nabla_x e_0^{k+1})^2 + \tilde{c}_9 \left|(D_1)_{m-\frac{1}{2}}\right| (\nabla_x e_{m-1}^{k+1})^2$$

$$+ c_9 \left|(D_2)_{m+\frac{1}{2}}\right| (\nabla_x e_m^{k+1})^2 + c_9 \left|(D_2)_{N-\frac{1}{2}}\right| (\nabla_x e_{N-1}^{k+1})^2]$$

$$\geq \quad K(1-\tau) h \sum_{j=0}^{N-1} (\nabla_x e_j^{k+1})^2. \tag{3.91}$$

Letting

$$E^k = \frac{\tilde{c}_9 h^2}{192\varepsilon} \sum_{p=0}^{k} \left[\left|(D_1)_{\frac{1}{2}}\right| (\nabla_x e_0^p)^2 + \left|(D_1)_{m-\frac{1}{2}}\right| (\nabla_x e_{m-1}^p)^2 \right]$$

$$+ \frac{c_9 h^2}{192\varepsilon} \sum_{p=0}^{k} \left[\left|(D_2)_{m+\frac{1}{2}}\right| (\nabla_x e_m^p)^2 + \left|(D_2)_{N-\frac{1}{2}}\right| (\nabla_x e_{N-1}^p)^2 \right], \tag{3.92}$$

and using Eq. (3.91) and the values of $r_j^{k+1/2}$, Eq. (3.89) can be simplified to

$$K(1-\tau)h\sum_{j=0}^{N-1}(\nabla_x e_j^{k+1})^2 + K\tau\sum_{p=0}^{k}||\delta_t e^{p+\frac{1}{2}}||^2$$

$$\leq \quad K\tau E^k + Kh\left|e_0^{k+1}\right|(\tau^2+h^3) + \sum_{p=1}^{k}K\tau h\left|e_0^{p}\right|(\tau^2+h^3) + Kh\left|e_m^{k+1}\right|(\tau^2+h^3)$$

$$+\sum_{p=1}^{k}K\tau h\left|e_m^{p}\right|(\tau^2+h^3) + Kh\left|e_N^{k+1}\right|(\tau^2+h^3) + \sum_{p=1}^{k}K\tau h\left|e_N^{p}\right|(\tau^2+h^3)$$

$$+K(\tau^2+h^4)^2. \tag{3.93}$$

As such, we obtain

$$K(1-\tau)h\sum_{j=0}^{N}(\nabla_x e_j^{k+1})^2 + K\tau\sum_{p=0}^{k}||\delta_t e^{p+\frac{1}{2}}||^2$$

$$\leq \quad K\tau E^k + K(\tau^2+h^4)||e^{k+1}||_\infty + K(\tau^2+h^4)\tau\sum_{p=0}^{k}||e^p||_\infty + K(\tau^2+h^4)^2. \tag{3.94}$$

Since $e_i^0 = 0$ and hence

$$e_i^{k+1} = \tau\sum_{p=0}^{k}\delta_t e_i^{p+\frac{1}{2}}, \tag{3.95}$$

we obtain

$$(e_i^{k+1})^2 = (\tau\sum_{p=0}^{k}\delta_t e_i^{p+\frac{1}{2}})^2 \leq T\tau\sum_{p=0}^{k}(\delta_t e_i^{p+\frac{1}{2}})^2. \tag{3.96}$$

Multiplying Eq. (3.96) by h and summing up for i, we have

$$\frac{1}{T}||e^{k+1}||^2 \leq \tau\sum_{p=0}^{k}||\delta_t e^{p+\frac{1}{2}}||^2, \tag{3.97}$$

implying that

$$K(1-\tau)||\nabla_x e^{k+1}||^2 + \frac{K}{T}||e^{k+1}||^2$$

$$\leq \quad K\tau E^k + K(\tau^2+h^4)||e^{k+1}||_\infty + K(\tau^2+h^4)\tau\sum_{p=0}^{k}||e^p||_\infty + K(\tau^2+h^4)^2. \quad (3.98)$$

By Lemma 3.3.2 and the Cauchy-Schwarz inequality, we obtain

$$K(1-\tau)||e^{k+1}||_\infty^2 \leq K\tau E^k + K(\tau^2+h^4)||e^{k+1}||_\infty + K(\tau^2+h^4)\tau\sum_{p=0}^{k}||e^p||_\infty$$

$$+ K(\tau^2+h^4)^2$$

$$\leq K\tau E^k + \frac{K}{2}||e^{k+1}||_\infty^2 + \frac{K}{2}(\tau^2+h^4)^2 + \frac{K}{2}\tau\sum_{p=0}^{k}||e^p||_\infty^2$$

$$+ \frac{K}{2}(\tau^2+h^4)^2 + K(\tau^2+h^4)^2. \quad (3.99)$$

Note that

$$h^2(\nabla_x e_j^p)^2 = h^2[\frac{e_{j+1}^p - e_j^p}{h}]^2$$

$$\leq 2[(e_{j+1}^p)^2 + (e_j^p)^2]$$

$$\leq 4||e^p||_\infty^2, \quad (3.100)$$

implying that $E^k \leq K\sum_{p=0}^{k}||e^p||_\infty^2$. Therefore, if τ is small enough such that $1-2K\tau \geq 1/2$, then we obtain from Eq. (3.99) that

$$||e^{k+1}||_\infty^2 \leq K\tau\sum_{p=0}^{k}||e^p||_\infty^2 + K(\tau^2+h^4)^2. \quad (3.101)$$

Thus, by the Gronwall inequality, we obtain

$$||e^{k+1}||_\infty^2 \leq K(\tau^2+h^4)^2[1+KT\exp(KT)], \quad (3.102)$$

and hence complete the proof.

It should be pointed out that Sun and Dai [115] obtained a fourth-order compact finite difference scheme for the heat conduction equation with constant coefficients in double layers. Although their scheme is unconditionally stable, there are some restrictions on coefficients (see Eqs. (4.1)-(4.3) in [62]). The present scheme does not have such restrictions on coefficients even if the coefficients are constants.

3.4 Numerical Example

To verify the accuracy of the present scheme in Eq. (3.56), we first consider a simple heat conduction problem as

$$u_t = \frac{\partial}{\partial x}\left(2(x^2+x+1)u_x\right) + F_1(x,t), \qquad 0 \le x \le 1, t > 0, \tag{3.103a}$$

$$u_t = \frac{\partial}{\partial x}\left((x^2+x+1)u_x\right) + F_2(x,t), \qquad 1 \le x \le 2, t > 0, \tag{3.103b}$$

where the source terms are

$$F_1(x,t) = [8\pi^2(x^2+x+1)-1]e^{-t}sin(2\pi x) - 4\pi(2x+1)e^{-t}cos(2\pi x), \tag{3.103c}$$

$$F_2(x,t) = [16\pi^2(x^2+x+1)-1]e^{-t}sin(4\pi x) - 4\pi(2x+1)e^{-t}cos(4\pi x), \tag{3.103d}$$

the initial and boundary conditions are given as

$$u(x,0) = sin(2\pi x), \;\; 0 \le x \le 1; \;\; u(x,0) = sin(4\pi x), \;\; 1 \le x \le 2; \tag{3.103e}$$

$$u_x(0,t) = 2\pi, \;\; u_x(2,t) = 4\pi, \;\; t \ge 0, \tag{3.103f}$$

and the interfacial condition at $x = 1$ is

$$u(1_-,t) = u(1_+,t), \; 6u_x(1_-,t) = 3u_x(1_+,t), \; t \ge 0. \tag{3.103g}$$

It can be seen that the analytical solution for the above problem is $u(x,t) = e^{-t}sin(2\pi x)$ when $0 \leq x \leq 1$, and $u(x,t) = e^{t}sin(4\pi x)$ when $1 \leq x \leq 2$.

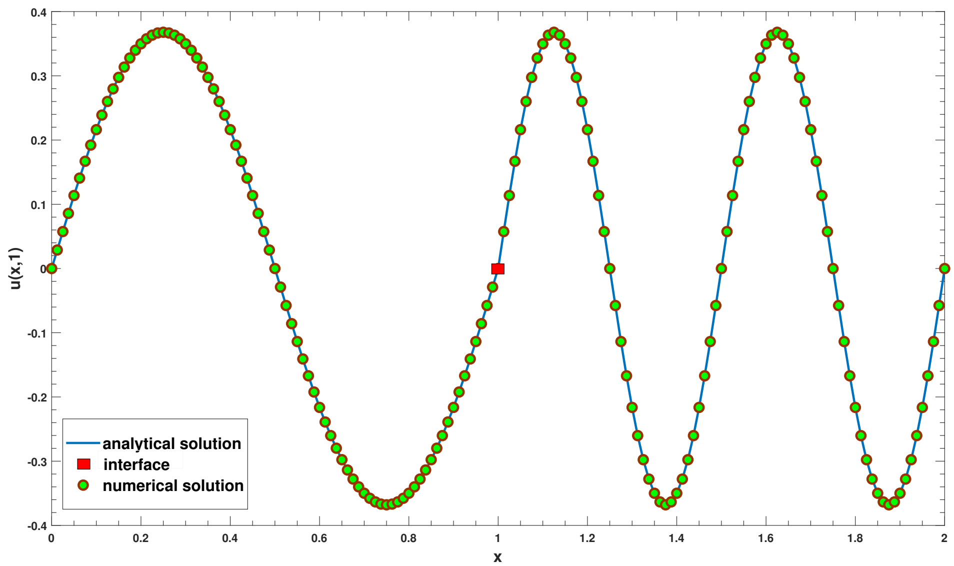


Figure 3.2: Solution profile at $t = 1$ along the x-direction.

Table 3.1: Maximum error and convergence order in space when $\tau = 10^{-6}$ and $0 \leq t \leq 1$.

h	$E(\tau,h)$	Convergence order
1/10	$2.96772209 \times 10^{-1}$	-
1/20	$1.83704916 \times 10^{-2}$	4.00
1/40	$1.14868809 \times 10^{-3}$	3.99
1/80	$7.18955235 \times 10^{-5}$	3.99

In our computation, we calculated the maximum error $E(\tau,h) = \max_{j,n} |u(x_j,t_n) -U_j^n|$ and the convergence orders in space and in time are based on $q = log(E(\tau,h_2)/E(\tau,h_1)) /log(h_2/h_1)$ and $q = log(E(\tau_2,h)/ E(\tau_1,h))/log(\tau_2/\tau_1)$, respectively. In particular, we computed the convergence order in space by choosing $h = 1/10$,$1/20$,$1/40$,$1/80$ and $\tau = 10^{-6}$, and on the other hand, we computed the convergence order in time by choosing $\tau = 1/10, 1/20, 1/40$, $1/80$ and $h = 10^{-5}$. Results are listed in Table 3.1 and

Table 3.2: Maximum error and convergence order in time when $h = 10^{-5}$ and $0 \leq t \leq 1$.

h	$E(\tau, h)$	Convergence order
$1/10$	$2.54812728 \times 10^{-3}$	-
$1/20$	$6.38902058 \times 10^{-4}$	1.99
$1/40$	$1.53023681 \times 10^{-4}$	2.06
$1/80$	$3.47321759 \times 10^{-5}$	2.13

3.2. From these two tables, one may see that the convergence orders in space and in time are about 4.0 and 2.0, respectively, which coincide with the theoretical analysis. Figure 3.2 shows the profile of the solution at $t = 1$ obtained based on the present scheme using $h = 1/80$ and $\tau = 10^{-6}$.

3.5 Summary

In this chapter, we have derived some important lemmas for developing a compact finite difference scheme for solving the heat conduction equation with variable coefficients. Based on these lemmas, we have derived a higher order finite difference method for solving the heat conduction with variable coefficients in double layers. The scheme has fourth-order accuracy in space and second-order accuracy in time. The stability and convergence of the scheme have been analyzed and proved using the discrete energy method. Finally, the scheme has been tested in an example to verify the accuracy and the convergence order. This work has already been published in the journal *Applied Mathematics and Computation*, volume 386, 2020 [172]. The doi for the article is https://doi.org/10.1016/j.amc.2020.125516.

CHAPTER 4

ARTIFICIAL NEURAL METHOD FOR DOUBLE-LAYERED STRUCTURES

4.1 Parabolic Two-Temperature Heat Conduction Equation in Double-Layered Structure

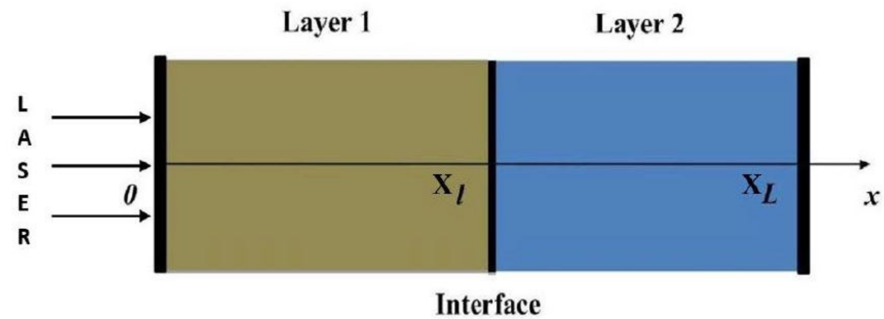


Figure 4.1: Schematic diagram for a double-layered film, where $0 \leq x \leq x_l$ represents the first layer and $x_l \leq x \leq x_L$ represents the second layer.

In this chapter, we consider a double-layered thin film exposed to an ultrashort-pulsed laser heating, where the governing equations are the parabolic two-step model as shown in 4.1:

$$C_e^{(m)}(T_e^{(m)})\frac{\partial T_e^{(m)}}{\partial t} = \frac{\partial}{\partial x}\left(k_e^{(m)}\frac{\partial T_e^{(m)}}{\partial x}\right) - G^{(m)}(T_e^{(m)} - T_l^{(m)}) + S^{(m)}(x,t), \quad (4.1a)$$

$$C_l^{(m)}\frac{\partial T_l^{(m)}}{\partial t} = G^{(m)}(T_e^{(m)} - T_l^{(m)}), \quad (4.1b)$$

where the heat source is given as

$$S^{(m)}(x,t) = 0.94\frac{1-R}{t_p\delta}J\exp\left(-\frac{x_m}{\delta} - 2.77(\frac{t-2t_p}{t_p})^2\right), \quad (4.2)$$

with the initial condition

$$T_e^{(m)}(x,0) = T_l^{(m)}(x,0) = T_0; \quad \frac{\partial T_e^{(m)}(x,0)}{\partial t} = \frac{\partial T_l^{(m)}(x,0)}{\partial t} = 0, \tag{4.3a}$$

the thermal insulated boundary condition

$$\frac{\partial T_e^{(m)}(0,t)}{\partial x} = \frac{\partial T_l^{(m)}(0,t)}{\partial x} = 0, \quad \frac{\partial T_e^{(m)}(x_L,t)}{\partial x} = \frac{\partial T_l^{(m)}(x_L,t)}{\partial x} = 0, \tag{4.3b}$$

and the perfectly thermal-contact interfacial condition

$$T_e^{(1)}(x_l,t) = T_e^{(2)}(x_l,t); \quad k_e^{(1)}\frac{\partial T_e^{(1)}}{\partial x} = k_e^{(2)}\frac{\partial T_e^{(2)}}{\partial x}. \tag{4.3c}$$

Here, $m = 1, 2$ represent the first layer ($0 \le x \le x_l$) and the second layer ($x_l \le x \le x_L$), respectively, and $0 \le t \le t_T$, T_0 is the initial temperature, $C_e^{(m)}(T_e^{(m)}) = C_{e0}^{(m)} T_e^{(m)}/T_0$, and $k_e^{(m)} = k_{e0}^{(m)} T_e^{(m)}/T_l^{(m)}$ where $C_{e0}^{(m)}$ and $k_{e0}^{(m)}$ are constant heat capacity and conductivity.

For the heat source, $R = 0.93$ is the reflectivity, $t_p = 0.1$ (ps) is the full-width-at-half-maximum duration of the laser pulse, $\delta = 15.3$ (nm) is the radiation penetration depth, $J = 500$ (Jm^2) is laser fluency.

The above parabolic two-temperature model and its extension to hyperbolic two-temperature models coupled with the insulated boundary condition, where $\partial T_e/\partial \vec{n} = \partial T_l/\partial \vec{n} = 0$ and $\vec{n}$ is the outward normal vector on the boundary, have been widely used in thermal analysis for micro/nanoscale heat conduction in thin films induced by the ultrashort-pulsed laser heating. We refer readers to references in the literature such as [172-189] and to Tzou[191] as well as to the references therein for details. In this study, we would like to develop a neural network method for solving the above model and predict the lattice and electron temperatures.

4.2 Neural Network Method

In order for the neural nets to learn better, we make the above equations in dimensionless form by introducing the following variables:

$$T_e^{*(m)} = \frac{T_e^{(m)} - T_0}{T_0}, \;\; T_l^{*(m)} = \frac{T_l^{(m)} - T_0}{T_0}, \tag{4.4a}$$

$$x^* = \frac{x}{x_L}, \quad \delta^* = \frac{\delta}{x_L}, \quad t^* = \frac{t}{t_T}, \quad t_p^* = \frac{t_p}{t_T}. \tag{4.4b}$$

Replacing $T_e^{(m)}$, $T_l^{(m)}$, x, t and t_p in Eqs. (4.1)-(4.3) based on Eq. (4.4), we obtain

$$C_e^{*(m)} \frac{\partial T_e^{*(m)}}{\partial t^*} = \frac{\partial}{\partial x^*}\left(k_e^{*(m)} \frac{\partial T_e^{*(m)}}{\partial x^*} \right) - G^{*(m)}(T_e^{*(m)} - T_l^{*(m)}) + S^{*(m)}(x_m^*, t^*), \tag{4.5a}$$

$$C_l^{(m)} \frac{\partial T_l^{*(m)}}{\partial t^*} = G^{*(m)}(T_e^{*(m)} - T_l^{*(m)}), \tag{4.5b}$$

where the heat source is given as

$$S^{*(m)}(x_m^*, t^*) = 0.94 \frac{1-R}{t_p^* \delta^*} J^* \exp\left(-\frac{x_m^*}{\delta^*} - 2.77 (\frac{t^* - 2t_p^*}{t_p^*})^2 \right), \tag{4.6}$$

with the initial condition

$$T_e^{*(m)}(x_m^*, 0) = T_l^{*(m)}(x_m^*, 0) = 0; \;\; \frac{\partial T_e^{*(m)}(x_m^*, 0)}{\partial t^*} = \frac{\partial T_l^{*(m)}(x_m^*, 0)}{\partial t^*} = 0, \tag{4.7}$$

the thermal insulated boundary condition

$$\frac{\partial T_e^{*(m)}(0, t^*)}{\partial x^*} = \frac{\partial T_l^{*(m)}(0, t^*)}{\partial x^*} = 0, \quad \frac{\partial T_e^{*(m)}(1, t^*)}{\partial x^*} = \frac{\partial T_l^{*(m)}(1, t^*)}{\partial x^*} = 0, \tag{4.8a}$$

and the perfectly thermal-contact interfacial condition

$$T_e^{*(1)}(x_l^*, t^*) = T_e^{*(2)}(x_l^*, t^*), \;\; k_e^{*(1)} \frac{\partial T_e^{*(1)}}{\partial x^*} = k_e^{*(2)} \frac{\partial T_e^{*(2)}}{\partial x^*}. \tag{4.8b}$$

Here, $C_e^{*(m)} = C_{e0}^{(m)}(T_e^{*(m)} + 1)$, $k_e^{*(m)} = t_T k_{e0}^{(m)}(T_e^{*(m)} + 1)/(x_L^2(T_l^{*(m)} + 1))$, $G^{*(m)} = t_T G^{(m)}$, $J^* = J/(x_L T_0)$, $0 < x^* = x_l/x_L < 1$. For simplicity, we omit asterisks in the following texts.

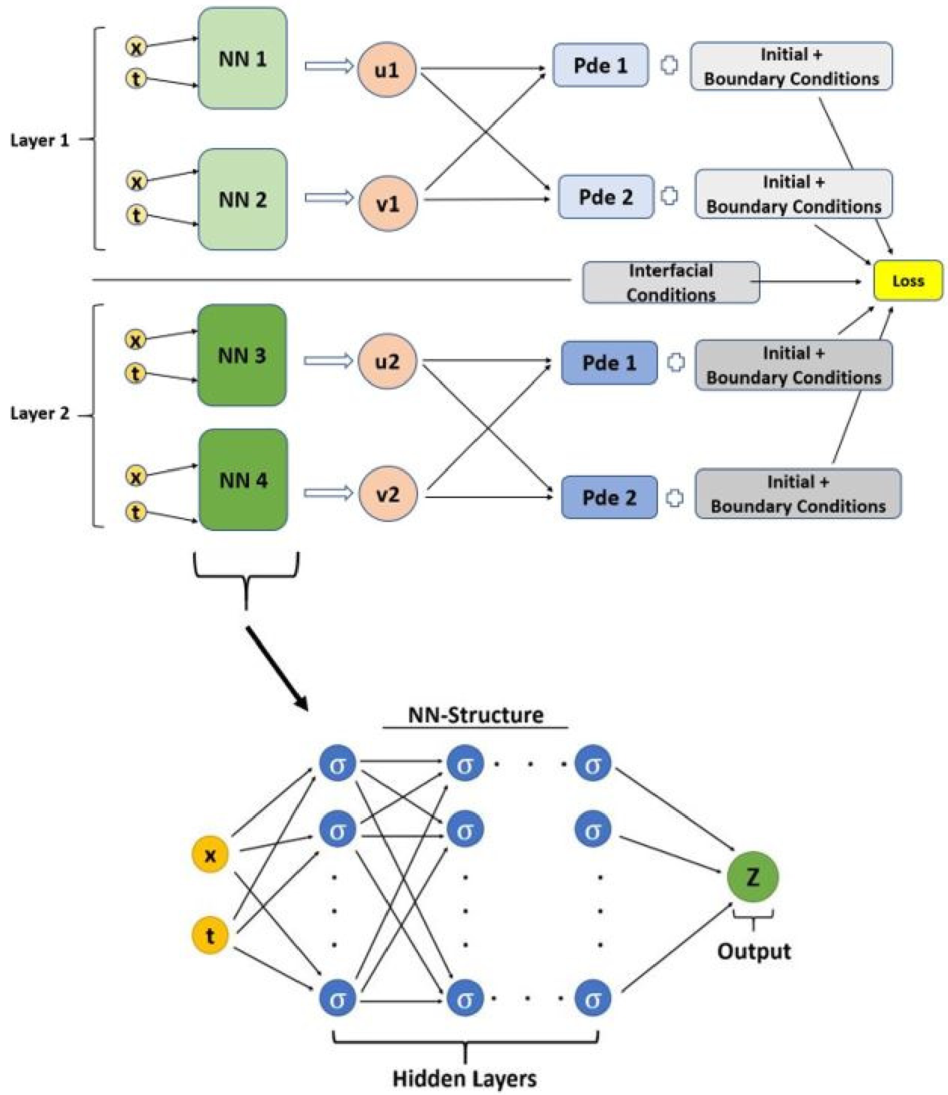


Figure 4.2: ANN schematic for solving the parabolic two-temperature model.

Since there are four unknowns (i.e.,$T_e^{(1)}$ $T_e^{(2)}, T_l^{(1)}, T_2^{(2)}$), we first design four individual neural nets and make all the neural nets fully connected, as shown in Figure 4.2. Here, u_1, v_1 represent the electron and lattice temperatures in the first layer, and u_2, v_2 represent the electron and lattice temperatures in the second layer, respectively. We assume that neural nets NN1, NN2, NN3, and NN4 have L_1, L_2, L_3 and L_4 number of hidden layers, and M_1, M_2, M_3 and M_4 number of hidden units, respectively. Let the input be $X = (\mathbf{x}, \mathbf{t})$. The output from NN1, which is the ANN solution u_1 can be expressed as

$$u_1(\mathbf{x}, t) = \sum_{i=1}^{M_1} W_i^{(1)} z_i^{(L_1)} + b^{(1)}, \tag{4.9a}$$

$$z_i^{(j)} = \sigma(\sum_{k=1}^{M_1} W_{k,i}^{(1,j)} z_k^{(j-1)} + b_i^{(1,j)}), \qquad j = 2, ..., L_1; \quad i = 1, 2, ..., M_1; \tag{4.9b}$$

$$z_i^{(1)} = \sigma(W_i^{(1,0)} x + W_i^{(1,1)} t + b_i^{(1,0)}), \qquad i = 1, 2, ..., M_1; \tag{4.9c}$$

where σ is the activation function which is the hyperbolic tangent function (i.e., $\sigma(y) = (e^y - e^{-y})/(e^y + e^{-y})$). Here, $W_i^{(1,0)}$, $W_i^{(1,1)}$ $W_{k,i}^{(1,j)}$, $W_i^{(1)}$, $b_i^{(1,0)}, b_i^{(1,j)}, b^{(1)}$ are weights and biases which are to be optimized. For simplicity, we list all the weights and biases into a vector and denote $\boldsymbol{\theta}^{(1)} = [W_1^{(1,0)}, \cdots, W_{M_1}^{(1,0)}, W_1^{(1,1)}, \cdots, W_{M_1}^{(1,1)}, \cdots, W_1^{(1)}, \cdots, W_{M_1}^{(1)}, \cdots, b_1^{(1,0)}, \cdots, b_{M_1}^{(1,0)}, \cdots, b^{(1)}]$. For the neural net NN2, NN3 and NN4, we have similar expressions for v_1, u_2, v_2 as the above u_1, and vectors $\boldsymbol{\theta}^{(2)}$, $\boldsymbol{\theta}^{(3)}$ and $\boldsymbol{\theta}^{(4)}$ as set of weights and biases for NN2, NN3 and NN4, respectively. Let $\boldsymbol{\theta} = [\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \boldsymbol{\theta}^{(3)}, \boldsymbol{\theta}^{(4)}]$. We define the loss function as

$$\begin{aligned} J_{Loss}(\boldsymbol{\theta}) = & Loss_{PDE1}^{(1)} + Loss_{PDE1}^{(2)} + Loss_{PDE2}^{(1)} + Loss_{PDE2}^{(2)} \\ & + Loss_{IC}^{(1)} + Loss_{IC}^{(2)} + Loss_{BC}^{(1)} + Loss_{BC}^{(2)} + Loss_{IF}, \end{aligned} \tag{4.10}$$

where $Loss_{PDE1}^{(m)}$, $Loss_{PDE2}^{(m)}$, $Loss_{IC}^{(m)}$, $Loss_{BC}^{(m)}$, and $Loss_{IF}$ are loss functions calculated based on PDEs in Eqs. (4.8a)-(4.8b), the initial condition in Eq. (4.7), the boundary condition in Eq. (4.8a), and the interface condition in Eq. (4.8b), respectively. These loss functions are expressed in l_2-norm as

$$Loss_{IC}^{(m)} = \frac{1}{N_{IC}^{(m)}} \sum_{i=1}^{N_{IC}^{(m)}} |u_m(x_i, 0) - T_e(x_i, 0)|^2, \tag{4.11a}$$

$$\begin{aligned} Loss_{BC}^{(m)} = & \frac{1}{N_{BC}^{(1,m)}} [\sum_{i=1}^{N_{BC}^{(1,m)}} |u_m(0, t_i) - T_e^{(m)}(0, t_i)|^2 + \sum_{i=1}^{N_{BC}^{(1,m)}} |v_m(0, t_i) - T_l^{(m)}(0, t_i)|^2] \\ & + \frac{1}{N_{BC}^{(2,m)}} [\sum_{i=1}^{N_{BC}^{(2,m)}} |u_m(1, t_i) - T_e^{(m)}(1, t_i)|^2 \\ & + \sum_{i=1}^{N_{BC}^{(2,m)}} |v_m(1, t_i) - T_l^{(m)}(1, t_i)|^2], \end{aligned} \tag{4.11b}$$

$$\begin{aligned} Loss_{IF} = & \frac{1}{N_{IF}} \sum_{i=1}^{N_{IF}} |u_1(l, t_i) - u_2(l, t_i)|^2 \\ & + \frac{1}{N_{IF}} \sum_{i=1}^{N_{IF}} |k_e^{(1)} \frac{\partial u_1(l, t_i)}{\partial x} - k_e^{(2)} \frac{\partial u_2(l, t_i)}{\partial x}|^2, \end{aligned} \tag{4.11c}$$

$$\begin{aligned} Loss_{PDE1}^{(m)} = & \frac{1}{N_{PDE}^{(m)}} \sum_{i=1}^{N_{PDE}^{(m)}} |C_e^{(m)} (u_m)_t(x_i, t_i) - (k_e^{(m)} (u_m)_x)_x(x_i, t_i) \\ & + G[u_m(x_i, t_i) - v_m(x_i, t_i)] - S^{(m)}(x_i, t_i)|^2, \end{aligned} \tag{4.11d}$$

$$Loss_{PDE2}^{(m)} = \frac{1}{N_{PDE}^{(m)}} \sum_{i=1}^{N_{PDE}^{(m)}} |C_l^{(m)} (v_m)_t(x_i, t_i) - G[u_m(x_i, t_i) - v_m(x_i, t_i)]|^2. \tag{4.11e}$$

Here, $m = 1, 2$ represents the first and second layers, respectively. $N_{IC}^{(m)}$, $N_{BC}^{(1,m)}$,$N_{BC}^{(1,m)}$, N_{IF} and $N_{PDE}^{(m)}$ are numbers of training points selected from the initial condition, boundary condition, interface condition and the interior domain.

To obtain the optimal $\boldsymbol{\theta}$ from $J_{Loss}(\boldsymbol{\theta})$, we use a combination of the Adam Optimization method [168] and the L-BFGS method [169] for optimizing the weights and biases. The Adam method is a gradient based optimization technique that uses for estimates of lower order moments while the L-BFGS is a quasi-Newton method that uses approximate Hessian matrix. We use the Adam Optimization method because of its efficiency in computation and fewer memory requirements and because it is invariant to diagonal re-scaling of the gradients and is well suited for large number of data points. On the other hand, the L-BFGS method is more stable than other optimization algorithms and can handle large batch sizes very well. And also since the L-BFGS uses curvature information from only the most recent iterations, it is more practical and saves time as well as storage space. The L-BFGS has a tendency to be attracted to saddle points, whereas the Adam method avoids saddle points. Therefore, we believe that a combination of both methods could be better. In this study, we first apply the Adam optimization method followed by the L-BFGS method to achieve better convergence for obtaining the optimal $\boldsymbol{\theta}$.

Note that the Adam Optimization method involves the hyper-parameters α, β_1, and β_2, where α is the learning rate, and β_1, β_2 are the exponential decay rates for the moment estimates. To start the Adam method, we initialize the first and second moment vector represented by $\mathbf{m}_0$ and $\mathbf{v}_0$ to $\mathbf{0}$. Let $\boldsymbol{\theta}_0$ be the starting initial weights and biases of the neural net. And let $(\theta_j^*)_0$ represent the individual elements of $\boldsymbol{\theta}_0$. We compute the gradient of $J_{Loss}(\boldsymbol{\theta})$ with respect to each component in $\boldsymbol{\theta}$ and store it as

$$(g_j)_1 = \nabla_{(\theta_j^*)_0} J_{Loss}((\theta_j^*)_0). \tag{4.12a}$$

We then update the first and second moment vectors as

$$(m_j)_1 = \beta_1 (m_j)_0 + (1-\beta_1)(g_j)_1, \qquad (v_j)_1 = \beta_2 (v_j)_0 + (1-\beta_2)(g_j)_1^2, \tag{4.12b}$$

After computing the bias corrected first and 2nd moments

$$(\bar{m}_j)_1 = \frac{(m_j)_1}{1-\beta_1^1}, \qquad (\bar{v}_j)_1 = \frac{(v_j)_1}{1-\beta_2^1}, \tag{4.12c}$$

and hence update $\boldsymbol{\theta}$ as

$$(\theta_j^*)_1 = (\theta_j^*)_0 - \alpha \cdot \frac{(\bar{m}_j)_1}{\sqrt{(\bar{v}_j)_1} + \epsilon}. \tag{4.12d}$$

The process continues till the maximum number N_1 of iterations is completed and the Adam optimization $\boldsymbol{\theta}$ is saved as $\boldsymbol{\theta}_{N_1}$.

We then start the L-BFGS method with $\bar{\boldsymbol{\theta}}_0 = \boldsymbol{\theta}_{N_1}$as

$$(\bar{\theta}_j^*)_{i+1} = (\bar{\theta}_j^*)_i - \alpha_i \mathbf{B}_i \nabla_{\theta_j^*} J_{Loss}(\bar{\boldsymbol{\theta}}_i), \qquad i = 0, 1, ..., N_2 - 1, \tag{4.13a}$$

where N_2 is the maximum number of iterations, α_i is the step length. Here, $\mathbf{B}_i$ represents the inverse Hessian approximation, which is given by

$$\mathbf{B}_i = \mathbf{V}_{i-1}^T \mathbf{B}_{i-1} \mathbf{V}_{i-1} + \boldsymbol{\rho}_{i-1} \mathbf{s}_{i-1} \mathbf{s}_{i-1}^T, \tag{4.13b}$$

where

$$\mathbf{s}_i = \bar{\boldsymbol{\theta}}_{i+1} - \bar{\boldsymbol{\theta}}_i,\ \mathbf{r}_i = \nabla_{\boldsymbol{\theta}} J_{Loss}(\bar{\boldsymbol{\theta}}_{i+1}) - \nabla_{\boldsymbol{\theta}} J_{Loss}(\bar{\boldsymbol{\theta}}_i), \boldsymbol{\rho}_i = \frac{1}{\mathbf{r}_i^T \mathbf{s}_i}, \tag{4.13c}$$

$$\mathbf{V}_i = \mathbf{I} - \boldsymbol{\rho}_i \mathbf{r}_i \mathbf{s}_i^T, \mathbf{B}_0 = \mathbf{I}, \text{ or } \mathbf{B}_0 = \frac{\mathbf{s}_0^T \mathbf{r}_0}{\mathbf{r}_0^T \mathbf{r}_0} \mathbf{I}. \tag{4.13d}$$

Note that we store only a limited number of B_i implicitly by storing say (w) pairs of $(\mathbf{s}_i,\mathbf{r}_i)$ used in the above equations.

It should be pointed out that because the pulse duration is very short, this requires the random learning points to be able to catch the pulse in order to obtain the accurate solution. In this study, we divide the whole time interval into several sub-intervals and then run the algorithm for the first sub-interval to obtain the neural network solutions. Once they are done, we use the solutions at the time end of the sub-interval as the initial condition for the next sub-interval and run the algorithm until all the solutions are obtained. In this way, we can obtain a more accurate as well as faster solution. The main idea is to select a high resolution of training points across time for the first interval. This ensures the proper catching of the pulse. The next time intervals can be of lower resolution, and by using the already captured result as an initial condition we can propagate on the next sub-interval rather quickly, utilizing fewer points for training. This technique increases the speed as well as accuracy of the neural network solution.

4.3 Algorithm

Algorithm: ANN for solving the parabolic two-temperature model

Step 1. Initialize weights and biases in $\boldsymbol{\theta}_0$; set maximum iterations $N_{max} = N_1 + N_2$; select training points $X = (\mathbf{x}, \mathbf{t})$ for input; calculate $J_{Loss}(\boldsymbol{\theta}_0)$

Step 2. Compare with tolerance ε (default machine floating-point precision) or threshold **if** $J_{Loss}(\boldsymbol{\theta}_0) \leq \varepsilon$ or iteration number $\geq N_{max}$ **return** $\boldsymbol{\theta}_0$

else

Step 3. **Adam Optimization method**

Set α (learning rate), β_1(=0.99), β_2(= 0.999), $\mathbf{m}_0 = 0, \mathbf{v}_0 = 0$ and k (iteration)

while $k \leq N_1$ **do**

$k \leftarrow k + 1$

$\mathbf{g}_1 \leftarrow \nabla_{\boldsymbol{\theta}} J_{Loss}(\boldsymbol{\theta}_{k-1})$; $\mathbf{m}_k \leftarrow \beta_1 \mathbf{m}_{k-1} + (1 - \beta_1)\mathbf{g}_k$;

$\mathbf{v}_k \leftarrow \beta_2 \mathbf{v}_{k-1} + (1 - \beta_1)\mathbf{g}_k^2$

$\bar{\mathbf{m}}_k \leftarrow \mathbf{m}_k/(1 - \beta_1^k)$; $\bar{\mathbf{v}}_k \leftarrow \mathbf{v}_k/(1 - \beta_2^k)$

$\boldsymbol{\theta}_k \leftarrow \boldsymbol{\theta}_{k-1} - \alpha \cdot \bar{\mathbf{m}}_k/(\sqrt{|\bar{\mathbf{v}}_k|} + \epsilon)$ $(\epsilon = 10^{-8})$

end while and **return** $\boldsymbol{\theta}_{N_1}$

Step 4. **L-BFGS method**

Set $\bar{\boldsymbol{\theta}}_0 = \boldsymbol{\theta}_{N_1}$, $w =$ integer, $i \leftarrow 0$

while $i \leq N_2$ **do**

set $\mathbf{B}_0 = \mathbf{I}$ for $i = 0$ and $\mathbf{B}_0 = \frac{\mathbf{s}_0^T \mathbf{r}_0}{\mathbf{r}_0^T \mathbf{r}_0} \mathbf{I}$ for $i > 0$

$\mathbf{p}_i \leftarrow -\mathbf{B}_i \nabla_{\boldsymbol{\theta}} J_{Loss}(\bar{\boldsymbol{\theta}}_i)$

$n \leftarrow min(i, w-1)$

$\mathbf{B}_i \leftarrow (\mathbf{V}_{i-1}^T ... \mathbf{V}_{i-1-n}^T)\mathbf{B}_0(\mathbf{V}_{i-1-n} ... \mathbf{V}_{i-1})$

$+\boldsymbol{\rho}_{i-1-n}(\mathbf{V}_{i-1}^T ... \mathbf{V}_{i-n}^T)\mathbf{s}_{i-1-n}\mathbf{s}_{i-n}^T(\mathbf{V}_{i-n} ... \mathbf{V}_{i-1})$

$+\boldsymbol{\rho}_{i-n}(\mathbf{V}_{i-1}^T ... \mathbf{V}_{i-n+1}^T)\mathbf{s}_{i-n}\mathbf{s}_{i-n}^T(\mathbf{V}_{i-n+1} ... \mathbf{V}_{i-1})$

$+ ... + \boldsymbol{\rho}_{i-1}\mathbf{s}_{i-1}\mathbf{s}_{i-1}^T$

$\bar{\boldsymbol{\theta}}_{i+1} = \bar{\boldsymbol{\theta}}_i + \alpha_i \mathbf{p}_i$

end while and **return** $\bar{\boldsymbol{\theta}}_{N_2}$

end if

Step 5. Input domain required for prediction; output u_1, v_1, u_2, v_2 based on $\bar{\boldsymbol{\theta}}_{N_2}$; transform u_1, v_1, u_2, v_2 back to the solutions with dimensions

4.4 Convergence Analysis

Since we require the loss function to be small in the artificial neural network computation, we may assume that $J_{Loss}(\boldsymbol{\theta}) \leq \varepsilon$, where ε is small value. Note that the neural network solutions, u_m, v_m, are composite functions of hyperbolic tangent functions, which are smooth functions. Thus, we may assume that they satisfy the following problem as

$$C_e^{(m)} \frac{\partial u_m}{\partial t} = k_e^{(m)} \frac{\partial^2 u_m}{\partial x^2} - G^{(m)}(u_m - v_m) + S^{(m)} + \varphi_e^{(m)}, \tag{4.14a}$$

$$C_l^{(m)} \frac{\partial v_m}{\partial t} = G^{(m)}(u_m - v_m) + \varphi_l^{(m)}, \tag{4.14b}$$

with initial and boundary condition

$$u_m(x,0) = T_0 + \eta_e^{(m)}, \qquad v_m(x,0) = T_0 + \eta_l^{(m)}, \tag{4.15a}$$

$$\frac{\partial u_1}{\partial x}(0,t) = \phi^{(1)}, \qquad \frac{\partial u_2}{\partial x}(1,t) = \phi^{(2)}, \tag{4.15b}$$

and the interfacial condition at $x = x_l$

$$k_e^{(1)}\frac{\partial u_1}{\partial x} = k_e^{(2)}\frac{\partial u_2}{\partial x} + \psi_1, \qquad U_e^{(1)} = U_e^{(2)} + \psi_2. \tag{4.16}$$

Here, $\varphi_e^{(m)}, \varphi_l^{(m)}, \eta_e^{(m)}, \phi^{(m)}, \psi_1$ and ψ_2 are functions which are assumed to satisfy

$$\max\{\left|\varphi_e^{(m)}\right|, \left|\varphi_l^{(m)}\right|, \left|\eta_e^{(m)}\right|, \left|\phi^{(m)}\right|, \left|\psi_1\right|, \left|\psi_1\right|\} \le \varepsilon, \tag{4.17}$$

where $m = 1$ for $0 \le x \le x_l$, $m = 2$ for $x_l \le x \le 1$, and $0 \le t \le 1$.

We would like to show the convergence of the ANN solution to the analytical solutions in Eqs. (4.5)-(4.8).

Theorem 4.4.1. Assume that the analytical solutions and the neural network solutions are smooth. The loss function $J_{Loss}(\boldsymbol{\theta}) \le \varepsilon$, where ε is small value. Assume that $C_e^{(m)}$ and $k_e^{(m)}$ $(m = 1, 2)$ are constants. Then, it holds

$$\int_0^1 \int_0^{x_l} [(E_e^{(1)})^2 + (E_l^{(1)})^2] dxdt + \int_0^1 \int_{x_l}^1 [(E_e^{(2)})^2 + (E_l^{(2)})^2] dxdt \le \varepsilon A, \tag{4.18}$$

where A is a constant.

Proof. Let $E_e^{(m)}(x,t) = u_m - T_e^{(m)}$ and $E_l^{(m)}(x,t) = v_m - T_l^{(m)}$. From Eqs. (4.5)-(4.8) and Eqs. (4.14)-(4.16), we obtain $E_e^{(m)}$ and $E_l^{(m)}$ satisfying

$$C_e^{(m)}\frac{\partial E_e^{(m)}}{\partial t} = k_e^{(m)}\frac{\partial^2 u_m}{\partial x^2} - G^{(m)}(E_e^{(m)} - E_l^{(m)}) + \varphi_e^{(m)}, \tag{4.19a}$$

$$C_l^{(m)}\frac{\partial E_l^{(m)}}{\partial t} = G^{(m)}(E_e^{(m)} - E_l^{(m)}) + \varphi_l^{(m)}, \tag{4.19b}$$

with initial and boundary conditions

$$E_e^{(m)}(x,0) = \eta_e^{(m)}, \qquad E_l^{(m)}(x,0) = \eta_l^{(m)}, \tag{4.20a}$$

$$\frac{\partial E_e^{(1)}}{\partial x}(0,t) = \phi^{(1)}, \qquad \frac{\partial E_e^{(2)}}{\partial x}(1,t) = \phi^{(2)}, \tag{4.20b}$$

and the interfacial condition at $x = x_l$

$$E_e^{(1)}(x_l,t) = E_e^{(2)}(x_l,t) + \psi_1, \quad k_e^{(1)}\frac{\partial E_e^{(1)}}{\partial x}(x_l,t) = k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}(x_l,t) + \psi_2. \tag{4.20c}$$

Multiplying Eq. (4.19a) by $E_e^{(m)}$ and Eq. (4.19b) by $E_l^{(m)}$, and integrating them with respect to x over $[0, x_l]$ for $m = 1$, and over $[x_l, 1]$ for $m = 2$, respectively, and then summing the results, we obtain

$$\begin{aligned}
&\frac{1}{2}\frac{d}{dt}\int_0^{x_l}[C_e^{(1)}(E_e^{(1)})^2 + C_l^{(1)}(E_e^{(1)})^2]dx + \frac{1}{2}\frac{d}{dt}\int_{x_l}^1[C_e^{(2)}(E_e^{(2)})^2 + C_l^{(2)}(E_e^{(2)})^2]dx \\
&+ \int_0^{x_l} k_e^{(1)}(\frac{\partial E_e^{(1)}}{\partial x})^2dx + \int_{x_l}^1 k_e^{(2)}(\frac{\partial E_e^{(2)}}{\partial x})^2dx + \int_0^{x_l} G^{(1)}[E_e^{(1)} - E_l^{(1)}]^2dx \\
&+ \int_{x_l}^1 G^{(2)}[E_e^{(2)} - E_l^{(2)}]^2dx \\
=&k_e^{(1)}\frac{\partial E_e^1}{\partial x}E_e^{(1)}|_0^{x_l} + k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}E_e^{(2)}|_{x_l}^{x_L} + \int_0^{x_l}\varphi_e^{(1)}E_e^{(1)}dx + \int_0^{x_l}\varphi_l^{(1)}E_l^{(1)}dx \\
&+ \int_{x_l}^1\varphi_e^{(1)}E_e^{(2)}dx + \int_{x_l}^1\varphi_l^{(2)}E_l^{(2)}dx.
\end{aligned} \tag{4.21}$$

Based on the interfacial condition and boundary condition Eqs. (4.20b)-(4.20c), we have

$$\begin{aligned}
\left|k_e^{(1)}\frac{\partial E_e^{(1)}}{\partial x}E_e^{(1)}|_0^{x_l} + k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}E_e^{(2)}|_{x_l}^1\right| \le &\left|[k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}E_e^{(2)} - k_e^{(1)}\frac{\partial E_e^{(1)}}{\partial x}E_e^{(1)}]|_{x_l}\right| \\
&+ \left|k_e^{(1)}\frac{\partial E_e^{(1)}}{\partial x}E_e^{(1)}|_0\right| + \left|k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}E_e^{(2)}|_1\right|
\end{aligned} \tag{4.22a}$$

$$\left|k_e^{(1)}\frac{\partial E_e^{(1)}}{\partial x}E_e^{(1)}|_0^{x_l}+k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}E_e^{(2)}|_{x_l}^1\right|$$

$$\leq\left|[k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}E_e^{(2)}-(k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}+\psi_2)(E_e^{(2)}+\psi_1)]|_{x_l}\right|+\left|k_e^{(1)}\phi^{(1)}E_e^{(1)}|_0\right|+\left|k_e^{(2)}\phi^{(2)}E_e^{(2)}|_1\right|$$

$$\leq\left|[k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}\psi_1+\psi_2(E_e^{(2)}+\psi_1)]|_{x_l}\right|+\left|k_e^{(1)}\phi^{(1)}E_e^{(1)}|_0\right|+\left|k_e^{(2)}\phi^{(2)}E_e^{(2)}|_1\right|. \tag{4.22b}$$

Using the smoothness of solutions and Eq. (4.17), Eq. (4.22) can be further simplified to

$$\left|k_e^{(1)}\frac{\partial E_e^{(1)}}{\partial x}E_e^{(1)}|_0^{x_l}+k_e^{(2)}\frac{\partial E_e^{(2)}}{\partial x}E_e^{(2)}|_{x_l}^1\right|\leq\varepsilon B_1, \tag{4.23}$$

where B_1 is a constant. Using the Cauchy-Schwarz inequality and Eq. (4.17), we have

$$\begin{aligned}&2\int_0^{x_l}\varphi_e^{(1)}E_e^{(1)}dx+2\int_0^{x_l}\varphi_l^{(1)}E_l^{(1)}dx+2\int_{x_l}^1\varphi_e^{(1)}E_e^{(2)}dx+2\int_{x_l}^1\varphi_l^{(2)}E_l^{(2)}dx\\ \leq\quad&\varepsilon^2B_2+\int_0^{x_l}[C_e^{(1)}(E_e^{(1)})^2+C_l^{(1)}(E_l^{(1)})^2]dx+\int_{x_l}^1[C_e^{(2)}(E_e^{(2)})^2+C_l^{(2)}(E_l^{(2)})^2]dx,\end{aligned} \tag{4.24}$$

where $B_2=\frac{1}{C_e^{(1)}}+\frac{1}{C_l^{(1)}}+\frac{1}{C_e^{(2)}}+\frac{1}{C_l^{(2)}}$. Substituting Eqs. (4.23) - (4.24) into Eq. (4.21) and then denoting

$$D(t)=\int_0^{x_l}[C_e^{(1)}(E_e^{(1)})^2+C_l^1(E_l^{(1)})^2]dx+\int_{x_l}^1[C_e^{(2)}(E_e^{(2)})^2+C_l^2(E_l^{(2)})^2]dx, \tag{4.25}$$

Eq. (4.21) becomes

$$\frac{d}{dt}D(t)\leq D(t)+\varepsilon B_1+\varepsilon^2B_2, \tag{4.26}$$

implying that

$$D(t)\leq e^t[\int_0^t e^{-\tau}(\varepsilon B_1+\varepsilon^2B_2)d\tau+D(0)]\leq e^t[\varepsilon B_1+\varepsilon^2B_2+D(0)]. \tag{4.27}$$

Thus, we obtain

$$\int_0^1 D(t)dt \leq [\varepsilon B_1 + \varepsilon^2 B_2 + D(0)] \int_0^1 e^t dt$$

$$\leq [\varepsilon B_1 + \varepsilon^2 B_2 + D(0)]e^1. \tag{4.28}$$

Note that

$$D(0) = \int_0^{x_l} [C_e^{(1)}(\eta_e^{(1)})^2 + C_l^{(1)}(\eta_l^{(1)})^2]dx + \int_{x_l}^1 [C_e^{(2)}(\eta_e^{(2)})^2 + C_l^{(2)}(\eta_l^{(2)})^2]dx$$

$$\leq \varepsilon^2 B_3, \tag{4.29}$$

where B_3 is a constant. We obtain

$$B_4 \int_0^1 \int_0^{x_l} [(E_e^{(1)})^2 + (E_l^{(1)})^2]dxdt + \int_0^1 \int_{x_l}^1 [(E_e^{(2)})^2 + (E_l^{(2)})^2]dxdt$$

$$\leq \int_0^1 D(t)dt \leq [\varepsilon B_1 + \varepsilon^2 B_2 + \varepsilon^2 B_3]e^1, \tag{4.30}$$

where $B_4 = \min\{C_e^{(1)}, C_e^{(2)}, C_l^{(1)}, C_l^{(2)}\}$, and hence Eq. (4.18) is obtained, where $A = e[B_1 + \varepsilon B_2 + \varepsilon B_3]/B_4$.

4.5 Summary

In this chapter, we have presented an artificial neural network (ANN) method and its algorithm for solving the parabolic two-temperature heat conduction equations in double-layered thin films exposed to ultrashort-pulsed lasers. Convergence of the ANN solution to the analytical solution has been analyzed theoretically. This work has already been published in the journal *International Journal of Heat and Mass Transfer*, volume 178, 2021 [196]. The doi for the article is https://doi.org/10.1016/.

CHAPTER 5

SIMULATION OF HEAT CONDUCTION IN GOLD-CHROMIUM THIN FILMS EXPOSED TO ULTRASHORT-PULSED LASERS

In this chapter, we will present our numerical results obtained based on the gradient preserved scheme and the neural network method developed in chapters 3 and 4 for solving parabolic two-temperature model in double-layered thin film exposed to ultrashort-pulse laser heating.

5.1 Results Obtained Based on Gradient Preserved Method

To show the applicability of the Gradient Preserved Method, we consider a 50-nm gold film padding on a 50-nm chromium film, which is exposed to the ultrashort-pulsed laser heating. This is a benchmark problem given in [80], and the temperature rise in these two films can be modeled by the well-known parabolic two-temperature heat conduction model as

$$C_e^m(T_e^{(m)})\frac{\partial T_e^{(m)}}{\partial t} = \frac{\partial}{\partial x}\left(k_m \frac{T_e^{(m)}}{T_l^{(m)}}\frac{\partial T_e^{(m)}}{\partial x}\right) - G^{(m)}(T_e^{(m)} - T_l^{(m)}) + S(x_m, t), \quad (5.1a)$$

$$C_l^{(m)}\frac{\partial T_l^{(m)}}{\partial t} = G^{(m)}(T_e^{(m)} - T_l^{(m)}), \quad (5.1b)$$

where the heat source is given as

$$S(x_m, t) = 0.94\frac{1-R}{t_p\delta}J\exp\left(-\frac{x_m}{\delta} - 2.77(\frac{t-2t_p}{t_p})^2\right), \quad (5.1c)$$

with the initial condition as

$$T_e^{(m)}(x,0) = T_l^{(m)}(x,0) = 300K, \tag{5.1d}$$

$$\frac{\partial T_e^{(m)}(x,0)}{\partial t} = \frac{\partial T_l^{(m)}(x,0)}{\partial t} = 0, \tag{5.1e}$$

the thermal insulated boundary condition as

$$\frac{\partial T_e^{(m)}(0,t)}{\partial x} = \frac{\partial T_l^{(m)}(0,t)}{\partial x} = 0, \qquad \frac{\partial T_e^{(m)}(L,t)}{\partial x} = \frac{\partial T_l^{(m)}(L,t)}{\partial x} = 0, \tag{5.1f}$$

and the thermal perfectly insulated interfacial condition as

$$k_1 \frac{T_e^{(1)}}{T_{(l)}^{(1)}} \frac{\partial T_e^{(1)}}{\partial x} = k_2 \frac{T_e^{(2)}}{T_{(l)}^{(2)}} \frac{\partial T_e^{(2)}}{\partial x}, \tag{5.1g}$$

$$T_e^{(1)}(l,t) = T_l^{(1)}(l,t). \tag{5.1h}$$

Here, $m = 1, 2$ represents the gold layer ($0 \le x \le l$) and the chromium layer ($l \le x \le L$), respectively, with $l = 50$ (nm) and $L = 100$ (nm). T_e and T_l are the electron and lattice temperatures, respectively, k_m is the thermal conductivity, $G^{(m)}$ is the electron-lattice coupling factor, $C_e^{(m)} = (C_e^0)^{(m)} T_e/T_0$ and $C_l^{(m)}$ are the electron heat capacity and the lattice heat capacity, respectively. The thermal properties of these parameters are listed in Table 5.1 [80].

Table 5.1: Thermal properties of gold and chromium.

Parameters	**Gold**	**Chromium**
G	2.6×10^{16} Wm^{-3}K^{-1}	42×10^{16} Wm^{-3}K^{-1}
C_e^0	2.1×10^4 Jm^{-3}K^{-1}	5.8×10^4 Jm^{-3}K^{-1}
C_l	2.5×10^6 Jm^{-3}K^{-1}	3.3×10^6 Jm^{-3}K^{-1}
k	315 Wm^{-1}K^{-1}	94 Wm^{-1}K^{-1}

For the heat source, $R = 0.93$ is the reflectivity; $t_p = 0.1$ (ps) is the full-width-at-half-maximum duration of the laser pulse; $\delta = 0.1$ (fs) is the radiation penetration depth, and $J = 500$ (Jm^2) is laser fluency.

It can be seen that Eq. (5.1a) is virtually a nonlinear equation. To apply our present method to the model, we employ three levels in time, $n-1$, n, $n+1$, where in our scheme Eq. (3.56), is set to be

$$k_m(x) = \frac{k_m T_e^{(m)}(x, t_n)}{T_l^{(m)}(x, t_n)}, \tag{5.2a}$$

$$C_e^{(m)}(T_e^{(m)}) = C_e^{(m)}(T_e^{(m)}(x, t_n)). \tag{5.2b}$$

. As such, the scheme for solving the above model can be expressed for the left hand-side boundary as

$$\begin{aligned} \alpha_1 \delta_t (T_e^{(1)})_0^n + \alpha_2 \delta_t (T_e^{(1)})_1^n = {} & \frac{1}{h}[(k_1)_{1/2} \frac{(\bar{T}_e^{(1)})_1^n - (\bar{T}_e^{(1)})_0^n}{h} - \alpha_3 (k_1)_0 \alpha(t_n)] \\ & + [-\frac{h}{12} + \frac{h^2}{24}\frac{(k_{1x})_0}{(k_1)_0}]\alpha'(t_n) + \beta_1 G^{(1)}(\bar{T}_e^{(1)} - \bar{T}_l^{(1)})_0^n \\ & + \beta_2 G^{(1)}(\bar{T}_e^{(1)} - \bar{T}_l^{(1)})_1^n + f_0^n; \end{aligned} \tag{5.3a}$$

for the first layer as

$$\begin{aligned} & (\alpha_4)_j \delta_t (T_e^{(1)})_{j-1}^n + (\alpha_5)_j \delta_t (T_e^{(1)})_j^n + (\alpha_6)_j \delta_t (T_e^{(1)})_{j+1}^n \\ & = \frac{1}{h^2}[(k_1)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_1)_{j+\frac{1}{2}}]((\bar{T}_e^{(1)})_{j+1}^n - (\bar{T}_e^{(1)})_j^n) - \frac{1}{h^2}[(k_1)_{j-\frac{1}{2}} \\ & - \frac{h^2}{24}(D_1)_{j-\frac{1}{2}}]((\bar{T}_e^{(1)})_j^n - (\bar{T}_e^{(1)})_{j-1}^n) + \beta_3 G^{(1)}(\bar{T}_e^{(1)} - \bar{T}_l^{(1)})_{j-1}^n \\ & \quad + \beta_4 G^{(1)}(\bar{T}_e^{(1)} - \bar{T}_l^{(1)})_j^n + \beta_5 G^{(1)}(\bar{T}_e^{(1)} - \bar{T}_l^{(1)})_{j+1}^n + f_j^n, 1 \le j \le l-1; \end{aligned} \tag{5.3b}$$

for the interface as

$$\alpha_7\delta_t(T_e^{(1)})_{l-1}^n + \alpha_{10}\delta_t(T_e^{(1)})_l^n + \tilde{\alpha_7}\delta_t(T_e^{(2)})_{l+1}^n$$

$$= \frac{1}{h^2}[\alpha_0(k_2)_{l+\frac{1}{2}}((\bar{T}_e^{(2)})_{l+1}^n - (\bar{T_e}^{(2)})_l^n) - (k_1)_{l-\frac{1}{2}}((\bar{T_e}^{(1)})_l^n - (\bar{T_e}^{(1)})_{l-1}^n)]$$

$$+ \beta_6 G^{(1)}(\bar{T}_e^{(1)} - \bar{T}_l^{(1)})_{l-}^n + \beta_7 G^{(1)}(\bar{T}_e^{(1)} - \bar{T}_l^{(1)})_{l-1}^n + \alpha_0\tilde{\beta}_6 G^{(2)}(\bar{T_e}^{(2)} - \bar{T_l}^{(2)})_{l+}^n$$

$$+ \alpha_0\tilde{\beta}_7 G^{(2)}(\bar{T}_e^{(2)} - \bar{T}_l^{(2)})_{l+1}^n + f_l^n; \tag{5.3c}$$

for the second layer as

$$(\tilde{\alpha}_4)_j\delta_t(T_e^{(2)})_{j-1}^n + (\tilde{\alpha}_5)_j\delta_t(T_e^{(2)})_j^n + (\tilde{\alpha}_6)_j\delta_t(T_e^{(2)})_{j+1}^n = \frac{1}{h^2}[(k_2)_{j+\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j+\frac{1}{2}}]$$

$$((\bar{T_e}^{(2)})_{j+1}^n - (\bar{T_e}^{(2)})_j^n) - \frac{1}{h^2}[(k_2)_{j-\frac{1}{2}} - \frac{h^2}{24}(D_2)_{j-\frac{1}{2}}]((\bar{T_e}^{(2)})_j^n - (\bar{T_e}^{(2)})_{j-1}^n)$$

$$+ \tilde{\beta}_3 G^{(2)}(\bar{T_e}^{(2)} - \bar{T_l}^{(2)})_{j-1}^n + \tilde{\beta}_4 G^{(2)}(\bar{T_e}^{(2)} - \bar{T_l}^{(2)})_j^n + \tilde{\beta}_5 G^{(2)}(\bar{T_e}^{(2)} - \bar{T_l}^{(2)})_{j+1}^n$$

$$+ \tilde{f}_j^n, l+1 \leq j \leq N-1; m \tag{5.3d}$$

for the right hand-side boundary as

$$\tilde{\alpha}_1\delta_t(T_e^{(2)})_N^n + \tilde{\alpha}_2\delta_t(T_e^{(2)})_{N-1}^n = \frac{1}{h}[\tilde{\alpha}_3(k_2)_N\beta(t_n) - (k_2)_{N-\frac{1}{2}}\frac{(\bar{T_e}^{(2)})_N^n - (\bar{T_e}^{(2)})_{N-1}^n}{h}]$$

$$+ [\frac{h}{12} + \frac{h^2}{24}\frac{(k_{2x})_N}{(k_2)_N}]\beta'(t_n) + \tilde{\beta}_1 G^{(2)}(\bar{T_e}^{(2)} - \bar{T_l}^{(2)})_N^n$$

$$+ \tilde{\beta}_2 G^{(2)}(\bar{T_e}^{(2)} - \bar{T_l}^{(2)})_{N-1}^n + f_N^n; \tag{5.3e}$$

where

$$\alpha_1 = (\frac{5}{12} - \frac{h}{12}\frac{(k_{1x})_0}{(k_1)_0} + \frac{h^2}{48}\frac{[4(k_{1x})_0]^2 - 3(k_1)_0(k_{1xx})_0}{[(k_1)_0]^2})C_e^{(1)}(T_e^{(1)})_0^n, \tag{5.4a}$$

$$\alpha_2 = (\frac{1}{12})C_e^{(1)}(T_e^{(1)})_0^n, \tag{5.4b}$$

$$\alpha_3 = (1 + \frac{h^2}{24(k_1)_0}[\frac{2[(k_{1x})_0]^2}{(k_1)_0} - (k_{1xxx})_0]$$

$$+ \frac{h^3}{48(k_1)_0}[\frac{5(k_1)_0(k_{1x})_0(k_{1xx})_0 - 4[(k_{1x})_0]^3}{[(k_1)_0]^2} - (k_{1xxx})_0])C_e^{(1)}(T_e^{(1)})_0^n, \tag{5.4c}$$

$$f_0^n = [\frac{1}{2} - \frac{h}{12}\frac{(k_{1x})_0}{(k_1)_0} - \frac{h^2}{48}\frac{4[(k_{1x})_0]^2 - 3(k_1)_0(k_{1xx})_0}{[(k_1)_0]^2}]S_0^n$$

$$+ [\frac{h}{6} - \frac{h^2}{24}\frac{(k_{1x})_0}{(k_1)_0}](S_x)_0^n + \frac{h^2}{24}(S_{xx})_0^n; \tag{5.4d}$$

$$\tilde{\alpha}_1 = (\frac{5}{12} + \frac{h}{12}\frac{(k_{2x})_N}{(k_2)_N} + \frac{h^2}{48}\frac{4[(k_{2x})_N]^2 - 3(k_2)_N(k_{2xx})_N}{[(k_2)_N]^2})C_e^{(2)}(T_e^{(2)})_N^n, \tag{5.5a}$$

$$\tilde{\alpha}_2 = (\frac{1}{12})C_e^{(2)}(T_e^{(2)})_N^n, \tag{5.5b}$$

$$\tilde{\alpha}_3 = (1 + \frac{h^2}{24(k_2)_N}[\frac{2[(k_{2x})_N]^2}{(k_2)_N} - (k_{2xxx})_N]$$

$$- \frac{h^3}{48(k_2)_N}[\frac{5(k_2)_N(k_{2x})_N(k_{2xx})_N - 4[(k_{2x})_N]^3}{[(k_2)_N]^2} - (k_{2xxx})_N])C_e^{(2)}(T_e^{(2)})_N^n, \tag{5.5c}$$

$$f_N^n = [\frac{1}{2} + \frac{h}{12}\frac{(k_{2x})_N}{(k_2)_N} + \frac{h^2}{48}\frac{4[(k_{2x})_N]^2 - 3(k_2)_N(k_{2xx})_N}{[(k_2)_N]^2}]S_N^n$$

$$- [\frac{h}{6} + \frac{h^2}{24}\frac{(k_{2x})_N}{(k_2)_N}](S_x)_N^n + \frac{h^2}{24}(S_{xx})_N^n; \tag{5.5d}$$

$$(\alpha_4)_j = (\frac{1}{12} + \frac{h}{24}\frac{(k_{1x})_{j-1}}{(k_1)_{j-1}})C_e^{(1)}(T_e^{(1)})_j^n, \tag{5.6a}$$

$$(\alpha_5)_j = (\frac{5}{6})C_e^{(1)}(T_e^{(1)})_j^n, \tag{5.6b}$$

$$(\alpha_6)_j = (\frac{1}{12} - \frac{h}{24}\frac{(k_{1x})_{j+1}}{(k_1)_{j+1}})C_e^{(1)}(T_e^{(1)})_j^n, \tag{5.6c}$$

$$(D_1)_{j+\frac{1}{2}} = 2\frac{[(k_{1x})_{j+\frac{1}{2}}]^2}{(k_1)_{j+\frac{1}{2}}} - (k_{1xx})_{j+\frac{1}{2}}, \tag{5.6d}$$

$$f_j^n = [1 - \frac{h^2}{12}\frac{(k_1)_j(k_{1xx})_j - (k_1)_j^2}{(k_1)_j^2}]S_j^n - \frac{h^2}{12}\frac{(k_{1x})_j}{(k_1)_j}(S_x)_j^n + \frac{h^2}{12}(S_{xx})_j^n; \qquad (5.6e)$$

$$(\tilde{\alpha}_4)_j = (\frac{1}{12} + \frac{h}{24}\frac{(k_{2x})_{j-1}}{(k_2)_{j-1}})C_e^{(2)}(T_e^{(2)})_j^n, \qquad (5.6a)$$

$$(\tilde{\alpha}_5)_j = (\frac{5}{6})C_e^{(2)}(T_e^{(2)})_j^n, \qquad (5.6b)$$

$$(\tilde{\alpha}_6)_j = (\frac{1}{12} - \frac{h}{24}\frac{(k_{2x})_{j+1}}{(k_2)_{j+1}})C_e^{(2)}(T_e^{(2)})_j^n, \qquad (5.7c)$$

$$(D_2)_{j+\frac{1}{2}} = 2\frac{[(k_{2x})_{j+\frac{1}{2}}]^2}{(k_2)_{j+\frac{1}{2}}} - (k_{2xx})_{j+\frac{1}{2}}, \qquad (5.7d)$$

$$\tilde{f}_j^n = [1 - \frac{h^2}{12}\frac{(k_2)_j(k_{2xx})_j - (k_2)_j^2}{(k_2)_j^2}]S_j^{n+\frac{1}{2}} - \frac{h^2}{12}\frac{(k_{2x})_j}{(k_2)_j}(S_x)_j^n + \frac{h^2}{12}(S_{xx})_j^{n+\frac{1}{2}}; \qquad (5.7e)$$

$$\alpha_9 = 1 + \frac{h^2}{12(k_1)_{l-\frac{1}{2}}}[\frac{(k_{1x})_{l-\frac{1}{2}}(k_{1x})_{l-\frac{1}{2}}}{(k_1)_{l-\frac{1}{2}}} - \frac{(k_1)_{l-2} - 2(k_1)_{l-1} + (k_1)_l}{2h^2}], \qquad (5.8a)$$

$$\tilde{\alpha}_9 = 1 + \frac{h^2}{12(k_2)_{l+\frac{1}{2}}}[\frac{(k_{2x})_{l+\frac{1}{2}}(k_{2x})_{l+\frac{1}{2}}}{(k_2)_{l+\frac{1}{2}}} - \frac{(k_1)_l - 2(k_1)_{l+1} + (k_1)_{l+2}}{2h^2}], \qquad (5.8b)$$

$$\alpha_0 = \frac{\alpha_9}{\tilde{\alpha}_9}, \qquad (5.8c)$$

$$\alpha_8 = [\frac{3}{8}\alpha_9 + \frac{h}{24}\frac{(k_{1x})_{l-\frac{1}{2}}}{(k_1)_{l-\frac{1}{2}}} - \frac{1}{24}]C_e^{(1)}(T_e^{(1)})_{l-\frac{1}{2}}^n, \qquad (5.8d)$$

$$\alpha_7 = [\frac{1}{8}\alpha_9 - \frac{h}{24}\frac{(k_{1x})_{l-\frac{1}{2}}}{(k_1)_{l-\frac{1}{2}}} + \frac{1}{24}]C_e^{(1)}(T_e^{(1)})_{l-\frac{1}{2}}^n, \qquad (5.8e)$$

$$\tilde{\alpha}_8 = [\frac{3}{8}\tilde{\alpha}_9 - \frac{h}{24}\frac{(k_{2x})_{l+\frac{1}{2}}}{(k_2)_{l+\frac{1}{2}}} + \frac{1}{24}]C_e^{(2)}(T_e^{(2)})_{l+\frac{1}{2}}^n, \qquad (5.8f)$$

$$\tilde{\alpha}_7 = [\frac{1}{8}\tilde{\alpha}_9 - \frac{h}{24}\frac{(k_{2x})_{l+\frac{1}{2}}}{(k_2)_{l+\frac{1}{2}}} - \frac{1}{24}]C_e^{(2)}(T_e^{(2)})_{l+\frac{1}{2}}^n, \qquad (5.8g)$$

$$\alpha_{10} = \alpha_8 + \alpha_0\tilde{\alpha}_8, \qquad (5.8h)$$

$$f_l^n = -(\frac{\alpha_9}{2} - \frac{h}{12}\frac{(k_{1x})_{l-\frac{1}{2}}}{(k_1)_{l-\frac{1}{2}}})S_{l-\frac{1}{2}}^{n+\frac{1}{2}} + (\frac{-\alpha_9 h}{8} + \frac{h}{24})(S_x)_{l-\frac{1}{2}}^n$$

$$+ \alpha_0[(\frac{\tilde{\alpha}_9}{2} + \frac{h}{12}\frac{(k_{2x})_{l+\frac{1}{2}}}{(k_2)_{l+\frac{1}{2}}})S_{l+\frac{1}{2}}^n + (\frac{\tilde{\alpha}_9 h}{8} - \frac{h}{24})(S_x)_{l+\frac{1}{2}}^n]; \tag{5.8i}$$

$$\beta_1 = \tilde{\beta}_1 = \frac{-5}{12}, \qquad \beta_2 = \tilde{\beta}_2 = \frac{-1}{12}, \tag{5.9a}$$

$$\beta_3 = \frac{-1}{12} - \frac{h}{24}\frac{k_{1x}(x)}{k_1(x)}, \tag{5.9b}$$

$$\beta_4 = \frac{-5}{12} + \frac{h^2}{12}\frac{k_1(x)k_{1xx}(x) - (k_{1x}(x))^2}{(k_1(x))^2}, \tag{5.9c}$$

$$\beta_5 = \frac{-1}{12} + \frac{h}{24}\frac{k_{1x}(x)}{k_1(x)}, \tag{5.9d}$$

$$\tilde{\beta}_3 = \frac{-1}{12} - \frac{h}{24}\frac{k_{2x}(x)}{k_2(x)}, \tag{5.9e}$$

$$\tilde{\beta}_4 = \frac{-5}{12} + \frac{h^2}{12}\frac{k_2(x)k_{2xx}(x) - (k_{2x}(x))^2}{(k_2(x))^2}, \tag{5.9f}$$

$$\tilde{\beta}_5 = \frac{-1}{12} + \frac{h}{24}\frac{k_{2x}(x)}{k_2(x)}, \tag{5.9g}$$

$$\beta_6 = -\frac{3}{8}\alpha_9 - \frac{h}{24}\frac{(k_{1x})_{l-\frac{1}{2}}}{(k_1)_{l-\frac{1}{2}}} - \frac{1}{24}, \qquad \tilde{\beta}_6 = -\frac{3}{8}\tilde{\alpha}_9 + \frac{h}{24}\frac{(k_{2x})_{l+\frac{1}{2}}}{(k_2)_{l+\frac{1}{2}}} + \frac{1}{24}, \tag{5.9h}$$

$$\beta_7 = -\frac{1}{8}\alpha_9 + \frac{h}{24}\frac{(k_{1x})_{l-\frac{1}{2}}}{(k_1)_{l-\frac{1}{2}}} - \frac{1}{24}, \qquad \tilde{\beta}_7 = -\frac{1}{8}\tilde{\alpha}_9 + \frac{h}{24}\frac{(k_{2x})_{l+\frac{1}{2}}}{(k_2)_{l+\frac{1}{2}}} - \frac{1}{24}. \tag{5.9i}$$

Here, we use the notations

$$\delta_t(T_e^{(m)})_j^n = \frac{(T_e^m)_j^{n+1} - (T_e^m)_j^{n-1}}{2\tau}, (\bar{T}_e^{(m)})_j^n = \frac{(T_e^m)_j^{n+1} + (T_e^m)_j^{n-1}}{2},$$

$$(\bar{T}_l^{(m)})_j^n = \frac{(T_l^m)_j^{n+1} + (T_l^m)_j^{n-1}}{2}.$$

In our computation, we chose 40 grid points in the x-direction with $h = 2.5 \times 10^{-6}$(mm) and $\tau = 0.001$(ps). Based on our scheme, we obtained that the maximum temperature rise is 3474.80 (K). Figure 5.1 represents electron temperature profiles and lattice temperature profiles respectively at various times in the 50-nm gold/ 50-nm chromium two-layer film during 0.1 (ps) laser pulse heating with $J = 500$ Jm^{-2}. As compared with Fig. 6 given in [80], our numerical results are not visibly different from those shown in Fig. 6 in [80]. In particular, we would like to point out that Fig. 6 in [80] was obtained based on the Cranck-Nicolson scheme on a 400-grid point mesh, which is a second order accurate scheme. On the other hand, our results were obtained based on a 40-grid point mesh, which is many fewer grid points. This indicates the advantage of our higher-order compact scheme.

5.2 Results Obtained Based on the Artificial Neural Network Method

To test the applicability of the present artificial neural network method and its algorithm, we consider the parabolic two-temperature heat conduction model in Eqs. (4.1)-(4.3) in three cases: (i) linear one dimension where $C_e^{(m)}$ and $k_e^{(m)}$ are constants; (ii) nonlinear one dimension where $C_e^{(m)} = C_{e0}^{(m)} T_e^{(m)}/T_0$ and $k_e^{(m)} = k_{e0}^{(m)} T_e^{(m)}/T_l^{(m)}$; and (iii) nonlinear two dimensions in x and y where $C_e^{(m)} = C_{e0}^{(m)} T_e^{(m)}/T_0$ and $k_e^{(m)} = k_{e0}^{(m)} T_e^{(m)}/T_l^{(m)}$. In the algorithm, we chose $\beta_1 = 0.99$, $\beta_1 = 0.99$, $\varepsilon = 10^{-8}$.

Case 1 (Linear one dimension). We consider a 50nm gold layer padding on a 50nm chromium layer exposed to a ultrashort-pulsed laser heating where the parameters in the energy absorption rate were chosen to be $J = 13.4$ (J/m^2), $R = 0.93$, $t_p = 100$ (fs), $\delta = 15.3$ (nm) [170], the thermal properties of gold and chromium were listed in Table 5.1 and $T_0 = 300$ (K). Since a lower laser fluence J was chosen, we assumed $C_e^{(m)} = C_{e0}^{(m)}$ and $k_e^{(m)} = k_{e0}^{(m)}$ in Eq. (4.1a), which becomes a linear equation.

For this case, we would compare our ANN solutions with those obtained in [116], where the time interval is $0 \leq t \leq 1$ (ps). To catch the pulse duration, we first divided the time interval into three sub-intervals as (i) $0 \leq t \leq 0.5$ and (ii) $0.5 \leq t \leq 1.0$. For $0 \leq t \leq 0.5$, we made a dimensionless transformation based on Eq. (4.4) where $t_T = 0.25$. We then discretized t and x into 400 and 400 grid points, respectively, and chose randomly 80 grid points as initial, boundary, and interface training points for Eqs. (4.7)-(4.8) from the discretized set of points.We further used the Latin hypercube sampling technique[86] to select 25,000 computer-generated random points as training points for the PDEs in Eq. (4.5) in each layer within the dimensionless domain of $0 \leq t \leq 1$ and $0 \leq x \leq 1$. Because the pulse occurs in this time sub-interval, we designed 11 hidden-layers with 150 units each for the neural nets in Figure 4.2 to capture the pulse. In our algorithm, we set the iteration number $N_1 = 50000$ for the Adam optimization method and $N_2 = 50000$ for the L-BFGS method. Based on the algorithm, we obtained the final value of the loss function to be 9.668900×10^{-6}, and hence obtained the values of weights and biases for u_1, v_1, u_2, and v_2 in this sub-interval. The training time for this interval was 20493.2312 seconds.

For $0.5 \leq t \leq 1.0$ (ps), we first used the obtained values of u_1, v_1, u_2, and v_2 at $t = 0.5$ (ps) as the initial values for the PDEs in Eq. (4.5). and made a dimensionless transformation based on Eq. (4.4), where $t_0 = 0.5$, $t_T = 1.0$ and

$$t^* = \frac{t - t_0}{t_T - t_0}, \quad t_p^* = \frac{t_p - t_0}{t_T - t_0}. \tag{5.9}$$

We then discretized t and x into 100 and 200 grid points, respectively, and chose randomly 50 grid points as initial, boundary, and interface training points for Eqs.

(4.7)-(4.8) from the discretized set of points, and then we selected 20,000 computer generated random points (using latin hyper cube sampling technique) as the training points for the PDEs in Eq. (4.5) in each layer. We kept the hidden layers and units as well as the number of iterations same as those used for the first time interval, and based on the algorithm, we obtained the final value of the loss function to be 7.312010 $\times$ 10^{-5}, and the values of u_1, v_1, u_2, and v_2, in this sub-interval. The training time for this interval was 19271.5810 seconds.

Figure 5.2 shows electron temperature profiles and lattice temperature profiles respectively at various times $t = 0.2, 0.25, 0.5, 1.0$ (ps), respectively. As compared with Figures 2 and 3 obtained in [170], we see from Figure 5.2 that our present results agree well with those in [170].

Case 2 (Nonlinear one dimension). We considered a 50nm gold layer padding on a 50nm chromium layer exposed to a ultrashort-pulsed laser heating where the parameters in the energy absorption rate were chosen to be $J = 500$ (J/m^2), $R = 0.93$, $t_p = 100$ (fs), $\delta = 15.3$ (nm) [80], and $T_0 = 300$ (K). Since laser fluence J is higher, we chose $C_e^{(m)} = C_{e0}^{(m)} T_e^{(m)}/T_0$ and $k_e^{(m)} = k_{e0}^{(m)} T_e^{(m)}/T_l^{(m)}$ in Eq. (4.1a), which becomes a nonlinear equation.

For this case, we would compare our ANN solutions with those obtained in [80]. Note that the initial condition in [80] was set at t = -2t_p. We chose the time interval to be $0 \leq t \leq 6.2$ (ps) in our computation. Once the solutions were obtained, we shifted the time back by -2t_p to compare with those in [80]. To simulate the nonlinear heat conduction, we divided the time interval into three intervals as: (i) $0 \leq t \leq 0.7$, (ii) $0.7 \leq t \leq 1.2$ and (iii) $1.2 \leq t \leq 6.2$. We discretized t and x into

500 and 400 grid points, respectively, and chose randomly 50 grid points as initial, boundary, and interface training points for Eqs. (4.7)-(4.8) from the discretized set of points. Then we selected 10,000 computer generated random points (using latin hyper cube sampling technique) as training points for the PDEs in Eq. (4.5) in each layer. Here, we used higher resolution points in t to capture the pulse properly. In our algorithm, we employed 4 hidden-layers with 100 units each for the neural nets in Figure 4.2 and set $N_1 = N_2 = 50000$. Based on the algorithm, we obtained the final values of the loss function at $t = 0.3$ (ps) and $t = 0.7$ (ps) to be 4.7904734×10^{-5}, and hence obtained the values of weights and biases for u_1, v_1, u_2, and v_2 in this sub-interval. The training time for this time interval was 4095.0951 seconds.

For $0.7 \leq t \leq 1.2$, we used 4 hidden layers with 80 hidden units each and discretized t and x into 300 and 200 grid points, respectively, with 40 grid points as initial, boundary and interficial training points and the same number of training points for the PDE's and the same number of iterations as used in the first time sub-interval. Based on the algorithm, we obtained the final loss function value to be 6.262183×10^{-5}, and hence obtained the values of weights and biases for u_1, v_1, u_2, and v_2 in this sub-interval. The training time for this tie interval was 4189.9203 seconds.

For $1.2 \leq t \leq 6.2$, we used 5 hidden layers with 120 units each and discretized into dimensionless intervals $0 \leq t \leq 1$ and $0 \leq x \leq 1$ into 400 and 200 grid points, respectively, with the same numbers of training points for initial, boundary, and interface as above but 20,000 training points for the PDEs in Eq. (4.5) for each layer. Here, we selected more training points of slightly higher resolution and more number of layers and units because the time interval is longer as compared to the other time intervals. Based on the algorithm, we obtained the final loss value to be 3.4241533×10^{-5}. The training time for this time interval was 4860.0788 seconds.

Figure 5.3 shows electron temperature profiles and lattice temperature profiles at various times $t = 0.1, 0.5, 1.0, 2.0, 6.0$ (ps), respectively, which were obtained based on the solutions of u_1, v_1, u_2 and v_2 after the time was shifted back by -$2t_p$. As compared with FIG. 6 obtained in [80], we see from Figure 5.3 that the present solutions agree well with those obtained in [80].

It should be pointed out that for the linear case, more hidden layers and units along with more training points were used as compared to the nonlinear case. This is because for the linear case, the lattice temperature remains almost same for most of the part except for a very small jump at the interface. To capture such a small jump, the number of layers and units needs to be high along with more training points. Furthermore, from our experience more high resolution training points are needed to capture the pulse. Once the pulse is captured, one may use fewer training points, layers and units for the other time intervals.

Case 3 (Nonlinear two dimensions). We extended the above Case 2 to a two-dimensional case, where the parabolic two-temperature model is given as

$$C_{e0}^{(m)} T_e^{(m)} \frac{\partial T_e^{(m)}}{\partial t} = \frac{\partial}{\partial x}\left(k_{e0}^{(m)} \frac{T_e^{(m)}}{T_l^{(m)}} \frac{\partial T_e^{(m)}}{\partial x}\right) + \frac{\partial}{\partial y}\left(k_{e0}^{(m)} \frac{T_e^{(m)}}{T_l^{(m)}} \frac{\partial T_e^{(m)}}{\partial y}\right)$$

$$- G^{(m)}(T_e^{(m)} - T_l^{(m)}) + S(x_m, y, t), \tag{5.10a}$$

$$C_l^{(m)} \frac{\partial T_l^{(m)}}{\partial t} = G^{(m)}(T_e^{(m)} - T_l^{(m)}), \tag{5.10b}$$

with the heat source

$$S(x_m, t) = 0.94\frac{1-R}{t_p\delta}J\exp\left(-\frac{x_m}{\delta} - \frac{(y-y_0)^2}{\omega^2} - 2.77(\frac{t-2t_p}{t_p})^2\right), \tag{5.10c}$$

the initial condition

$$T_e^{(m)}(x,y,0) = T_l^{(m)}(x,y,0) = T_0, \quad \frac{\partial T_e^{(m)}(x,y,0)}{\partial t} = \frac{\partial T_l^{(m)}(x,y,0)}{\partial t} = 0, \tag{5.10d}$$

the thermal insulated boundary conditions

$$\frac{\partial T_e^{(m)}(0,y,t)}{\partial x} = \frac{\partial T_l^{(m)}(0,y,t)}{\partial x} = 0, \quad \frac{\partial T_e^{(m)}(x_L,y,t)}{\partial x} = \frac{\partial T_l^{(m)}(x_L,y,t)}{\partial x} = 0, \tag{5.10e}$$

$$\frac{\partial T_e^{(m)}(x,0,t)}{\partial x} = \frac{\partial T_l^{(m)}(x,0,t)}{\partial x} = 0, \quad \frac{\partial T_e^{(m)}(x,y_L,t)}{\partial x} = \frac{\partial T_l^{(m)}(x,y_L,t)}{\partial x} = 0, \tag{5.10f}$$

and the interfacial condition at $x = x_l$

$$T_e^{(1)} = T_e^{(2)}, \quad k_{e0}^{(1)}\frac{T_e^{(1)}}{T_l^{(1)}}\frac{\partial T_e^{(1)}}{\partial x} = k_{e0}^{(2)}\frac{T_e^{(2)}}{T_l^{(2)}}\frac{\partial T_e^{(2)}}{\partial x}. \tag{5.10g}$$

Here, $m = 1, 2$ represents the first layer ($0 \leq x \leq x_l$, $0 \leq y \leq y_L$) and the second layer ($x_l \leq x \leq x_L$, $0 \leq y \leq y_L$), respectively.

We would like to point out that in the two dimension case, $z_i^{(1)}$ in Eq. (4.9c) changes to

$$z_i^{(1)} = \sigma(W_i^{(1,0)}x + \bar{W}_i^{(1,0)}y + W_i^{(1,1)}t + b_i^{(1,0)}), \qquad i = 1, 2, ..., M_1; \tag{5.11}$$

Also, the loss function in Eq. (4.10) will have additional loss terms from the other boundaries as shown in Eq. (5.10).

For this case, we considered a 50nm× 1000nm gold layer padding on a 50nm×1000nm chromium layer exposed to a ultrashort-pulsed laser heating where the parameters in the energy absorption rate were chosen to be $J = 500$ (J/m^2), $R = 0.93$, $t_p = 100$ (fs), $\delta = 15.3$ (nm) [80], $T_0 = 300$ (K), and $\omega = 10^{-6}$ (m) [171]. We compared our ANN solutions with those obtained in [80], where the time interval is $0 \leq t \leq 6.2$ (ps).

Again we first divided the time interval into four subintervals as: (i) $0 \leq t \leq 0.7$, (ii) $0.7 \leq t \leq 1.2$, (iii) $1.2 \leq t \leq 2.2$ and (iv) $2.2 \leq t \leq 6.2$. For $0 \leq t \leq 0.7$, we employed 5 hidden layers with 100 units each for the neural nets in Figure 4.2 . We discretized the dimensionless intervals $0 \leq t \leq 1$, $0 \leq x \leq 1$ and $0 \leq y \leq 10$ into 500, 1200 and 500 grid points, respectively, and chose randomly 400 grid points as initial, boundary, and interface training points for Eqs. (5.10d)-(5.10g) from the discretized set of points. Then we selected 35,000 computer generated random points (using latin hyper cube sampling technique) as training points for the PDEs in Eq. (5.10a)-(5.10b) in each layer within the dimensionless domain of $0 \leq t \leq 1$, $0 \leq x \leq 1$ and $0 \leq y \leq 10$. In our algorithm, we set the iteration number $N_1 = 100000$ for Adam Optimization method with learning rate of 0.0001 and $N_2 = 50000$ for L-BFGS method. Based on the algorithm, we obtained the final value of the loss function to be 5.89723×10^{-6} for $t = 0.3$ and $t = 0.7$ (ps). The training time for this time interval was 21584.0974.

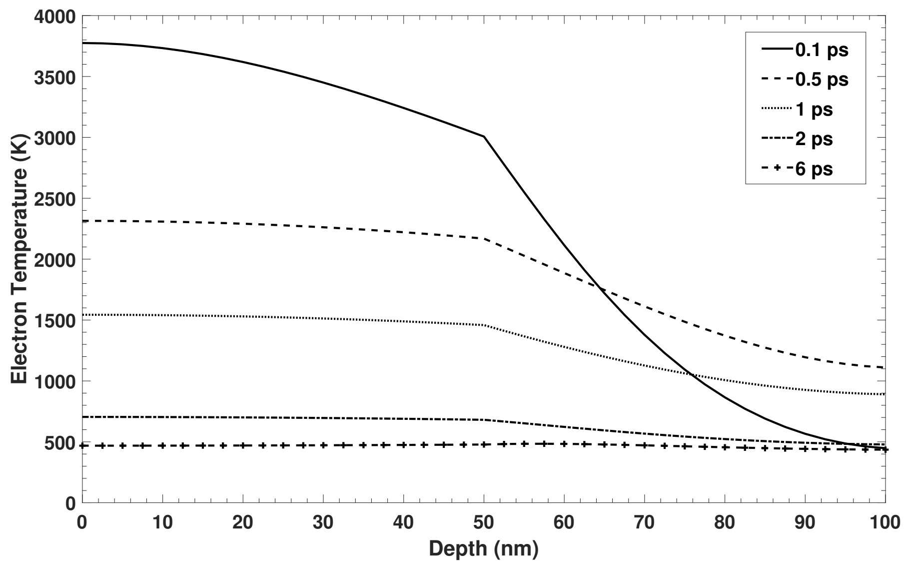
For $0.7 \leq t \leq 1.2$, we kept the same neural net structure and discretized t , x and y into 100, 600 and 400 grid points with selection of 150 random grid points as initial, boundary, and interface training points for Eqs. (5.10d)-(5.10g) from the discretized set of points. We further selected 10000 computer generated random points (using latin hyper cube sampling technique) as training points for the PDEs in Eq. (5.10a)-(5.10b) in each layer within the dimensionless domain of $0 \leq t \leq 1$, $0 \leq x \leq 1$ and $0 \leq y \leq 10$. In our algorithm, we set the iteration number $N_1 = N_2 = 50000$ for both Adam optimization and L-BFGS method. Based on the algorithm, we obtained the loss function value to be 2.5809063×10^{-6}. The training time for this time interval was 14599.0725 seconds.

For the rest of the time subintervals, we kept the same configuration and number of training points used for $0.7 \leq t \leq 1.2$. Based on the algorithm, we obtained the values of loss function to be 2.5809063×10^{-5} and 6.2730214×10^{-5}, respectively. The training time for all these time intervals was quite close to the previous time interval.

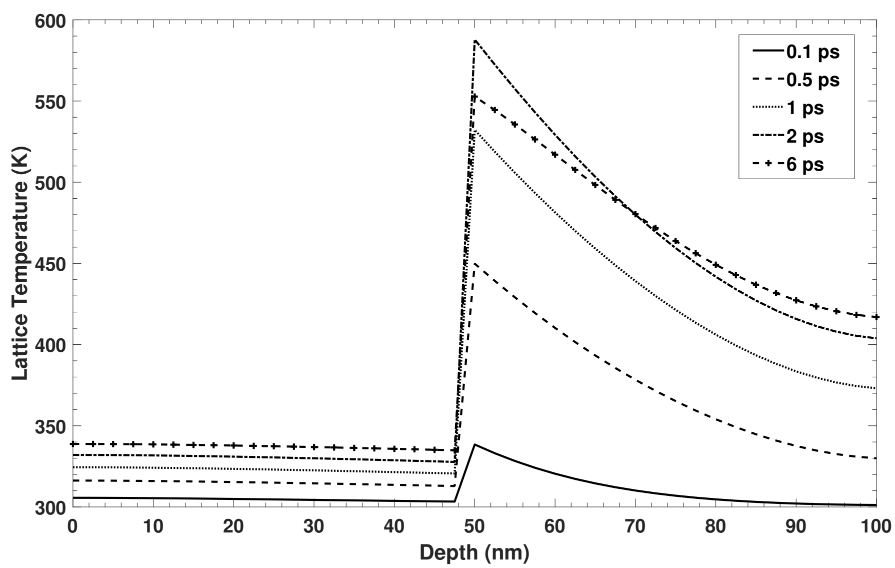
Figure 5.4 shows the electron temperature profiles and lattice temperature profiles at the cross-section $y = 500$ (nm). One may see from the figure that the result agrees well with that in the nonlinear case and that obtained in FIG. 6 obtained in [80]. Figure 5.5 and Figure 5.6 display contours of electron temperature distributions and lattice temperature distribution at various times $t = 0.1, 0.5, 1.0, 2.0, 6.0$ (ps), respectively.

5.3 Summary

In this chapter, we have applied the Gradient Preserved Method and Artificial Neural Network Method to solve the well-known parabolic two-temperature heat conduction model and predict the electron and lattice temperatures of a 50-nm gold film padding on a 50-nm chromium film, which is exposed to the ultrashort-pulsed laser heating. We have then compared both the results with each other and also with the benchmark results. Results show that they agree well with each other.

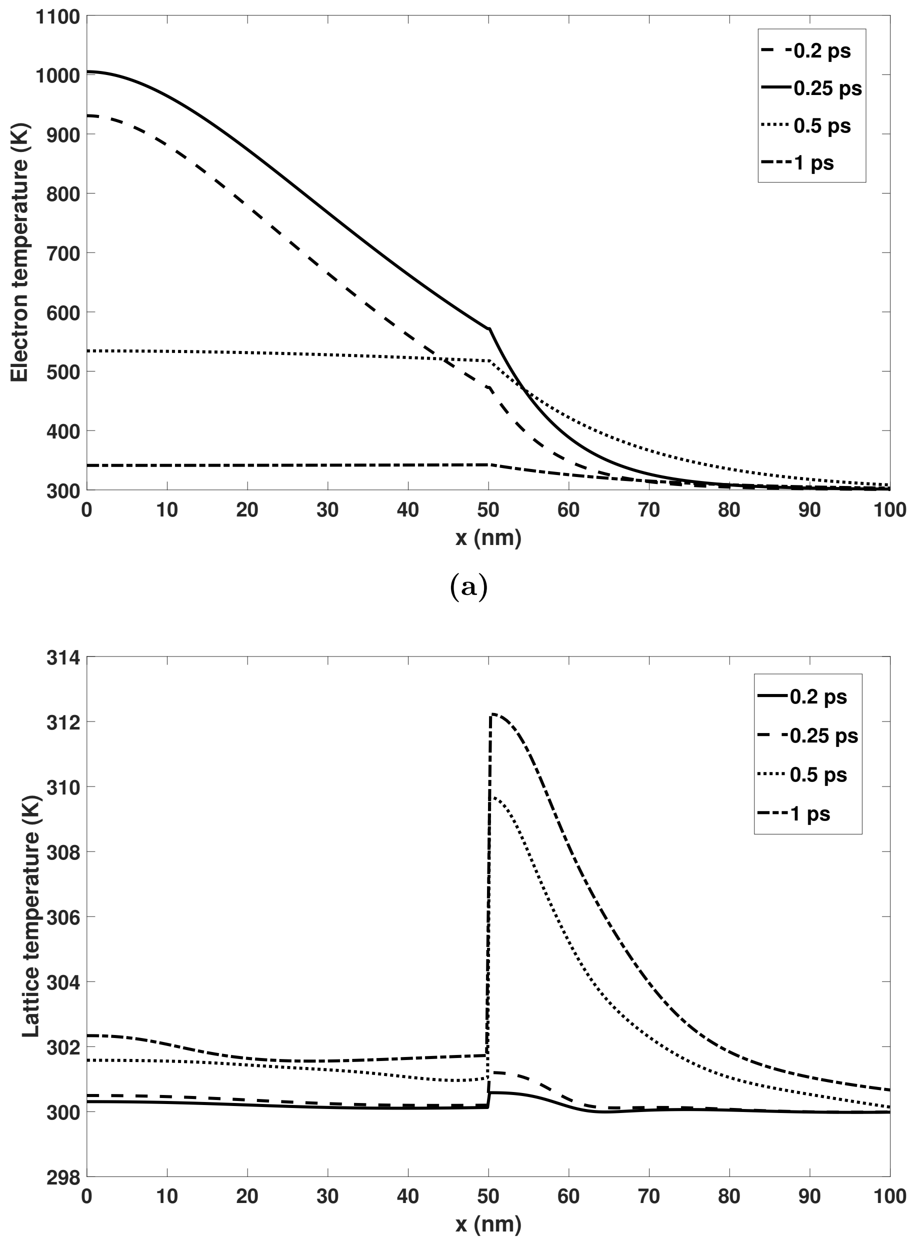


(a)



(b)

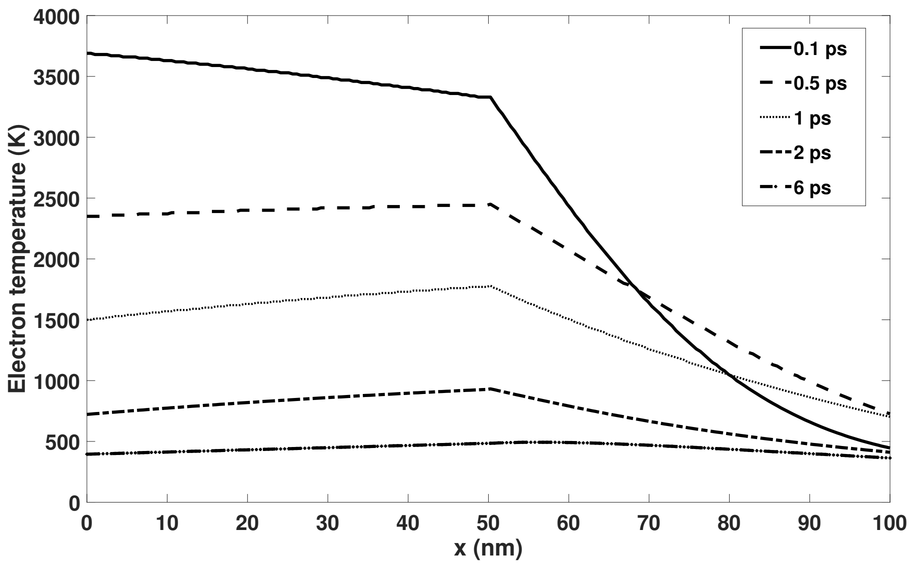
Figure 5.1: (a) Electron temperature and (b) lattice temperature profiles at various times in a 50-nm gold/ 50-nm chromium two-layer film, with interface at 50-nm, during 0.1 (ps) ultrashort-pulsed laser heating at a fluence of 500 Jm^{-2} .



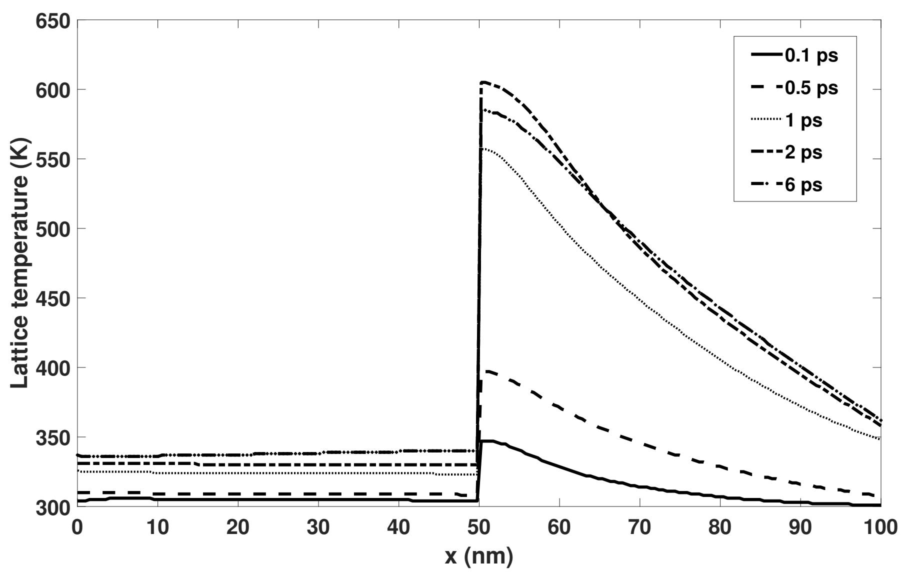
(a)

(b)

Figure 5.2: (a) Electron temperature and (b) lattice temperature profiles at various times in a 50-nm gold/ 50-nm chromium two-layer film, with interface at 50-nm, during 0.1 (ps) ultrashort-pulsed laser heating at a fluence of 13.4 Jm^{-2}.

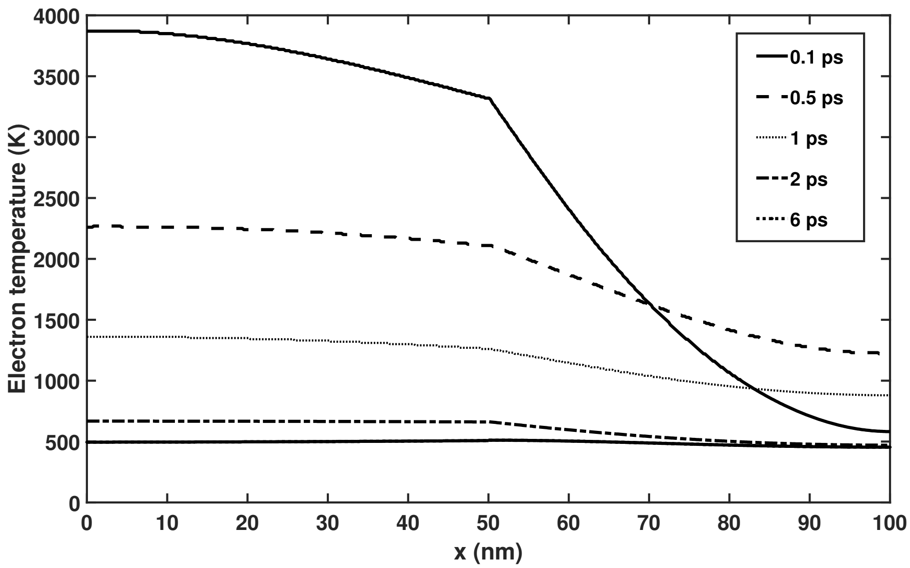


(a)

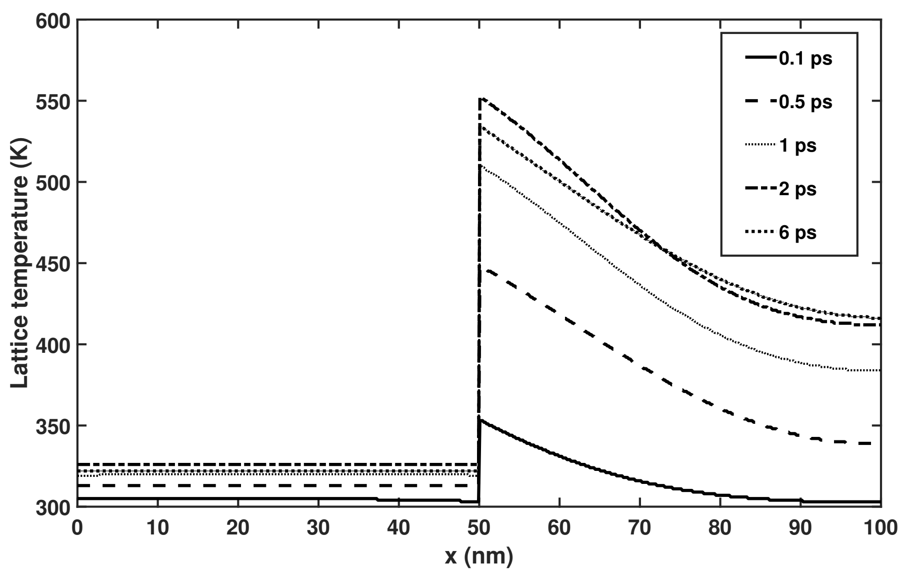


(b)

Figure 5.3: (a) Electron temperature and (b) lattice temperature profiles at various times in a 50-nm gold/ 50-nm chromium two-layer film, with interface at 50-nm, during 0.1 (ps) ultrashort-pulsed laser heating at a fluence of 500 Jm^{-2} .

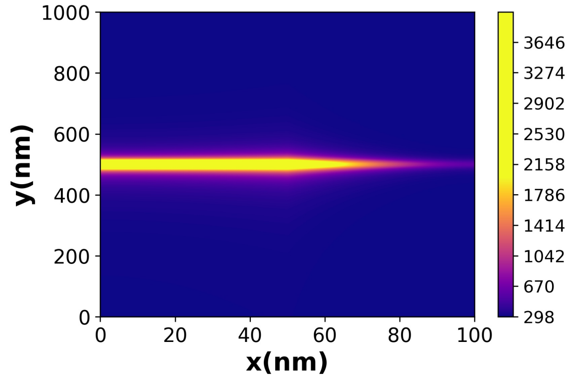


(a)

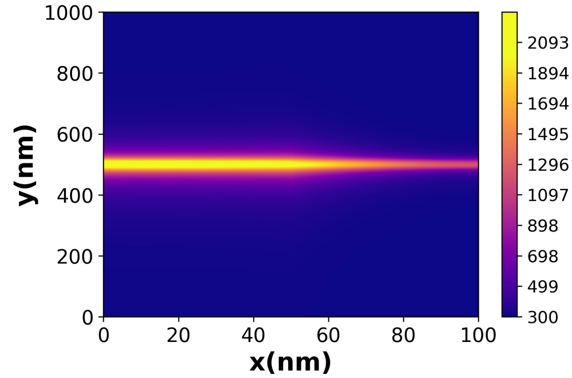


(b)

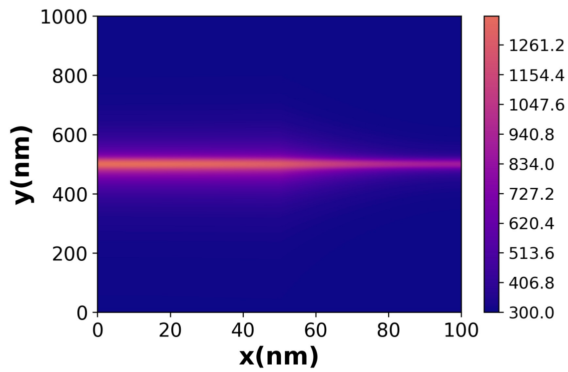
Figure 5.4: (a) Electron temperature and (b) lattice temperature profiles at various times in a 1000-nm wide, 50-nm gold/50-nm chromium two-layer film, with interface at 50-nm along the x-axis, during 0.1 (ps) ultrashort-pulsed laser heating at a fluence of 500 Jm^{-2} at the cross section y= 500(nm).



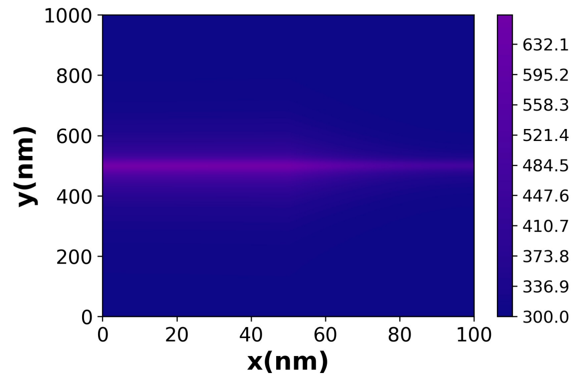
(a) t = 0.1 ps



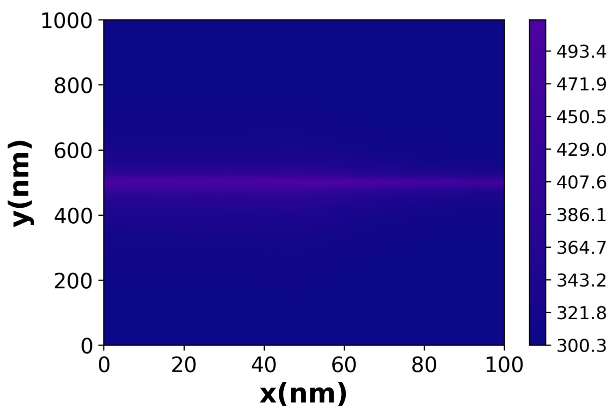
(b) t= 0.5 ps



(c) t = 1 ps

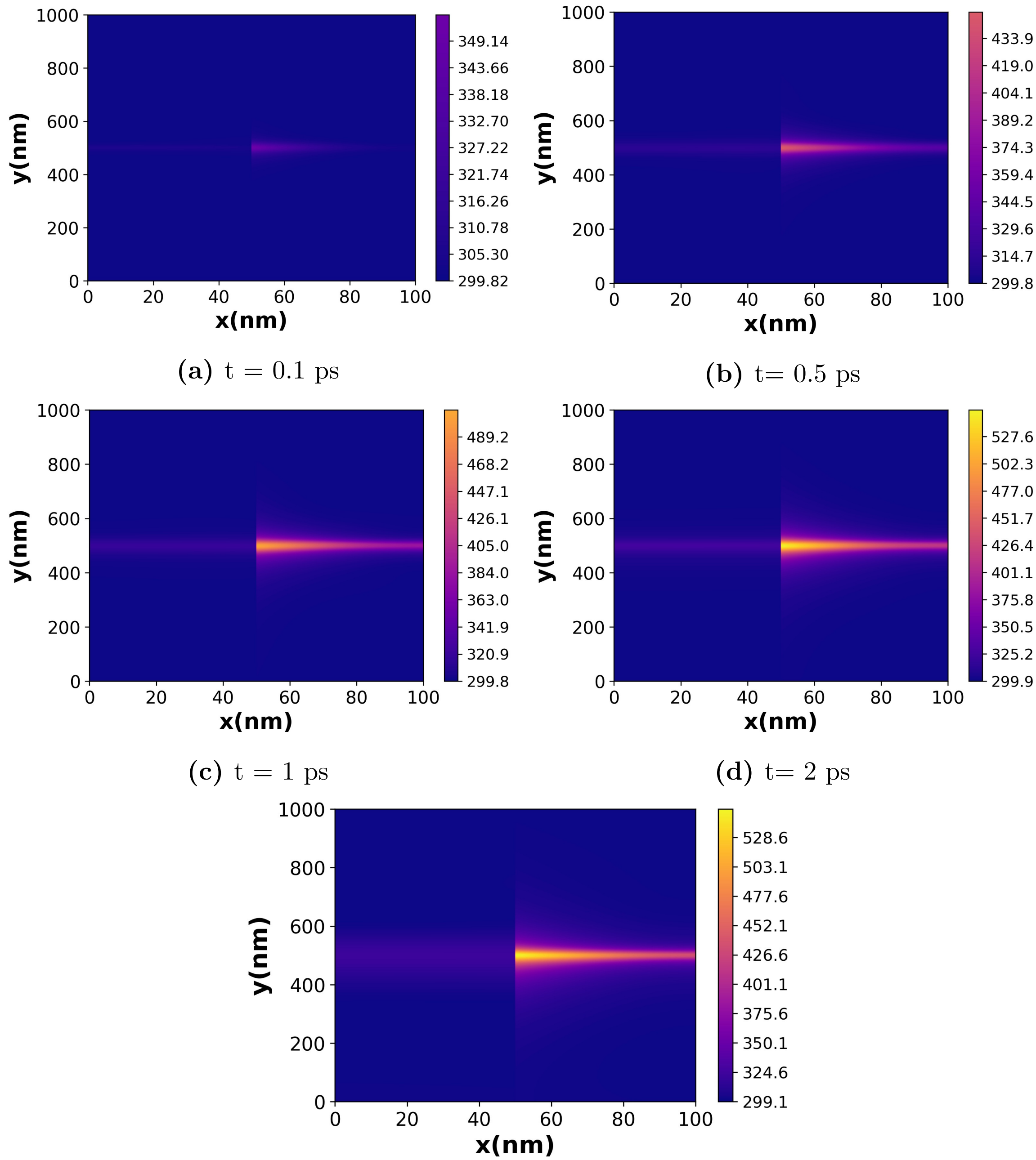


(d) t= 2 ps



(e) t= 6 ps

Figure 5.5: Contours of electron temperature distributions at various times in a 1000 nm wide, 50-nm gold/ 50-nm chromium two-layered thin film, with interface at x = 50-nm, during 0.1 (ps) ultrashort-pulsed laser heating at a fluence of 500 Jm^{-2} .



(a) t = 0.1 ps

(b) t= 0.5 ps

(c) t = 1 ps

(d) t= 2 ps

(e) t= 6 ps

Figure 5.6: Contours of lattice temperature distributions at various times in a 1000 nm wide, 50-nm gold/ 50-nm chromium two-layered thin film, with interface at $x =$ 50-nm, during 0.1 (ps) ultrashort-pulsed laser heating at a fluence of 500 Jm^{-2} .

CHAPTER 6

CONCLUSIONS

In this dissertation, we have developed two computational methods for solving heat conduction in double layers. In the first method, we have presented an accurate compact finite difference scheme for solving the heat conduction equation with spatially variable coefficients in double layers. The scheme is obtained based on the three-grid point Compact Finite Difference Method. While deriving the scheme, we have preserved the derivative term on the boundary and interface and hence we call it the Gradient Preserved Method. We have then shown by the Discrete Energy Method that the scheme is unconditionally stable and fourth order accurate in space and second order accurate in time. The method is tested by an example to verify the convergence order and accuracy. This method is extended to deal with the temaperature-dependent coefficient case, which is shown with the applicability of the present method in predicting the temperature profiles when a gold-chromium micro-scale double layer is exposed to an ultrashort pulsed laser heating. We compared the result with the benchmark in[80] and showed that with our method we can achieve the same result using 1/10th of the grid-points used in that benchmark.

In the second method, we have presented an Artificial Neural Network (ANN) method and its algorithm for solving the parabolic two-temperature heat conduction equations in double-layered thin films exposed to ultrashort-pulsed lasers. Convergence of the ANN solution to the analytical solution has been theoretically analyzed. We have tested the ANN method and its algorithm in three cases for predicting the electron and lattice temperatures in a gold layer padding on a chromium layer exposed to ultrashort-pulsed lasers. Results show that the present ANN method is promising. This present ANN method and its algorithm can be easily extended to three-dimensional cases or to deal with the parabolic N-temperature model where there are N energy carriers in the materials such as cells exposed to ultrashort-pulsed lasers [192, 193, 194, 195]. It should be pointed out that when N is large, solving the N-temperature model using the common numerical methods will be very tedious. However, the present ANN method can solve the N-temperature model more effectively with the aid of GPU computing. Also, this method shows in general the way to capture very high shock values using neural networks.

In the future, the research will focus on thermal analysis in three-dimensional multi-layered thin films, and/or more complicated geometric materials, as well as other models related to the ultrashort-pulsed laser heating, especially when the mean free path of the electron is larger than the length of the material. Also, further research will be directed towards: (i) making the ANN method faster and (ii) combining the numerical method and the neural network method to come up with a hybrid technique.

BIBLIOGRAPHY

[1] A. Abbassi, J. Ghazanfarian, Z. Shomali, Macro to nanoscale heat transfer: the lagging behaviour, Int. J. Thermophys. 36 (2015) 1416-1467.

[2] C. Kuo, J. Ho, W. Jiaung, Study of heat transfer in multilayered structure within the framework of dual-phase-lag heat conduction model using lattice Boltzmann method, Int. J. Heat Mass Trans. 43 (2003) 55-69.

[3] K. Liu, Analysis of dual-phase-lag thermal behavior in layered films with temperature-dependent interface thermal resistance, J. Phys. D: Applied Physics 38 (2005) 3722-3732.

[4] M. Shen, P. Keblinski, Ballistic vs. diffusive heat transfer across nanoscopic films of layered crystals, J. Appl. Phys. 115 (2014) 4870940.

[5] M. Pillers, M. Lieberman, Rapid thermal processing of dna origami on silicon creates embedded silicon carbide replicas, 13th Annual Conference on Foundations of Nanoscience, Snowbird, Utah, 11-16 April (2016).

[6] T. Tsai, Y. Lee, Analysis of microscale heat transfer and ultrafast thermoelasticity in a multi-layered metal film with nonlinear thermal boundary resistance, Int. J. Heat Mass Trans. 62 (2013) 87-98.

[7] S. Sadasivam, T.S. Fisher, U.V. Waghmare, Electron-phonon coupling and thermal conductance at metal-semiconductor interface: first-principles analysis, J. Appl. Phys. 117 (2015) 4916729.

[8] D.Y. Tzou, Macro-to Microscale Heat Transfer: The Lagging Behavior. Taylor & Francis, Washington DC (1997).

[9] Y. Mao, M. Xu, Lattice Boltzmann numerical analysis of heat transfer in nano-scale silicon films induced by ultra-fast laser heating, Int. J. Therm. Sci. 89 (2015) 210-221.

[10] G. Udupas, K.V. Gangadharan, S. Rao, Functionally graded composite materials: an overview, Procedia Materials Science 5 (2014) 1291-1299.

[11] L.W. Byrd, V. Birman, Modeling and analysis of functionally graded materials and structures, Appl. Mech. Rev. 60 (2007) 195-216.

[12] M.J. Pinder, Y. Zhong, Efficient reformulation of HOTFGM: heat conduction with variable thermal conductivity, Report No. NASA/CR-211910, 2002.

[13] B.E. Griffith, C.S. Peskin, On the order of accuracy of the immersed boundary method: higher order convergence rates for sufficiently smooth problems, J. Comput. Phys. 208 (2005) 75-105.

[14] C.S. Peskin, M. Lai, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2000) 705-719.

[15] C.S. Peskin, Numerical analysis of blood flow in heart, J. Comput. Phys. 25 (1997) 220-252.

[16] C.S. Peskin, Lectures on mathematical aspects of physiology. Lectures in Applied Mathematics 19 (1981) 69-109.

[17] C.S. Peskin, D.M. McQueen, A 3-dimensional computational method for blood-flow in the heart I. Immersed elastic fibers in a viscous incompressible fluid, J. Comput. Phys. 81 (1989) 372-405.

[18] C.S. Peskin, B.F. Printz, Improved volume conservation in the computation of flows with immersed elastic boundaries, J. Comput. Phys. 105 (1993) 33-46.

[19] E.A. Fadlun, J.M. Yusof, P. Orlandi, R. Verzicco, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 30-60 .

[20] L. Adams, Z. Li, The immersed interface/multigrid methods for interface problems, SIAM J. Sci. Comput. 24 (2002) 463-479.

[21] K. Ito, S. Deng, Z. Li, Three-dimensional elliptic solvers for interface problems and applications, J. Comput. Phys. 184 (2003) 215-243.

[22] A. Wiegmann, J.A. Sethian, Structural boundary design via level set and immersed interface methods, J. Comput. Phys. 163 (2000) 489-528.

[23] Leveque, R.J., Li, Z., The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (1994) 1019-1044.

[24] K. Ito, Z. Li, Maximum principle preserving schemes for interface problems with discontinuous coefficients, SIAM J. Sci. Comput. 23 (2001) 339-361.

[25] Z. Li, An overview of the immersed interface method and its application, Taiwanese Journal of Mathematics 7 (2003) 1-49.

[26] Z. Li, A fast iterative algorithm for elliptic interface problems, SIAM J. Numer. Anal. 35 (1998) 230-254.

[27] A. Wiegmann, K.P. Bube, The explicit-jump immersed interface method: Finite difference methods for pdes with piecewise smooth solutions. SIAM J. Numer. Anal. 37 (2000) 827-862.

[28] Z. Li, The immersed interface method using a finite element formulation, Applied Numerical Mathematics 27 (1998) 253-267.

[29] H. Huang, Z. Li, Convergence analysis of the immersed interface method, IMA J. Numer. Anal. 19 (1999) 583-608.

[30] J.D. Kandilarov, Immersed interface method for a reaction-diffusion equation with a moving own concentrated source, Lecture Notes in Computer Science 2542 (2003) 506-513.

[31] K. Ito, Z. Li, The Immeresed Interface Method, SIAM Frontiers in Applied Mathematics, Philadelphia (2006).

[32] B. Merriman, R.P. Fedkiw, S. Osher, T. Aslam, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method), J. Comput. Phys. 152 (1999) 457-492.

[33] M. Kang, R.P. Fedkiw, X. Liu, A boundary condition capturing method for Poisson equation on irregular domains, J. Comput. Phys. 160 (2000) 151-178.

[34] M. Kang, R.P. Fedkiw, X. Liu, A boundary condition capturing method for multiphase incompressible flow, J. Sci. Comput. 15 (2000) 323-360.

[35] G. Wei, S. Yu, Y. Zhou, Matched interface and boundary (MIB) method for elliptic problems with sharp-edged interfaces, J. Comput. Phys. 224 (2007) 729-756.

[36] G. Wei, S. Yu, Three-dimensional method interface and boundary (MIB) method for treating geometric singularities, J. Comput. Phys. 227 (2007) 602-623.

[37] G. Wei, S. Zhao, High order FDTD methods via derivative matching for maxwell equations with material interfaces, J. Comput. Phys. 200 (2004) 60-103.

[38] G. Wei, M. Feig, S. Zhao, Y. Zhou, High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources, J. Comput. Phys. 213 (2006) 1-30.

[39] G. Wei, Y. Zhou, On the fictitious-domain and interpolation formulations of the matched interface and boundary method, J. Comput. Phys. 219 (2006) 228-246.

[40] G. Wei, S. Zhao, Matched interface and boundary (MIB) for the implementation of boundary conditions in high-order central finite differences, Int. J. Numer. Methods Eng. 77 (2009) 1690-1730.

[41] H. Hu, K. Pan, Y. Tan, An interpolation matched interface and boundary method for elliptic interface problems, J. Comput. Appl. Math. 234 (2010) 73-94.

[42] G. Wei, Y. Zhou, On the fictitious-domain and interpolation formulations of the matched interface and boundary (MIB) method, J. Comput. Phys. 219 (2016) 228-246 .

[43] I. Babuska, The finite element method for elliptic equations with discontinuous coefficients, Computing 5 (1970) 207-213.

[44] J. Zou, Z. Chen, Finite element methods and their convergence for elliptic and parabolic interface problems, Numerische Mathematik 79 (1998) 175-202.

[45] J.H. Bramble, J.T. King, A finite element method for interface problems in domains with smooth boundaries and interfaces, Advances in Computational Mathematics 6 (1996) 109-138.

[46] T. Lin, X. Wu, , Z. Li, New Cartesian grid methods for interface problems using the finite element formulation, Numerische Mathematik 96 (2003) 61-98.

[47] C. Bajaj, D. Farrell, H. Hsua, J. Hong, J. Lee, W.K. Liu, , L. Zhang, N. Patankar, X. Chen, X.S. Wang, Y. Fukui, Y. Liu, Y. Zhang, Immersed finite element method and its applications to biological systems, Computer Methods in Applied Mechanics and Engineering 195 (2006) 1722-1749.

[48] B. Deka, R.K. Sinha, On the convergence of finite element method for second order elliptic interface problems. Numer. Funct. Anal. Opt. 27 (2006) 99-115.

[49] J. Huang, J. Zou, A mortar element method for elliptic problems with discontinuous coefficients, IMA J. Numer. Anal. 22 (2002) 549-576 .

[50] S. Hou, X. Liu, A numerical method for solving variable coefficient elliptic equation with interfaces, J. Comput. Phys. 202 (2005) 411-445.

[51] S. Hou, P. Song, H-K. Zhao, L. Wang, A weak formulation for solving elliptic interface problems without body fitted grid, J. Comput. Phys. 249 (2013) 80-95.

[52] L. Wang, S. Hou, L. Shi, An improved non-traditional finite element formulation for solving the elliptic interface problems, J. Comput. Math. 32 (2014) 39-57.

[53] L. Wang, S. Hou, L. Shi, A weak formulation for solving elliptic interface problems with imperfect contact, Appl. Math. Mech. 9 (2017) 1189-1205.

[54] W. Wang, A jump condition capturing finite difference scheme for elliptic interface problems, SIAM J. Sci. Comput. 25 (2004) 1479-1496.

[55] I. Chern, M. Lai, W. Wang, Z. Li, New formulations for interface problems in polar coordinates, SIAM J. Sci. Comput. 25 (2003) 224-245.

[56] B. Strand, Summation by parts for finite difference approximations for d/dx, J. Comput. Phys. 110 (1994) 47-67.

[57] J. Nordström, K. Mattsson, Summation by parts operators for finite difference approximations of second derivatives, J. Comput. Phys. 199 (2004) 503-540.

[58] B. Gustafsson, High order difference methods for time dependent PDE, Springer, Berlin, 2008.

[59] D.C.D.R. Fernández, D. Zingg, J. Hicken, Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations, Computers & Fluids 95 (2014) 171-196.

[60] J. Nordström, M. Svärd, Review of summation-by-parts schemes for initial–boundary-value problems, J. Comput. Phys. 268 (2014) 17-38.

[61] L. Wu, W. Dai, S. Zhai, Y. Yan, Accurate gradient preserved method for solving heat conduction equations in double layers, Appl. Math. Comput. 354 (2019) 58-85.

[62] S.K. Lele, Compact finite difference schemes with spectral-like resolution, J. Comput. Phys. 103 (1992) 16-42.

[63] Z.Z. Sun, W. Dai, A new higher-order accurate numerical method for solving heat conduction in a double-layered film with the neumann boundary condition, Numer Methods Partial Differ Equ. (2014) 1291–1314.

[64] A. Elshorbagy, G. Corzo, S. Srinivasulu, D. P. Solomatine, Experimental investigation of the predictive capabilities of data driven modeling techniques in hydrology - Part 1: Concepts and methodology, Hydrology and Earth System Sciences 14 (2010) 1931-1941.

[65] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems 25 (2012) 1097-1105.

[66] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2 (1989) 359-366.

[67] H. Owhadi, Bayesian numerical homogenization, Multiscale Modelling and Simulation 13 (2015) 812-828.

[68] M. Raissi, P. Perdikaris, G.E. Karniadakis, Inferring solutions of differential equations using noisy multi-fidelity data, J. of Comput. Phy. 335 (2017) 736-746.

[69] M. Raissi, P. Perdikaris, G.E. Karniadakis, Machine learning of linear differential equations using Gaussian processes, J. of Comput. Phy. 348 (2017) 683-693.

[70] M. Raissi, P. Perdikaris, G.E. Karniadakis, Numerical Gaussian processes for time-dependent and non-linear partial differential equations. (2017). arXiv:1703.10230.

[71] M. Raissi, G.E. Karniadakis, Hidden physics models: machine learning of nonlinear partial differential equations (2017). arXiv:1708.00588.

[72] C.E. Rasmussen, C.K. Williams, Gaussian Processes for Machine Learning, vol. 1, MIT Press, Cambridge (2006).

[73] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. of Comput. Phy. 378 (2019) 686-707.

[74] J.K. Chen, J.E. Beraun, Numerical study of ultrashort laser pulse interactions with metal films, Numerical Heat Transfer, Part A 40 (2001) 1-20.

[75] Jean-Baptiste-Joseph Fourier, Théorie analytique de la chaleur, Paris : F. Didot (1822).

[76] J. H. Leinhard IV, J. H. Leinhard V, A heat transfer textbook, Dover Publications 4 (2013).

[77] R. B. Guenther, J. W. Lee, Partial differential equations of mathematical physics and integral equations, Dover Publications (1996).

[78] T. Myint-U, L. Debnath, Linear partial differential equations for scientists and engineers, Birkhäuser, 4 (2006).

[79] Y. Zhong, M. Pindera, S. Arnold, Efficient reformulation of HOTFGM: heat conduction with variable thermal conductivity, NASA/CR–2002–211909 (2002).

[80] T. Qui, C. Tien, Femtosecond laser heating of multi-layer metals-1.Analysis, Int. J. Mass Transfer 37 (1994) 2789-2797.

[81] E. E. Joseph and L. Preziosi, Heat waves, Rev. Mod. Phys. 61 (1989) 41-73.

[82] W. S. Kim, L. G. Hector and M. N. Ozisik, Hyperbolic heat conduction due to axisymmetric continuous or pulsed surface heat sources, J. Appl. Phys. 6S,5478-5485 (1990).

[83] A. Kar, C. L. Chan and J. Mazumder, Comparative studies on nonlinear hyperbolic and parabolic heat conduction for various boundary conditions : analytic and numerical solutions, J. Heat Transfer 114 (1992) 14-20 .

[84] D. Y. Tzou, Thermal shock phenomena under high rate response in solids. In Annual Review of Heat Transfer (Edited by C. L. Tien), Vol. IV, pp. 11 l-186. Hemisphere Publishing, Washington (1992).

[85] L. G. Hector, W. S. Kim and M. N. Ozisik, Hyperbolic heat conduction due to a mode locked laser pulse train, bat. J. Engng Sci. 30 (1992) 1731-1744 .

[86] R. J. von Gutfeld and A. J. Netherot, Heat pulses in quartz and sapphire at low temperatures, Phys. Rev. Lett. 12, (1964) 641-644.

[87] W. E. Bron, Y. B. Levinson and J. M. O'Connor, Phonon propagation by quasidiffusion, Phys. Rev. Lett. 49, (1982).

[88] W. E. Bron, Phonon generation, transport and detection through electronic states in solids. In Non equilibrium Phonons in Nonmetallic Crystals (Edited by W. Eisenmenger and A. A. Kaplyanskii), Elsevier (1986) 227-273.

[89] W. E. Bron, Phonon transport-experiment. In Dynamical Properties ofSolids (Edited by G. K. Horton and A. A. Maradudin), Elsevier (1990) l-64.

[90] T. I. Galkina, Generation and propagation of phonons in crystalline and amorphous silicon under optical excitation. In Physics of Phonons (Edited by T. Paszkiewicz), pp. 410-418. Springer-Verlag, Berlin (1987).

[91] M. I. Kaganov, I. M. Lifshitz and L. V. Tanatarov, Relaxation between electrons and crystalline lattices, Sov. Phys. JETP 4 (1957) 173-178.

[92] S. I. Anisimov, B. L. Kapeliovich and T. L. Perel'man, Electron emission from metal surfaces exposed to ultrashort laser pulses, Sov. Phys. JETP 39 (1974) 375377.

[93] G. L. Eesley, Observation of nonequilibrium electron heating in copper, Phys. Rev. Lett. 51 (1983) 2140-2143.

[94] S. D. Brorson, J. G. Fujimoto and E. P. Ippen, Femtosecond electronic heat-transfer dynamics in thin gold film, Phys. Rev. Lett. 59 (1987) 1062-196.

[95] H. E. Elsayed-Ali, T. Juhasz, G. 0. Smith and W. E. Bron, Femtosecond thermoreflectivity and thermotransmissivity of polycrystalline and single-crystalline gold films, Phys. Rev. B 43 (1991) 4488-4491.

[96] T. Juhasz, H. E. Elsayed-Ali, X. H. Hu and W. E. Bron, Time-resolved thermoreflectivity of thin gold films and its dependence on the ambient temperature, Phys. Rev. B45 (1992) 13819-13822.

[97] H. E. Elsayed-Ali and T. Jushasz, Femtosecond timeresolved thermomodulation of thin gold films with different crystal structures, Phys. Rev. B 47 13 (1993) 599-610.

[98] T. Q. Qiu and C. L. Tien, Short-pulse laser heating on metals, Int. J. Heat Mass Transfer 35 (1992) 719-726.

[99] M. J. Maurer, Relaxation model for heat conduction in metals, J. Appl. Phys. 40 5 (1969) 123-130.

[100] T. Q. Qiu and C. L. Tien, Heat transfer mechanisms during short-pulse laser heating of metals, J. Heat Transfer 115 (1994) 835-841.

[101] K. Morton, D. Mayers, Numerical solution of partial differential equations: an Introduction, Cambridge: Cambridge University Press (2005). doi:10.1017/CBO9780511812248.

[102] M. Bader, Scientific computing I, module 6: the 1D heat equation $https://www5.in.tum.de/lehre/vorlesungen/sci_comp/slides/heateq.pdf$.

[103] M. Francois, W. Shyy, Computations of drop dynamics with the immersed boundary method, part 2: drop impact and heat transfer, Numerical Heat Transfer 44 (2003) 119-143.

[104] G. Iaccarino, R. Verzicco, Immersed boundary technique for turbulent flow simulations, Applied Mechanics Reviews 56 (2003) 331-347.

[105] A. Tornberg, B. Engquist, Numerical approximations of singular source terms in differential equations, Journal of Computational Physics, 200 (2004) 462-488.

[106] T. Hou, Z. Li, S. Osher, H. Zhao, A hybrid method for moving interface problems with application to the Hele–Shaw flow, Journal of Computational Physics, 134 (1997) 236-252.

[107] J. Jin, X. Wang, Robust numerical simulation of porosity evolution in chemical vapor infiltration II. Two-dimensional anisotropic fronts, Journal of Computational Physics, 179 (2002) 557-577.

[108] S. Zhao, G. Wei, Tensor product derivative matching for wave propagation in inhomogeneous media, Microwave and Optical Technology Letters, 43 (2004) 69-77.

[109] K. Pan, Y. Tan, H. Hu, An interpolation matched interface and boundary method for elliptic interface problems, Journal of Computational and Applied Mathematics, 234 (2010) 73-94.

[110] C. David, D. Fernández, J. Hicken, D. Zingg, Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations, Computers and Fluids Elsevier, 95 (2014) 171-196.

[111] M. Svard, J. Nordström, Review of summation-by-parts schemes for initial–boundary-value problems, Journal of Computational Physics, 268 (2014) 17-38.

[112] M. Giles, Stability analysis of numerical interface conditions in fluid-structure thermal analysis, International Journal for Numerical Methods in Fluids, 25 (1997) 421-436.

[113] B. Roe, R. Jaiman, A. Haselbacher, P.H. Geubelle, Combined interface boundary condition method for coupled thermal simulations, International Journal for Numerical Methods in Fluids, 57 (2008) 329-354.

[114] W. Henshaw, K. Chand, A composite grid solver for conjugate heat transfer in fluid-structure systems, Journal of Computational Physics, 228 (2009) 3708-3741.

[115] W. Dai, Z. Sun, A new accurate numerical method for solving heat conduction in a double-layered film with the neumann boundary condition, Numer. Methods Partial Differ. Equ. 30 (2014) 1291-1314.

[116] W. Dai, H. Yu, R. Nassar, A fourth-order compact finite difference scheme for solving a 1-D Pennes' bioheat transfer equation in a triple-layered skin structure, Numerical Heat Transfer, Part B, 46 (2004) 447-461.

[117] W. Dai, R. Nassar, A hybrid finite element-finite difference method for solving three-dimensional heat transport equations in a double-layered thin film with microscale thickness, Numerical Heat Transfer, Part A, 38 (2000) 573-588.

[118] W. Dai, T. Niu, A finite difference method for solving nonlinear hyperbolic twostep model in a double-layered thin film exposed to ultrashort pulsed lasers with nonlinear interfacial conditions, Nonlinear Analysis: Hybrid Systems, 2 (2008) 121-143.

[119] T. Niu, W. Dai, A hyperbolic two-step model based finite difference scheme for studying thermal deformation in a double-layered thin film exposed to ultrashort pulsed lasers, International Journal of Thermal Sciences, 48 (2009) 34-49.

[120] H. Wang, W. Dai, L. Hewavitharana, A finite difference method for studying thermal deformation in a double-layered thin film with imperfect interfacial contact exposed to ultrashort pulsed lasers, International Journal of Thermal Sciences, 47 (2008) 7-24.

[121] H. Sun, Z. Sun, W. Dai, A second-order finite difference scheme for solving the dual-phase-lagging equation in a double-layered nanoscale thin film, Numerical Methods for Partial Differential Equations, 33 (2017) 142-173.

[122] S. Shen, W. Dai, J. Cheng, Fractional parabolic two-step model and its accurate numerical scheme for nanoscale heat conduction, Journal of Computational and Applied Mathematics, 375 (2020) 112812.

[123] H. A. Schwarz, Über ein flächen kleinsten flächeninhalts betreffendes problem der variationsrechnung, Acta Societatis Scientiarum Fennicae, XV: 318 (1888).

[124] T.H. Gronwall, Note on the derivatives with respect to a parameter of the solutions of a system of differential equations, Ann. of Math., 20 (1919) 292–296.

[125] W. Dai, R. Nassar, A compact finite-difference scheme for solving a one-dimensional heat transport equation at the microscale, Journal of Computational and Applied Mathematics, 132 (2001) 431-441.

[126] W. S. Mcculloch, W. Pitts, A logical calculus of the ideas immanent in nervous activity, Bulletien of mathematical biophysics 5 (1943) 115-133.

[127] Neural Networks, History 1940-1970, online article, Stanford University https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/History/history1.html.

[128] D. O. Hebb, The Organization of Behavior: A neuropsychological theory. New York: Wiley, (1949).

[129] K. C. Fukushima, A self-organizing multilayered neural network. Biol. Cybernetics 20 (1975) 121–136 . https://doi.org/10.1007/BF00342633.

[130] J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences of the United States of America, 79 (8) (1982) 2554-2558.

[131] Neural Networks, History 1980-present, online article, Stanford University https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/History/history2.html.

[132] D.L. Reilly, L.N. Cooper, C. Elbaum, A Neural Model for Category Learning, Biological Cybernetics, 45 (1982) 35–41.

[133] D. Rumelhart, G. Hinton, R. Williams, Learning representations by back-propagating errors. Nature 323 (1986) 533–536. https://doi.org/10.1038/323533a0.

[134] G. Pang, L. Lu, G.E. Karniadakis, fPINNs: fractional physics-informed neural networks, SIAM J. Sci. Comput. 41 (2019) A2603-A2626.

[135] L. Yang, Highly-scalable physics-informed GANs for learning solutions of stochastic PDEs, IEEE/ACM Third Workshop on Deep Learning on Supercomputers (2019) 1-11.

[136] X.I.A. Yang, S. Zafar, J.-X. Wang, H. Xiao, Predictive large-eddy-simulation wall modeling via physics-informed neural networks, Phys. Rev. Fluids 4 (2019) 034602.

[137] G.P.P. Pun, R. Batra, R. Ramprasad, Physically informed artificial neural networks for atomistic modeling of materials, Nat. Commun. 10 (2019) 2339.

[138] Y. Chen, L. Lu, G.E. Karniadakis, L.D. Negro, Physics-informed neural networks for inverse problems in nano-optics and metamaterials, Opt. Express 28 (2020) 11618-11633.

[139] T. Kadeethum, TM. Jørgensen, H.M. Nick, Physics-informed neural networks for solving nonlinear diffusivity and Biot's equations, PLOS ONE 15 (2020) 0232683.

[140] S. Goswami, C. Anitescu, S. Chakraborty, T. Rabczuk. Transfer learning enhanced physics informed neural network for phase-field modeling of fracture, Theoretical and Applied Fracture Mechanics 106 (2020) 102447.

[141] Q. He, D.B. Solano, G. Tartakovsky, A.M. Tartakovsky, Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport, Advances in Water Resources 141 (2020) 103610.

[142] Z. Fang, J. Zhan, A physics-informed neural network framework for PDEs on 3D surfaces: time independent problems, IEEE Access 8 (2020) 26328-26335.

[143] G. Kissas, Y. Yang, E. Hwuang, W.R. Witschey, J.A. Detre, P. Perdikaris, Machine learning in cardiovascular flows modeling: predicting arterial blood pressure from noninvasive 4D flow MRI data using physics-informed neural networks, Comput. Methods Appl. Mech. Eng. 358 (2020) 112623.

[144] A.M. Tartakovsky, C.O.Marrero, P. Perdikaris, G.D. Tartakovsky, D.B. Solano, Physicsinformed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems, Water Resources Research 56 (2020) 026731.

[145] C. Rao, H. Sun, Y. Liu, Physics-informed deep learning for incompressible laminar flows, Theoretical and Applied Mechanics Letters 10 (2020) 207-212.

[146] V. Dwivedi, B. Srinivasan, Physics informed extreme learning machine (PIELM) a rapid method for the numerical solution of partial differential equations, Neurocomputing 391 (2020) 96-118.

[147] H. Wessels, C. Weißenfels, P. Wriggers, The neural particle method - an updated Lagrangian physics informed neural network for computational fluid dynamics, Comput. Methods Appl. Mech. Eng. 368 (2020) 113127.

[148] A. Ghaderi, V. Morovati, R. Dargazany, A physics-informed assembly of feed-forward neural network engines to predict inelasticity in cross-linked polymers, Polymers 12 (2020) 2628.

[149] R. Zhang, Y. Liu, H. Sun, Physics-informed multi-LSTM networks for metamodeling of nonlinear structures, Comput. Methods Appl. Mech. Eng. 369 (2020) 113226.

[150] K. Champion, P. Zheng, A.Y. Aravkin, S.L. Brunton, J.N. Kutz, A unified sparse optimization framework to learn parsimonious physics-informedmodels from data, IEEE Access 8 (2020) 169259-169271.

[151] A. Li, R. Chen, A.B. Farimani, Reaction diffusion system prediction based on convolutional neural network, Sci. Rep. 10 (2020) 3894.

[152] S. Pan, K. Duraisamy, Physics-informed probabilistic learning of linear embeddings of nonlinear dynamics with guaranteed stability, SIAM J. Appl. Dyn. Syst. 19 (2020) 480-509.

[153] A.T. Mohan, D. Tretiak, M. Chertkov, D. Livescu, Spatio-temporal deep learning models of 3D turbulence with physics informed diagnostics, Journal of Turbulence 21 (2020) 484-524.

[154] X. Zhao, K. Shirvan, R. K. Salko, F. Guo, On the prediction of critical heat flux using a physics-informed machine learning-aided framework, Appl. Therm. Eng. 164 (2020) 114540.

[155] E. Qian, B. Kramer, B. Peherstorfer, K. Willcox, Lift and Learn: Physics-informed machine learning for large-scale nonlinear dynamical systems, Physica D: Nonlinear Phenomena 406 (2020) 132401.

[156] L. Lu, X. Meng, Z. Mao, G.E. Karniadakis, DeepXDE: A deep learning library for solving differential equations, SIAM Review 63 (2021) 208-228.

[157] H. Haghighat, R. Juanes, SciANN: A Keras/TensorFlow wrapper for scientific computations and physics-informed deep learning using artificial neural networks, Comput. Methods Appl. Mech. Eng. 373 (2021) 113552.

[158] C.L.Wight, J. Zhao, Solving Allen-Cahn and Cahn-Hilliard equations using the adaptive physics informed neural networks, Commun. Comput. Phys. 29 (2021) 930-954.

[159] H. Gao, L. Sun, J.X. Wang, PhyGeoNet: Physics-informed geometry-adaptive convolutional neural networks for solving parameterized steady-state PDEs on irregular domain, J. Comput. Phys. 428 (2021) 110079.

[160] C. Song, T. Alkhalifah, U. B.Waheed, Solving the frequency-domain acoustic VTI wave equation using physics-informed neural networks, Geophysical Journal International 225 (2021) 846-859.

[161] Q. Zhu, Z. Liu, J. Yan, Machine learning for metal additive manufacturing: predicting temperature and melt pool fluid dynamics using physics-informed neural networks, Comput. Mech. 67 (2021) 619-635.

[162] C. Cheng, G.T. Zhang, Deep learning method based on physics informed neural network with resnet block for solving fluid flow problems, Water 13 (2021) 3040423.

[163] X. Lei, Z. Yang, J. Yu, J. Zhao, Q. Gao, H. Yu, Data-driven optimal power flow: a physics-informed machine learning approach, IEEE Transactions on Power Systems 36 (2021) 346-354.

[164] Lu, L., Jin, P., Pang, G. et al. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. Nat Mach Intell 3 (2021) 218–229. https://doi.org/10.1038/s42256-021-00302-5

[165] Z.Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, A. Anandkumar, Neural operator: graph kernel network for partial differential equations, (2020) arXiv:2003.03485.

[166] Z.Li, N. Kovachki, K. Azizzadenesheli, B. Liu, A. Stuart, K. Bhattacharya, A. Anandkumar, Multipole Graph Neural Operator for Parametric Partial Differential Equations (2020) arXiv:2006.09535.

[167] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, A. Anandkumar, Fourier neural operator for parametric partial differential equations, (2021) arXiv:2010.08895.

[168] D. Kingma, J. Ba, Adam: A method for stochastic optimization, 3rd International Conference on Learning Representations (2015). http://arxiv.org/abs/1412.6980

[169] D.C. Liu, J. Nocedal, On the limited memory BFGS method for large scale optimization, Mathematical Programming 45 (1989) 503–528.

[170] W. Dai, Q. Li, R. Nassar, L. Shen, An unconditionally stable three level finite difference scheme for solving parabolic two-step micro heat transport equations in a three-dimensional double-layered thin film, Int. J. for Numerical Methods in Engineering 59 (2004) 493-509.

[171] H. Wang, W. Dai, L. G. Hewavitharana, A finite difference method for studying thermal deformation in a double-layered thin film with imperfect interfacial contact exposed to ultrashort pulsed lasers, Int. J. Therm. Sci. 47 (2008) 7-24.

[172] A. Bora, W. Dai, Gradient preserved method for solving heat conduction equation with variable coefficients in double layers, Applied Mathematics and Computation 386 (2020) 125516.

[173] M.A. Al-Nimr, O.M. Haddad, V.S. Arpaci, Thermal behavior of metal films - a hyperbolic two-step model, Heat and Mass Transfer 35 (1999) 459-464.

[174] M.A. Al-Nimr, M. Hader, M. Naji, Use of the microscopic parabolic heat conduction model in place of the macroscopic model validation criterion under harmonic boundary heating, Int. J. Heat Mass Transfer 46 (2003) 333-339.

[175] M.A. Al-Nimr, S. Kiwan, Effect of thermal losses on the microscopic two-step heat conduction model, Int. J. Heat Mass Transfer 43 (2001) 1013-1018.

[176] M. Al-Odat, M.A. Al-Nimr, M. Hamdan, Thermal stability of superconductors under the effect of a two-dimensional hyperbolic heat conduction model, Int. J. Numer. Meth. Heat and Fluid Flow 12 (2002) 163-177.

[177] J.K. Chen, J.E. Beraun, C.L. Tham, Investigation of thermal response caused by pulsed laser heating, Numerical Heat Transfer, Part A 44 (2003) 705-722.

[178] J.K. Chen, W.P. Latham, J.E. Beraun, Axisymmetric modeling of femtosecond-pulse laser heating on metal films, Numerical Heat Transfer, Part B 42 (2002) 1-17.

[179] J.K. Chen, D.Y. Tzou, J.E. Beraun, Numerical investigation of ultrashort laser damage in semiconductors, In. J. Heat Mass Transfer 48 (2005) 501-509.

[180] W. Dai, C. Johnson, A new accurate finite-difference scheme for the thermal analysis of one-dimensional microspheres exposed to ultrashort-pulsed lasers, Numerical Heat Transfer, Part B 57 (2010) 241-259.

[181] W. Dai, T. Niu, A finite difference method for solving nonlinear hyperbolic two-step model in a double-layered thin film exposed to ultrashort pulsed lasers with nonlinear interfacial conditions, Nonlinear Anal. Hybrid Sys. 2 (2008) 121-143.

[182] W. Dai, G. Li, A finite difference scheme for solving parabolic two-step micro heat transport equations in a double-layered micro sphere heated by ultrashort-pulsed lasers, Numer. Methods Partial Differential Eq. 22 (2006) 1396-1417.

[183] W. Dai, H. Song, S. Su, R. Nassar, A stable finite difference scheme for solving a hyperbolic two-step model in a 3D micro sphere exposed to ultrashort-pulsed lasers, Int. J. Numer. Meth. Heat Fluid Flow 16 (2006) 693-717.

[184] W. Dai, Q. Li, R. Nassar, L. Shen, An unconditionally stable three level finite difference scheme for solving parabolic two-step micro heat transport equations in a three dimensional double-layered thin film, Int. J. for Numerical Methods in Engineering 59 (2004) 493-509.

[185] A. Giri, J.T. Gaskins, B.F. Donovan, C. Szwejkowski, R.J. Warzoha, M.A. Rodriguez, J. Ihlefeld, P.E. Hopkins, Mechanisms of nonequilibrium electron-phonon coupling and thermal conductance at interfaces, J. Appl. Phys. 117 (2015) 105105.

[186] I.K. Kaba, W. Dai, A stable three-level finite difference scheme for solving the parabolic two-step model in a 3D micro-sphere heated by ultrashort-pulsed lasers , J. Comp. Appl. Math. 181 (2005) 125-147.

[187] M. Mozafarifard, D. Toghraie, Time-fractional subdiffusion model for thin metal films under femtosecond laser pulses based on Caputo fractional derivative to examine anomalous diffusion process, Int. J. Heat Mass Transfer 153 (2020) 119592.

[188] K.M. Murat Tunc, M. Erdem Gunay, F. Bayata, Analysis of the thermalization dynamics of two-layer thin films irradiated by femtosecond laser, Optik 208 (2020) 164137.

[189] S. Shen, W. Dai, J. Cheng, Fractional parabolic two-step model and its accurate numerical scheme for nanoscale heat conduction, J. Comp. Appl. Math. 375 (2020) 112812.

[190] S. Zhai, L. Huang, Z. Weng, W. Dai, Parabolic two-step model and accurate numerical scheme for nanoscale heat conduction induced by ultrashort-pulsed laser heating, J. Comp. Appl. Math. 369 (2020) 112591.

[191] D.Y. Tzou, Macro-to-Microscale Heat Transfer: the Lagging Behavior, John Wiley and Sons, New York, 2015.

[192] D.Y. Tzou, W. Dai, Thermal lagging in multi-carrier system, Int. J. Heat Mass Transfer 52 (2009) 1206-1213.

[193] W. Dai, A hyperbolic microscopic model and its numerical scheme for thermal analysis in an N-carrier system, Int. J. Heat Mass Transfer 52 (2009) 2379-2389.

[194] W. Dai, F. Zhu, D.Y. Tzou, A stable finite difference scheme for thermal analysis in an N-carrier system, Int. J. Therm. Sci. 48 (2009) 1530-1541

[195] D. Zhao, W. Dai, Accurate finite difference schemes for solving a 3D micro heat transfer model in an N-carrier system with the Neumann boundary condition in spherical coordinates, J. Comp. Appl. Math. 235 (2010) 850-869.

[196] A. Bora, W. Dai, J.P. Wilson, J.C. Boyt, Neural network method for solving parabolic two-temperature microscale heat conduction in double-layered thin films exposed to ultrashort-pulsed lasers, Int. J. Heat Mass Transfer 178 (2021) 121616.