Fall 2008

A hyperbolic two-step model based finite difference method for studying thermal deformation in three-dimensional micro spheres exposed to ultrashort-pulsed lasers

Pan Wang

A HYPERBOLIC TWO-STEP MODEL BASED FINITE DIFFERENCE METHOD FOR STUDYING THERMAL DEFORMATION IN 3D MICRO SPHERES EXPOSED TO ULTRASHORT-PULSED LASERS

by

Pan Wang, B.S., M.S.

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

COLLEGE OF ENGINEERING AND SCIENCE
LOUISIANA TECH UNIVERSITY

November 2008

LOUISIANA TECH UNIVERSITY

THE GRADUATE SCHOOL

10/16/2008

Date

We hereby recommend that the dissertation prepared under our supervision by Pan Wang

entitled A HYPERBOLIC TWO-STEP MODEL BASED FINITE DIFFERENCE METHOD FOR STUDYING THERMAL DEFORMATION IN 3D MICRO SPHERES EXPOSED TO ULTRASHORT-PULSED LASERS

be accepted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy

Weizhong Dai

Supervisor of Dissertation Research

Weizhong Dai

Head of Department

Computational Analysis & Modeling

Department

Recommendation concurred in:

R Nassar

Advisory Committee

Audrey Rorrer

Don Liu

Approved:

Director of Graduate Studies

Approved:

Dean of the Graduate School

Dean of the College

GS Form 13a
(6/07)

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 Pam Day

Date 10/22/2008

ABSTRACT

Ultrashort-pulsed lasers with pulse durations of the order of sub-picoseconds to femtoseconds possess the capabilities in limiting the undesirable spread of the thermal process zone in a heated sample. Because of this, ultrashort-pulsed lasers have been attracting worldwide interest in science and engineering. The success of ultrashort-pulsed lasers in real application relies on: (1) well characterized pulse width, intensity and experimental techniques; (2) reliable microscale heat transfer models; and (3) prevention of thermal damage. Laser damage induced by ultrashort-pulsed lasers occurs after the heating pulse is over, since the pulse duration time is extremely short and the heat flux is essentially limited to the region within the electron thermal diffusion length. In contrast with long-pulse lasers, laser damage is caused by melting temperature, resulting from continuous pulse of energy. Therefore, in order to apply ultrashort-pulsed lasers successfully, one must study the thermal deformation to prevent the thermal damage.

In the previous research, the parabolic two-step micro heat transport equations have been widely applied in microscale heat transfer. However, when the laser pulse duration is much shorter than the electron thermal relaxation time for the activation of ballistic behavior in the electron gas, the parabolic two-step model may be inadequate to describe the continuous energy flow from hot electrons to lattices during non-equilibrium heating, as pointed out in the literature.

To our knowledge, it has not been seen in the literature that the hyperbolic two-step model is used for studying thermal deformation in micro spheres exposed to ultrashort-pulsed lasers. Micro spheres are considered because they are of interest related to micro resonators in optical applicants, such as ultra-low-threshold lasing, sensing, optoelectronic microdevices, cavity quantum electrodynamics, and their potential in quantum information processing. Hence, the purpose of this dissertation is to employ the hyperbolic two-step model with temperature-dependent thermal properties for obtaining temperature distribution in micro spheres induced by ultrashort-pulsed lasers. This model is coupled with the dynamic equations of motions for studying thermal deformation in micro spheres. To achieve this goal, we first employ an implicit finite difference scheme for solving the hyperbolic two-step model with temperature-dependent thermal properties. We then apply it to studying thermal deformations in Three-Dimensional (3D) micro spheres exposed to ultrashort-pulsed lasers. For this method, staggered grids are designed, and the coupling effect between lattice temperature and strain rate, as well as the hot electron blast effect in momentum transfer, are considered. As such, this obtained method allows us to avoid non-physical oscillations in the solution.

To demonstrate the applicability of the method, we test two physical cases, (1) 3D micro sphere irradiated by ultrashort-pulsed lasers, and (2) 3D double-layered micro sphere with perfectly thermal contacted interface irradiated by ultrashort-pulsed lasers. Results show that the micro spheres expand, and there are some differences between the hyperbolic two-step model and the parabolic two-step model. Particularly, one may see the differences between these two models in the change in electron temperature $(\Delta T_e /(\Delta T_e)_{\max})$.

TABLE OF CONTENTS

ABSTRACT iii

LIST OF TABLES viii

LIST OF FIGURES ix

NOMENCLATURE xiii

ACKNOWLEDGEMENTS xvi

CHAPTER 1.0 INTRODUCTION 1

1.1 General Overview 1

1.2 Research Objectives 3

1.3 Organization of the Dissertation 4

CHAPTER 2.0 BACKGROUND AND PREVIOUS WORK 5

2.1 Macroscopic Heat Transfer Model 5

2.2 Microscale Heat Transfer Model 6

2.3 Previous Work 8

2.3.1 2D Mathematical Model for Thermal Deformation 9

2.3.2 3D Mathematical Model for Thermal Deformation 12

2.3.3 2D Hyperbolic Two-Step Model for Thermal Deformation 14

2.3.4 Other Work 16

CHAPTER 3.0 THREE-DIMENSIONAL MATHEMATICAL MODEL AND FINITE DIFFERENCE SCHEME 19

3.1 Single-Layered Case 19

3.1.1 Governing Equations 19

3.1.2 Boundary Conditions and Initial Conditions 23

3.1.3 Notations 24

3.1.4 Finite Difference Scheme 27

3.1.5 Algorithm 34

3.2 Double-Layered Case 35

3.2.1 Governing Equations 36

3.2.2 Boundary Conditions and Initial Conditions 38

3.2.3 Notations 39

3.2.4 Finite Difference Scheme 42

3.2.5 Algorithm 49

CHAPTER 4.0 NUMERICAL EXAMPLES 51

4.1 Three-Dimensional Single-Layered Case 51

4.1.1 Example Description 51

4.1.2 Results and Discussions 53

4.2 Three-Dimensional Double-Layered Case 67

4.2.1 Example Description 67

4.2.2 Results and Discussions 69

CHAPTER 5.0 CONCLUSION AND FUTURE WORK 83

APPENDIX A. SOURCE CODE FOR 3D MICRO SPHERE CASE 85

APPENDIX B. SOURCE CODE FOR 3D DOUBLE-LAYERED CASE 104

REFERENCES 122

LIST OF TABLES

Table 2.1 Electron-lattice coupling factor G, for some noble and transition metals [Qiu 1992].....8

Table 4.1 Thermophysical properties of gold [Tzou 2002, Chen 2002, Kaye 1973].....53

Table 4.2 Thermophysical properties of gold and chromium [Toulou 1970a, 1970b, Chen 2003, Tzou 1996].....68

LIST OF FIGURES

Figure 2.1 A metal thin film exposed to ultrashort-pulsed lasers.....9

Figure 2.2 A 3D thin film irradiated by ultrashort-pulsed lasers.....12

Figure 3.1 A 3D micro sphere irradiated by ultrashort-pulsed lasers.....20

Figure 3.2 A 3D staggered grid and locations of variables for a micro sphere.....26

Figure 3.3 A 3D double-layered metal micro sphere.....35

Figure 4.1 A 3D micro sphere with a radius $L = 0.1\ \mu m$.....52

Figure 4.2 Change in electron temperature at the top point of the sphere versus time with a laser fluence (J) of 500 J/m^2 and three different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, and (c) $t_p = 0.01ps$.....55

Figure 4.3 Change in displacement (u_r) at the point ($r = L/2, \theta = 0, \varphi = 0$) of the sphere versus time with a laser fluence (J) of 500 J/m^2 and three different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, and (c) $t_p = 0.01ps$.....56

Figure 4.4 Electron temperature profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200J/m^2$, $500J/m^2$ and $1000J/m^2$ when $t_p = 0.01ps$.....57

Figure 4.5 Lattice temperature profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200J/m^2$, $500J/m^2$ and $1000J/m^2$ when $t_p = 0.01ps$.....59

Figure 4.6 Normal stress (σ_r) profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200 J/m^2$, $500 J/m^2$ and $1000 J/m^2$ when $t_p = 0.01ps$ 61

Figure 4.7 Contours of the electron temperature distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500 J/m^2$ when $t_p = 0.01ps$ 62

Figure 4.8 Contours of the lattice temperature distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500 J/m^2$ when $t_p = 0.01ps$ 63

Figure 4.9 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500 J/m^2$ when $t_p = 0.1ps$ 64

Figure 4.10 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500 J/m^2$ when $t_p = 0.04ps$ 65

Figure 4.11 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500 J/m^2$ when $t_p = 0.01ps$ 66

Figure 4.12 Configuration of a 3D double-layered micro sphere.....67

Figure 4.13 Change in electron temperature at the top point of the sphere versus time with a laser fluence (J) of $500\ J/m^2$ and four different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, (c) $t_p = 0.01ps$ and (d) $t_p = 0.006ps$ 70

Figure 4.14 Change in displacement (u_r) at the point ($r = L/2, \theta = 0, \varphi = 0$) of the sphere versus time with a laser fluence (J) of 500 J/m^2 and four different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, (c) $t_p = 0.01ps$ and (d) $t_p = 0.006ps$ 71

Figure 4.15 Electron temperature profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 0.015ps$, (b) $t = 0.03ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200J/m^2$, $500J/m^2$ and $1000J/m^2$ when $t_p = 0.006ps$ 73

Figure 4.16 Lattice temperature profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 0.015ps$, (b) $t = 0.03ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200J/m^2$, $500J/m^2$ and $1000J/m^2$ when $t_p = 0.006ps$ 74

Figure 4.17 Normal stress (σ_r) profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200J/m^2$, $500J/m^2$ and $1000J/m^2$ when $t_p = 0.006ps$ 76

Figure 4.18 Contours of the electron temperature distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 0.015ps$, (b) $t = 0.03ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.006ps$ 77

Figure 4.19 Contours of the lattice temperature distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.006ps$ 78

Figure 4.20 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.1ps$ 79

Figure 4.21 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.04ps$ 80

Figure 4.22 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.01ps$ 81

Figure 4.23 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.006ps$ 82

NOMENCLATURE

C_{e0}	electron heat capcity, $J/(m^3K)$
C_l	lattice heat capcity, $J/(m^3K)$
G	electron-lattice coupling factor, $W/(m^3K)$
J	laser fluence, J/m^2
K	bulk modulus, Pa
K_e	thermal conductivity, $W/(mK)$
N_r	number of grid points in the x-direction
N_θ	number of grid points in the θ-direction
N_φ	number of grid points in the φ-direction
Q	heat source, W/m^2
$\vec{q}_e$	electron heat flux, W/m^2
$\vec{q}_l$	lattice heat flux, W/m^2
q_e^r	component of electron heat flux in the r-direction, W/m^2
q_e^θ	component of electron heat flux in the θ-direction, W/m^2
q_e^φ	component of electron heat flux in the φ-direction, W/m^2
q_l^r	component of lattice heat flux in the r-direction, W/m^2
q_l^θ	component of lattice heat flux in the θ-direction, W/m^2
q_l^φ	component of lattice heat flux in the φ-direction, W/m^2
R	surface reflectivity
T_0	surrounding temperature, K
T_e	electron temperature, K

T_l	lattice temperature, K
t, t_n	time, s
t_p	laser pulse duration, s
u_r	displacement in the r-direction , m
u_θ	displacement in the θ-direction , m
u_φ	displacement in the φ-direction , m
v_r	velocity component in the r-direction , m/s
v_θ	velocity component in the θ-direction , m/s
v_φ	velocity component in the φ-direction , m/s
(r, θ, φ)	spherical coordinates

Greek Symbols

Δt	time increment, s
Δr	spatial grid size in the r-direction, m
$\Delta\theta$	spatial grid size in the θ-direction, m
$\Delta\varphi$	spatial grid size in the φ-direction, m
ζ	optical penetration depth, m
α_T	thermal expansion coefficient
Δ_{-t}	finite difference operator
δ_r	finite difference operator
δ_θ	finite difference operator
δ_φ	finite difference operator
Δ_r	finite difference operator
$\Delta_{\bar{r}}$	finite difference operator
τ_e	electron relaxation time, ps
τ_l	lattice relaxation time, ps
ε_r	normal strain in the r-direction

ε_θ	normal strain in the θ-direction
ε_φ	normal strain in the φ-direction
$\varepsilon_{r\theta}$	shear strain in the $r\theta$-direction
$\varepsilon_{r\varphi}$	shear strain in the $r\varphi$-direction
$\varepsilon_{\varphi\theta}$	shear strain in the $\varphi\theta$-direction
Λ	electron-blast coefficient, $J/(m^3K^2)$
λ	Lame's constant, Pa
μ	Lame's constant, Pa
ρ	density, kg/m^3
δ	penetration depth nm
σ	Stefan-Boltzmann's constant
σ_r	normal stress in the r-direction
σ_θ	normal stress in the θ-direction
σ_φ	normal stress in the φ-direction
$\sigma_{r\theta}$	shear stress in the $r\theta$-direction
$\sigma_{r\varphi}$	shear stress in the $r\varphi$-direction
$\sigma_{\varphi\theta}$	shear stress in the $\varphi\theta$-direction

Subscripts and Superscripts

0	initial value at $t = 0$
e	electron
i	grid index in the r-direction
j	grid index in the θ-direction
k	grid index in the φ-direction
l	lattice
n	time level

ACKNOWLEDGEMENTS

This dissertation owes great appreciation to many people for their support and encouragement. I express my sincere gratitude to my advisor, Dr. Weizhong Dai, for his patient guidance, generous help, and abundant encouragement. It is my great honor to be his student. Without his advices and suggestions, this dissertation could not be completed. I would like to thank Dr. Raja Nassar for his trust and help. In his class, he showed me beauty of statistics. Sincere acknowledgement is also extended to Dr. Andrei Paun, Dr. Don Liu and Dr. Mihaela Paun for teaching and directing me.

Furthermore, I owe a lot of thanks to my parents for their unconditional love and selfless sacrifice. They trust, respect, and support me and do their best to make me happy. I would also like to thank my friend, Fei Zhu, who accompanies me always on the way to my goal. Finally, I want to express my appreciation to all my friends. This dissertation is dedicated to all those mentioned above.

CHAPTER 1

INTRODUCTION

1.1 General Overview

Heat transport through micro thin films plays a very important role in microtechnology applications [Joseph 1989, Joshi 1993]. Many microelectronic devices have metal thin films as their key components. Microscale heat transfer is also important for the thermal processing of materials [Qiu 1992, 1993], including laser micromachining, laser patterning, laser synthesis, and laser surface hardening.

Ultrafast lasers with pulse lengths in the pico- to femtosecond range enable outstanding spatial and temporal resolution in the heated sample [Tzou 2002]. The applications of ultrashort-pulsed lasers, such as laser micromachining and patterning [Elliot 1989], structural monitoring of thin metal films [Mandelis 1992, Opsal 1991], laser synthesis and processing in thin-film deposition [Narayan 1991], and structural tailoring of microfilms [Grigorop 1994], become more and more mature. Recently, ultrashort-pulsed lasers have been applied in different disciplines such as physics, chemistry, biology, medicine, and optical technology [Liu 2000, Shirk 1998]. The non-contact nature of femtosecond lasers has made them an ideal candidate for precise thermal processing of functional nanophase materials [Tzou 2002].

At the micro scale, the heating induced by an ultrashort-pulsed laser involves the excitation of electrons and heat transfer from electrons to lattices in short time [Dai 2005]. The heat transport primarily depends on the collision between the energy carriers in conducting medium. In metal, electrons and phonons are the main energy carriers. Because phonons are more inclined to scatter than electrons, the mechanism of heating electron gas is more efficient than that of heating phonon gas [Touloukian 1970a, 1970b], which results in that free electrons absorb the laser energy primarily when the metal is heated. Then, the electrons temperature increases to several hundred or thousand degrees, which makes them more active to diffuse heat. Through electron-phonon interactions, thermal equilibrium between the electrons and lattices is reached [Tzou 1996].

There has been considerable interest in the development of models that focus on heat transfer in the context of ultrashort-pulsed lasers. However, only a few mathematical models for studying thermal deformation induced by ultrashort-pulsed lasers have been developed [Chen 2002a, 2003, Wang 2002a, 2006b, 2008a]. As presented in [Wang 2008b], "Tzou and associates [Tzou 2002] presented a one-dimensional model in a double-layered thin film. The model was solved using a differential-difference approach. Chen and associates [Chen 2002a] considered a two-dimensional axisymmetric cylindrical thin film and proposed an explicit finite difference method by adding an artificial viscosity term to eliminate numerical oscillations. Dai and associates [Wang 2002a, 2006b, 2008a] developed a new method for studying thermal deformation in 2D thin films exposed to ultrashort-pulsed lasers, which is based on the parabolic two-step heat transport equations and implicit finite-difference scheme on a staggered grid." This is our motivation to study the thermal deformation induced by ultrashort-pulsed lasers

and to extend our research to 3D micro spheres because they are of interest related to micro resonators in optical applications, such as ultra-low-threshold lasing, sensing, optoelectronic microdevices, cavity quantum electrodynamics and their potential in quantum information processing.

1.2 Research Objectives

The objective of this dissertation is to develop a new numerical method for studying thermal deformation in three-dimensional single-layered and double-layered micro spheres exposed to ultrashort-pulsed lasers in which the numerical method is obtained based on the dynamic equations of motion and hyperbolic two-step heat transport equations. To achieve this objective, a series of steps should be followed.

Step 1: Introduce velocity components into the model and re-write the dynamic equations of motion.

Step 2: Construct a staggered grid.

Step 3: Develop a fourth-order compact finite difference scheme for obtaining stress derivatives and shear stress derivatives in the dynamic equations of motion. As such, the third-order derivatives of stresses and shear stresses are disappeared and hence non-physical oscillations in the solutions are eliminated.

Step 4: Develop a finite difference scheme for obtaining temperatures, stress, strain, and displacement distributions in a 3D single-layered micro sphere heated by ultrashort-pulsed lasers. This scheme is based on the hyperbolic two-step model.

Step 5: Develop a finite difference scheme for obtaining temperatures, stress, strain and displacement distributions in a 3D double-layered micro sphere with perfectly thermal contacted interface based on the hyperbolic two-step model.

Step 6: Test the method by some numerical examples.

1.3 Organization of the Dissertation

In Chapter 1, a general review of the main idea of our work is given, and the objective of this dissertation is proposed. Chapter 2 provides some background for this research. Heat transfer at macro scale, micro scale, the parabolic two-step model, and the hyperbolic two-step model for micro thin films, as well as a review of previous works, will be reviewed in this chapter.

Chapter 3 describes the mathematical models for 3D single-layered and double-layered micro spheres irradiated by ultrashort pulsed lasers. The governing equations will be set up. Meanwhile, the finite difference scheme will be developd for both single-layered and double-layered micro spheres for 3D cases.

In Chapter 4, we will design the numerical algorithms for computing the temperature, displacement, stress and strain distributions and show the numerical results based on the developed numerical methods in a micro sphere exposed to an ultrashort pulsed laser. Two cases will be focused on, which are a 3D micro sphere, and a 3D double-layered micro sphere with perfectly thermal contacted interface. Various mesh sizes will be chosen to test the convergence of the method. Also, the electron temperature, the lattice temperature, the displacements, and the stresses will be showed and discussed.

Lastly, in Chapter 5, we give the conclusions of our work and suggest future research work.

CHAPTER 2

BACKGROUND AND PREVIOUS WORK

2.1 Macroscopic Heat Transfer Model

Heat can flow from a region with higher temperature to a region with lower temperature. Only by convection, conduction and radiation, can thermal energy be transferred [Mills 1994]. In our research, heat transfer on 3D micro spheres irradiated by ultrashort-pulsed lasers ocurred by radiation, whereas heat transfer across the micro spheres ocurred by conduction.

Heat conduction at the macro scale describes macroscopic behavior averaged over many grains. Heat is transferred through conduction when free electrons diffuse in metal. In the classical theory of heat transfer, the heat conduction is governed by Fourier's law. It is a constitutive equation that depicts the time rate of heat transfer, which is directed by the temperature gradient. It is necessary, along with the conservation of energy law, to derive the heat transport equations. This is consistent with the second law of thermodynamics [Kaba 2005].

Fourier's law of heat conduction can be expressed as Equation (2.1) [Kaba 2005, Wang 2007]

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

where $\vec{r}$ denotes the position vector of the material volume, k is the thermal conductivity of a particular substance, and t is the physical time.

The energy equation derived from the first law of thermodynamics is Equation (2.2) [Kaba 2005, Wang 2007]

$$-\nabla\vec{q} = C_p \frac{\partial T}{\partial t} - Q, \tag{2.2}$$

where C_p is the volumetric heat capacity and Q is the heat source. Taking the divergence of Equation (2.1) and substituting it into Equation (2.2), we obtain the traditional heat conduction equation as Equation (2.3) [Kaba 2005, Wang 2007]

$$C_p \frac{\partial T}{\partial t} = \nabla \cdot (k\nabla T) + Q. \tag{2.3}$$

2.2 Microscale Heat Transfer Model

We will give a brief review of the microscopic two-step model (phonon-electron interaction model), which emphasizes the special behavior depicted by the model that might reveal possible lagging behavior.

The model of traditional heat conduction can be described as a parabolic one-step equation because of the assumptions it makes that heat energy is converted to lattice energy instantaneously and that heat energy is assumed to be a diffusive process [Qiu 1993]. Other non-Fourier models have been proposed to deal with the failings of the Fourier model on a microscale. One model is based on the modified flux law expressed as Equation (2.4) [Tzou 1993]

$$\vec{q} + \tau \frac{\partial \vec{q}}{\partial t} = -k\nabla T, \tag{2.4}$$

where τ is the relaxation time and $\vec{q}$ is the heat flux. The heat flux vector in this case maintains a memory of the time-history of the temperature gradient. Relaxation time is the effective mean free path l, divided by the phonon speed v_s. Mathematically, $\tau = l / v_s$.

When Equation (2.4) is combined with Equation (2.3), we obtain the hyperbolic heat equation expressed as Equations (2.5)

$$C\frac{\partial T}{\partial t} = -\nabla\vec{q} + Q, \tag{2.5a}$$

$$\tau\frac{\partial \vec{q}}{\partial t} + k\nabla T + \vec{q} = 0. \tag{2.5b}$$

This equation is known as a hyperbolic equation because of the additional term that modifies the parabolic Fourier heat Equation (2.3) [Tang 1996]. This modification predicts a finite speed of heat propagation because of the relaxation time τ, associated with heat transfer. Typical wave speeds in metals are on the order of $10^5 m/s$ [Ozisik 1994].

While the hyperbolic model answers some issues arising from a microscale examination of heat transfer, it still leaves some questions. It is not based on the details of energy transport in the material, such as the interaction of electrons and phonons [Qiu 1993].

The parabolic two-step model depicts heating of the electron gas and the metal lattice by a two-step process for metals. The equations can be written as Equations (2.6)-(2.7) [Kaganov 1957]

$$\text{Step 1:} \quad C_e\frac{\partial T_e}{\partial t} = \nabla\cdot(K\nabla T_e) - G(T_e - T_l), \tag{2.6}$$

$$\text{Step 2:} \qquad C_l \frac{\partial T_l}{\partial t} = G(T_e - T_l), \tag{2.7}$$

where C denotes the volumetric heat capacity, K the thermal conductivity of electron gas, G is the phonon-electron coupling factor, and subscripts e and l represent the electron and metal lattice, respectively. Equation (2.6) is the mathematical representation for the first step heating of the electron gas, and Equation (2.7) is for the second step heating of the metal lattice.

The phonon-electron coupling factor is calculated, and experimentally measured values are listed in Table 2.1 for comparison.

Table 2.1 Phonon-electron coupling factor G, for some noble and transition metals [Qiu 1992].

Metal	Calculated, $\times 10^{16} w/m^3 k$	Measured, $\times 10^{16} w/m^3 k$
Cu	14	4.8 ± 0.7 [Brorson 1990] 10 [Elsayed-Ali 1987]
Ag	3.1	2.8 [Groeneveld 1990]
Au	2.6	2.8 ± 0.5 [Brorson 1990]
Cr	45 ($n_e / n_a = 0.5$)	42 ± 5 [Brorson 1990]
W	27 ($n_e / n_a = 1.0$)	26 ± 3 [Brorson 1990]
V	648 ($n_e / n_a = 2.0$)	523 ± 37 [Brorson 1990]
Nb	138 ($n_e / n_a = 2.0$)	387 ± 36 [Brorson 1990]
Pb	62	12.4 ± 1.4 [Brorson 1990]
Ti	202 ($n_e / n_a = 1.0$)	185 ± 16 [Brorson 1990]

2.3 Previous Work

Up to now, there are many researchers studying heat transfer models related to ultrashort-pulsed lasers [Tzou 1993, 1994, 1995a, 1995b, 1995c, 1995d, 1997, 1999, 2000a, 2000b, 2001, 2002], [Ozisik 1994], [Chiffell 1994], [Wang 2000, 2001a, 2001b,

2002], [Antaki 1998, 2000, 2002], [Dai 1999, 2000a, 2000b, 2000c, 2001a, 2001b, 2002, 2004a, 2004b], [Qiu 1992, 1993, 1994a, 1994b], [Joshi 1993], [Chen 1999a, 1999b, 2000a, 2000b, 2001, 2003, 2005], [Al-Nimr 1997a, 1997b, 1999, 2000a, 2000b, 2000c, 2001, 2003], [Tsai 2003], [Falkovsky 1999], [Fushinobu 1999], [Hoashi 2002], and [Lee 2003, 2005]. Some detailed parabolic models will be reviewed in this section first.

2.3.1 2D Mathematical Model for Thermal Deformation

In 2006, Dai and associates [Wang 2006a] considered a 2D thin film exposed to ultrashort-pulsed lasers as shown in Figure 2.1, and developed a finite difference method for studying thermal deformation in the 2D thin film exposed to ultrashort-pulsed lasers.

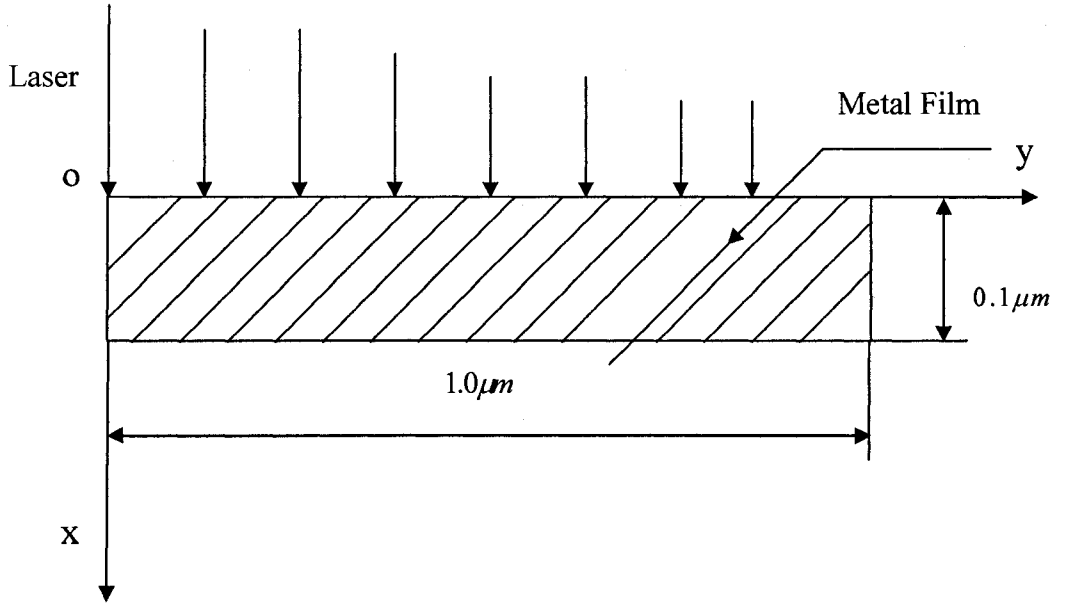


Figure 2.1 A metal thin film exposed to ultrashort-pulsed lasers.

The 2D mathematical model is obtained based on the parabolic two-step model coupled with the dynamic equations of motion, which can be expressed as follows

(1) Dynamic equations of motion can be expressed as Equations (2.8)-(2.14) [Tzou 2002, Chen 2002a, Brorson 1987]

$$\rho\frac{\partial^2 u}{\partial t^2}=\frac{\partial\sigma_x}{\partial x}+\frac{\partial\sigma_{xy}}{\partial y}+2\Lambda T_e\frac{\partial T_e}{\partial x}, \tag{2.8}$$

$$\rho\frac{\partial^2 v}{\partial t^2}=\frac{\partial\sigma_{xy}}{\partial x}+\frac{\partial\sigma_y}{\partial y}+2\Lambda T_e\frac{\partial T_e}{\partial y}, \tag{2.9}$$

where

$$\sigma_x=\lambda(\varepsilon_x+\varepsilon_y)+2\mu\varepsilon_x-(3\lambda+2\mu)\alpha_T(T_l-T_0), \tag{2.10}$$

$$\sigma_y=\lambda(\varepsilon_x+\varepsilon_y)+2\mu\varepsilon_y-(3\lambda+2\mu)\alpha_T(T_l-T_0), \tag{2.11}$$

$$\sigma_{xy}=\mu\varepsilon_{xy}, \tag{2.12}$$

$$\varepsilon_x=\frac{\partial u}{\partial x},\ \varepsilon_y=\frac{\partial u}{\partial y},\ \varepsilon_{xy}=\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}, \tag{2.13}$$

$$\lambda=K-\frac{2}{3}\mu. \tag{2.14}$$

Here, u ,v are the displacement in the x and y directions, respectively; ε_x and ε_y are the normal strains in the x and y directions, respectively; σ_x and σ_y are the normal stresses in the x and y directions, respectively; σ_{xy} is the shear stress in the xy - plane; T_e and T_l are electron and lattice temperatures, respectively; T_0 is the initial temperature; ρ is the density; Λ is the electron-blast coefficient; λ ($=K-\frac{2}{3}\mu$ [Reismann 1980]) is Lame constant; K is the bulk modulus; μ is the shear modulus; and α_T is the thermal expansion coefficient.

(2) Energy equations can be expressed as Equations (2.15)-(2.16) [Tzou 2002, Chen 2002a, Qiu 1992]

$$C_e(T_e)\frac{\partial T_e}{\partial t}=\frac{\partial}{\partial x}\left[k_e(T_e,T_l)\frac{\partial T_e}{\partial x}\right]+\frac{\partial}{\partial y}\left[k_e(T_e,T_l)\frac{\partial T_e}{\partial y}\right]$$

$$-G(T_e-T_l)+Q, \tag{2.15}$$

$$C_l\frac{\partial T_l}{\partial t}=G(T_e-T_l)-(3\lambda+2\mu)\alpha_T T_0\frac{\partial}{\partial t}(\varepsilon_x+\varepsilon_y), \tag{2.16}$$

where the heat source is given by Equation (2.17)

$$Q=0.94J\frac{1-R}{t_p x_s}\exp\left[-\frac{x}{x_s}-\left(\frac{y}{y_s}\right)^2-2.77\left(\frac{t-2t_p}{t_p}\right)^2\right]. \tag{2.17}$$

Here, $C_e(T_e)=C_{e0}\left(\frac{T_e}{T_0}\right)$ is the electron heat capacity, $k_e(T_e,T_l)=k_0\left(\frac{T_e}{T_l}\right)$ is the thermal conductivity, G is the electron–lattice coupling factor, C_l is the lattice heat capacity, respectively; Q is the energy absorption rate; J is the laser fluence; R is the surface reflectivity; t_p is the laser pulse duration.

The boundary conditions are assumed to be stress free and thermally insulated, which can be shown as Equations (2.18)-(2.20)

$$\sigma_x=0,\sigma_{xy}=0,\text{ at } x=0,L_x, \tag{2.18}$$

$$\sigma_y=0,\sigma_{xy}=0,\text{ at } y=0,L_y, \tag{2.19}$$

$$\frac{\partial T_e}{\partial \vec{n}}=0,\frac{\partial T_l}{\partial \vec{n}}=0. \tag{2.20}$$

The initial conditions are assumed to be Equations (2.21)

$$T_e=T_l=T_0,u=v=0,u_t=v_t=0,\text{ at}t=0. \tag{2.21}$$

Numerical results in Wang [Wang 2006a] show that the method allows avoiding non-physical oscillations in the solution.

2.3.2 3D Mathematical Model for Thermal Deformation

A finite difference method for studying thermal deformation in 3D thin films exposed to ultrashort-pulsed lasers based on the parabolic two-step model, as shown in Figure 2.2, was developed recently [Zhang 2008].

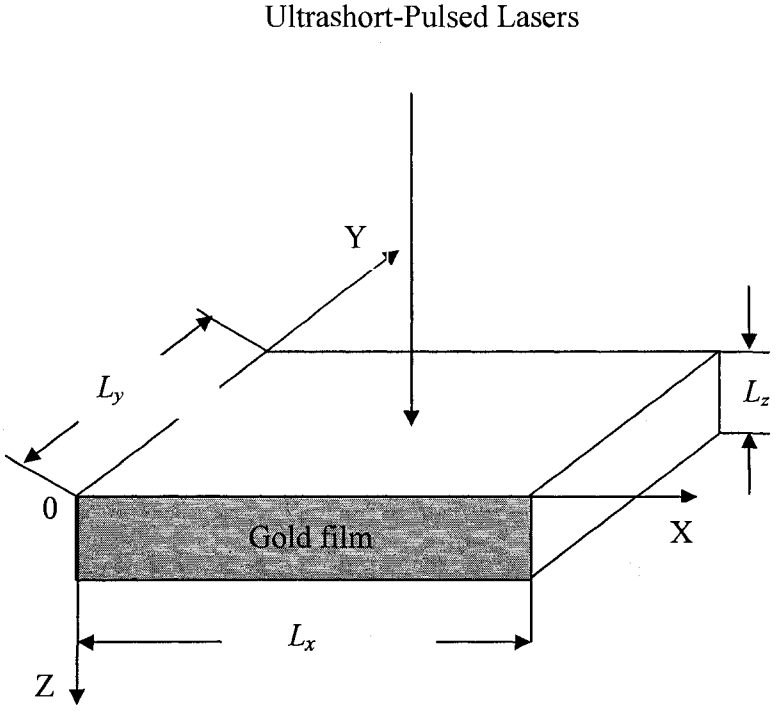


Figure 2.2 A 3D thin film irradiated by ultrashort-pulsed lasers.

The governing equations can be expressed as follows

(1) Dynamic Equations of Motion can be expressed as Equations (2.22)-(2.30) [Brorson 1987, Chen 2002, Tzou 2002, Wang 2006a]

$$\rho\frac{\partial^2 u}{\partial t^2}=\frac{\partial \sigma_x}{\partial x}+\frac{\partial \sigma_{xy}}{\partial y}+\frac{\partial \sigma_{xz}}{\partial z}+2\Lambda T_e\frac{\partial T_e}{\partial x}, \tag{2.22}$$

$$\rho\frac{\partial^2 v}{\partial t^2}=\frac{\partial \sigma_{xy}}{\partial x}+\frac{\partial \sigma_y}{\partial y}+\frac{\partial \sigma_{yz}}{\partial z}+2\Lambda T_e\frac{\partial T_e}{\partial y}, \tag{2.23}$$

$$\rho\frac{\partial^2 w}{\partial t^2}=\frac{\partial\sigma_{xz}}{\partial x}+\frac{\partial\sigma_{yz}}{\partial y}+\frac{\partial\sigma_z}{\partial z}+2\Lambda T_e\frac{\partial T_e}{\partial z}, \tag{2.24}$$

$$\sigma_x=\lambda(\varepsilon_x+\varepsilon_y+\varepsilon_z)+2\mu\varepsilon_x-(3\lambda+2\mu)\alpha_T(T_l-T_0), \tag{2.25}$$

$$\sigma_y=\lambda(\varepsilon_x+\varepsilon_y+\varepsilon_z)+2\mu\varepsilon_y-(3\lambda+2\mu)\alpha_T(T_l-T_0), \tag{2.26}$$

$$\sigma_z=\lambda(\varepsilon_x+\varepsilon_y+\varepsilon_z)+2\mu\varepsilon_z-(3\lambda+2\mu)\alpha_T(T_l-T_0), \tag{2.27}$$

$$\sigma_{xy}=\mu\gamma_{xy},\sigma_{xz}=\mu\gamma_{xz},\sigma_{yz}=\mu\gamma_{yz}, \tag{2.28}$$

$$\varepsilon_x=\frac{\partial u}{\partial x},\varepsilon_y=\frac{\partial v}{\partial y},\varepsilon_z=\frac{\partial w}{\partial z}, \tag{2.29}$$

$$\gamma_{xy}=\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x},\gamma_{xz}=\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x},\gamma_{yz}=\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}. \tag{2.30}$$

(2) Energy Equations can be expressed as Equations (2.31)-(2.32) [Chen 2002, Qiu 1992, Tzou 2002, Wang 2006a]

$$C_e(T_e)\frac{\partial T_e}{\partial t}=\frac{\partial}{\partial x}\left[k_e(T_e,T_l)\frac{\partial T_e}{\partial x}\right]+\frac{\partial}{\partial y}\left[k_e(T_e,T_l)\frac{\partial T_e}{\partial y}\right]+\frac{\partial}{\partial z}\left[k_e(T_e,T_l)\frac{\partial T_e}{\partial z}\right] -G(T_e-T_l)+Q, \tag{2.31}$$

$$C_l\frac{\partial T_l}{\partial t}=G(T_e-T_l)+(3\lambda+2\mu)\alpha_T T_0\frac{\partial}{\partial t}(\varepsilon_x+\varepsilon_y+\varepsilon_z), \tag{2.32}$$

where the heat source is considered to be a Gaussian distribution and is given by Equation (2.33) [Tzou 1997]

$$Q(x,y,z,t)=0.94J\frac{1-R}{t_p\delta}\exp[-\frac{z}{\delta}-\frac{(x-x_0)^2+(y-y_0)^2}{r_s^2}-2.77(\frac{t-2t_p}{t_p})^2]. \tag{2.33}$$

The boundary conditions are assumed to be stress free and thermally insulated, which are given by Equations (2.34)-(2.37) [Bruno 1997, Chen 2002, Swartz 1989]

$$\sigma_x=0,\ \ \sigma_{xy}=0,\ \ \sigma_{xz}=0,\ \text{at}\ x=0,L_x, \tag{2.34}$$

$$\sigma_y = 0, \quad \sigma_{xy} = 0, \quad \sigma_{yz} = 0, \text{ at } y = 0, L_y, \tag{2.35}$$

$$\sigma_z = 0, \quad \sigma_{xz} = 0, \quad \sigma_{yz} = 0, \text{ at } z = 0, L_z, \tag{2.36}$$

$$\frac{\partial T_e}{\partial \vec{n}} = 0, \frac{\partial T_l}{\partial \vec{n}} = 0. \tag{2.37}$$

To avoid numerical oscillations, the velocity components were introduced into the dynamic equation of motion and replaced the displacement components, a staggered grid was designed so that the checker-board solution could be eliminated, and a fourth-order compact scheme was employed to calculate stresses derivatives in the dynamic equations of motion [Zhang 2008].

2.3.3 2D Hyperbolic Two-Step Model for Thermal Deformation

Recently, Dai and associates have further developed a numerical method based on hyperbolic two-step model for studying thermal deformation in 2D thin films exposed to ultrashort-pulsed lasers [Niu 2008a, 2008b]. The governing equations for studying thermal deformation in the thin film can be expressed as

(1) Dynamic Equations of Motion are given by Equations (2.38)-(2.44) [Brorson 1987, Chen 2002a, Tzou 2002]

$$\rho \frac{\partial^2 u}{\partial t^2} = \frac{\partial \sigma_x}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + 2\Lambda T_e \frac{\partial T_e}{\partial x}, \tag{2.38}$$

$$\rho \frac{\partial^2 v}{\partial t^2} = \frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + 2\Lambda T_e \frac{\partial T_e}{\partial y}, \tag{2.39}$$

where

$$\sigma_x = \lambda(\varepsilon_x + \varepsilon_y) + 2\mu\varepsilon_x - (3\lambda + 2\mu)\alpha_T(T_l - T_0), \tag{2.40}$$

$$\sigma_y = \lambda(\varepsilon_x + \varepsilon_y) + 2\mu\varepsilon_y - (3\lambda + 2\mu)\alpha_T(T_l - T_0), \tag{2.41}$$

$$\sigma_{xy} = \mu\varepsilon_{xy}, \tag{2.42}$$

$$\varepsilon_x = \frac{\partial u}{\partial x}, \varepsilon_y = \frac{\partial u}{\partial y}, \varepsilon_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}, \tag{2.43}$$

$$\lambda = K - \frac{2}{3}\mu. \tag{2.44}$$

(2) Energy Equations are given by Equations (2.45)-(2.50) [Chen 2002a, Qiu 1992, Tzou 2002]

$$C_e \frac{\partial T_e}{\partial t} = -\frac{\partial q_e^x}{\partial x} - \frac{\partial q_e^y}{\partial y} - G(T_e - T_l) + Q, \tag{2.45}$$

$$\tau_e \frac{\partial q_e^x}{\partial t} + q_e^x = -k_e \frac{\partial T_e}{\partial x}, \tag{2.46}$$

$$\tau_e \frac{\partial q_e^y}{\partial t} + q_e^y = -k_e \frac{\partial T_e}{\partial y}, \tag{2.47}$$

$$C_l \frac{\partial T_l}{\partial t} = -\frac{\partial q_l^x}{\partial x} - \frac{\partial q_l^y}{\partial y} + G(T_e - T_l) - (3\lambda + 2\mu)\alpha_T T_0 \frac{\partial}{\partial t}(\varepsilon_x + \varepsilon_y), \tag{2.48}$$

$$\tau_l \frac{\partial q_l^x}{\partial t} + q_l^x = -k_l \frac{\partial T_l}{\partial x}, \tag{2.49}$$

$$\tau_l \frac{\partial q_l^y}{\partial t} + q_l^y = -k_l \frac{\partial T_l}{\partial y}. \tag{2.50}$$

These energy equations are referred to as hyperbolic two-step heat transport equations. The boundary conditions are assumed to be stress free and thermally insulated, which are given by Equations (2.51)-(2.53)

$$\sigma_x = 0, \sigma_{xy} = 0, \text{ at } x = 0, L_x, \tag{2.51}$$

$$\sigma_y = 0, \sigma_{xy} = 0, \text{ at } y = 0, L_y, \tag{2.52}$$

$$\frac{\partial T_e}{\partial \vec{n}} = 0, \frac{\partial T_l}{\partial \vec{n}} = 0. \tag{2.53}$$

The initial conditions are assumed to be Equations (2.54)

$$T_e = T_l = T_0, u = v = 0, u_t = v_t = 0, \text{at} t = 0. \tag{2.54}$$

If τ_e, τ_l and k_e are zeros, the hyperbolic two-step model can be reduced to the parabolic two-step model.

Numerical results show that the method in Wang [Wang 2006a, 2006b] and Niu [Niu 2008a, 2008b] allows avoiding non-physical oscillations in the solution.

2.3.4 Other Work

Although the parabolic two-step micro heat transport equations have been widely applied in analysis of microscale heat transfer, the model may be inadequate to describe the continuous energy flow from hot electrons to lattices during non-equilibrium heating when the characteristic heating time is much shorter than the electron relaxation time of free electrons in a metal [Qiu 1994]. Qiu and Tien developed the hyperbolic two-step heat transport equations based on the macroscopic averages of the electric and heat currents carried by electrons in the momentum space [Qiu 1993]. The generalized hyperbolic two-step model can be written as Equations (2.55)-(2.58) [Chen 2001], [Qiu 1994], and [Chen 2003]

$$C_e \frac{\partial T_e}{\partial t} = -\nabla \cdot \vec{q}_e - G(T_e - T_l) + S, \tag{2.55}$$

$$\tau_e \frac{\partial \vec{q}_e}{\partial t} + \vec{q}_e = -k_e \nabla T_e, \tag{2.56}$$

$$C_l \frac{\partial T_l}{\partial t} = -\nabla \cdot \vec{q}_l + G(T_e - T_l), \tag{2.57}$$

$$\tau_l \frac{\partial \vec{q}_l}{\partial t} + \vec{q}_l = -k_l \nabla T_l . \tag{2.58}$$

Here, $\tau_e = \frac{1}{A_e T_e^2 + B_l T_l}$ is the electron relaxation time, and $\tau_l = \frac{3k_l}{C_l v_s^2}$ is the lattice relaxation time. It can be seen that if τ_e, τ_l and k_l are zero, the hyperbolic two-step model will reduce to the parabolic two-step model.

Al-Nimr and associates [Al-Nimr 1999, 2000, 2003], [Al-Odat 2002], [Naji 2003] studied the thermal behavior of thin metal films in the hyperbolic two-step model with constant thermal properties. Chen and associates [Chen 2001, 2003] employed finite-difference and finite-element methods to solve the hyperbolic two-step model for investigation of thermal response in a single-layered metal thin film caused by pulse laser heating.

Tzou and Chiu studied the temperature-dependent thermal lagging [Tzou 2001], developing an explicit finite difference algorithm to perform the nonlinear analysis. As Antaki and associates pointed out, using the temperature-dependent conductivity instead of constant conductivity can gain better agreement between temperature predictions and corresponding measurements for short-pulse laser heating [Antaki 2002].

Wang and associates [Wang 2000, 2001] gave methods of measuring the phase-lags of the heat flux and the temperature gradient and obtained analytical solutions for 1D, 2D and 3D heat conduction domains under essentially arbitrary initial and boundary conditions. They provided solution system for both mixed and Cauchy problems of Dual-Phase-Lagging (DPL) heat conduction equations. The DPL heat conduction equation is well-posed in a bounded 1D region under Dirichlet, Neumann or Robin boundary conditions [Wang 2000]. Under linear boundary conditions, two solution systems exist.

For a finite region of dimension $n\,(n \geq 2)$ under Dirichlet, Neumann or Robin boundary conditions, the DPL heat conduction equation has a unique solution, which is stable regarding to the initial conditions [Wang 2001].

In order to solve the transient heat conduction problems in finite rigid slabs irradiated by short-pulse lasers, Tang and Araki [Tang 1999] developed a generalized macroscopic model with the solution derived from Green's function method and finite integral transform technique.

To study the deformation caused by ultrashort-pulsed lasers [Tzou 2002, Chen 2002a, 2002b, 2003]. Tzou [Tzou 2002] developed a one-dimensional model in a double-layered thin film and solved it using the differential-difference approach. Chen and his co-workers [Chen 2002a, 2002b, 2003] considered a two-dimensional axisymmetric cylindrical thin film and applied an explicit finite difference method. They added an artificial viscosity term to eliminate numerical oscillations.

Recently, Dai and Niu [Niu 2008a, 2008b] considered 3D thin films induced by ultrashort-pulsed lasers and employed hyperbolic model and dynamic equations of motion for studying thermal deformation. In this dissertation, we will extend the previous research to the micro sphere case, where the laser pulse duration being shorter than the electron thermal relaxation time is considered.

CHAPTER 3

THREE-DIMENSIONAL MATHEMATICAL MODEL AND FINITE DIFFERENCE SCHEME

In this chapter, we consider 3D micro spheres exposed to ultrashort-pulsed lasers. Mathematical models and their numerical solutions will be set up and developed.

3.1 Single-Layered Case

3.1.1 Governing Equations

Figure 3.1 shows a metal micro sphere in a three-dimensional coordinates system which is exposed to ultrashort-pulsed lasers. Consider the spherical coordinates system, r is the radial distance, ranging from 0 to ∞; θ is the azimuthal angle ranging from 0 to 2π; φ is the zenith angle ranging from 0 to π.

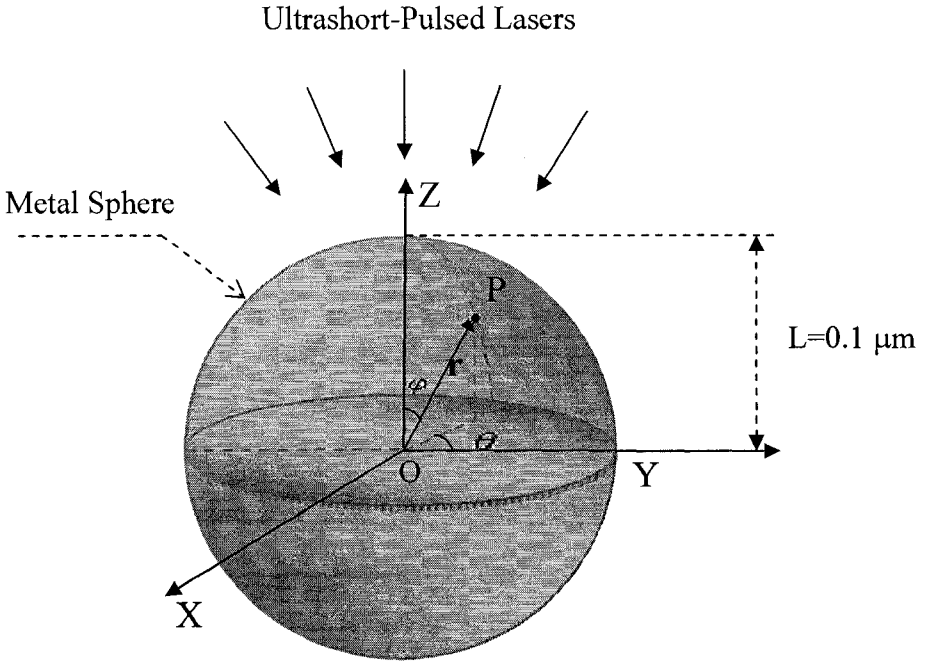


Figure 3.1 A 3D micro sphere irradiated by ultrashort-pulsed lasers.

The governing equations for studying thermal deformation in the micro sphere under spherical coordinates can be expressed as follows:

(1) Dynamic equations of motion can be written as Equations (3.1)-(3.4) [Tzou 2002, Chen 2002a, Brorson 1987, Wang 2006a, Reismann 1980, Zhang 2008]

$$\rho\frac{\partial^2 u_r}{\partial t^2}=\frac{\partial \sigma_r}{\partial r}+\frac{1}{r}\frac{\partial \sigma_{r\varphi}}{\partial \varphi}+\frac{1}{r\sin\varphi}\frac{\partial \sigma_{r\theta}}{\partial \theta}+\frac{1}{r}(2\sigma_r-\sigma_\varphi-\sigma_\theta+\sigma_{r\varphi}\cot\varphi)+2\Lambda T_e\frac{\partial T_e}{\partial r}, \quad (3.1)$$

$$\rho\frac{\partial^2 u_\varphi}{\partial t^2}=\frac{\partial \sigma_{r\varphi}}{\partial r}+\frac{1}{r}\frac{\partial \sigma_\varphi}{\partial \varphi}+\frac{1}{r\sin\varphi}\frac{\partial \sigma_{\varphi\theta}}{\partial \theta}+\frac{1}{r}[(\sigma_\varphi-\sigma_\theta)\cot\varphi+3\sigma_{r\varphi}]+2\Lambda T_e\frac{1}{r}\frac{\partial T_e}{\partial \varphi}, \quad (3.2)$$

$$\rho\frac{\partial^2 u_\theta}{\partial t^2}=\frac{\partial \sigma_{r\theta}}{\partial r}+\frac{1}{r}\frac{\partial \sigma_{\varphi\theta}}{\partial \varphi}+\frac{1}{r\sin\varphi}\frac{\partial \sigma_\theta}{\partial \theta}+\frac{1}{r}(2\sigma_{\varphi\theta}\cot\varphi+3\sigma_{r\theta})+2\Lambda T_e\frac{1}{r\sin\varphi}\frac{\partial T_e}{\partial \theta}, \quad (3.3)$$

where

$$\sigma_r = \lambda(\varepsilon_r + \varepsilon_\varphi + \varepsilon_\theta) + 2\mu\varepsilon_r - (3\lambda + 2\mu)\alpha_T(T_l - T_0), \tag{3.4a}$$

$$\sigma_\varphi = \lambda(\varepsilon_r + \varepsilon_\varphi + \varepsilon_\theta) + 2\mu\varepsilon_\varphi - (3\lambda + 2\mu)\alpha_T(T_l - T_0), \tag{3.4b}$$

$$\sigma_\theta = \lambda(\varepsilon_r + \varepsilon_\varphi + \varepsilon_\theta) + 2\mu\varepsilon_\theta - (3\lambda + 2\mu)\alpha_T(T_l - T_0), \tag{3.4c}$$

$$\sigma_{r\varphi} = 2\mu\varepsilon_{r\varphi}, \sigma_{r\theta} = 2\mu\varepsilon_{r\theta}, \sigma_{\varphi\theta} = 2\mu\varepsilon_{\varphi\theta}, \tag{3.4d}$$

$$\varepsilon_r = \frac{\partial u_r}{\partial r}, \tag{3.4e}$$

$$\varepsilon_\varphi = \frac{1}{r}(u_r + \frac{\partial u_\varphi}{\partial \varphi}), \tag{3.4f}$$

$$\varepsilon_\theta = \frac{1}{r\sin\varphi}(\frac{\partial u_\theta}{\partial \theta} + u_r \sin\varphi + u_\varphi \cos\varphi), \tag{3.4g}$$

$$\varepsilon_{r\varphi} = \frac{1}{2}(\frac{1}{r}\frac{\partial u_r}{\partial \varphi} + \frac{\partial u_\varphi}{\partial r} - \frac{u_\varphi}{r}), \tag{3.4h}$$

$$\varepsilon_{\varphi\theta} = \frac{1}{2r}(\frac{1}{\sin\varphi}\frac{\partial u_\varphi}{\partial \theta} + \frac{\partial u_\theta}{\partial \varphi} - u_\theta \cot\varphi), \tag{3.4i}$$

$$\varepsilon_{r\theta} = \frac{1}{2}(\frac{1}{r\sin\varphi}\frac{\partial u_r}{\partial \theta} + \frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r}). \tag{3.4j}$$

Here, u_r, u_θ, u_φ are the displacements in the r, θ, φ directions, respectively; $\varepsilon_r, \varepsilon_\theta, \varepsilon_\varphi$ are the normal strains in the r, θ, φ directions, respectively; $\varepsilon_{r\varphi}$ is the shear strain in the $r\varphi$ plane, $\varepsilon_{r\theta}$ is the shear strain in the $r\theta$ plane, and $\varepsilon_{\varphi\theta}$ is the shear strain in the $\varphi\theta$ plane; $\sigma_r, \sigma_\theta, \sigma_\varphi$ are the normal stresses in the r, θ, φ directions, respectively; $\sigma_{r\varphi}$ is the shear stress in the $r\varphi$ plane, $\sigma_{r\theta}$ is the shear stress in the $r\theta$ plane, and $\sigma_{\varphi\theta}$ is the shear stress in the $\varphi\theta$ plane; T_e and T_l are electron and lattice temperatures, respectively; T_0 is

the initial temperature; ρ is density; Λ is the electron-blast coefficient; $\lambda = K - \frac{2}{3}\mu$ [Reismann 1980] and μ are Lame's coefficients; K is bulk modulus; and α_T is the thermal expansion coefficient.

(2) Energy equations can be written as Equations (3.5)-(3.12) [Chen 2001, Chen 2002a, Chen 2002b, Chen 2003, Du 2008, Niu 2008a, 2008b]

$$C_e(T_e)\frac{\partial T_e}{\partial t} = -\frac{1}{r^2}\frac{\partial}{\partial r}(r^2 q_e^r) - \frac{1}{r\sin\varphi}\frac{\partial q_e^\theta}{\partial \theta} - \frac{1}{r\sin\varphi}\frac{\partial}{\partial \varphi}(q_e^\varphi \sin\varphi) - G(T_e - T_l) + Q, \tag{3.5}$$

$$\tau_e \frac{\partial q_e^r}{\partial t} + q_e^r = -k_e \frac{\partial T_e}{\partial r}, \tag{3.6}$$

$$\tau_e \frac{\partial q_e^\theta}{\partial t} + q_e^\theta = -\frac{k_e}{r\sin\varphi}\frac{\partial T_e}{\partial \theta}, \tag{3.7}$$

$$\tau_e \frac{\partial q_e^\varphi}{\partial t} + q_e^\varphi = -\frac{k_e}{r}\frac{\partial T_e}{\partial \varphi}, \tag{3.8}$$

$$\begin{aligned} C_l \frac{\partial T_l}{\partial t} = &-\frac{1}{r^2}\frac{\partial}{\partial r}(r^2 q_l^r) - \frac{1}{r\sin\varphi}\frac{\partial q_l^\theta}{\partial \theta} - \frac{1}{r\sin\varphi}\frac{\partial}{\partial \varphi}(q_l^\varphi \sin\varphi) \\ &+ G(T_e - T_l) - (3\lambda + 2\mu)\alpha_T T_l \frac{\partial}{\partial t}(\varepsilon_r + \varepsilon_\varphi + \varepsilon_\theta), \end{aligned} \tag{3.9}$$

$$\tau_l \frac{\partial q_l^r}{\partial t} + q_l^r = -k_l \frac{\partial T_l}{\partial r}, \tag{3.10}$$

$$\tau_l \frac{\partial q_l^\theta}{\partial t} + q_l^\theta = -\frac{k_l}{r\sin\varphi}\frac{\partial T_l}{\partial \theta}, \tag{3.11}$$

$$\tau_l \frac{\partial q_l^\varphi}{\partial t} + q_l^\varphi = -\frac{k_l}{r}\frac{\partial T_l}{\partial \varphi}, \tag{3.12}$$

where the heat source introduced by [Chen 2002] is extended for a Gaussian laser beam on the upper hemisphere as Equation (3.13)

$$Q(r,\theta,\varphi,t) = 0.94J\frac{1-R}{t_p\zeta}\exp[-\frac{L-r}{\zeta} - 2.77(\frac{t-2t_p}{t_p})^2]\cos\varphi. \tag{3.13}$$

Here, $C_e(T_e)=C_{e0}\left(\frac{T_e}{T_0}\right)$ is the electron heat capacity, $k_e(T_e,T_l)=k_0\left(\frac{T_e}{T_l}\right)$ is the thermal conductivity, G is the electron–lattice coupling factor, C_l is the lattice heat capacity, respectively; Q is the energy absorption rate; J is the laser fluence; R is the surface reflectivity; t_p is the laser pulse duration; L is the radius of the micro sphere; ζ is the optical penetration depth; $\vec{q}_e=(q_e^r,q_e^\theta,q_e^\varphi)$ and $\vec{q}_l=(q_l^r,q_l^\theta,q_l^\varphi)$ are the heat fluxes associated with electrons and the lattice, respectively; τ_e is the electron relaxation time and τ_l is the lattice relaxation time. It can be seen that the hyperbolic two-step model can be reduced to the parabolic two-step model if τ_e, τ_l and k_l are zeros.

3.1.2 Boundary Conditions and Initial Conditions

The boundary conditions are assumed to be stress free and thermally insulated as Equations (3.14) [Tzou 2002, Chen 2002]

$$\sigma_r=0,\quad \sigma_{r\varphi}=0,\quad \sigma_{r\theta}=0,\qquad at\ r=L, \tag{3.14a}$$

$$\sigma_\theta=\sigma_{\theta+2\pi},\quad \sigma_{r\theta}=\sigma_{r\theta+2\pi},\quad \sigma_{\varphi\theta}=\sigma_{\varphi\theta+2\pi}, \tag{3.14b}$$

$$\frac{\partial T_e}{\partial r}=0,\quad \frac{\partial T_l}{\partial r}=0,\qquad at\ r=L, \tag{3.14c}$$

$$T_e(r,\theta,\varphi,t)=T_e(r,\theta+2\pi,\varphi,t),\ T_l(r,\theta,\varphi,t)=T_l(r,\theta+2\pi,\varphi,t). \tag{3.14d}$$

Without loss of generality, we assume for simplicity as Equations (3.15)

$$\frac{\partial \sigma_r}{\partial r}=0,\quad \frac{\partial \sigma_{r\theta}}{\partial r}=0,\quad \frac{\partial \sigma_{r\varphi}}{\partial r}=0,\quad \frac{\partial T_e}{\partial r}=0,\quad \frac{\partial T_l}{\partial r}=0,\qquad at\ \ r=0, \tag{3.15a}$$

$$\frac{\partial \sigma_\varphi}{\partial \varphi}=0,\quad \frac{\partial \sigma_{\varphi\theta}}{\partial \varphi}=0,\quad \frac{\partial \sigma_{r\varphi}}{\partial \varphi}=0,\quad \frac{\partial T_e}{\partial \varphi}=0,\quad \frac{\partial T_l}{\partial \varphi}=0,\qquad at\ \ \varphi=0,\pi, \tag{3.15b}$$

$$\frac{\partial T_e}{\partial \varphi}=0,\quad \frac{\partial T_l}{\partial \varphi}=0,\qquad at\ \ \varphi=0,\pi. \tag{3.15c}$$

The initial conditions are assumed to be Equations (3.16)

$$T_e=T_l=T_0,\quad u_r=u_\varphi=u_\theta=0,\quad \frac{\partial u_r}{\partial t}=\frac{\partial u_\theta}{\partial t}=\frac{\partial u_\varphi}{\partial t}=0,\qquad at\ \ t=0. \tag{3.16}$$

3.1.3 Notations

To avoid the nonphysical oscillations in the numerical solutions, we employ the idea in [Wang 2006a, Wang 2007, Niu 2008a, 2008b, Zhang 2008] and introduce three velocity components v_r, v_θ and v_φ into the dynamic equations of motion, and hence, the Equations (3.1)-(3.4) are rewritten as Equations (3.17)-(3.21)

$$v_r=\frac{\partial u_r}{\partial t},\quad v_\theta=\frac{\partial u_\theta}{\partial t},\quad v_\varphi=\frac{\partial u_\varphi}{\partial t}, \tag{3.17}$$

$$\rho\frac{\partial v_r}{\partial t}=\frac{\partial \sigma_r}{\partial r}+\frac{1}{r}\frac{\partial \sigma_{r\varphi}}{\partial \varphi}+\frac{1}{r\sin\varphi}\frac{\partial \sigma_{r\theta}}{\partial \theta}+\frac{1}{r}(2\sigma_r-\sigma_\varphi-\sigma_\theta+\sigma_{r\varphi}\cot\varphi)+2\Lambda T_e\frac{\partial T_e}{\partial r}, \tag{3.18}$$

$$\rho\frac{\partial v_\varphi}{\partial t}=\frac{\partial \sigma_{r\varphi}}{\partial r}+\frac{1}{r}\frac{\partial \sigma_\varphi}{\partial \varphi}+\frac{1}{r\sin\varphi}\frac{\partial \sigma_{\varphi\theta}}{\partial \theta}+\frac{1}{r}[(\sigma_\varphi-\sigma_\theta)\cot\varphi+3\sigma_{r\varphi}]+2\Lambda T_e\frac{1}{r}\frac{\partial T_e}{\partial \varphi}, \tag{3.19}$$

$$\rho\frac{\partial v_\theta}{\partial t}=\frac{\partial \sigma_{r\theta}}{\partial r}+\frac{1}{r}\frac{\partial \sigma_{\varphi\theta}}{\partial \varphi}+\frac{1}{r\sin\varphi}\frac{\partial \sigma_\theta}{\partial \theta}+\frac{1}{r}(2\sigma_{\varphi\theta}\cot\varphi+3\sigma_{r\theta})+2\Lambda T_e\frac{1}{r\sin\varphi}\frac{\partial T_e}{\partial \theta}, \tag{3.20}$$

$$\sigma_r=\lambda(\varepsilon_r+\varepsilon_\varphi+\varepsilon_\theta)+2\mu\varepsilon_r-(3\lambda+2\mu)\alpha_T(T_l-T_0), \tag{3.21a}$$

$$\sigma_\varphi=\lambda(\varepsilon_r+\varepsilon_\varphi+\varepsilon_\theta)+2\mu\varepsilon_\varphi-(3\lambda+2\mu)\alpha_T(T_l-T_0), \tag{3.21b}$$

$$\sigma_\theta=\lambda(\varepsilon_r+\varepsilon_\varphi+\varepsilon_\theta)+2\mu\varepsilon_\theta-(3\lambda+2\mu)\alpha_T(T_l-T_0), \tag{3.21c}$$

$$\sigma_{r\varphi} = 2\mu\varepsilon_{r\varphi}, \sigma_{r\theta} = 2\mu\varepsilon_{r\theta}, \sigma_{\varphi\theta} = 2\mu\varepsilon_{\varphi\theta}, \tag{3.21d}$$

$$\frac{\partial \varepsilon_r}{\partial t} = \frac{\partial v_r}{\partial r}, \tag{3.21e}$$

$$\frac{\partial \varepsilon_\varphi}{\partial t} = \frac{1}{r}(v_r + \frac{\partial v_\varphi}{\partial \varphi}), \tag{3.21f}$$

$$\frac{\partial \varepsilon_\theta}{\partial t} = \frac{1}{r\sin\varphi}(\frac{\partial v_\theta}{\partial \theta} + v_r \sin\varphi + v_\varphi \cos\varphi), \tag{3.21g}$$

$$\frac{\partial \varepsilon_{r\varphi}}{\partial t} = \frac{1}{2}(\frac{1}{r}\frac{\partial v_r}{\partial \varphi} + \frac{\partial v_\varphi}{\partial r} - \frac{v_\varphi}{r}), \tag{3.21h}$$

$$\frac{\partial \varepsilon_{\varphi\theta}}{\partial t} = \frac{1}{2r}(\frac{1}{\sin\varphi}\frac{\partial v_\varphi}{\partial \theta} + \frac{\partial v_\theta}{\partial \varphi} - v_\theta \cot\varphi), \tag{3.21i}$$

$$\frac{\partial \varepsilon_{r\theta}}{\partial t} = \frac{1}{2}(\frac{1}{r\sin\varphi}\frac{\partial v_r}{\partial \theta} + \frac{\partial v_\theta}{\partial r} - \frac{v_\theta}{r}). \tag{3.21j}$$

To develop the finite difference schemes, we need to construct a staggered grid shown as Figure 3.2, where v_r, q_e^r and q_l^r are placed at $(r_{i+1/2}, \theta_j, \varphi_k)$, v_θ, q_e^θ and q_l^θ are placed at $(r_i, \theta_{j+1/2}, \varphi_k)$, v_φ, q_e^φ and q_l^φ are placed at $(r_i, \theta_j, \varphi_{k+1/2})$, $\varepsilon_{r\theta}$ and $\sigma_{r\theta}$ are placed at $(r_{i+1/2}, \theta_{j+1/2}, \varphi_k)$, $\varepsilon_{r\varphi}$ and $\sigma_{r\varphi}$ are placed at $(r_{i+1/2}, \theta_j, \varphi_{k+1/2})$, $\varepsilon_{\varphi\theta}$ and $\sigma_{\varphi\theta}$ are placed at $(r_i, \theta_{j+1/2}, \varphi_{k+1/2})$, while $\varepsilon_r, \varepsilon_\theta, \varepsilon_\varphi, \sigma_r, \sigma_\theta, \sigma_\varphi, T_e$ and T_l are at $(r_i, \theta_j, \varphi_k)$. Here i, j and k are indices with $1 \le i \le N_r + 1$, $1 \le j \le N_\theta + 1$, $1 \le k \le N_\varphi + 1$.

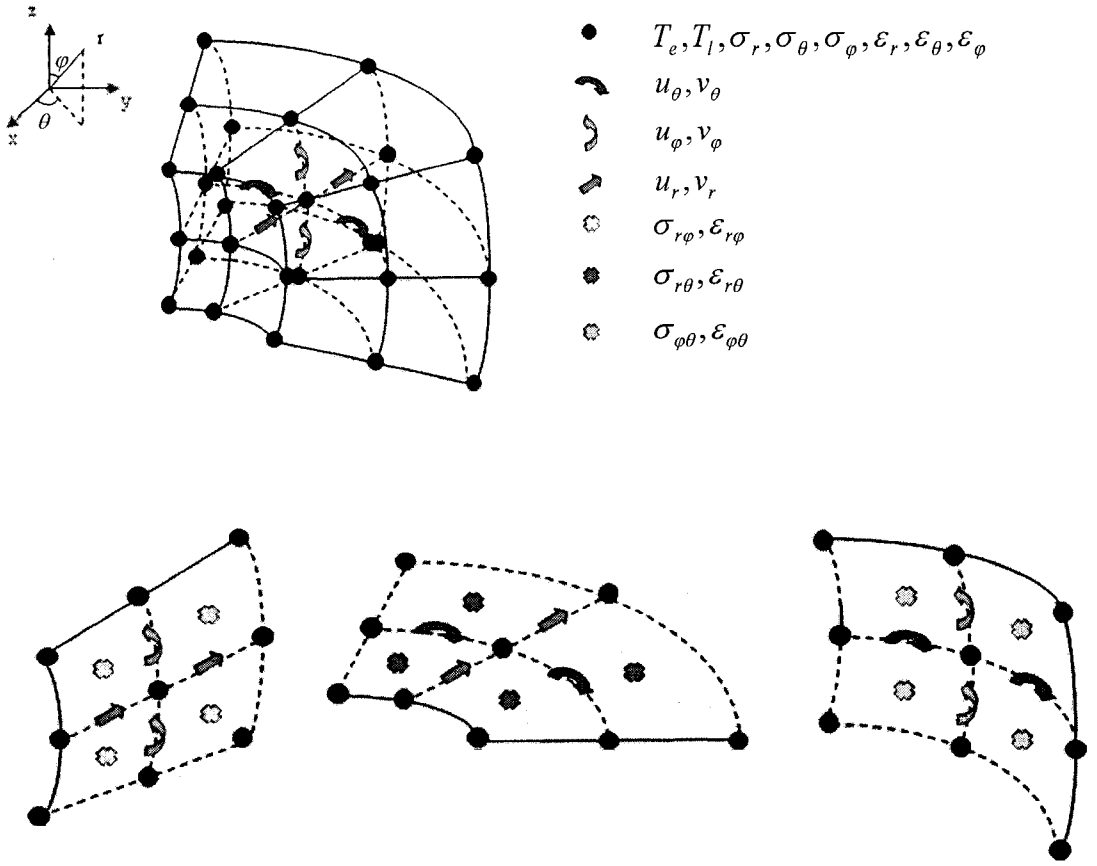


Figure 3.2 A 3D staggered grid and locations of variables for a micro sphere [Du 2008].

The staggered grid method is often employed in computational fluid dynamics to prevent the solution from oscillations [Patanka 1980]. For example, if v_r and ε_r in Equation (3.21e) are placed at the same location, employing a central finite difference scheme may produce a velocity component v_1, a wave solution, implying oscillation.

Here, we denote $(v_r)^n_{i+\frac{1}{2},j,k}$, $(v_\theta)^n_{i,j+\frac{1}{2},k}$ and $(v_\varphi)^n_{i,j,k+\frac{1}{2}}$ as numerical approximations of $v_r((i+1/2)\Delta r, j\Delta\theta, k\Delta\varphi, n\Delta t)$, $v_\theta(i\Delta r, (j+1/2)\Delta\theta, k\Delta\varphi, n\Delta t)$ and $v_\varphi(i\Delta r, j\Delta\theta, (k+1/2)\Delta\varphi, n\Delta t)$, respectively, where $\Delta t, \Delta r, \Delta\theta$, and $\Delta\varphi$ are time

increment and spatial step sizes, respectively. Similar notations are used for other variables. Furthermore, we introduce the finite difference operators, Δ_{-t}, δ_r, δ_θ, δ_φ, ∇_r, $\nabla_{\bar{r}}$, ∇_θ, $\nabla_{\bar{\theta}}$, ∇_φ and $\nabla_{\bar{\varphi}}$, as follows

$$\Delta_{-t}u^n_{i,j,k} = u^n_{i,j,k} - u^{n-1}_{i,j,k},$$

$$\delta_r u^n_{i,j,k} = u^n_{i+\frac{1}{2},j,k} - u^n_{i-\frac{1}{2},j,k},\ \delta_\theta u^n_{i,j,k} = u^n_{i,j+\frac{1}{2},k} - u^n_{i,j-\frac{1}{2},k},\ \delta_\varphi u^n_{i,j,k} = u^n_{i,j,k+\frac{1}{2}} - u^n_{i,j,k-\frac{1}{2}}$$

$$\nabla_r u^n_{i,j,k} = \frac{u^n_{i+1,j,k} - u^n_{i,j,k}}{\Delta r},\ \nabla_{\bar{r}} u^n_{i,j,k} = \frac{u^n_{i,j,k} - u^n_{i-1,j,k}}{\Delta r},$$

$$\nabla_\theta u^n_{i,j,k} = \frac{u^n_{i,j+1,k} - u^n_{i,j,k}}{\Delta \theta},\ \nabla_{\bar{\theta}} u^n_{i,j,k} = \frac{u^n_{i,j,k} - u^n_{i,j-1,k}}{\Delta \theta},$$

$$\nabla_\varphi u^n_{i,j,k} = \frac{u^n_{i,j,k+1} - u^n_{i,j,k}}{\Delta \varphi},\ \nabla_{\bar{\varphi}} u^n_{i,j,k} = \frac{u^n_{i,j,k} - u^n_{i,j,k-1}}{\Delta \varphi}.$$

3.1.4 Finite Difference Scheme

To avoid non-physical oscillations in the solution, we further follow the idea in [Wang 2006a, Wang 2006b, Wang 2008, Du 2008, Niu 2008a, 2008b, Zhang 2008] and employ a fourth-order compact finite difference scheme for stress derivatives $\frac{\partial \sigma_r}{\partial r}, \frac{\partial \sigma_{r\varphi}}{\partial \varphi}, \frac{\partial \sigma_{\varphi\theta}}{\partial \theta}$, etc. in Equations (3.18)-(3.21). For example, $\frac{\partial \sigma_r}{\partial r}$ can be obtained by solving the following tridiagonal system as Equations (3.22)-(3.23) (indices j and k are omitted).

$$\frac{1}{24}\frac{\partial(\sigma_r)_{i-1}}{\partial r} + \frac{11}{12}\frac{\partial(\sigma_r)_i}{\partial r} + \frac{1}{24}\frac{\partial(\sigma_r)_{i+1}}{\partial r} = \frac{(\sigma_r)_{i+1/2} - (\sigma_r)_{i-1/2}}{\Delta r},\ 2+\frac{1}{2} \le i \le N_r - \frac{1}{2}, \quad (3.22)$$

where

$$\frac{\partial(\sigma_r)_{3/2}}{\partial r} = \frac{(\sigma_r)_2 - (\sigma_r)_1}{\Delta r}, \qquad \frac{\partial(\sigma_r)_{N_r+1/2}}{\partial r} = \frac{(\sigma_r)_{N_r+1} - (\sigma_r)_{N_r}}{\Delta r}. \quad (3.23)$$

As such, the implicit finite difference schemes for solving Equations (3.18)-(3.21) can be written as Equations (3.24)-(3.30)

$$\rho\frac{1}{\Delta t}\Delta_{-t}(v_r)^{n+1}_{i+\frac{1}{2},j,k}=\frac{\partial(\sigma_r)^{n+1}_{i+\frac{1}{2},j,k}}{\partial r}+\frac{1}{r_{i+1/2}}\frac{\partial(\sigma_{r\varphi})^{n+1}_{i+\frac{1}{2},j,k}}{\partial\varphi}+\frac{1}{r_{i+1/2}\sin\varphi_k}\frac{\partial(\sigma_{r\theta})^{n+1}_{i+\frac{1}{2},j,k}}{\partial\theta}$$
$$+\frac{1}{r_{i+1/2}}[2\frac{(\sigma_r)^{n+1}_{i,j,k}+(\sigma_r)^{n+1}_{i+1,j,k}}{2}-\frac{(\sigma_\varphi)^{n+1}_{i,j,k}+(\sigma_\varphi)^{n+1}_{i+1,j,k}}{2}$$
$$-\frac{(\sigma_\theta)^{n+1}_{i,j,k}+(\sigma_\theta)^{n+1}_{i+1,j,k}}{2}+\frac{(\sigma_{r\varphi})^{n+1}_{i+\frac{1}{2},j,k+\frac{1}{2}}+(\sigma_{r\varphi})^{n+1}_{i+\frac{1}{2},j,k-\frac{1}{2}}}{2}\cot\varphi_k]$$
$$+\Lambda\frac{1}{\Delta r}\delta_r((T_e)^{n+1}_{i+\frac{1}{2},j,k})^2, \tag{3.24}$$

$$\rho\frac{1}{\Delta t}\Delta_{-t}(v_\varphi)^{n+1}_{i,j,k+\frac{1}{2}}=\frac{\partial(\sigma_{r\varphi})^{n+1}_{i,j,k+\frac{1}{2}}}{\partial r}+\frac{1}{r_i}\frac{\partial(\sigma_\varphi)^{n+1}_{i,j,k+\frac{1}{2}}}{\partial\varphi}+\frac{1}{r_i\sin\varphi_{k+1/2}}\frac{\partial(\sigma_{\varphi\theta})^{n+1}_{i,j,k+\frac{1}{2}}}{\partial\theta}$$
$$+\frac{1}{r_i}[(\frac{(\sigma_\varphi)^{n+1}_{i,j,k+1}+(\sigma_\varphi)^{n+1}_{i,j,k}}{2}-\frac{(\sigma_\theta)^{n+1}_{i,j,k+1}+(\sigma_\theta)^{n+1}_{i,j,k}}{2})\cot\varphi_{k+1/2}$$
$$+3\frac{(\sigma_{r\varphi})^{n+1}_{i+\frac{1}{2},j,k+\frac{1}{2}}+(\sigma_{r\varphi})^{n+1}_{i-\frac{1}{2},j,k+\frac{1}{2}}}{2}]+\Lambda\frac{1}{r_i\Delta\varphi}\delta_\varphi((T_e)^{n+1}_{i,j,k+\frac{1}{2}})^2, \tag{3.25}$$

$$\rho\frac{1}{\Delta t}\Delta_{-t}(v_\theta)^{n+1}_{i,j+\frac{1}{2},k}=\frac{\partial(\sigma_{r\theta})^{n+1}_{i,j+\frac{1}{2},k}}{\partial r}+\frac{1}{r_i}\frac{\partial(\sigma_{\varphi\theta})^{n+1}_{i,j+\frac{1}{2},k}}{\partial\varphi}+\frac{1}{r_i\sin\varphi_k}\frac{\partial(\sigma_\theta)^{n+1}_{i,j+\frac{1}{2},k}}{\partial\theta}$$
$$+\frac{1}{r_i}[2\frac{(\sigma_{\varphi\theta})^{n+1}_{i,j+\frac{1}{2},k+\frac{1}{2}}+(\sigma_{\varphi\theta})^{n+1}_{i,j+\frac{1}{2},k-\frac{1}{2}}}{2}\cot\varphi_k$$
$$+3\frac{(\sigma_{r\theta})^{n+1}_{i+\frac{1}{2},j+\frac{1}{2},k}+(\sigma_{r\theta})^{n+1}_{i-\frac{1}{2},j+\frac{1}{2},k}}{2}]+\Lambda\frac{1}{r_i\sin\varphi_k\Delta\theta}\delta_\theta((T_e)^{n+1}_{i,j+\frac{1}{2},k})^2, \tag{3.26}$$

$$(\sigma_r)^{n+1}_{i,j,k}=\lambda[(\varepsilon_r)^{n+1}_{i,j,k}+(\varepsilon_\theta)^{n+1}_{i,j,k}+(\varepsilon_\varphi)^{n+1}_{i,j,k}]+2\mu(\varepsilon_r)^{n+1}_{i,j,k}-(3\lambda+2\mu)\alpha_T[(T_l)^{n+1}_{i,j,k}-T_0], \tag{3.27a}$$

$$(\sigma_\varphi)^{n+1}_{i,j,k}=\lambda[(\varepsilon_r)^{n+1}_{i,j,k}+(\varepsilon_\theta)^{n+1}_{i,j,k}+(\varepsilon_\varphi)^{n+1}_{i,j,k}]+2\mu(\varepsilon_\varphi)^{n+1}_{i,j,k}-(3\lambda+2\mu)\alpha_T[(T_l)^{n+1}_{i,j,k}-T_0], \tag{3.27b}$$

$$(\sigma_\theta)_{i,j,k}^{n+1}=\lambda[(\varepsilon_r)_{i,j,k}^{n+1}+(\varepsilon_\theta)_{i,j,k}^{n+1}+(\varepsilon_\varphi)_{i,j,k}^{n+1}]+2\mu(\varepsilon_\theta)_{i,j,k}^{n+1}-(3\lambda+2\mu)\alpha_T[(T_l)_{i,j,k}^{n+1}-T_0], \tag{3.27c}$$

$$(\sigma_{r\varphi})_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1}=2\mu(\varepsilon_{r\varphi})_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1}, \tag{3.28a}$$

$$(\sigma_{r\theta})_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+1}=2\mu(\varepsilon_{r\theta})_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+1}, \tag{3.28b}$$

$$(\sigma_{\varphi\theta})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1}=2\mu(\varepsilon_{\varphi\theta})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1}, \tag{3.28c}$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_r)_{i,j,k}^{n+1}=\frac{1}{\Delta r}\delta_r(v_r)_{i,j,k}^{n+1}, \tag{3.29a}$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_\varphi)_{i,j,k}^{n+1}=\frac{1}{r_i}[\frac{(v_r)_{i+\frac{1}{2},j,k}^{n+1}+(v_r)_{i-\frac{1}{2},j,k}^{n+1}}{2}+\frac{1}{\Delta\varphi}\delta_\varphi(v_\varphi)_{i,j,k}^{n+1}], \tag{3.29b}$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_\theta)_{i,j,k}^{n+1}=\frac{1}{r_i\sin\varphi_k}[\frac{(v_r)_{i+\frac{1}{2},j,k}^{n+1}+(v_r)_{i-\frac{1}{2},j,k}^{n+1}}{2}\sin\varphi_k$$

$$+\frac{(v_\varphi)_{i,j,k+\frac{1}{2}}^{n+1}+(v_\varphi)_{i,j,k-\frac{1}{2}}^{n+1}}{2}\cos\varphi_k+\frac{1}{\Delta\theta}\delta_\theta(v_\theta)_{i,j,k}^{n+1}], \tag{3.29c}$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_{r\varphi})_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1}=\frac{1}{2}[\frac{1}{r_{i+1/2}\Delta\varphi}\delta_\varphi(v_r)_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1}+\frac{1}{\Delta r}\delta_r(v_\varphi)_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1}$$

$$-\frac{1}{r_{i+1/2}}\frac{(v_\varphi)_{i+1,j,k+\frac{1}{2}}^{n+1}+(v_\varphi)_{i,j,k+\frac{1}{2}}^{n+1}}{2}], \tag{3.30a}$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_{\varphi\theta})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1}=\frac{1}{2r_i}[\frac{1}{\sin\varphi_{k+1/2}\Delta\theta}\delta_\theta(v_\varphi)_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1}+\frac{1}{\Delta\varphi}\delta_\varphi(v_\theta)_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1}$$

$$-\frac{(v_\theta)_{i,j+\frac{1}{2},k+1}^{n+1}+(v_\theta)_{i,j+\frac{1}{2},k}^{n+1}}{2}\cot\varphi_{k+1/2}], \tag{3.30b}$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_{r\theta})^{n+1}_{i+\frac{1}{2},j+\frac{1}{2},k}=\frac{1}{2}[\frac{1}{r_{i+1/2}\sin\varphi_k\Delta\theta}\delta_\theta(v_r)^{n+1}_{i+\frac{1}{2},j+\frac{1}{2},k}+\frac{1}{\Delta r}\delta_r(v_\theta)^{n+1}_{i+\frac{1}{2},j+\frac{1}{2},k}$$

$$-\frac{1}{r_{i+1/2}}\frac{(v_\theta)^{n+1}_{i+1,j+\frac{1}{2},k}+(v_\theta)^{n+1}_{i,j+\frac{1}{2},k}}{2}]. \tag{3.30c}$$

We can obtain the displacements, u and v, by the Euler backward scheme of Equations (3.17) as Equations (3.31)

$$\frac{1}{\Delta t}\Delta_{-t}(u_r)^{n+1}_{i+\frac{1}{2},j,k}=(v_r)^{n+1}_{i+\frac{1}{2},j,k}, \tag{3.31a}$$

$$\frac{1}{\Delta t}\Delta_{-t}(u_\theta)^{n+1}_{i,j+\frac{1}{2},k}=(v_\theta)^{n+1}_{i,j+\frac{1}{2},k}, \tag{3.31b}$$

$$\frac{1}{\Delta t}\Delta_{-t}(u_\varphi)^{n+1}_{i,j,k+\frac{1}{2}}=(v_\varphi)^{n+1}_{i,j,k+\frac{1}{2}}. \tag{3.31c}$$

On the other hand, we discretize the energy equations, Equations (3.5)-(3.12), as Equations (3.32)-(3.40)

$$C_{e0}\frac{\left|(T_e)^{n+1}_{i,j,k}\right|^3-\left|(T_e)^{n}_{i,j,k}\right|^3}{\frac{3}{2}T_0\Delta t\left[(T_e)^{n+1}_{i,j,k}+(T_e)^{n}_{i,j,k}\right]}$$

$$=Q^{n+\frac{1}{2}}_{i,j,k}-\frac{\nabla_{\bar{r}}}{r_i^2}\left[r_i^2\frac{(q_e^r)^{n+1}_{i,j,k}+(q_e^r)^{n}_{i,j,k}}{2}\right]-\frac{\nabla_{\bar{\varphi}}}{r_i\sin\varphi_k}\left[\sin\varphi_k\frac{(q_e^\varphi)^{n+1}_{i,j,k}+(q_e^\varphi)^{n}_{i,j,k}}{2}\right]$$

$$-\frac{\nabla_{\bar{\theta}}}{r_i\sin\varphi_k}\left[\frac{(q_e^\theta)^{n+1}_{i,j,k}+(q_e^\theta)^{n}_{i,j,k}}{2}\right]-G\left[\frac{(T_e)^{n+1}_{i,j,k}+(T_e)^{n}_{i,j,k}}{2}-\frac{(T_l)^{n+1}_{i,j,k}+(T_l)^{n}_{i,j,k}}{2}\right], \tag{3.32}$$

$$\tau_e\frac{(q_e^r)^{n+1}_{i,j,k}-(q_e^r)^{n}_{i,j,k}}{\Delta t}+\frac{(q_e^r)^{n+1}_{i,j,k}+(q_e^r)^{n}_{i,j,k}}{2}=-(k_e)^{n+\frac{1}{2}}_{i-\frac{1}{2},j,k}\cdot\nabla_r\left[\frac{(T_e)^{n+1}_{i,j,k}+(T_e)^{n}_{i,j,k}}{2}\right], \tag{3.33}$$

$$\tau_e\frac{(q_e^\theta)^{n+1}_{i,j,k}-(q_e^\theta)^{n}_{i,j,k}}{\Delta t}+\frac{(q_e^\theta)^{n+1}_{i,j,k}+(q_e^\theta)^{n}_{i,j,k}}{2}=-(k_e)^{n+\frac{1}{2}}_{i,j-\frac{1}{2},k}\cdot\frac{\nabla_\theta}{r_i\sin\varphi_k}\left[\frac{(T_e)^{n+1}_{i,j,k}+(T_e)^{n}_{i,j,k}}{2}\right], \tag{3.34}$$

$$\tau_e \frac{(q_e^{\varphi})_{i,j,k}^{n+1}-(q_e^{\varphi})_{i,j,k}^{n}}{\Delta t}+\frac{(q_e^{\varphi})_{i,j,k}^{n+1}+(q_e^{\varphi})_{i,j,k}^{n}}{2}=-(k_e)_{i,j,k-\frac{1}{2}}^{n+\frac{1}{2}}\cdot\frac{\nabla_{\varphi}}{r_i}\left[\frac{(T_e)_{i,j,k}^{n+1}+(T_e)_{i,j,k}^{n}}{2}\right], \tag{3.35}$$

$$\begin{aligned}
&C_l\frac{(T_l)_{i,j,k}^{n+1}-(T_l)_{i,j,k}^{n}}{\Delta t}\\
&=-\frac{\nabla_{\bar{r}}}{r_i^2}\left[r_i^2\frac{(q_l^r)_{i,j,k}^{n+1}+(q_l^r)_{i,j,k}^{n}}{2}\right]-\frac{\nabla_{\bar{\varphi}}}{r_i\sin\varphi_k}\left[\sin\varphi_k\frac{(q_l^{\varphi})_{i,j,k}^{n+1}+(q_l^{\varphi})_{i,j,k}^{n}}{2}\right]\\
&-\frac{\nabla_{\bar{\theta}}}{r_i\sin\varphi_k}\left[\frac{(q_l^{\theta})_{i,j,k}^{n+1}+(q_l^{\theta})_{i,j,k}^{n}}{2}\right]+G\left[\frac{(T_e)_{i,j,k}^{n+1}+(T_e)_{i,j,k}^{n}}{2}-\frac{(T_l)_{i,j,k}^{n+1}+(T_l)_{i,j,k}^{n}}{2}\right]\\
&-(3\lambda+2\mu)\alpha_T\frac{(T_l)_{i,j,k}^{n+1}+(T_l)_{i,j,k}^{n}}{2}\cdot\frac{\Delta_{-t}(\varepsilon_r)_{i,j,k}^{n+1}+\Delta_{-t}(\varepsilon_{\theta})_{i,j,k}^{n+1}+\Delta_{-t}(\varepsilon_{\varphi})_{i,j,k}^{n+1}}{\Delta t},
\end{aligned} \tag{3.36}$$

$$\tau_l \frac{(q_l^r)_{i,j,k}^{n+1}-(q_l^r)_{i,j,k}^{n}}{\Delta t}+\frac{(q_l^r)_{i,j,k}^{n+1}+(q_l^r)_{i,j,k}^{n}}{2}=-k_0\cdot\nabla_r\left[\frac{(T_l)_{i,j,k}^{n+1}+(T_l)_{i,j,k}^{n}}{2}\right], \tag{3.37}$$

$$\tau_l \frac{(q_l^{\theta})_{i,j,k}^{n+1}-(q_l^{\theta})_{i,j,k}^{n}}{\Delta t}+\frac{(q_l^{\theta})_{i,j,k}^{n+1}+(q_l^{\theta})_{i,j,k}^{n}}{2}=-k_0\cdot\frac{\nabla_{\theta}}{r_i\sin\varphi_k}\left[\frac{(T_l)_{i,j,k}^{n+1}+(T_l)_{i,j,k}^{n}}{2}\right], \tag{3.38}$$

$$\tau_l \frac{(q_l^{\varphi})_{i,j,k}^{n+1}-(q_l^{\varphi})_{i,j,k}^{n}}{\Delta t}+\frac{(q_l^{\varphi})_{i,j,k}^{n+1}+(q_l^{\varphi})_{i,j,k}^{n}}{2}=-k_0\cdot\frac{\nabla_{\varphi}}{r_i}\left[\frac{(T_l)_{i,j,k}^{n+1}+(T_l)_{i,j,k}^{n}}{2}\right]. \tag{3.39}$$

Here, $i=2,...,N_r$, $j=2,...,N_{\theta}$ and $k=2,...,N_{\varphi}$ in Equations (3.32)-(3.39) and

$$(k_e)_{i-\frac{1}{2},j,k}^{n+\frac{1}{2}}=\frac{1}{2}k_0\left|\frac{(T_e)_{i,j,k}^{n+1}+(T_e)_{i,j,k}^{n}}{(T_l)_{i,j,k}^{n+1}+(T_l)_{i,j,k}^{n}}\right|+\frac{1}{2}k_0\left|\frac{(T_e)_{i-1,j,k}^{n+1}+(T_e)_{i-1,j,k}^{n}}{(T_l)_{i-1,j,k}^{n+1}+(T_l)_{i-1,j,k}^{n}}\right|, \tag{3.40a}$$

$$(k_e)_{i,j-\frac{1}{2},k}^{n+\frac{1}{2}}=\frac{1}{2}k_0\left|\frac{(T_e)_{i,j,k}^{n+1}+(T_e)_{i,j,k}^{n}}{(T_l)_{i,j,k}^{n+1}+(T_l)_{i,j,k}^{n}}\right|+\frac{1}{2}k_0\left|\frac{(T_e)_{i,j-1,k}^{n+1}+(T_e)_{i,j-1,k}^{n}}{(T_l)_{i,j-1,k}^{n+1}+(T_l)_{i,j-1,k}^{n}}\right|, \tag{3.40b}$$

$$(k_e)_{i,j,k-\frac{1}{2}}^{n+\frac{1}{2}}=\frac{1}{2}k_0\left|\frac{(T_e)_{i,j,k}^{n+1}+(T_e)_{i,j,k}^{n}}{(T_l)_{i,j,k}^{n+1}+(T_l)_{i,j,k}^{n}}\right|+\frac{1}{2}k_0\left|\frac{(T_e)_{i,j,k-1}^{n+1}+(T_e)_{i,j,k-1}^{n}}{(T_l)_{i,j,k-1}^{n+1}+(T_l)_{i,j,k-1}^{n}}\right|. \tag{3.40c}$$

It should be pointed out that Equation (3.32) is obtained based on the recently developed numerical method [Niu 2008b] for studying the thermal deformation of thin films under

the Cartesian coordinates. The scheme has been shown to satisfy a discrete analogous of the energy estimate and it is unconditionally stable.

To complete the formulation of our numerical method, we now turn our attention to the approximations of boundary and initial conditions given by Equations (3.41)-(3.45)

$$(\sigma_r)^n_{N_r+1,j,k} = 0, \quad 1 \le j \le N_\theta + 1, \quad 1 \le k \le N_\varphi + 1; \tag{3.41a}$$

$$(\sigma_r)^n_{1,j,k} = (\sigma_r)^n_{2,j,k}, \quad 1 \le j \le N_\theta + 1, \quad 1 \le k \le N_\varphi + 1; \tag{3.41b}$$

$$(\sigma_{r\theta})^n_{N_r+\frac{1}{2},j+\frac{1}{2},k} = 0, \quad 1 \le j \le N_\theta, \quad 1 \le k \le N_\varphi + 1; \tag{3.41c}$$

$$(\sigma_{r\theta})^n_{1+\frac{1}{2},j+\frac{1}{2},k} = (\sigma_{r\theta})^n_{2+\frac{1}{2},j+\frac{1}{2},k}, \quad 1 \le j \le N_\theta, \quad 1 \le k \le N_\varphi + 1; \tag{3.41d}$$

$$(\sigma_{r\varphi})^n_{N_r+\frac{1}{2},j,k+\frac{1}{2}} = 0, \quad 1 \le j \le N_\theta + 1, \quad 1 \le k \le N_\varphi; \tag{3.41e}$$

$$(\sigma_{r\varphi})^n_{1+\frac{1}{2},j,k+\frac{1}{2}} = (\sigma_{r\varphi})^n_{2+\frac{1}{2},j,k+\frac{1}{2}}, \quad 1 \le j \le N_\theta + 1, \quad 1 \le k \le N_\varphi; \tag{3.41f}$$

$$(\sigma_\theta)^n_{i,j,k} = (\sigma_\theta)^n_{i,j+N_\theta,k}, \quad 1 \le i \le N_r + 1, \quad 1 \le k \le N_\varphi + 1; \tag{3.42a}$$

$$(\sigma_{r\theta})^n_{i+\frac{1}{2},j+\frac{1}{2},k} = (\sigma_{r\theta})^n_{i+\frac{1}{2},j+N_\theta+\frac{1}{2},k}, \quad 1 \le i \le N_r, \quad 1 \le k \le N_\varphi + 1; \tag{3.42b}$$

$$(\sigma_{\varphi\theta})^n_{i,j+\frac{1}{2},k+\frac{1}{2}} = (\sigma_{\varphi\theta})^n_{i,j+N_\theta+\frac{1}{2},k+\frac{1}{2}}, \quad 1 \le i \le N_r + 1, \quad 1 \le k \le N_\varphi; \tag{3.42c}$$

$$(\sigma_\varphi)^n_{i,j,1} = (\sigma_\varphi)^n_{i,j,2}, \quad 1 \le i \le N_r + 1, \quad 1 \le j \le N_\theta + 1; \tag{3.43a}$$

$$(\sigma_\varphi)^n_{i,j,N_\varphi+1} = (\sigma_\varphi)^n_{i,j,N_\varphi}, \quad 1 \le i \le N_r + 1, \quad 1 \le j \le N_\theta + 1; \tag{3.43b}$$

$$(\sigma_{r\varphi})^n_{i+\frac{1}{2},j,1+\frac{1}{2}} = (\sigma_{r\varphi})^n_{i+\frac{1}{2},j,2+\frac{1}{2}}, \quad 1 \le i \le N_r, \quad 1 \le j \le N_\theta + 1; \tag{3.43c}$$

$$(\sigma_{r\varphi})^n_{i+\frac{1}{2},j,N_\varphi+\frac{1}{2}} = (\sigma_{r\varphi})^n_{i+\frac{1}{2},j,N_\varphi-\frac{1}{2}}, \quad 1 \le i \le N_r, \quad 1 \le j \le N_\theta + 1; \tag{3.43d}$$

$$(\sigma_{\varphi\theta})^n_{i,j+\frac{1}{2},1+\frac{1}{2}} = (\sigma_{\varphi\theta})^n_{i,j+\frac{1}{2},2+\frac{1}{2}}, \quad 1 \le i \le N_r + 1, \quad 1 \le j \le N_\theta; \tag{3.43e}$$

$$(\sigma_{\varphi\theta})^n_{i,j+\frac{1}{2},N_\varphi+\frac{1}{2}} = (\sigma_{\varphi\theta})^n_{i,j+\frac{1}{2},N_\varphi-\frac{1}{2}}, \quad 1 \le i \le N_r+1, \quad 1 \le j \le N_\theta; \tag{3.43f}$$

$$(T_e)^n_{1,j,k} = (T_e)^n_{2,j,k}, \quad (T_e)^n_{N_r+1,j,k} = (T_e)^n_{N_r,j,k}; \tag{3.44a}$$

$$(T_e)^n_{i,j,k} = (T_e)^n_{i,j+N_\theta,k}; \tag{3.44b}$$

$$(T_e)^n_{i,j,1} = (T_e)^n_{i,j,2}, \quad (T_e)^n_{i,j,N_\varphi+1} = (T_e)^n_{i,j,N_\varphi}; \tag{3.44c}$$

$$(T_l)^n_{1,j,k} = (T_l)^n_{2,j,k}, \quad (T_l)^n_{N_r+1,j,k} = (T_l)^n_{N_r,j,k}; \tag{3.45a}$$

$$(T_l)^n_{i,j,k} = (T_l)^n_{i,j+N_\theta,k}; \tag{3.45b}$$

$$(T_l)^n_{i,j,1} = (T_l)^n_{i,j,2}, \quad (T_l)^n_{i,j,N_\varphi+1} = (T_l)^n_{i,j,N_\varphi}; \tag{3.45c}$$

where $1 \le i \le N_r+1$, $1 \le j \le N_\theta+1$, $1 \le k \le N_\varphi+1$, for any time level *n*. The initial conditions are approximated as Equations (3.46)

$$(u_r)^0_{i+\frac{1}{2},j,k} = (u_\theta)^0_{i,j+\frac{1}{2},k} = (u_\varphi)^0_{i,j,k+\frac{1}{2}} = 0, \tag{3.46a}$$

$$(v_r)^0_{i+\frac{1}{2},j,k} = (v_\theta)^0_{i,j+\frac{1}{2},k} = (v_\varphi)^0_{i,j,k+\frac{1}{2}} = 0, \tag{3.46b}$$

$$(T_e)^0_{i,j,k} = (T_l)^0_{i,j,k} = T_0, \tag{3.46c}$$

$$(\overrightarrow{q_e})^0 = (\overrightarrow{q_l})^0 = 0, \tag{3.46d}$$

$$(\varepsilon_r)^0_{i,j,k} = (\varepsilon_\theta)^0_{i,j,k} = (\varepsilon_\varphi)^0_{i,j,k} = 0, \tag{3.46e}$$

$$(\sigma_r)^0_{i,j,k} = (\sigma_\theta)^0_{i,j,k} = (\sigma_\varphi)^0_{i,j,k} = 0, \tag{3.46f}$$

$$(\sigma_{r\theta})^0_{i+\frac{1}{2},j+\frac{1}{2},k} = (\varepsilon_{r\theta})^0_{i+\frac{1}{2},j+\frac{1}{2},k} = 0, \tag{3.46g}$$

$$(\sigma_{r\varphi})^0_{i+\frac{1}{2},j,k+\frac{1}{2}} = (\varepsilon_{r\varphi})^0_{i+\frac{1}{2},j,k+\frac{1}{2}} = 0, \tag{3.46h}$$

$$(\sigma_{\varphi\theta})^0_{i,j+\frac{1}{2},k+\frac{1}{2}} = (\varepsilon_{\varphi\theta})^0_{i,j+\frac{1}{2},k+\frac{1}{2}} = 0. \tag{3.46i}$$

where $1 \le i \le N_r + 1,\ 1 \le j \le N_\theta + 1,\ 1 \le k \le N_\varphi + 1$.

3.1.5 Algorithm

We should notice that Equations (3.24)-(3.26) are nonlinear since they contain nonlinear terms $\delta_r((T_e)^{n+1}_{i+\frac{1}{2},j,k})^2$, $\delta_\theta((T_e)^{n+1}_{i,j+\frac{1}{2},k})^2$ and $\delta_\varphi((T_e)^{n+1}_{i,j,k+\frac{1}{2}})^2$. Similarly, Equations (3.32)-(3.39) are also nonlinear. Therefore, the scheme must be solved iteratively. The following algorithm is an iterative method for solving the finite difference scheme at time level $n+1$ developed in the previous section.

Step 1: Set the values $(\varepsilon_r)^{n+1}$, $(\varepsilon_\theta)^{n+1}$, $(\varepsilon_\varphi)^{n+1}$, $(\varepsilon_{r\theta})^{n+1}$, $(\varepsilon_{r\varphi})^{n+1}$ and $(\varepsilon_{\varphi\theta})^{n+1}$. Solve $(q_e^r)^{n+1}$, $(q_e^\theta)^{n+1}$, $(q_e^\varphi)^{n+1}$, $(q_l^r)^{n+1}$, $(q_l^\theta)^{n+1}$ and $(q_l^\varphi)^{n+1}$ from Equations (3.33)-(3.35) and (3.37)-(3.39), respectively, and substitute them into Equations (3.32) and (3.36). Then solve for $(T_e)^{n+1}$ and $(T_l)^{n+1}$ iteratively.

Step 2: Solve for $(\sigma_r)^{n+1}$, $(\sigma_\theta)^{n+1}$, $(\sigma_\varphi)^{n+1}$, $(\sigma_{r\theta})^{n+1}$, $(\sigma_{r\varphi})^{n+1}$ and $(\sigma_{\varphi\theta})^{n+1}$ using Equations (3.27)-(3.28).

Step 3: Solve for derivatives of $(\sigma_r)^{n+1}$, $(\sigma_\theta)^{n+1}$, $(\sigma_\varphi)^{n+1}$, $(\sigma_{r\theta})^{n+1}$, $(\sigma_{r\varphi})^{n+1}$ and $(\sigma_{\varphi\theta})^{n+1}$ using Equations (3.22)-(3.23) or similar equations.

Step 4: Solve for $(v_r)^{n+1}$, $(v_\theta)^{n+1}$, and $(v_\varphi)^{n+1}$ using Equations (3.24)-(3.26).

Step 5: Update $(\varepsilon_r)^{n+1}$, $(\varepsilon_\theta)^{n+1}$, $(\varepsilon_\varphi)^{n+1}$, $(\varepsilon_{r\theta})^{n+1}$, $(\varepsilon_{r\varphi})^{n+1}$ and $(\varepsilon_{\varphi\theta})^{n+1}$ by using Equations (3.29)-(3.30).

Given the required accuracy E_1 (for temperature) and E_2 (for strain), repeat the

above steps until a convergent solution is obtained based on the following criteria (3.47)

$$|(T_e)_{i,j,k}^{n+1(new)} - (T_e)_{i,j,k}^{n+1(old)}| \le E_1, \tag{3.47a}$$

$$|(\varepsilon_r)_{i,j,k}^{n+1(new)} - (\varepsilon_r)_{i,j,k}^{n+1(old)}| \le E_2, \quad |(\varepsilon_\theta)_{i,j,k}^{n+1(new)} - (\varepsilon_\theta)_{i,j,k}^{n+1(old)}| \le E_2, \tag{3.47b}$$

$$|(\varepsilon_\varphi)_{i,j,k}^{n+1(new)} - (\varepsilon_\varphi)_{i,j,k}^{n+1(old)}| \le E_2, \quad |(\varepsilon_{r\theta})_{i,j,k}^{n+1(new)} - (\varepsilon_{r\theta})_{i,j,k}^{n+1(old)}| \le E_2, \tag{3.47c}$$

$$|(\varepsilon_{r\varphi})_{i,j,k}^{n+1(new)} - (\varepsilon_{r\varphi})_{i,j,k}^{n+1(old)}| \le E_2, \quad |(\varepsilon_{\varphi\theta})_{i,j,k}^{n+1(new)} - (\varepsilon_{\varphi\theta})_{i,j,k}^{n+1(old)}| \le E_2. \tag{3.47d}$$

3.2 Double-Layered Case

In this section, we consider a three-dimensional double-layered micro sphere exposed to ultrashort pulsed lasers, which has a perfectly thermal contact interface, as shown in Figure 3.3.

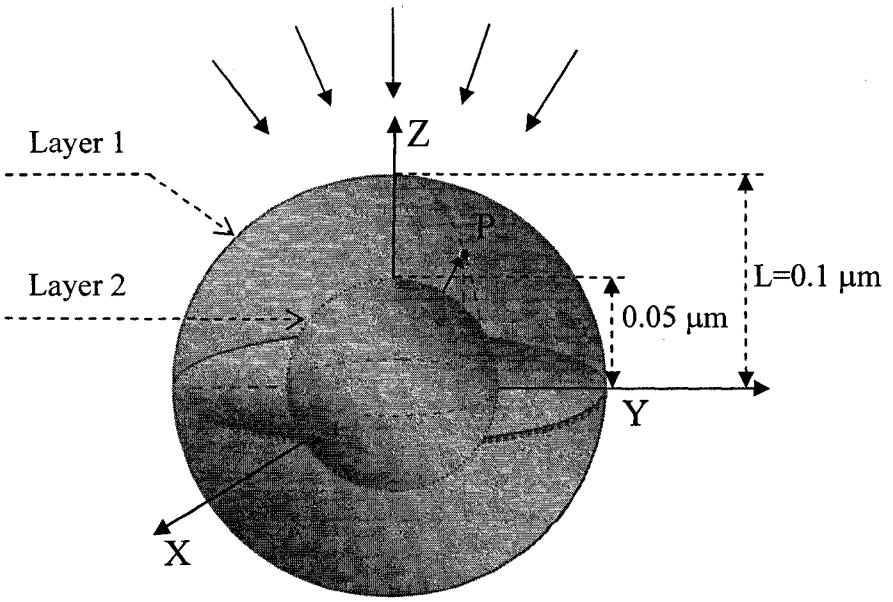


Figure 3.3 A 3D double-layered metal micro sphere.

3.2.1 Governing Equations

We extend the governing equations described in the previous section to the double-layered micro sphere case as follows

(1) Dynamic equations of motion can be written as Equations (3.48)-(3.51)

$$\rho^{(m)}\frac{\partial^2 u_r^{(m)}}{\partial t^2}=\frac{\partial \sigma_r^{(m)}}{\partial r}+\frac{1}{r}\frac{\partial \sigma_{r\varphi}^{(m)}}{\partial \varphi}+\frac{1}{r\sin\varphi}\frac{\partial \sigma_{r\theta}^{(m)}}{\partial \theta}+\frac{1}{r}(2\sigma_r^{(m)}-\sigma_\varphi^{(m)}-\sigma_\theta^{(m)}+\sigma_{r\varphi}^{(m)}\cot\varphi)+2\Lambda^{(m)}T_e^{(m)}\frac{\partial T_e^{(m)}}{\partial r}, \tag{3.48}$$

$$\rho^{(m)}\frac{\partial^2 u_\varphi^{(m)}}{\partial t^2}=\frac{\partial \sigma_{r\varphi}^{(m)}}{\partial r}+\frac{1}{r}\frac{\partial \sigma_\varphi^{(m)}}{\partial \varphi}+\frac{1}{r\sin\varphi}\frac{\partial \sigma_{\varphi\theta}^{(m)}}{\partial \theta}+\frac{1}{r}[(\sigma_\varphi^{(m)}-\sigma_\theta^{(m)})\cot\varphi+3\sigma_{r\varphi}^{(m)}]+2\Lambda^{(m)}T_e^{(m)}\frac{1}{r}\frac{\partial T_e^{(m)}}{\partial \varphi}, \tag{3.49}$$

$$\rho^{(m)}\frac{\partial^2 u_\theta^{(m)}}{\partial t^2}=\frac{\partial \sigma_{r\theta}^{(m)}}{\partial r}+\frac{1}{r}\frac{\partial \sigma_{\varphi\theta}^{(m)}}{\partial \varphi}+\frac{1}{r\sin\varphi}\frac{\partial \sigma_\theta^{(m)}}{\partial \theta}+\frac{1}{r}(2\sigma_{\varphi\theta}^{(m)}\cot\varphi+3\sigma_{r\theta}^{(m)})+2\Lambda^{(m)}T_e^{(m)}\frac{1}{r\sin\varphi}\frac{\partial T_e^{(m)}}{\partial \theta}, \tag{3.50}$$

where

$$\sigma_r^{(m)}=\lambda^{(m)}(\varepsilon_r^{(m)}+\varepsilon_\varphi^{(m)}+\varepsilon_\theta^{(m)})+2\mu^{(m)}\varepsilon_r^{(m)}-(3\lambda^{(m)}+2\mu^{(m)})\alpha_T^{(m)}(T_l^{(m)}-T_0), \tag{3.51a}$$

$$\sigma_\varphi^{(m)}=\lambda^{(m)}(\varepsilon_r^{(m)}+\varepsilon_\varphi^{(m)}+\varepsilon_\theta^{(m)})+2\mu^{(m)}\varepsilon_\varphi^{(m)}-(3\lambda^{(m)}+2\mu^{(m)})\alpha_T^{(m)}(T_l^{(m)}-T_0), \tag{3.51b}$$

$$\sigma_\theta^{(m)}=\lambda^{(m)}(\varepsilon_r^{(m)}+\varepsilon_\varphi^{(m)}+\varepsilon_\theta^{(m)})+2\mu^{(m)}\varepsilon_\theta^{(m)}-(3\lambda^{(m)}+2\mu^{(m)})\alpha_T^{(m)}(T_l^{(m)}-T_0), \tag{3.51c}$$

$$\sigma_{r\varphi}^{(m)}=2\mu^{(m)}\varepsilon_{r\varphi}^{(m)},\ \sigma_{r\theta}^{(m)}=2\mu^{(m)}\varepsilon_{r\theta}^{(m)},\ \sigma_{\varphi\theta}^{(m)}=2\mu^{(m)}\varepsilon_{\varphi\theta}^{(m)}, \tag{3.51d}$$

$$\varepsilon_r^{(m)}=\frac{\partial u_r^{(m)}}{\partial r}, \tag{3.51e}$$

$$\varepsilon_\varphi^{(m)}=\frac{1}{r}(u_r^{(m)}+\frac{\partial u_\varphi^{(m)}}{\partial \varphi}), \tag{3.51f}$$

$$\varepsilon_{\theta}^{(m)} = \frac{1}{r\sin\varphi}\left(\frac{\partial u_{\theta}^{(m)}}{\partial\theta} + u_{r}^{(m)}\sin\varphi + u_{\varphi}^{(m)}\cos\varphi\right), \tag{3.51g}$$

$$\varepsilon_{r\varphi}^{(m)} = \frac{1}{2}\left(\frac{1}{r}\frac{\partial u_{r}^{(m)}}{\partial\varphi} + \frac{\partial u_{\varphi}^{(m)}}{\partial r} - \frac{u_{\varphi}^{(m)}}{r}\right), \tag{3.51h}$$

$$\varepsilon_{\varphi\theta}^{(m)} = \frac{1}{2r}\left(\frac{1}{\sin\varphi}\frac{\partial u_{\varphi}^{(m)}}{\partial\theta} + \frac{\partial u_{\theta}^{(m)}}{\partial\varphi} - u_{\theta}^{(m)}\cot\varphi\right), \tag{3.51i}$$

$$\varepsilon_{r\theta}^{(m)} = \frac{1}{2}\left(\frac{1}{r\sin\varphi}\frac{\partial u_{r}^{(m)}}{\partial\theta} + \frac{\partial u_{\theta}^{(m)}}{\partial r} - \frac{u_{\theta}^{(m)}}{r}\right). \tag{3.51j}$$

Here, $m = 1,\ 2$ denote layer 1 and layer 2, respectively.

(2) Energy equations can be written as Equations (3.52)-(3.59)

$$C_{e}^{(m)}(T_{e})\frac{\partial T_{e}^{(m)}}{\partial t} = -\frac{1}{r^{2}}\frac{\partial}{\partial r}(r^{2}q_{e}^{r(m)}) - \frac{1}{r\sin\varphi}\frac{\partial q_{e}^{\theta(m)}}{\partial\theta} - \frac{1}{r\sin\varphi}\frac{\partial}{\partial\varphi}(q_{e}^{\varphi(m)}\sin\varphi) - G^{(m)}(T_{e}^{(m)} - T_{l}^{(m)}) + Q, \tag{3.52}$$

$$\tau_{e}^{(m)}\frac{\partial q_{e}^{r(m)}}{\partial t} + q_{e}^{r(m)} = -k_{e}^{(m)}\frac{\partial T_{e}^{(m)}}{\partial r}, \tag{3.53}$$

$$\tau_{e}^{(m)}\frac{\partial q_{e}^{\theta(m)}}{\partial t} + q_{e}^{\theta(m)} = -\frac{k_{e}^{(m)}}{r\sin\varphi}\frac{\partial T_{e}^{(m)}}{\partial\theta}, \tag{3.54}$$

$$\tau_{e}^{(m)}\frac{\partial q_{e}^{\varphi(m)}}{\partial t} + q_{e}^{\varphi(m)} = -\frac{k_{e}^{(m)}}{r}\frac{\partial T_{e}^{(m)}}{\partial\varphi}, \tag{3.55}$$

$$C_{l}^{(m)}\frac{\partial T_{l}^{(m)}}{\partial t} = -\frac{1}{r^{2}}\frac{\partial}{\partial r}(r^{2}q_{l}^{r(m)}) - \frac{1}{r\sin\varphi}\frac{\partial q_{l}^{\theta(m)}}{\partial\theta} - \frac{1}{r\sin\varphi}\frac{\partial}{\partial\varphi}(q_{l}^{\varphi(m)}\sin\varphi) + G^{(m)}(T_{e}^{(m)} - T_{l}^{(m)}) - (3\lambda^{(m)} + 2\mu^{(m)})\alpha_{T}^{(m)}T_{l}^{(m)}\frac{\partial}{\partial t}(\varepsilon_{r}^{(m)} + \varepsilon_{\varphi}^{(m)} + \varepsilon_{\theta}^{(m)}), \tag{3.56}$$

$$\tau_{l}^{(m)}\frac{\partial q_{l}^{r(m)}}{\partial t} + q_{l}^{r(m)} = -k_{l}^{(m)}\frac{\partial T_{l}^{(m)}}{\partial r}, \tag{3.57}$$

$$\tau_l^{(m)} \frac{\partial q_l^{\theta(m)}}{\partial t} + q_l^{\theta(m)} = -\frac{k_l^{(m)}}{r \sin\varphi} \frac{\partial T_l^{(m)}}{\partial \theta}, \tag{3.58}$$

$$\tau_l^{(m)} \frac{\partial q_l^{\varphi(m)}}{\partial t} + q_l^{\varphi(m)} = -\frac{k_l^{(m)}}{r} \frac{\partial T_l^{(m)}}{\partial \varphi}, \tag{3.59}$$

where the heat source is given by Equation (3.60)

$$Q(r,\theta,\varphi,t) = 0.94J \frac{1-R}{t_p \zeta} \exp[-\frac{L-r}{\zeta} - 2.77(\frac{t-2t_p}{t_p})^2]\cos\varphi. \tag{3.60}$$

Here, $m = 1, 2$ denote layer 1 and layer 2, respectively. $C_e^{(m)}(T_e) = C_{e0}^{(m)}\left(\frac{T_e^{(m)}}{T_0}\right)$ is the electron heat capacity, $k_e^{(m)} = k_0^{(m)}\left(\frac{T_e^{(m)}}{T_l^{(m)}}\right)$ is the thermal conductivity, $G^{(m)}$ is the electron–lattice coupling factor, $C_l^{(m)}$ is the lattice heat capacity, respectively; Q is the energy absorption rate; J is the laser fluence; R is the surface reflectivity; t_p is the laser pulse duration; L is the radius of the micro sphere; ζ is the optical penetration depth; $\tau_e^{(m)}$ is the electron relaxation time and $\tau_l^{(m)}$ is the lattice relaxation time.

3.2.2 Boundary Conditions and Initial Conditions

The boundary conditions are assumed to be stress free and thermally insulated as Equations (3.61)

$$\sigma_r^{(1)} = 0, \quad \sigma_{r\varphi}^{(1)} = 0, \quad \sigma_{r\theta}^{(1)} = 0, \qquad at \ r = L, \tag{3.61a}$$

$$\sigma_\theta^{(m)} = \sigma_{\theta+2\pi}^{(m)}, \quad \sigma_{r\theta}^{(m)} = \sigma_{r\theta+2\pi}^{(m)}, \quad \sigma_{\varphi\theta}^{(m)} = \sigma_{\varphi\theta+2\pi}^{(m)}, \tag{3.61b}$$

$$\frac{\partial T_e^{(1)}}{\partial r} = 0, \quad \frac{\partial T_l^{(1)}}{\partial r} = 0, \qquad at \ r = L, \tag{3.61c}$$

$$(T_e^{(m)})_{r,\theta,\varphi,t} = (T_e^{(m)})_{r,\theta+2\pi,\varphi,t}, (T_l^{(m)})_{r,\theta,\varphi,t} = (T_l^{(m)})_{r,\theta+2\pi,\varphi,t}. \quad (3.61d)$$

Without loss of generality, we assume for simplicity as Equations (3.62)

$$\frac{\partial \sigma_r^{(2)}}{\partial r} = 0, \quad \frac{\partial \sigma_{r\theta}^{(2)}}{\partial r} = 0, \quad \frac{\partial \sigma_{r\varphi}^{(2)}}{\partial r} = 0, \quad \frac{\partial T_e^{(2)}}{\partial r} = 0, \quad \frac{\partial T_l^{(2)}}{\partial r} = 0, \qquad at \ r = 0, \quad (3.62a)$$

$$\frac{\partial \sigma_\varphi^{(m)}}{\partial \varphi} = 0, \quad \frac{\partial \sigma_{\varphi\theta}^{(m)}}{\partial \varphi} = 0, \quad \frac{\partial \sigma_{r\varphi}^{(m)}}{\partial \varphi} = 0, \quad \frac{\partial T_e^{(m)}}{\partial \varphi} = 0, \quad \frac{\partial T_l^{(m)}}{\partial \varphi} = 0, \qquad at \ \varphi = 0, \pi, \quad (3.62b)$$

$$\frac{\partial T_e^{(m)}}{\partial \varphi} = 0, \quad \frac{\partial T_l^{(m)}}{\partial \varphi} = 0, \qquad at \ \varphi = 0, \pi. \quad (3.62c)$$

The initial conditions are assumed to be Equations (3.63)

$$T_e^{(m)} = T_l^{(m)} = T_0, \quad u_r^{(m)} = u_\varphi^{(m)} = u_\theta^{(m)} = 0, \quad \frac{\partial u_r^{(m)}}{\partial t} = \frac{\partial u_\varphi^{(m)}}{\partial t} = \frac{\partial u_\theta^{(m)}}{\partial t} = 0, \qquad at \ t = 0. \quad (3.63)$$

where $m = 1, 2$.

The interfacial equations are assumed to be perfectly thermal contacted at $r = L/2$, which are shown as Equations (3.64)

$$u^{(1)} = u^{(2)}, \ v^{(1)} = v^{(2)}, \quad (3.64a)$$

$$\sigma_r^{(1)} = \sigma_r^{(2)}, \ \sigma_{r\theta}^{(1)} = \sigma_{r\theta}^{(2)}, \ \sigma_{r\varphi}^{(1)} = \sigma_{r\varphi}^{(2)} \quad (3.64b)$$

$$T_e^{(1)} = T_e^{(2)}, \ q_e^{r(1)} = q_e^{r(2)}, \ T_l^{(1)} = T_l^{(2)}, \ q_l^{r(1)} = q_l^{r(2)}. \quad (3.64c)$$

3.2.3 Notations

Again, we employ the idea in [Wang 2006b, Wang 2007] and introduce three velocity components, $v_r^{(m)}$, $v_\theta^{(m)}$ and $v_\varphi^{(m)}$, into the model to prevent the solution from oscillations. The dynamic equations of motion, Equations (3.48)-(3.51), are rewritten as Equations (3.65)-(3.69)

$$v_r^{(m)} = \frac{\partial u_r^{(m)}}{\partial t}, \quad v_\theta^{(m)} = \frac{\partial u_\theta^{(m)}}{\partial t}, \quad v_\varphi^{(m)} = \frac{\partial u_\varphi^{(m)}}{\partial t}, \tag{3.65}$$

$$\rho^{(m)} \frac{\partial v_r^{(m)}}{\partial t} = \frac{\partial \sigma_r^{(m)}}{\partial r} + \frac{1}{r}\frac{\partial \sigma_{r\varphi}^{(m)}}{\partial \varphi} + \frac{1}{r\sin\varphi}\frac{\partial \sigma_{r\theta}^{(m)}}{\partial \theta}$$

$$+\frac{1}{r}(2\sigma_r^{(m)} - \sigma_\varphi^{(m)} - \sigma_\theta^{(m)} + \sigma_{r\varphi}^{(m)}\cot\varphi) + 2\Lambda^{(m)}T_e^{(m)}\frac{\partial T_e^{(m)}}{\partial r}, \tag{3.66}$$

$$\rho^{(m)} \frac{\partial v_\varphi^{(m)}}{\partial t} = \frac{\partial \sigma_{r\varphi}^{(m)}}{\partial r} + \frac{1}{r}\frac{\partial \sigma_\varphi^{(m)}}{\partial \varphi} + \frac{1}{r\sin\varphi}\frac{\partial \sigma_{\varphi\theta}^{(m)}}{\partial \theta}$$

$$+\frac{1}{r}[(\sigma_\varphi^{(m)} - \sigma_\theta^{(m)})\cot\varphi + 3\sigma_{r\varphi}^{(m)}] + 2\Lambda^{(m)}T_e^{(m)}\frac{1}{r}\frac{\partial T_e^{(m)}}{\partial \varphi}, \tag{3.67}$$

$$\rho^{(m)} \frac{\partial v_\theta^{(m)}}{\partial t} = \frac{\partial \sigma_{r\theta}^{(m)}}{\partial r} + \frac{1}{r}\frac{\partial \sigma_{\varphi\theta}^{(m)}}{\partial \varphi} + \frac{1}{r\sin\varphi}\frac{\partial \sigma_\theta^{(m)}}{\partial \theta}$$

$$+\frac{1}{r}(2\sigma_{\varphi\theta}^{(m)}\cot\varphi + 3\sigma_{r\theta}^{(m)}) + 2\Lambda^{(m)}T_e^{(m)}\frac{1}{r\sin\varphi}\frac{\partial T_e^{(m)}}{\partial \theta}, \tag{3.68}$$

$$\sigma_r^{(m)} = \lambda^{(m)}(\varepsilon_r^{(m)} + \varepsilon_\varphi^{(m)} + \varepsilon_\theta^{(m)}) + 2\mu^{(m)}\varepsilon_r^{(m)} - (3\lambda^{(m)} + 2\mu^{(m)})\alpha_T^{(m)}(T_l^{(m)} - T_0), \tag{3.69a}$$

$$\sigma_\varphi^{(m)} = \lambda^{(m)}(\varepsilon_r^{(m)} + \varepsilon_\varphi^{(m)} + \varepsilon_\theta^{(m)}) + 2\mu^{(m)}\varepsilon_\varphi^{(m)} - (3\lambda^{(m)} + 2\mu^{(m)})\alpha_T^{(m)}(T_l^{(m)} - T_0), \tag{3.69b}$$

$$\sigma_\theta^{(m)} = \lambda^{(m)}(\varepsilon_r^{(m)} + \varepsilon_\varphi^{(m)} + \varepsilon_\theta^{(m)}) + 2\mu^{(m)}\varepsilon_\theta^{(m)} - (3\lambda^{(m)} + 2\mu^{(m)})\alpha_T^{(m)}(T_l^{(m)} - T_0), \tag{3.69c}$$

$$\sigma_{r\varphi}^{(m)} = 2\mu^{(m)}\varepsilon_{r\varphi}^{(m)}, \sigma_{r\theta}^{(m)} = 2\mu^{(m)}\varepsilon_{r\theta}^{(m)}, \sigma_{\varphi\theta}^{(m)} = 2\mu^{(m)}\varepsilon_{\varphi\theta}^{(m)}, \tag{3.69d}$$

$$\frac{\partial \varepsilon_r^{(m)}}{\partial t} = \frac{\partial v_r^{(m)}}{\partial r}, \tag{3.69e}$$

$$\frac{\partial \varepsilon_\varphi^{(m)}}{\partial t} = \frac{1}{r}(v_r^{(m)} + \frac{\partial v_\varphi^{(m)}}{\partial \varphi}), \tag{3.69f}$$

$$\frac{\partial \varepsilon_\theta^{(m)}}{\partial t} = \frac{1}{r\sin\varphi}(\frac{\partial v_\theta^{(m)}}{\partial \theta} + v_r^{(m)}\sin\varphi + v_\varphi^{(m)}\cos\varphi), \tag{3.69g}$$

$$\frac{\partial \varepsilon_{r\varphi}^{(m)}}{\partial t} = \frac{1}{2}\left(\frac{1}{r}\frac{\partial v_r^{(m)}}{\partial \varphi} + \frac{\partial v_\varphi^{(m)}}{\partial r} - \frac{v_\varphi^{(m)}}{r}\right), \tag{3.69h}$$

$$\frac{\partial \varepsilon_{\varphi\theta}^{(m)}}{\partial t} = \frac{1}{2r}\left(\frac{1}{\sin\varphi}\frac{\partial v_\varphi^{(m)}}{\partial \theta} + \frac{\partial v_\theta^{(m)}}{\partial \varphi} - v_\theta^{(m)}\cot\varphi\right), \tag{3.69i}$$

$$\frac{\partial \varepsilon_{r\theta}^{(m)}}{\partial t} = \frac{1}{2}\left(\frac{1}{r\sin\varphi}\frac{\partial v_r^{(m)}}{\partial \theta} + \frac{\partial v_\theta^{(m)}}{\partial r} - \frac{v_\theta^{(m)}}{r}\right). \tag{3.69j}$$

We apply the staggered grid shown as Figure 3.2, where $v_r^{(m)}$, $q_e^{r(m)}$ and $q_l^{r(m)}$ are placed at $(r_{i+1/2},\theta_j,\varphi_k)$, $v_\theta^{(m)}$, $q_e^{\theta(m)}$ and $q_l^{\theta(m)}$ are placed at $(r_i,\theta_{j+1/2},\varphi_k)$, $v_\varphi^{(m)}$, $q_e^{\varphi(m)}$ and $q_l^{\varphi(m)}$ are placed at $(r_i,\theta_j,\varphi_{k+1/2})$, $\varepsilon_{r\theta}^{(m)}$ and $\sigma_{r\theta}^{(m)}$ are placed at $(r_{i+1/2},\theta_{j+1/2},\varphi_k)$, $\varepsilon_{r\varphi}^{(m)}$ and $\sigma_{r\varphi}^{(m)}$ are placed at $(r_{i+1/2},\theta_j,\varphi_{k+1/2})$, $\varepsilon_{\varphi\theta}^{(m)}$ and $\sigma_{\varphi\theta}^{(m)}$ are placed at $(r_i,\theta_{j+1/2},\varphi_{k+1/2})$, while $\varepsilon_r^{(m)}, \varepsilon_\theta^{(m)}, \varepsilon_\varphi^{(m)}, \sigma_r^{(m)}, \sigma_\theta^{(m)}, \sigma_\varphi^{(m)}, T_e^{(m)}$ and $T_l^{(m)}$ are at (r_i,θ_j,φ_k). Here i, j and k are indices with $1 \le i \le N_r+1$, $1 \le j \le N_\theta+1$, $1 \le k \le N_\varphi+1$.

We denote $(v_r^{(m)})^n_{i+\frac{1}{2},j,k}$, $(v_\theta^{(m)})^n_{i,j+\frac{1}{2},k}$ and $(v_\varphi^{(m)})^n_{i,j,k+\frac{1}{2}}$ as numerical approximations of $v_r^{(m)}((i+1/2)\Delta r, j\Delta\theta, k\Delta\varphi, n\Delta t)$, $v_\theta^{(m)}(i\Delta r,(j+1/2)\Delta\theta,k\Delta\varphi,n\Delta t)$ and $v_\varphi^{(m)}(i\Delta r, j\Delta\theta,(k+1/2)\Delta\varphi,n\Delta t)$, respectively, where Δt, Δr, $\Delta\theta$, and $\Delta\varphi$ are time increment and spatial step sizes, respectively. Similar notations are used for other variables. Furthermore, we introduce the finite difference operators, Δ_{-t}, δ_r, δ_θ, δ_φ, ∇_r, $\nabla_{\bar{r}}$, ∇_θ, $\nabla_{\bar{\theta}}$, ∇_φ and $\nabla_{\bar{\varphi}}$, as following equations

$$\Delta_{-t}u^n_{i,j,k} = u^n_{i,j,k} - u^{n-1}_{i,j,k},$$

$$\delta_r u^n_{i,j,k} = u^n_{i+\frac{1}{2},j,k} - u^n_{i-\frac{1}{2},j,k},\ \delta_\theta u^n_{i,j,k} = u^n_{i,j+\frac{1}{2},k} - u^n_{i,j-\frac{1}{2},k},\ \delta_\varphi u^n_{i,j,k} = u^n_{i,j,k+\frac{1}{2}} - u^n_{i,j,k-\frac{1}{2}}$$

$$\nabla_r u^n_{i,j,k} = \frac{u^n_{i+1,j,k} - u^n_{i,j,k}}{\Delta r},\ \nabla_{\bar{r}} u^n_{i,j,k} = \frac{u^n_{i,j,k} - u^n_{i-1,j,k}}{\Delta r},$$

$$\nabla_\theta u^n_{i,j,k} = \frac{u^n_{i,j+1,k} - u^n_{i,j,k}}{\Delta \theta},\ \nabla_{\bar{\theta}} u^n_{i,j,k} = \frac{u^n_{i,j,k} - u^n_{i,j-1,k}}{\Delta \theta},$$

$$\nabla_\varphi u^n_{i,j,k} = \frac{u^n_{i,j,k+1} - u^n_{i,j,k}}{\Delta \varphi},\ \nabla_{\bar{\varphi}} u^n_{i,j,k} = \frac{u^n_{i,j,k} - u^n_{i,j,k-1}}{\Delta \varphi}.$$

3.2.4 Finite Difference Scheme

To avoid non-physical oscillations in the solution, we employ the same fourth-order compact finite difference scheme for stress derivatives $\frac{\partial \sigma_r}{\partial r}, \frac{\partial \sigma_{r\varphi}}{\partial \varphi}, \frac{\partial \sigma_{\varphi\theta}}{\partial \theta}$, etc. as in Section 3.1.4.

As such, the implicit finite difference schemes for solving Equations (3.66)-(3.69) can be written as Equations (3.70)-(3.76)

$$\begin{aligned}\rho^{(m)} \frac{1}{\Delta t} \Delta_{-t} (v_r^{(m)})^{n+1}_{i+\frac{1}{2},j,k} = & \frac{\partial (\sigma_r^{(m)})^{n+1}_{i+\frac{1}{2},j,k}}{\partial r} + \frac{1}{r_{i+1/2}} \frac{\partial (\sigma_{r\varphi}^{(m)})^{n+1}_{i+\frac{1}{2},j,k}}{\partial \varphi} + \frac{1}{r_{i+1/2} \sin \varphi_k} \frac{\partial (\sigma_{r\theta}^{(m)})^{n+1}_{i+\frac{1}{2},j,k}}{\partial \theta} \\ & + \frac{1}{r_{i+1/2}} [2 \frac{(\sigma_r^{(m)})^{n+1}_{i,j,k} + (\sigma_r^{(m)})^{n+1}_{i+1,j,k}}{2} - \frac{(\sigma_\varphi^{(m)})^{n+1}_{i,j,k} + (\sigma_\varphi^{(m)})^{n+1}_{i+1,j,k}}{2} \\ & - \frac{(\sigma_\theta^{(m)})^{n+1}_{i,j,k} + (\sigma_\theta^{(m)})^{n+1}_{i+1,j,k}}{2} + \frac{(\sigma_{r\varphi}^{(m)})^{n+1}_{i+\frac{1}{2},j,k+\frac{1}{2}} + (\sigma_{r\varphi}^{(m)})^{n+1}_{i+\frac{1}{2},j,k-\frac{1}{2}}}{2} \cot \varphi_k] \\ & + \Lambda^{(m)} \frac{1}{\Delta r} \delta_r ((T_e^{(m)})^{n+1}_{i+\frac{1}{2},j,k})^2, \end{aligned} \tag{3.70}$$

$$\begin{aligned}\rho^{(m)} \frac{1}{\Delta t} \Delta_{-t} (v_\varphi^{(m)})^{n+1}_{i,j,k+\frac{1}{2}} = & \frac{\partial (\sigma_{r\varphi}^{(m)})^{n+1}_{i,j,k+\frac{1}{2}}}{\partial r} + \frac{1}{r_i} \frac{\partial (\sigma_\varphi^{(m)})^{n+1}_{i,j,k+\frac{1}{2}}}{\partial \varphi} + \frac{1}{r_i \sin \varphi_{k+1/2}} \frac{\partial (\sigma_{\varphi\theta}^{(m)})^{n+1}_{i,j,k+\frac{1}{2}}}{\partial \theta} \\ & + \frac{1}{r_i} [(\frac{(\sigma_\varphi^{(m)})^{n+1}_{i,j,k+1} + (\sigma_\varphi^{(m)})^{n+1}_{i,j,k}}{2} - \frac{(\sigma_\theta^{(m)})^{n+1}_{i,j,k+1} + (\sigma_\theta^{(m)})^{n+1}_{i,j,k}}{2}) \cot \varphi_{k+1/2} \end{aligned}$$

$$+3\frac{(\sigma_{r\varphi}^{(m)})_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1}+(\sigma_{r\varphi}^{(m)})_{i-\frac{1}{2},j,k+\frac{1}{2}}^{n+1}}{2}]+\Lambda^{(m)}\frac{1}{r_i\Delta\varphi}\delta_{\varphi}((T_e^{(m)})_{i,j,k+\frac{1}{2}}^{n+1})^2, \tag{3.71}$$

$$\rho^{(m)}\frac{1}{\Delta t}\Delta_{-t}(v_{\theta}^{(m)})_{i,j+\frac{1}{2},k}^{n+1}=\frac{\partial(\sigma_{r\theta}^{(m)})_{i,j+\frac{1}{2},k}^{n+1}}{\partial r}+\frac{1}{r_i}\frac{\partial(\sigma_{\varphi\theta}^{(m)})_{i,j+\frac{1}{2},k}^{n+1}}{\partial\varphi}+\frac{1}{r_i\sin\varphi_k}\frac{\partial(\sigma_{\theta}^{(m)})_{i,j+\frac{1}{2},k}^{n+1}}{\partial\theta}$$

$$+\frac{1}{r_i}[2\frac{(\sigma_{\varphi\theta}^{(m)})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1}+(\sigma_{\varphi\theta}^{(m)})_{i,j+\frac{1}{2},k-\frac{1}{2}}^{n+1}}{2}\cot\varphi_k+3\frac{(\sigma_{r\theta}^{(m)})_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+1}+(\sigma_{r\theta}^{(m)})_{i-\frac{1}{2},j+\frac{1}{2},k}^{n+1}}{2}]$$

$$+\Lambda^{(m)}\frac{1}{r_i\sin\varphi_k\Delta\theta}\delta_{\theta}((T_e^{(m)})_{i,j+\frac{1}{2},k}^{n+1})^2, \tag{3.72}$$

$$(\sigma_r^{(m)})_{i,j,k}^{n+1}=\lambda^{(m)}[(\varepsilon_r^{(m)})_{i,j,k}^{n+1}+(\varepsilon_{\theta}^{(m)})_{i,j,k}^{n+1}+(\varepsilon_{\varphi}^{(m)})_{i,j,k}^{n+1}]$$

$$+2\mu^{(m)}(\varepsilon_r^{(m)})_{i,j,k}^{n+1}-(3\lambda^{(m)}+2\mu^{(m)})\alpha_T^{(m)}[(T_l)_{i,j,k}^{n+1}-T_0], \tag{3.73a}$$

$$(\sigma_{\varphi}^{(m)})_{i,j,k}^{n+1}=\lambda^{(m)}[(\varepsilon_r^{(m)})_{i,j,k}^{n+1}+(\varepsilon_{\theta}^{(m)})_{i,j,k}^{n+1}+(\varepsilon_{\varphi}^{(m)})_{i,j,k}^{n+1}]$$

$$+2\mu^{(m)}(\varepsilon_{\varphi}^{(m)})_{i,j,k}^{n+1}-(3\lambda^{(m)}+2\mu^{(m)})\alpha_T^{(m)}[(T_l)_{i,j,k}^{n+1}-T_0], \tag{3.73b}$$

$$(\sigma_{\theta}^{(m)})_{i,j,k}^{n+1}=\lambda^{(m)}[(\varepsilon_r^{(m)})_{i,j,k}^{n+1}+(\varepsilon_{\theta}^{(m)})_{i,j,k}^{n+1}+(\varepsilon_{\varphi}^{(m)})_{i,j,k}^{n+1}]$$

$$+2\mu^{(m)}(\varepsilon_{\theta}^{(m)})_{i,j,k}^{n+1}-(3\lambda^{(m)}+2\mu^{(m)})\alpha_T^{(m)}[(T_l)_{i,j,k}^{n+1}-T_0], \tag{3.73c}$$

$$(\sigma_{r\varphi}^{(m)})_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1}=2\mu^{(m)}(\varepsilon_{r\varphi}^{(m)})_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1}, \tag{3.74a}$$

$$(\sigma_{r\theta}^{(m)})_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+1}=2\mu^{m)}(\varepsilon_{r\theta}^{(m)})_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+1}, \tag{3.74b}$$

$$(\sigma_{\varphi\theta}^{(m)})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1}=2\mu^{(m)}(\varepsilon_{\varphi\theta}^{(m)})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1}, \tag{3.74c}$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_r^{(m)})_{i,j,k}^{n+1}=\frac{1}{\Delta r}\delta_r(v_r^{(m)})_{i,j,k}^{n+1}, \tag{3.75a}$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_{\varphi}^{(m)})_{i,j,k}^{n+1}=\frac{1}{r_i}[\frac{(v_r^{(m)})_{i+\frac{1}{2},j,k}^{n+1}+(v_r^{(m)})_{i-\frac{1}{2},j,k}^{n+1}}{2}+\frac{1}{\Delta\varphi}\delta_{\varphi}(v_{\varphi}^{(m)})_{i,j,k}^{n+1}], \tag{3.75b}$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_{\theta}^{(m)})_{i,j,k}^{n+1} = \frac{1}{r_i \sin\varphi_k}[\frac{(v_r^{(m)})_{i+\frac{1}{2},j,k}^{n+1} + (v_r^{(m)})_{i-\frac{1}{2},j,k}^{n+1}}{2}\sin\varphi_k$$

$$+\frac{(v_{\varphi}^{(m)})_{i,j,k+\frac{1}{2}}^{n+1} + (v_{\varphi}^{(m)})_{i,j,k-\frac{1}{2}}^{n+1}}{2}\cos\varphi_k + \frac{1}{\Delta\theta}\delta_{\theta}(v_{\theta}^{(m)})_{i,j,k}^{n+1}], \quad (3.75c)$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_{r\varphi}^{(m)})_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1} = \frac{1}{2}[\frac{1}{r_{i+1/2}\Delta\varphi}\delta_{\varphi}(v_r^{(m)})_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1} + \frac{1}{\Delta r}\delta_r(v_{\varphi}^{(m)})_{i+\frac{1}{2},j,k+\frac{1}{2}}^{n+1}$$

$$-\frac{1}{r_{i+1/2}}\frac{(v_{\varphi}^{(m)})_{i+1,j,k+\frac{1}{2}}^{n+1} + (v_{\varphi}^{(m)})_{i,j,k+\frac{1}{2}}^{n+1}}{2}], \quad (3.76a)$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_{\varphi\theta}^{(m)})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1} = \frac{1}{2r_i}[\frac{1}{\sin\varphi_{k+1/2}\Delta\theta}\delta_{\theta}(v_{\varphi}^{(m)})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1} + \frac{1}{\Delta\varphi}\delta_{\varphi}(v_{\theta}^{(m)})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1}$$

$$-\frac{(v_{\theta}^{(m)})_{i,j+\frac{1}{2},k+1}^{n+1} + (v_{\theta}^{(m)})_{i,j+\frac{1}{2},k}^{n+1}}{2}\cot\varphi_{k+1/2}], \quad (3.76b)$$

$$\frac{1}{\Delta t}\Delta_{-t}(\varepsilon_{r\theta}^{(m)})_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+1} = \frac{1}{2}[\frac{1}{r_{i+1/2}\sin\varphi_k\Delta\theta}\delta_{\theta}(v_r^{(m)})_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+1} + \frac{1}{\Delta r}\delta_r(v_{\theta}^{(m)})_{i+\frac{1}{2},j+\frac{1}{2},k}^{n+1}$$

$$-\frac{1}{r_{i+1/2}}\frac{(v_{\theta}^{(m)})_{i+1,j+\frac{1}{2},k}^{n+1} + (v_{\theta}^{(m)})_{i,j+\frac{1}{2},k}^{n+1}}{2}]. \quad (3.76c)$$

We can obtain the displacements, $u^{(m)}$ and $v^{(m)}$, by the Euler backward scheme of Equations (3.65) as Equations (3.77)

$$\frac{1}{\Delta t}\Delta_{-t}(u_r^{(m)})_{i+\frac{1}{2},j,k}^{n+1} = (v_r^{(m)})_{i+\frac{1}{2},j,k}^{n+1}, \quad (3.77a)$$

$$\frac{1}{\Delta t}\Delta_{-t}(u_{\theta}^{(m)})_{i,j+\frac{1}{2},k}^{n+1} = (v_{\theta}^{(m)})_{i,j+\frac{1}{2},k}^{n+1} \quad (3.77b)$$

$$\frac{1}{\Delta t}\Delta_{-t}(u_{\varphi}^{(m)})_{i,j,k+\frac{1}{2}}^{n+1} = (v_{\varphi}^{(m)})_{i,j,k+\frac{1}{2}}^{n+1} \quad (3.77c)$$

On the other hand, we discretize the energy equations, Equations (3.5)-(3.12), as Equations (3.78)-(3.86)

$$C_{e0}^{(m)}\frac{\left|(T_e^{(m)})_{i,j,k}^{n+1}\right|^3-\left|(T_e^{(m)})_{i,j,k}^{n}\right|^3}{\frac{3}{2}T_0\Delta t\left[(T_e^{(m)})_{i,j,k}^{n+1}+(T_e^{(m)})_{i,j,k}^{n}\right]}$$
$$=-\frac{\nabla_{\bar{r}}}{r_i^2}\left[r_i^2\frac{(q_e^{r(m)})_{i,j,k}^{n+1}+(q_e^{r(m)})_{i,j,k}^{n}}{2}\right]-\frac{\nabla_{\bar{\theta}}}{r_i\sin\varphi_k}\left[\frac{(q_e^{\theta(m)})_{i,j,k}^{n+1}+(q_e^{\theta(m)})_{i,j,k}^{n}}{2}\right]$$
$$-\frac{\nabla_{\bar{\varphi}}}{r_i\sin\varphi_k}\left[\sin\varphi_k\frac{(q_e^{\varphi(m)})_{i,j,k}^{n+1}+(q_e^{\varphi(m)})_{i,j,k}^{n}}{2}\right]+Q_{i,j,k}^{n+\frac{1}{2}}$$
$$-G^{(m)}\left[\frac{(T_e^{(m)})_{i,j,k}^{n+1}+(T_e^{(m)})_{i,j,k}^{n}}{2}-\frac{(T_l^{(m)})_{i,j,k}^{n+1}+(T_l^{(m)})_{i,j,k}^{n}}{2}\right], \tag{3.78}$$

$$\tau_e^{(m)}\frac{(q_e^{r(m)})_{i,j,k}^{n+1}-(q_e^{r(m)})_{i,j,k}^{n}}{\Delta t}+\frac{(q_e^{r(m)})_{i,j,k}^{n+1}+(q_e^{r(m)})_{i,j,k}^{n}}{2}$$
$$=-(k_e^{(m)})_{i-\frac{1}{2},j,k}^{n+\frac{1}{2}}\cdot\nabla_r\left[\frac{(T_e^{(m)})_{i,j,k}^{n+1}+(T_e^{(m)})_{i,j,k}^{n}}{2}\right], \tag{3.79}$$

$$\tau_e^{(m)}\frac{(q_e^{\theta(m)})_{i,j,k}^{n+1}-(q_e^{\theta(m)})_{i,j,k}^{n}}{\Delta t}+\frac{(q_e^{\theta(m)})_{i,j,k}^{n+1}+(q_e^{\theta(m)})_{i,j,k}^{n}}{2}$$
$$=-(k_e^{(m)})_{i,j-\frac{1}{2},k}^{n+\frac{1}{2}}\cdot\frac{\nabla_\theta}{r_i\sin\varphi_k}\left[\frac{(T_e^{(m)})_{i,j,k}^{n+1}+(T_e^{(m)})_{i,j,k}^{n}}{2}\right], \tag{3.80}$$

$$\tau_e^{(m)}\frac{(q_e^{\varphi(m)})_{i,j,k}^{n+1}-(q_e^{\varphi(m)})_{i,j,k}^{n}}{\Delta t}+\frac{(q_e^{\varphi(m)})_{i,j,k}^{n+1}+(q_e^{\varphi(m)})_{i,j,k}^{n}}{2}$$
$$=-(k_e^{(m)})_{i,j,k-\frac{1}{2}}^{n+\frac{1}{2}}\cdot\frac{\nabla_\varphi}{r_i}\left[\frac{(T_e^{(m)})_{i,j,k}^{n+1}+(T_e^{(m)})_{i,j,k}^{n}}{2}\right], \tag{3.81}$$

$$C_l^{(m)}\frac{(T_l^{(m)})_{i,j,k}^{n+1}-(T_l^{(m)})_{i,j,k}^{n}}{\Delta t}$$
$$=-\frac{\nabla_{\bar{r}}}{r_i^2}\left[r_i^2\frac{(q_l^{r(m)})_{i,j,k}^{n+1}+(q_l^{r(m)})_{i,j,k}^{n}}{2}\right]-\frac{\nabla_{\bar{\varphi}}}{r_i\sin\varphi_k}\left[\sin\varphi_k\frac{(q_l^{\varphi(m)})_{i,j,k}^{n+1}+(q_l^{\varphi(m)})_{i,j,k}^{n}}{2}\right]$$
$$-\frac{\nabla_{\bar{\theta}}}{r_i\sin\varphi_k}\left[\frac{(q_l^{\theta(m)})_{i,j,k}^{n+1}+(q_l^{\theta(m)})_{i,j,k}^{n}}{2}\right]+G^{(m)}\left[\frac{(T_e^{(m)})_{i,j,k}^{n+1}+(T_e^{(m)})_{i,j,k}^{n}}{2}-\frac{(T_l^{(m)})_{i,j,k}^{n+1}+(T_l^{(m)})_{i,j,k}^{n}}{2}\right]$$
$$-(3\lambda^{(m)}+2\mu^{(m)})\alpha_T^{(m)}\frac{(T_l^{(m)})_{i,j,k}^{n+1}+(T_l^{(m)})_{i,j,k}^{n}}{2}\cdot\frac{\Delta_{-t}(\varepsilon_r^{(m)})_{i,j,k}^{n+1}+\Delta_{-t}(\varepsilon_\theta^{(m)})_{i,j,k}^{n+1}+\Delta_{-t}(\varepsilon_\varphi^{(m)})_{i,j,k}^{n+1}}{\Delta t},$$

(3.82)

$$\tau_l^{(m)}\frac{(q_l^{r(m)})_{i,j,k}^{n+1}-(q_l^{r(m)})_{i,j,k}^{n}}{\Delta t}+\frac{(q_l^{r(m)})_{i,j,k}^{n+1}+(q_l^{r(m)})_{i,j,k}^{n}}{2}$$
$$=-k_0\cdot\nabla_r\left[\frac{(T_l^{(m)})_{i,j,k}^{n+1}+(T_l^{(m)})_{i,j,k}^{n}}{2}\right],\tag{3.83}$$

$$\tau_l^{(m)}\frac{(q_l^{\theta(m)})_{i,j,k}^{n+1}-(q_l^{\theta(m)})_{i,j,k}^{n}}{\Delta t}+\frac{(q_l^{\theta(m)})_{i,j,k}^{n+1}+(q_l^{\theta(m)})_{i,j,k}^{n}}{2}$$
$$=-k_0\cdot\frac{\nabla_\theta}{r_i\sin\varphi_k}\left[\frac{(T_l^{(m)})_{i,j,k}^{n+1}+(T_l^{(m)})_{i,j,k}^{n}}{2}\right],\tag{3.84}$$

$$\tau_l^{(m)}\frac{(q_l^{\varphi(m)})_{i,j,k}^{n+1}-(q_l^{\varphi(m)})_{i,j,k}^{n}}{\Delta t}+\frac{(q_l^{\varphi(m)})_{i,j,k}^{n+1}+(q_l^{\varphi(m)})_{i,j,k}^{n}}{2}$$
$$=-k_0\cdot\frac{\nabla_\varphi}{r_i}\left[\frac{(T_l^{(m)})_{i,j,k}^{n+1}+(T_l^{(m)})_{i,j,k}^{n}}{2}\right].\tag{3.85}$$

$$(k_e^{(m)})_{i-\frac{1}{2},j,k}^{n+\frac{1}{2}}=\frac{1}{2}k_0\left|\frac{(T_e^{(m)})_{i,j,k}^{n+1}+(T_e^{(m)})_{i,j,k}^{n}}{(T_l^{(m)})_{i,j,k}^{n+1}+(T_l^{(m)})_{i,j,k}^{n}}\right|+\frac{1}{2}k_0\left|\frac{(T_e^{(m)})_{i-1,j,k}^{n+1}+(T_e^{(m)})_{i-1,j,k}^{n}}{(T_l^{(m)})_{i-1,j,k}^{n+1}+(T_l^{(m)})_{i-1,j,k}^{n}}\right|,\tag{3.86a}$$

$$(k_e^{(m)})_{i,j-\frac{1}{2},k}^{n+\frac{1}{2}}=\frac{1}{2}k_0\left|\frac{(T_e^{(m)})_{i,j,k}^{n+1}+(T_e^{(m)})_{i,j,k}^{n}}{(T_l^{(m)})_{i,j,k}^{n+1}+(T_l^{(m)})_{i,j,k}^{n}}\right|+\frac{1}{2}k_0\left|\frac{(T_e^{(m)})_{i,j-1,k}^{n+1}+(T_e^{(m)})_{i,j-1,k}^{n}}{(T_l^{(m)})_{i,j-1,k}^{n+1}+(T_l^{(m)})_{i,j-1,k}^{n}}\right|,\tag{3.86b}$$

$$(k_e^{(m)})_{i,j,k-\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{2}k_0\left|\frac{(T_e^{(m)})_{i,j,k}^{n+1}+(T_e^{(m)})_{i,j,k}^{n}}{(T_l^{(m)})_{i,j,k}^{n+1}+(T_l^{(m)})_{i,j,k}^{n}}\right| + \frac{1}{2}k_0\left|\frac{(T_e^{(m)})_{i,j,k-1}^{n+1}+(T_e^{(m)})_{i,j,k-1}^{n}}{(T_l^{(m)})_{i,j,k-1}^{n+1}+(T_l^{(m)})_{i,j,k-1}^{n}}\right|. \tag{3.86c}$$

where $i = 2,..., N_r$, $j = 2,..., N_\theta$, $k = 2,..., N_\varphi$ and $m = 1,\ 2$ for any time level n.

To complete the formulation of our numerical method, the boundary conditions are discretized as Equations (3.87)-(3.91)

$$(\sigma_r^{(1)})_{N_r+1,j,k}^{n} = 0, \quad 1 \le j \le N_\theta + 1, \quad 1 \le k \le N_\varphi + 1; \tag{3.87a}$$

$$(\sigma_r^{(2)})_{1,j,k}^{n} = (\sigma_r^{(2)})_{2,j,k}^{n}, \quad 1 \le j \le N_\theta + 1, \quad 1 \le k \le N_\varphi + 1; \tag{3.87b}$$

$$(\sigma_{r\theta}^{(1)})_{N_r+\frac{1}{2},j+\frac{1}{2},k}^{n} = 0, \quad 1 \le j \le N_\theta, \quad 1 \le k \le N_\varphi + 1; \tag{3.87c}$$

$$(\sigma_{r\theta}^{(2)})_{1+\frac{1}{2},j+\frac{1}{2},k}^{n} = (\sigma_{r\theta}^{(2)})_{2+\frac{1}{2},j+\frac{1}{2},k}^{n}, \quad 1 \le j \le N_\theta, \quad 1 \le k \le N_\varphi + 1; \tag{3.87d}$$

$$(\sigma_{r\varphi}^{(1)})_{N_r+\frac{1}{2},j,k+\frac{1}{2}}^{n} = 0, \quad 1 \le j \le N_\theta + 1, \quad 1 \le k \le N_\varphi; \tag{3.87e}$$

$$(\sigma_{r\varphi}^{(2)})_{1+\frac{1}{2},j,k+\frac{1}{2}}^{n} = (\sigma_{r\varphi}^{(2)})_{2+\frac{1}{2},j,k+\frac{1}{2}}^{n}, \quad 1 \le j \le N_\theta + 1, \quad 1 \le k \le N_\varphi; \tag{3.87f}$$

$$(\sigma_\theta^{(m)})_{i,j,k}^{n} = (\sigma_\theta^{(m)})_{i,j+N_\theta,k}^{n}, \quad 1 \le i \le N_r + 1, \quad 1 \le k \le N_\varphi + 1; \tag{3.88a}$$

$$(\sigma_{r\theta}^{(m)})_{i+\frac{1}{2},j+\frac{1}{2},k}^{n} = (\sigma_{r\theta}^{(m)})_{i+\frac{1}{2},j+N_\theta+\frac{1}{2},k}^{n}, \quad 1 \le i \le N_r, \quad 1 \le k \le N_\varphi + 1; \tag{3.88b}$$

$$(\sigma_{\varphi\theta}^{(m)})_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n} = (\sigma_{\varphi\theta}^{(m)})_{i,j+N_\theta+\frac{1}{2},k+\frac{1}{2}}^{n}, \quad 1 \le i \le N_r + 1, \quad 1 \le k \le N_\varphi; \tag{3.88c}$$

$$(\sigma_\varphi^{(m)})_{i,j,1}^{n} = (\sigma_\varphi^{(m)})_{i,j,2}^{n}, \quad 1 \le i \le N_r + 1, \quad 1 \le j \le N_\theta + 1; \tag{3.89a}$$

$$(\sigma_\varphi^{(m)})_{i,j,N_\varphi+1}^{n} = (\sigma_\varphi^{(m)})_{i,j,N_\varphi}^{n}, \quad 1 \le i \le N_r + 1, \quad 1 \le j \le N_\theta + 1; \tag{3.89b}$$

$$(\sigma_{r\varphi}^{(m)})_{i+\frac{1}{2},j,1+\frac{1}{2}}^{n} = (\sigma_{r\varphi}^{(m)})_{i+\frac{1}{2},j,2+\frac{1}{2}}^{n}, \quad 1 \le i \le N_r, \quad 1 \le j \le N_\theta + 1; \tag{3.89c}$$

$$(\sigma_{r\varphi}^{(m)})_{i+\frac{1}{2},j,N_\varphi+\frac{1}{2}}^{n} = (\sigma_{r\varphi}^{(m)})_{i+\frac{1}{2},j,N_\varphi-\frac{1}{2}}^{n}, \quad 1 \le i \le N_r, \quad 1 \le j \le N_\theta + 1; \tag{3.89d}$$

$$(\sigma_{\varphi\theta}^{(m)})^n_{i,j+\frac{1}{2},1+\frac{1}{2}} = (\sigma_{\varphi\theta}^{(m)})^n_{i,j+\frac{1}{2},2+\frac{1}{2}}, \quad 1 \le i \le N_r+1, \quad 1 \le j \le N_\theta; \tag{3.89e}$$

$$(\sigma_{\varphi\theta}^{(m)})^n_{i,j+\frac{1}{2},N_\varphi+\frac{1}{2}} = (\sigma_{\varphi\theta}^{(m)})^n_{i,j+\frac{1}{2},N_\varphi-\frac{1}{2}}, \quad 1 \le i \le N_r+1, \quad 1 \le j \le N_\theta; \tag{3.89f}$$

$$(T_e^{(2)})^n_{1,j,k} = (T_e^{(2)})^n_{2,j,k}, \quad (T_e^{(1)})^n_{N_r+1,j,k} = (T_e^{(1)})^n_{N_r,j,k}; \tag{3.90a}$$

$$(T_e^{(m)})^n_{i,j,k} = (T_e^{(m)})^n_{i,j+N_\theta,k}; \tag{3.90b}$$

$$(T_e^{(m)})^n_{i,j,1} = (T_e^{(m)})^n_{i,j,2}, \quad (T_e^{(m)})^n_{i,j,N_\varphi+1} = (T_e^{(m)})^n_{i,j,N_\varphi}; \tag{3.90c}$$

$$(T_l^{(2)})^n_{1,j,k} = (T_l^{(2)})^n_{2,j,k}, \quad (T_l^{(1)})^n_{N_r+1,j,k} = (T_l^{(1)})^n_{N_r,j,k}; \tag{3.91a}$$

$$(T_l^{(m)})^n_{i,j,k} = (T_l^{(m)})^n_{i,j+N_\theta,k}; \tag{3.91b}$$

$$(T_l^{(m)})^n_{i,j,1} = (T_l^{(m)})^n_{i,j,2}, \quad (T_l^{(m)})^n_{i,j,N_\varphi+1} = (T_l^{(m)})^n_{i,j,N_\varphi}; \tag{3.91c}$$

where $1 \le i \le N_r+1$, $1 \le j \le N_\theta+1$, $1 \le k \le N_\varphi+1$ and $m = 1, 2$, for any time level n. The initial conditions are approximated as Equations (3.92)

$$(u_r^{(m)})^0_{i+\frac{1}{2},j,k} = (u_\theta^{(m)})^0_{i,j+\frac{1}{2},k} = (u_\varphi^{(m)})^0_{i,j,k+\frac{1}{2}} = 0, \tag{3.92a}$$

$$(v_r^{(m)})^0_{i+\frac{1}{2},j,k} = (v_\theta^{(m)})^0_{i,j+\frac{1}{2},k} = (v_\varphi^{(m)})^0_{i,j,k+\frac{1}{2}} = 0, \tag{3.92b}$$

$$(T_e^{(m)})^0_{i,j,k} = (T_l^{(m)})^0_{i,j,k} = T_0, \tag{3.92c}$$

$$(\overrightarrow{q_e^{(m)}})^0 = (\overrightarrow{q_l^{(m)}})^0 = 0, \tag{3.92d}$$

$$(\varepsilon_r^{(m)})^0_{i,j,k} = (\varepsilon_\theta^{(m)})^0_{i,j,k} = (\varepsilon_\varphi^{(m)})^0_{i,j,k} = 0, \tag{3.92e}$$

$$(\sigma_r^{(m)})^0_{i,j,k} = (\sigma_\theta^{(m)})^0_{i,j,k} = (\sigma_\varphi^{(m)})^0_{i,j,k} = 0, \tag{3.92f}$$

$$(\sigma_{r\theta}^{(m)})^0_{i+\frac{1}{2},j+\frac{1}{2},k} = (\varepsilon_{r\theta}^{(m)})^0_{i+\frac{1}{2},j+\frac{1}{2},k} = 0, \tag{3.92g}$$

$$(\sigma_{r\varphi}^{(m)})^0_{i+\frac{1}{2},j,k+\frac{1}{2}} = (\varepsilon_{r\varphi}^{(m)})^0_{i+\frac{1}{2},j,k+\frac{1}{2}} = 0, \tag{3.92h}$$

$$(\sigma_{\varphi\theta}^{(m)})^0_{i,j+\frac{1}{2},k+\frac{1}{2}} = (\varepsilon_{\varphi\theta}^{(m)})^0_{i,j+\frac{1}{2},k+\frac{1}{2}} = 0, \tag{3.92i}$$

where $1 \le i \le N_r + 1,\ 1 \le j \le N_\theta + 1,\ 1 \le k \le N_\varphi + 1$, and $m = 1,\ 2$.

The interfacial conditions in Equations (3.64) are employed and then discretized in this way. First, the interfacial condition for velocity components $v_r^{(1)} = v_r^{(2)}$, $v_\theta^{(1)} = v_\theta^{(2)}$, and $v_\varphi^{(1)} = v_\varphi^{(2)}$, which can be expressed in Equations (3.93)

$$(v_r^{(1)})^{n+1}_{1+\frac{1}{2},j,k} = (v_r^{(2)})^{n+1}_{N_r+\frac{1}{2},j,k}, \tag{3.93a}$$

$$(v_\theta^{(1)})^{n+1}_{1,j+\frac{1}{2},k} = (v_\theta^{(2)})^{n+1}_{N_r+1,j+\frac{1}{2},k}, \tag{3.93b}$$

$$(v_\varphi^{(1)})^{n+1}_{1,j,k+\frac{1}{2}} = (v_\varphi^{(2)})^{n+1}_{N_r+1,j,k+\frac{1}{2}}, \tag{3.93c}$$

$$(\sigma_r^{(1)})^{n+1}_{1,j,k} = (\sigma_r^{(2)})^{n+1}_{N_r+1,j,k}, \tag{3.93d}$$

$$(\sigma_{r\theta}^{(1)})^{n+1}_{1+\frac{1}{2},j+\frac{1}{2},k} = (\sigma_{r\theta}^{(2)})^{n+1}_{N_r+\frac{1}{2},j+\frac{1}{2},k}, \tag{3.93e}$$

$$(\sigma_{r\varphi}^{(1)})^{n+1}_{1+\frac{1}{2},j,k+\frac{1}{2}} = (\sigma_{r\varphi}^{(2)})^{n+1}_{N_r+\frac{1}{2},j,k+\frac{1}{2}}, \tag{3.93f}$$

$$(T_e^{(1)})^{n+1}_{1,j,k} = (T_e^{(2)})^{n+1}_{N_r+1,j,k},\ (T_l^{(1)})^{n+1}_{1,j,k} = (T_l^{(2)})^{n+1}_{N_r+1,j,k}, \tag{3.93g}$$

$$(q_e^{r(1)})^{n+1}_{1+\frac{1}{2},j,k} = (q_e^{r(2)})^{n+1}_{N_r+\frac{1}{2},j,k},\ (q_l^{r(1)})^{n+1}_{1+\frac{1}{2},j,k} = (q_l^{r(2)})^{n+1}_{N_r+\frac{1}{2},j,k}, \tag{3.93h}$$

where $1 \le j \le N_\theta + 1,\ 1 \le k \le N_\varphi + 1$.

3.2.5 Algorithm

It should be pointed out that Equations (3.70)-(3.72) are nonlinear since they contain nonlinear terms $\delta_r((T_e)^{n+1}_{i+\frac{1}{2},j,k})^2$, $\delta_\theta((T_e)^{n+1}_{i,j+\frac{1}{2},k})^2$ and $\delta_\varphi((T_e)^{n+1}_{i,j,k+\frac{1}{2}})^2$. Similarly, Equations (3.78)-(3.85) are also nonlinear. Therefore, the scheme must be solved

iteratively. An iterative method for solving the above scheme at time level n + 1 is developed as follows:

Step 1: Set the values $(\varepsilon_r^{(m)})^{n+1}$, $(\varepsilon_\theta^{(m)})^{n+1}$, $(\varepsilon_\varphi^{(m)})^{n+1}$, $(\varepsilon_{r\theta}^{(m)})^{n+1}$, $(\varepsilon_{r\varphi}^{(m)})^{n+1}$ and $(\varepsilon_{\varphi\theta}^{(m)})^{n+1}$. Solve $(q_e^{r(m)})^{n+1}$, $(q_e^{\theta(m)})^{n+1}$, $(q_e^{\varphi(m)})^{n+1}$, $(q_l^{r(m)})^{n+1}$, $(q_l^{\theta(m)})^{n+1}$ and $(q_l^{\varphi(m)})^{n+1}$ from Equations (3.79)-(3.81) and (3.83)-(3.85), respectively, and substitute them into Equations (3.78) and (3.82). Then solve for $(T_e^{(m)})^{n+1}$ and $(T_l^{(m)})^{n+1}$ iteratively.

Step 2: Solve for $(\sigma_r^{(m)})^{n+1}$, $(\sigma_\theta^{(m)})^{n+1}$, $(\sigma_\varphi^{(m)})^{n+1}$, $(\sigma_{r\theta}^{(m)})^{n+1}$, $(\sigma_{r\varphi}^{(m)})^{n+1}$ and $(\sigma_{\varphi\theta}^{(m)})^{n+1}$ using Equations (3.73)-(3.74).

Step 3: Solve for derivatives of $(\sigma_r^{(m)})^{n+1}$, $(\sigma_\theta^{(m)})^{n+1}$, $(\sigma_\varphi^{(m)})^{n+1}$, $(\sigma_{r\theta}^{(m)})^{n+1}$, $(\sigma_{r\varphi}^{(m)})^{n+1}$ and $(\sigma_{\varphi\theta}^{(m)})^{n+1}$ using Equations (3.22)-(3.23) or similar equations.

Step 4: Solve for $(v_r^{(m)})^{n+1}$, $(v_\theta^{(m)})^{n+1}$, and $(v_\varphi^{(m)})^{n+1}$ using Equations (3.70)-(3.72).

Step 5: Update $(\varepsilon_r^{(m)})^{n+1}$, $(\varepsilon_\theta^{(m)})^{n+1}$, $(\varepsilon_\varphi^{(m)})^{n+1}$, $(\varepsilon_{r\theta}^{(m)})^{n+1}$, $(\varepsilon_{r\varphi}^{(m)})^{n+1}$ and $(\varepsilon_{\varphi\theta}^{(m)})^{n+1}$ using Equations (3.75)-(3.76).

Given the required accuracy E_1 (for temperature) and E_2 (for strain), repeat the above steps until a convergent solution is obtained based on the following criteria (3.94)

$$|(T_e^{(m)})_{i,j,k}^{n+1(new)} - (T_e^{(m)})_{i,j,k}^{n+1(old)}| \le E_1, \tag{3.94a}$$

$$|(\varepsilon_r^{(m)})_{i,j,k}^{n+1(new)} - (\varepsilon_r^{(m)})_{i,j,k}^{n+1(old)}| \le E_2, \quad |(\varepsilon_\theta^{(m)})_{i,j,k}^{n+1(new)} - (\varepsilon_\theta^{(m)})_{i,j,k}^{n+1(old)}| \le E_2, \tag{3.94b}$$

$$|(\varepsilon_\varphi^{(m)})_{i,j,k}^{n+1(new)} - (\varepsilon_\varphi^{(m)})_{i,j,k}^{n+1(old)}| \le E_2, \quad |(\varepsilon_{r\theta}^{(m)})_{i,j,k}^{n+1(new)} - (\varepsilon_{r\theta}^{(m)})_{i,j,k}^{n+1(old)}| \le E_2, \tag{3.94c}$$

$$|(\varepsilon_{r\varphi}^{(m)})_{i,j,k}^{n+1(new)} - (\varepsilon_{r\varphi}^{(m)})_{i,j,k}^{n+1(old)}| \le E_2, \quad |(\varepsilon_{\varphi\theta}^{(m)})_{i,j,k}^{n+1(new)} - (\varepsilon_{\varphi\theta}^{(m)})_{i,j,k}^{n+1(old)}| \le E_2. \tag{3.94d}$$

CHAPTER 4

NUMERICAL EXAMPLES

In this chapter, we will give the numerical examples based on the developed numerical schemes for thermal deformation in a gold micro sphere with a radius of 0.1 μm and in a double-layered micro sphere consisting of a gold layer on a chromium padding layer with a radius of 0.05 μm for each layer subjected to ultrashort pulsed lasers, respectively. In both examples, the laser irradiates a portion ($0 \le \varphi \le \pi/4$) of the upper hemisphere.

4.1 Three-Dimensional Single-Layered Case

4.1.1 Example Description

To demonstrate the applicability of the numerical scheme mentioned in Section 3.1, we investigate the temperature rise and thermal deformation in a three-dimensional micro sphere with a radius of 0.1 μm as shown in Figure 4.1.

Ultrashort-Pulsed Lasers

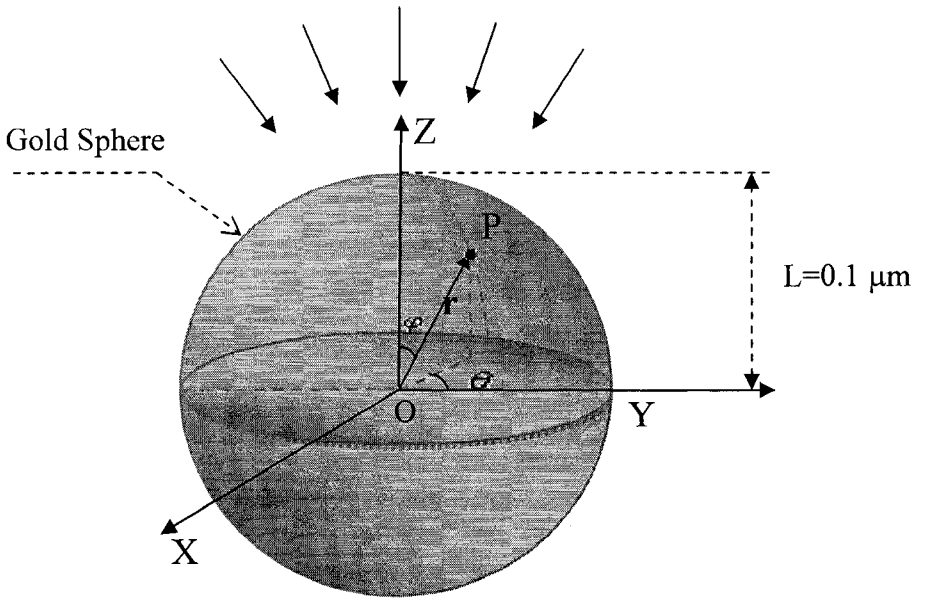


Figure 4.1 A 3D micro sphere with a radius $L = 0.1\ \mu m$.

The heat source was chosen to be Equation (4.1)

$$Q(r,\theta,\varphi,t) = 0.94J\frac{1-R}{t_p\zeta}\exp[-\frac{L-r}{\zeta} - 2.77(\frac{t-2t_p}{t_p})^2]\cos\varphi, \qquad (4.1)$$

where $L = 0.1\mu m$; $\zeta = 15.3nm$; $R = 0.93$; $J = 200\,J/m^2$, $500\,J/m^2$, $1000\,J/m^2$; and $t_p = 0.1ps$, $0.04ps$, $0.01ps$. The initial temperature T_0 for T_e and T_l is chosen to be $300K$.

The thermophysical properties for gold are listed in Table 4.1 [Tzou 2002, Chen 2002, Kaye 1973].

Table 4.1 Thermophysical properties of gold [Tzou 2002, Chen 2002, Kaye 1973].

Properties	Unit	Gold
ρ	kg/m^3	19300
Λ	J/m^3K^2	70
λ	Pa	199×10^9
μ	Pa	27×10^9
α_T	K^{-1}	14.2×10^{-6}
C_{e0}	J/m^3K^2	2.1×10^4
C_l	J/m^3K	2.5×10^6
G	W/m^3K	2.6×10^{16}
τ_e	ps	0.04
τ_l	ps	0.8
k_0	W/mK	315

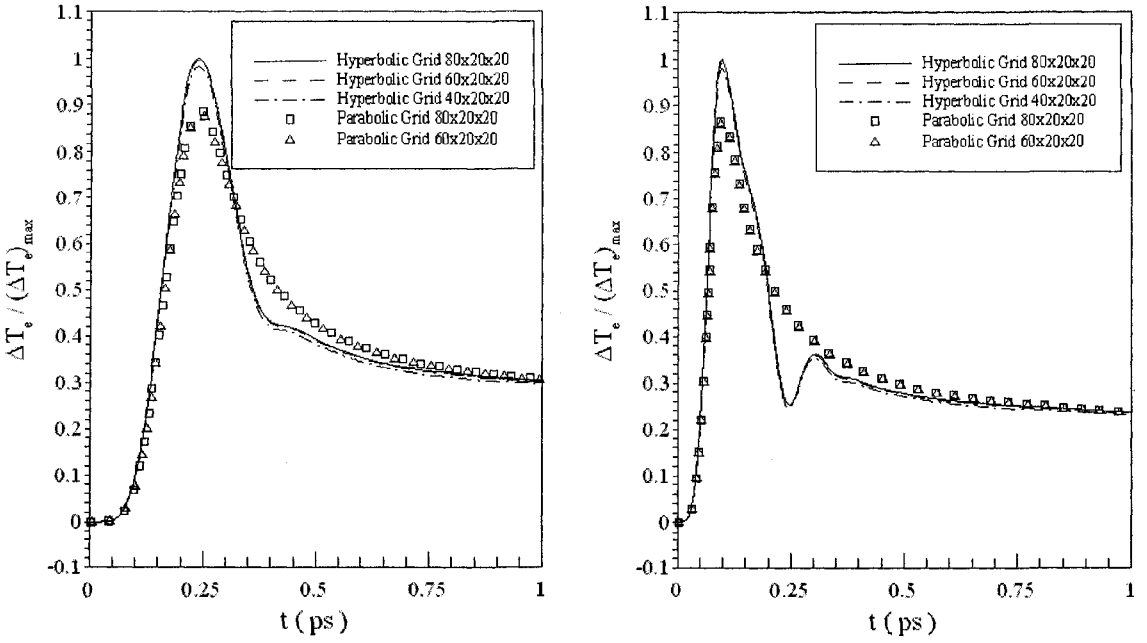
4.1.2 Results and Discussions

We assumed that the laser irradiates a portion $(0<r<L, 0\le\theta\le2\pi, 0\le\varphi\le\pi/4)$ of the upper hemisphere. Three meshes of $40\times20\times20$, $60\times20\times20$, $80\times20\times20$ in (r,θ,φ) were chosen in order to test the convergence of the scheme. The time increment was chosen to be $0.0025ps$. Three different values of laser fluences ($J=200J/m^2$, $500J/m^2$, $1000J/m^2$) were chosen to study the hot electron blast force. The

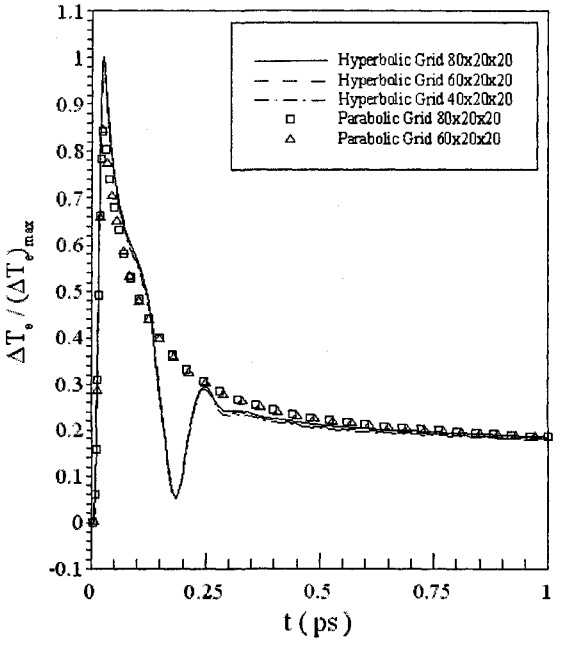
convergence criteria were chosen to be $E_1 = 10^{-5}$ for temperature and $E_1 = 10^{-12}$ for deformation.

Figure 4.2 shows the change in electron temperature ($\Delta T_e / (\Delta T_e)_{\max}$) at the point ($r = L,\ \theta = 0,\ \varphi = 0$) for various meshes ($40\times20\times20,\ 60\times20\times20,\ 80\times20\times20$) with laser fluence $J = 500 J/m^2$ and three different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, and (c) $t_p = 0.01ps$, respectively. The maximum temperature rise of T_e (i.e., $(\Delta T_e)_{\max}$) when $t_p = 0.1ps$ is about $4047\,K$. The electron temperature ($3552\,K$) at $t = 0.3ps$ when $t_p = 0.1ps$ is close to $3727\,K$ at $t = 0.1ps$ (starting at $t = -0.2ps$), which was obtained by Tzou et al. [Tzou 2002] who employed a parabolic two-step model. It is noted that the electron thermal relaxation time for gold is $0.04ps$. It can be seen from Figure 4.2a that when $t_p = 0.1ps$, there is not much difference in the solution between the hyperbolic model and the parabolic model. A small kink appears around $t = 0.35ps$ in the solution obtained by the hyperbolic model. This small kink can be also seen in the papers [Qiu 1993, Chen 2001]. From Figure 4.2b and 4.2c, we see that the kink becomes severe with decrease of the laser pulse duration.

Figure 4.3 shows the displacement (u_r) at the point ($r = L/2,\ \theta = 0,\ \varphi = 0$) versus time with three different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, and (c) $t_p = 0.01ps$, respectively. It can be seen from both Figures 4.2 and 4.3 that mesh size had no significant effect on the solution and hence the solution is convergent.

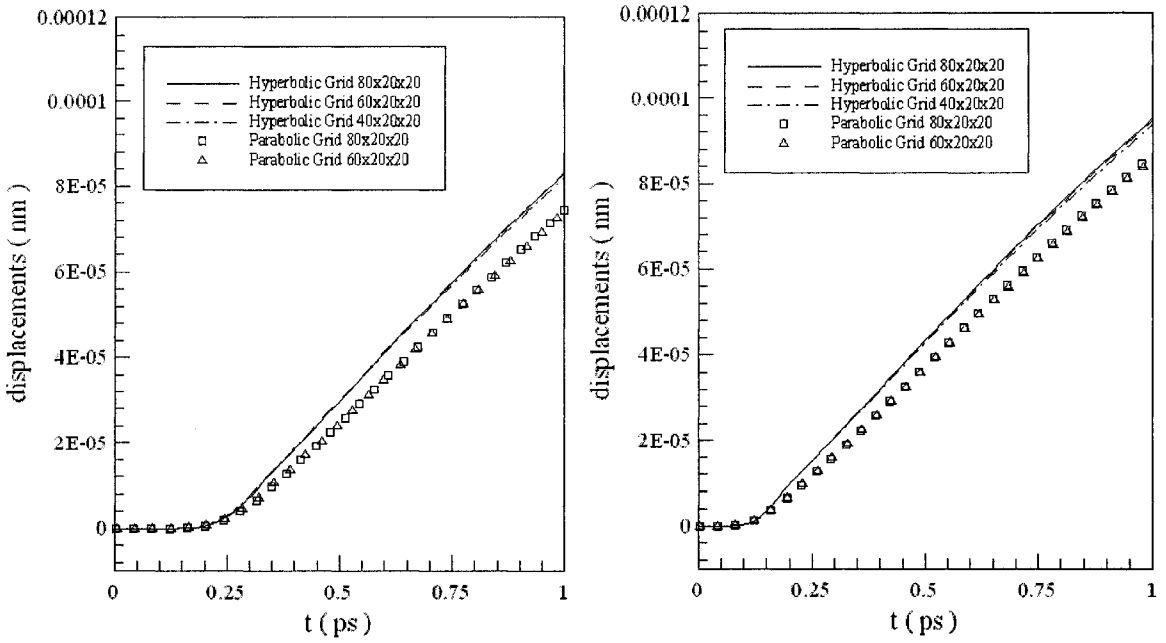


4.2(a) 4.2(b)



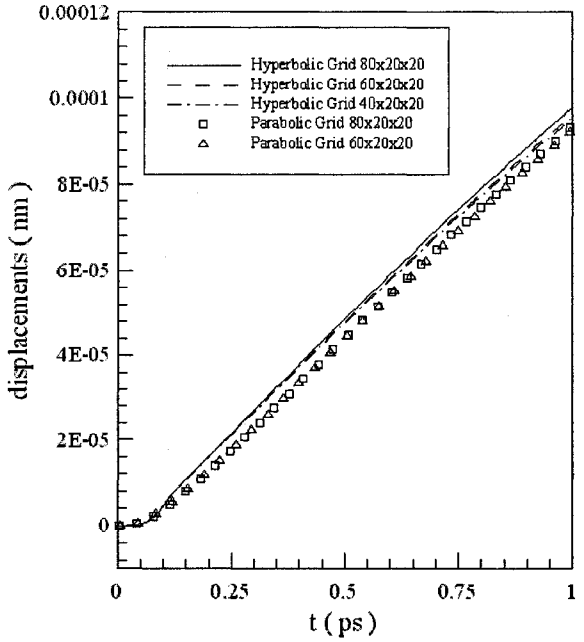
4.2(c)

Figure 4.2 Change in electron temperature at the top point of the sphere versus time with a laser fluence (J) of 500 J/m^2 and three different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, and (c) $t_p = 0.01ps$.



4.3(a)

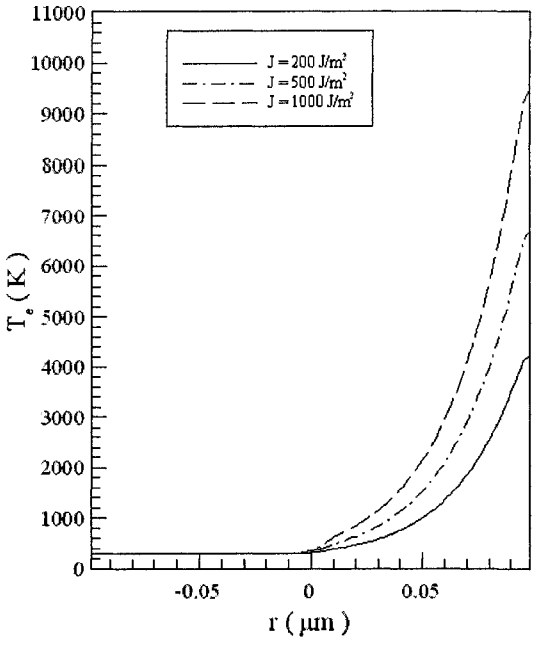
4.3(b)



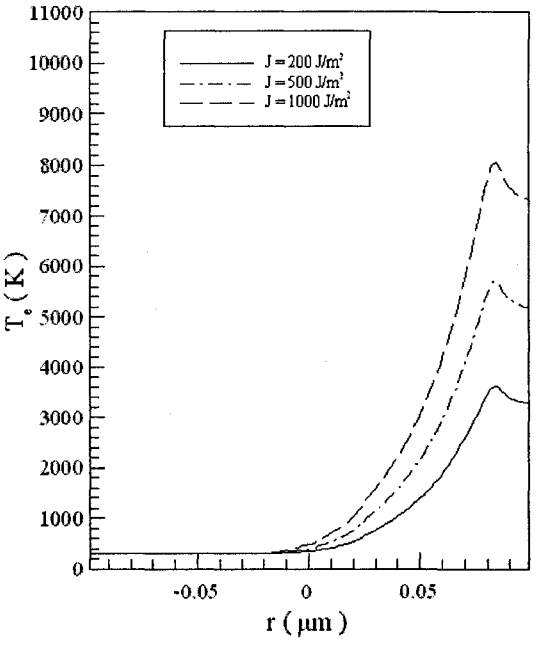
4.3(c)

Figure 4.3 Change in displacement (u_r) at the point ($r = L/2, \theta = 0, \varphi = 0$) of the sphere versus time with a laser fluence (J) of 500 J/m^2 and three different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, and (c) $t_p = 0.01ps$.

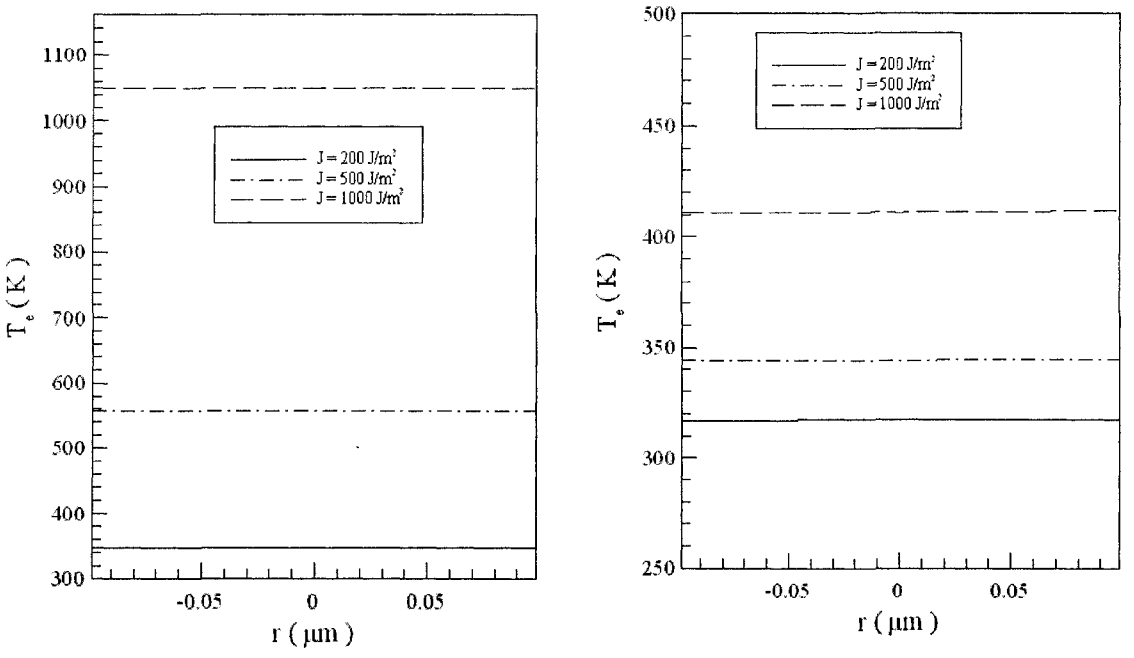
Figures 4.4 and 4.5 show electron temperature and lattice temperature along the diameter at $\varphi = 0$ and $\varphi = \pi$ with three different laser fluences of $J = 200 J/m^2$, $500 J/m^2$ and $1000 J/m^2$ at different times (a) $t = 0.025 ps$, (b) $t = 0.05 ps$, (c) $t = 5 ps$, and (d) $t = 10 ps$, respectively when laser pulse duration $t_p = 0.01 ps$. It can be seen that the electron temperature rises to its maximum at the beginning and then decreases while the lattice temperature rises gradually with time, and the electron temperature is almost uniform after $t = 5 ps$ along the diameter.



4.4(a)



4.4(b)



4.4(c) 4.4(d)

Figure 4.4 Electron temperature profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200 J/m^2$, $500 J/m^2$ and $1000 J/m^2$ when $t_p = 0.01ps$.

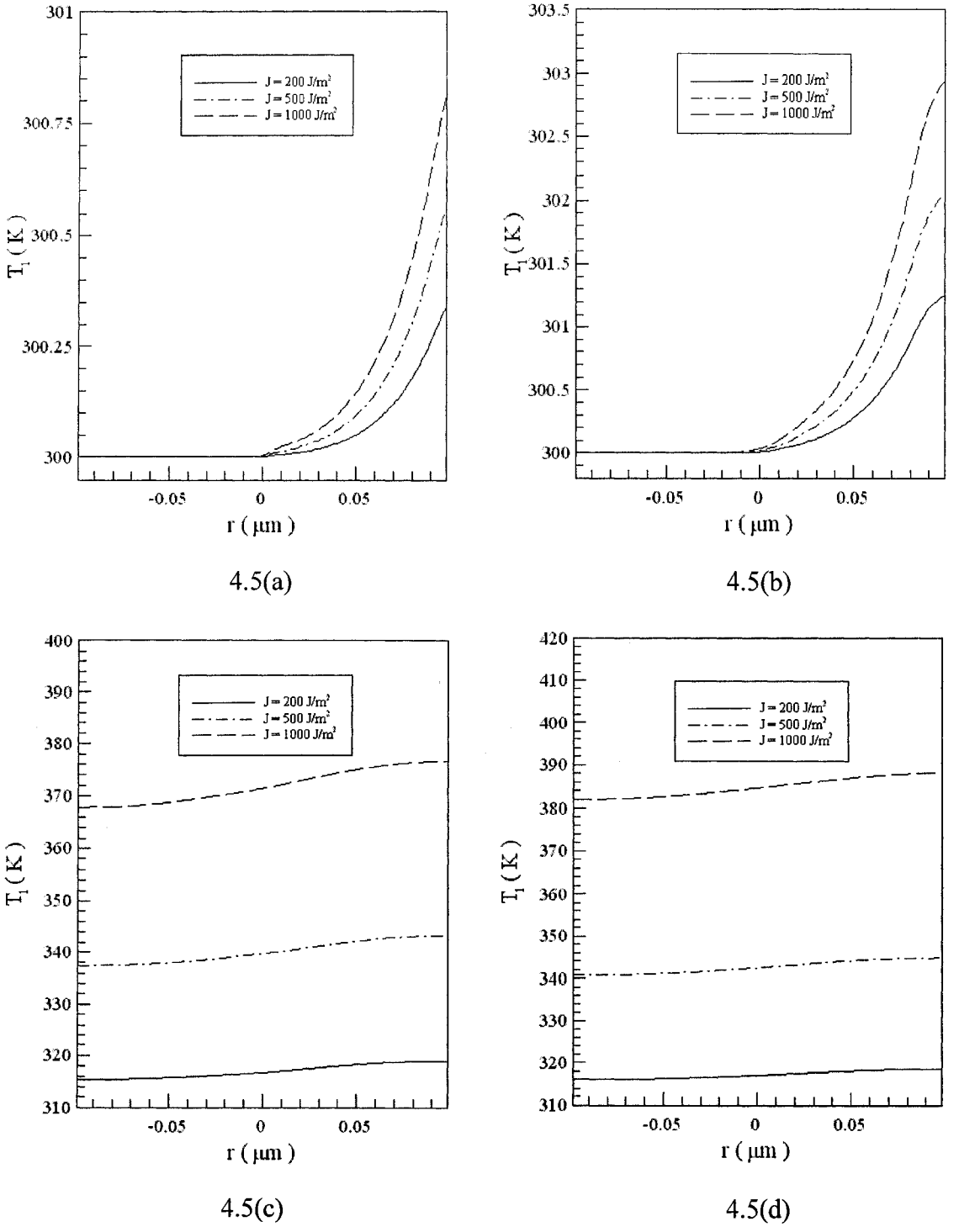


Figure 4.5 Lattice temperature profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200J/m^2$, $500J/m^2$ and $1000J/m^2$ when $t_p = 0.01ps$.

Figure 4.6 shows normal stress σ_r along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of 80×20×20 and three different laser fluences ($J = 200J/m^2$, $500J/m^2$, $1000J/m^2$) when laser pulse duration $t_p = 0.01ps$. Usually, numerical oscillations appear near the peak of the curve [Wang 2006a]. It can be seen from Figure 4.6 (particularly, Figure 4.6b-Figure 4.6d) that the normal stress σ_r does not show non-physical oscillations near the two peaks of the curve.

Figures 4.7-4.11 were plotted based on the results obtained in a mesh of 80×20×20 with a laser fluence of $J = 500J/m^2$. Figures 4.7 and 4.8 show contours of electron temperature distributions and lattice temperature distributions in the cross section of $\theta = 0$ and $\theta = \pi$ at $t_p = 0.01ps$ and four different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$, respectively. It can be seen from both figures that the heat is mainly transferred from the top surface to the bottom. Figures 4.9-4.11 show contours of displacement (u_r) in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$, respectively with three different laser pulse durations $t_p = 0.1ps$, $t_p = 0.04ps$, and $t_p = 0.01ps$. It can be seen from Figures 4.9-4.11 that the sphere is expanding along the r direction and the top portion of the sphere expands much more compared with the bottom. This is because the laser irradiates the top portion of the surface. Although the changes in electron temperature in Figure 4.2 are different with various pulse durations, the displacements shown in Figures 4.9-4.11 do not seem much different except for a slightly different scale.

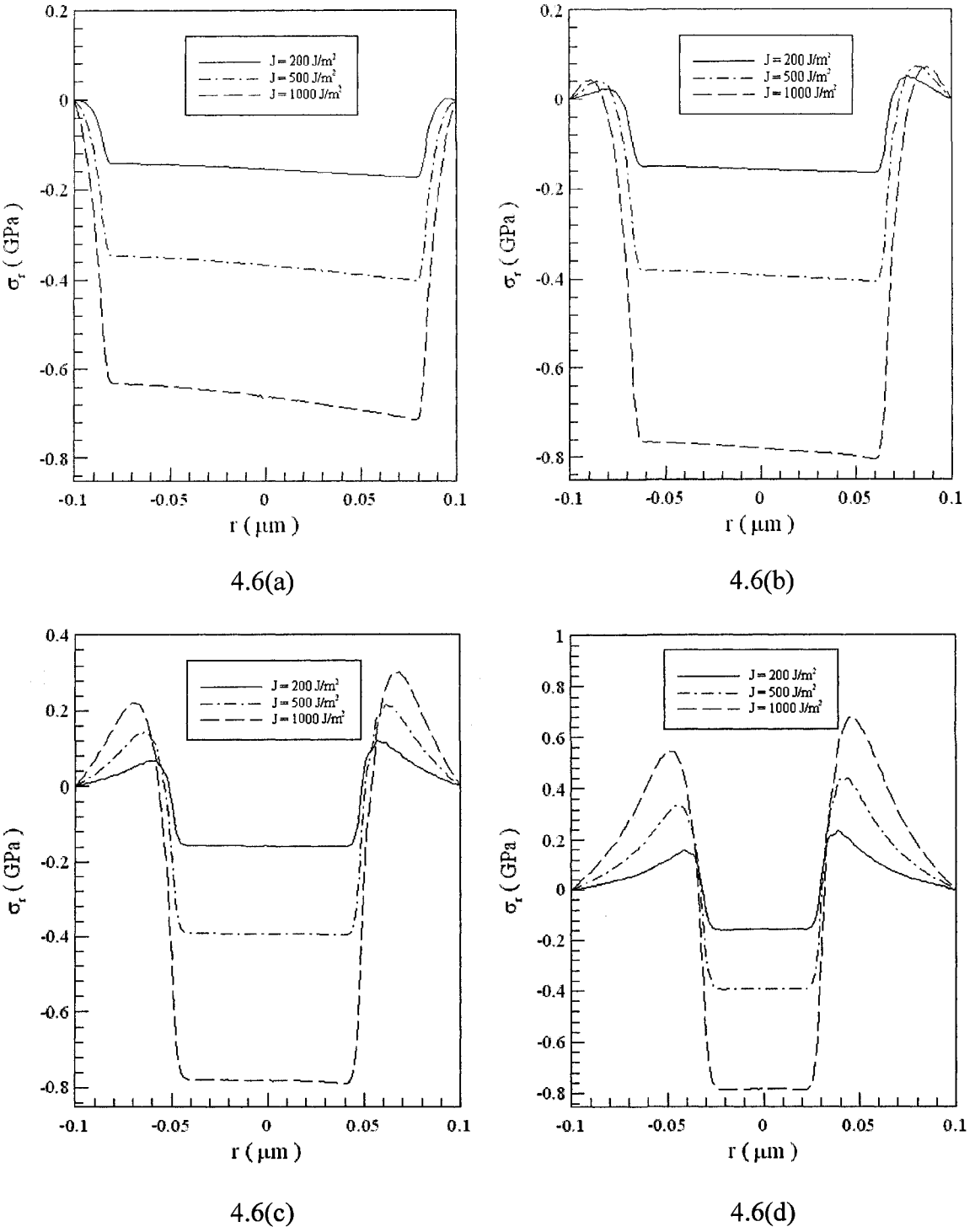
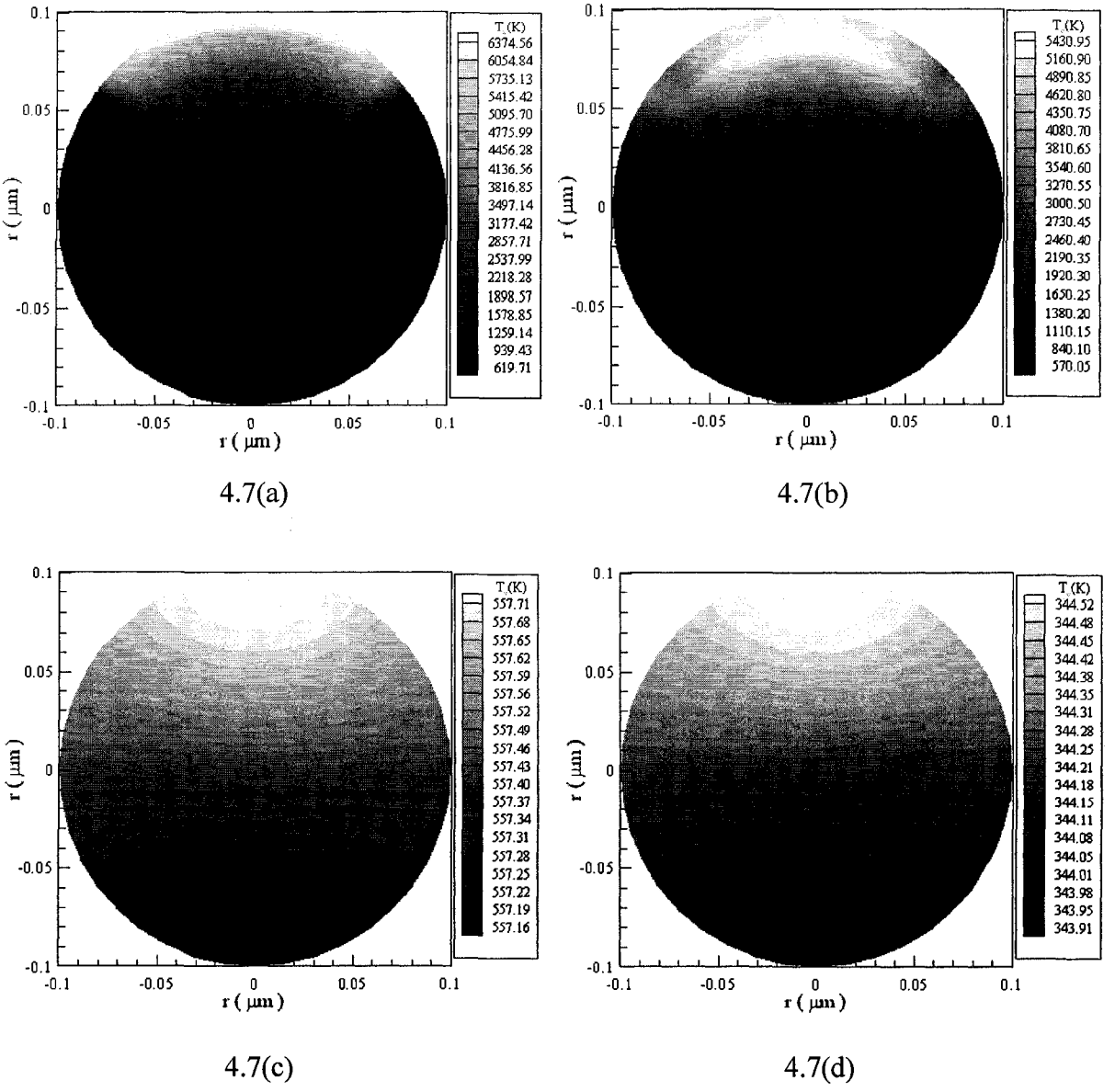


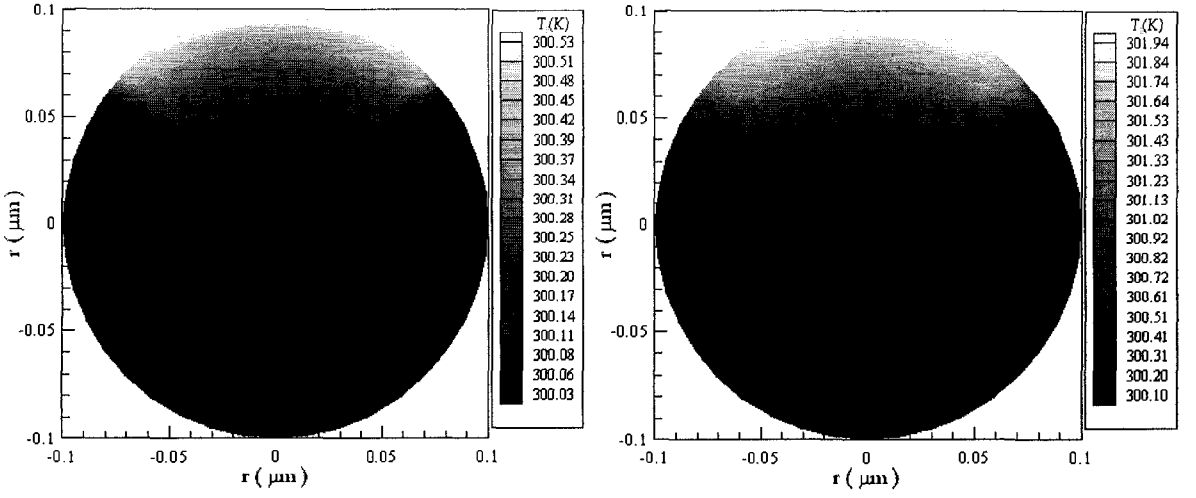
Figure 4.6 Normal stress (σ_r) profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200 J/m^2$, $500 J/m^2$ and $1000 J/m^2$ when $t_p = 0.01ps$.



4.7(a) 4.7(b)

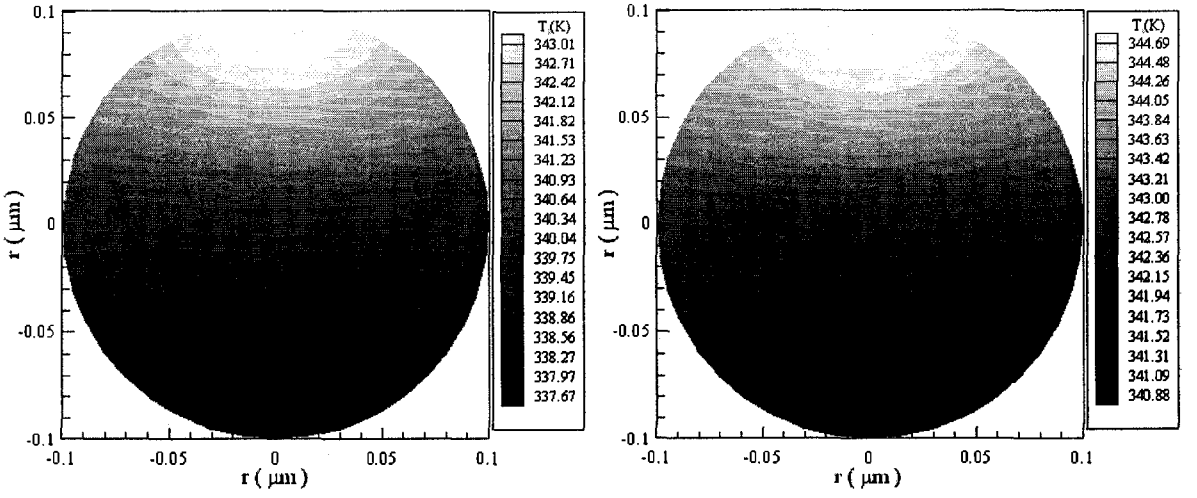
4.7(c) 4.7(d)

Figure 4.7 Contours of the electron temperature distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.01ps$.



4.8(a)

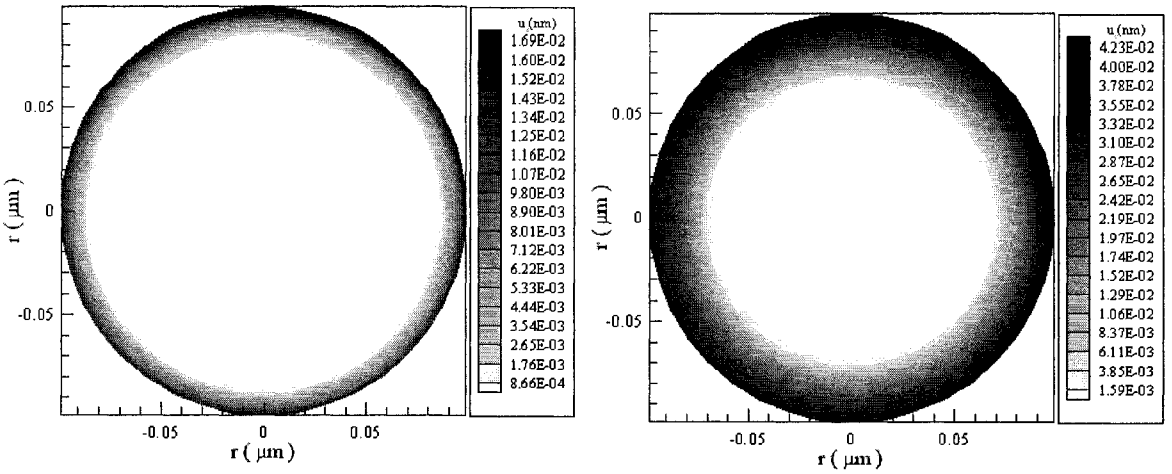
4.8(b)



4.8(c)

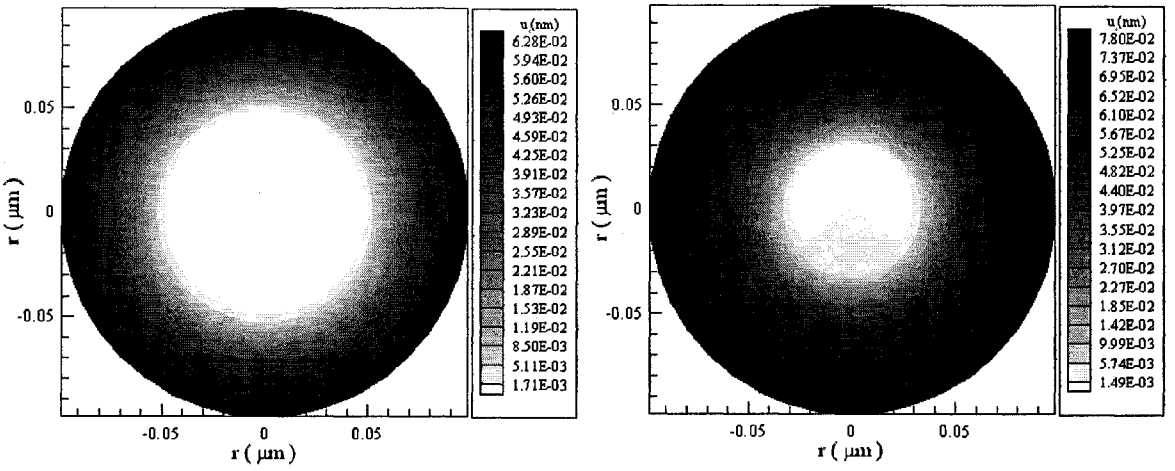
4.8(d)

Figure 4.8 Contours of the lattice temperature distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.01ps$.



4.9(a)

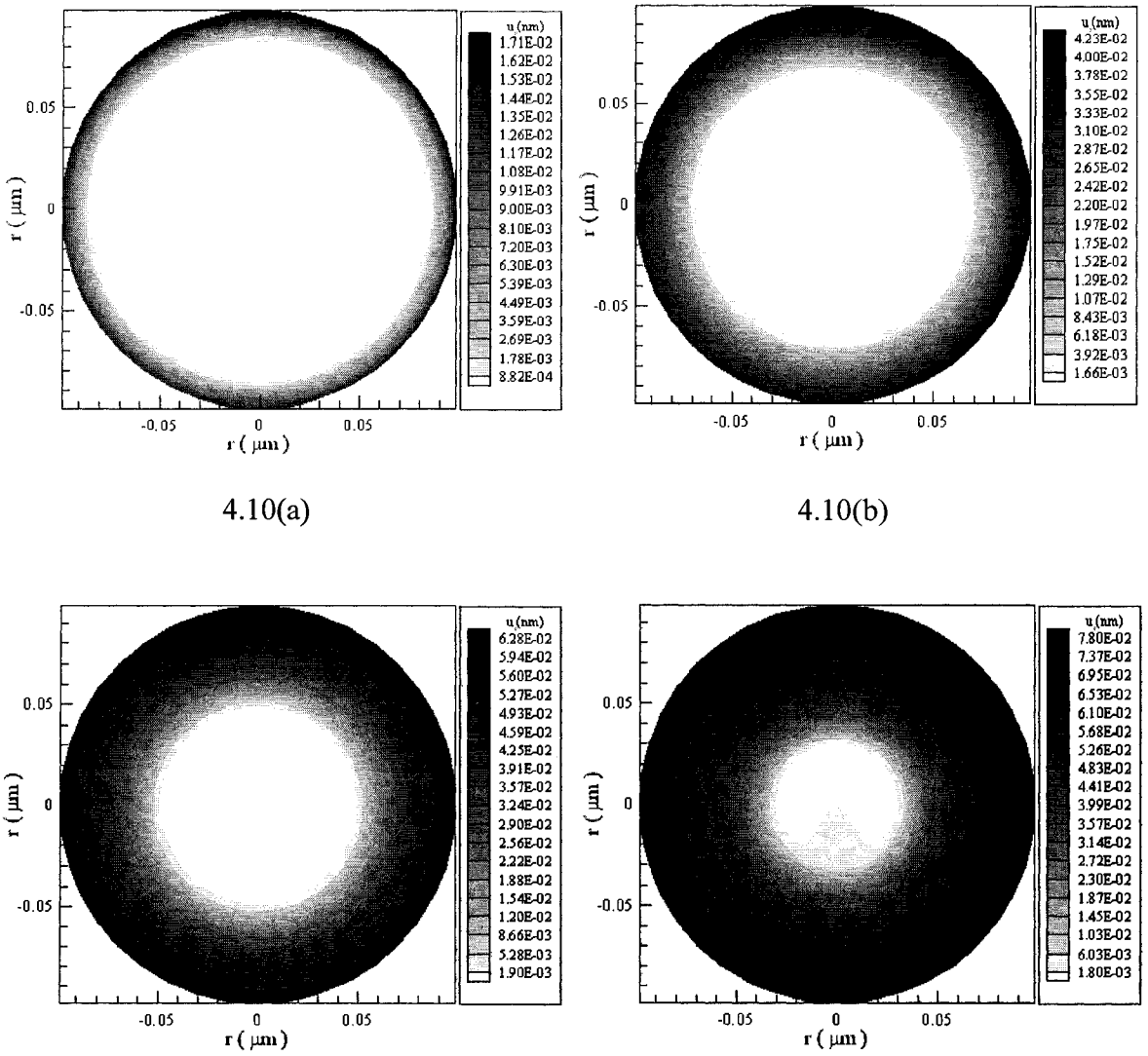
4.9(b)



4.9(c)

4.9(d)

Figure 4.9 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.1ps$.



4.10(a) 4.10(b)

4.10(c) 4.10(d)

Figure 4.10 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.04ps$.

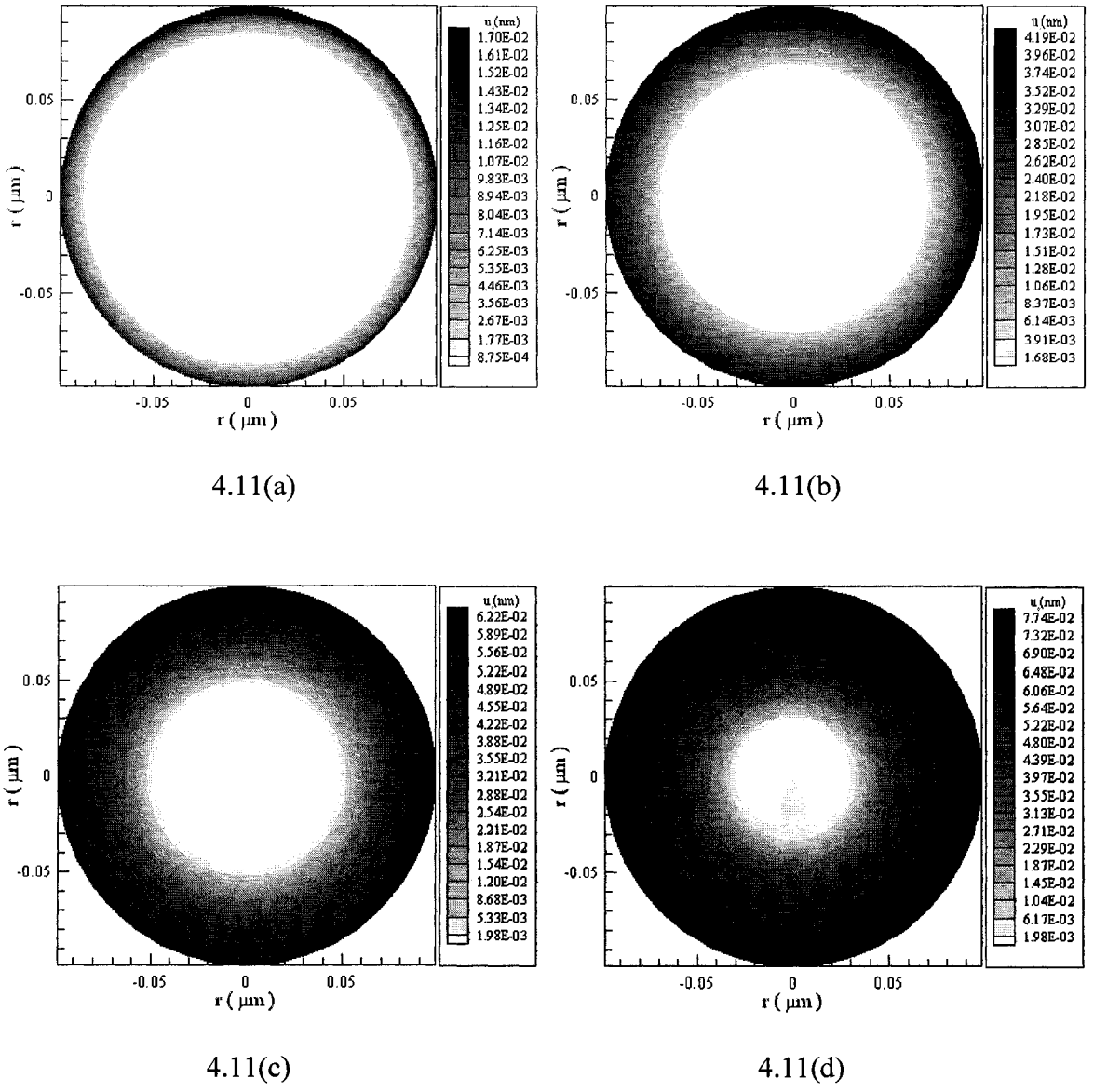


Figure 4.11 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.01ps$.

4.2 Three-Dimensional Double-Layered Case

4.2.1 Example Description

In this example, we want to test the applicability of the finite different scheme developed in Section 3.2, Equations (3.70)-(3.93). For this purpose, we investigate the temperature rise and deformation in a three-dimensional double-layered micro sphere which consists of a gold layer on a chromium padding layer with a radius of $0.05\mu m$ for each layer, as shown in Figure 4.12.

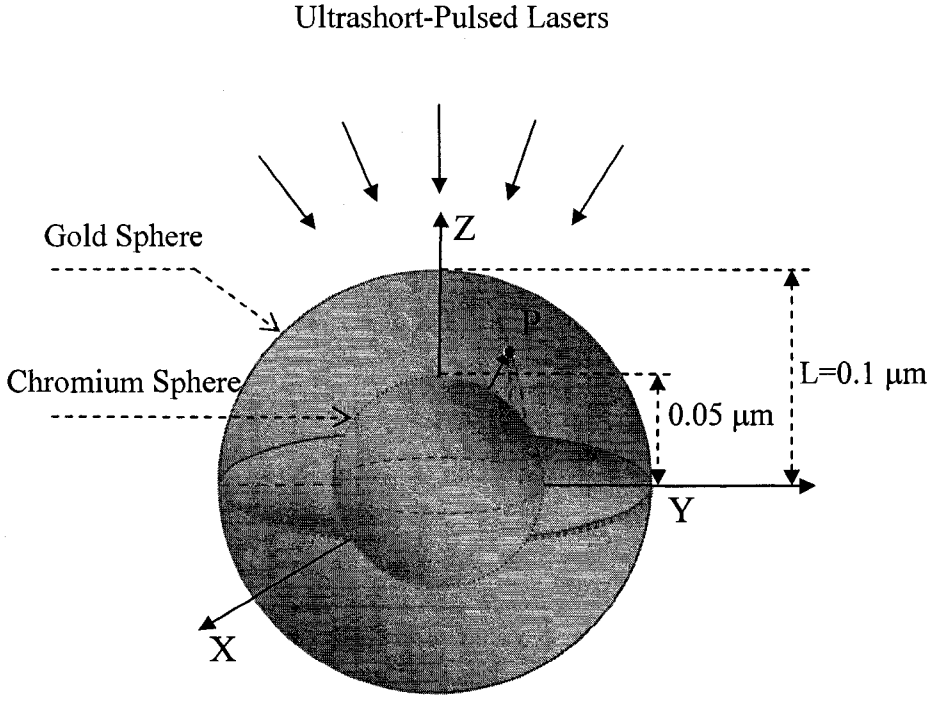


Figure 4.12 Configuration of a 3D double-layered micro sphere.

The heat source is given by Equation (4.2)

$$Q(r,\theta,\varphi,t) = 0.94J\frac{1-R}{t_p\zeta}\exp[-\frac{L-r}{\zeta} - 2.77(\frac{t-2t_p}{t_p})^2]\cos\varphi. \qquad (4.2)$$

where $L = 0.1\mu m$; $\zeta = 15.3nm$; $R = 0.93$; $J = 200\,J/m^2$, $500\,J/m^2$, $1000\,J/m^2$; and $t_p = 0.1ps, 0.04ps, 0.01ps, 0.006ps$.

The thermophysical properties for gold and chromium are listed in Table 4.2 [Toulou 1970a, b, Chen 2003, Tzou 1996].

Table 4.2 Thermophysical properties of gold and chromium [Toulou 1970a, b, Chen 2003, Tzou 1996].

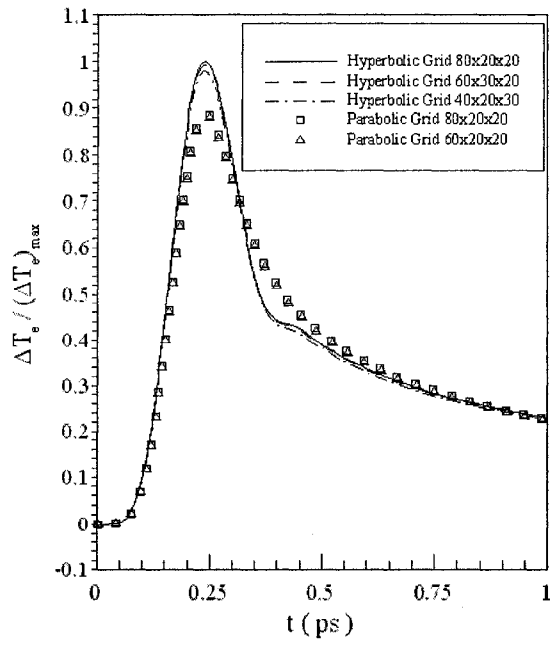
Properties	Unit	Gold	Chromium
ρ	kg/m^3	19300	7190
Λ	J/m^3K^2	70	193.3
λ	Pa	199.0×10^9	83.3×10^9
μ	Pa	27×10^9	115.0×10^9
α_T	K^{-1}	14.2×10^{-6}	4.9×10^{-6}
C_{e0}	J/m^3K^2	2.1×10^4	5.8×10^4
C_l	J/m^3K	2.5×10^6	3.3×10^6
G	W/m^3K	2.6×10^{16}	42×10^{16}
τ_e	ps	0.04	0.0068
τ_l	ps	0.8	0.136
k_0	W/mK	315	94

In this example, we assumed that the laser irradiates a portion $(0 < r < L, 0 \le \theta \le 2\pi, 0 \le \varphi \le \pi/4)$ of the upper hemisphere. Three meshes of

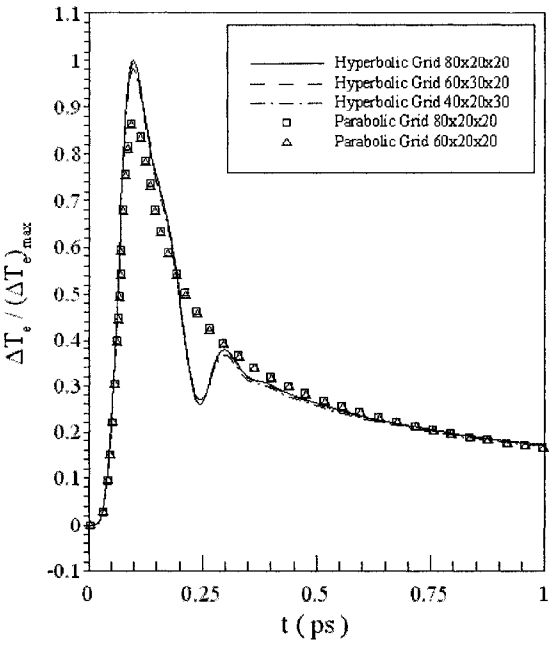
$40\times20\times30$, $60\times30\times20$, $80\times20\times20$ in (r,θ,φ) were chosen in order to test the convergence of the scheme. The time increment was chosen to be $0.0025ps$. Three different values of laser fluences ($J=200J/m^2$, $500J/m^2$, $1000J/m^2$) were chosen to study the hot electron blast force. The convergence criteria were chosen to be $E_1=10^{-5}$ for temperature and $E_1=10^{-12}$ for deformation.

4.2.2 Results and Discussions

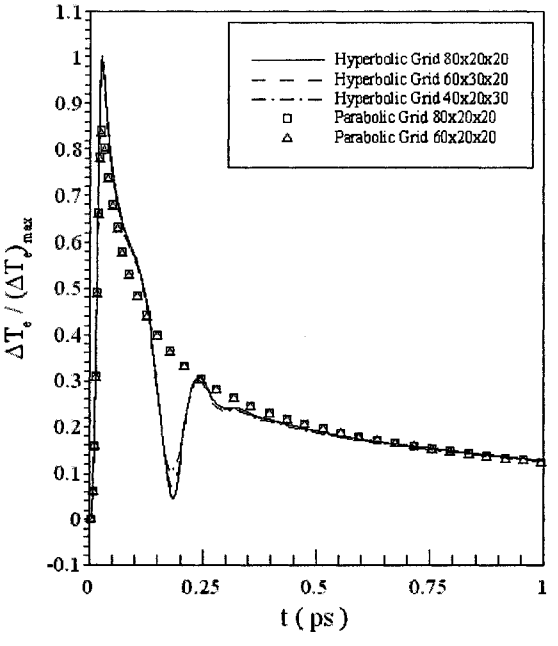
Figure 4.13 shows the change in electron temperature ($\Delta T_e/(\Delta T_e)_{\max}$) at the point ($r=L$, $\theta=0$, $\varphi=0$) for various meshes ($40\times20\times30$, $60\times30\times20$, $80\times20\times20$) with laser fluence $J=500J/m^2$ and four different laser pulse durations (a) $t_p=0.1ps$, (b) $t_p=0.04ps$, (c) $t_p=0.01ps$, and $t_p=0.006ps$ respectively. The maximum temperature rise of T_e (i.e., $(\Delta T_e)_{\max}$) when $t_p=0.1ps$ is about $4036\,K$, which is close to that obtained by Qiu and Tien [Qiu 1994]. It is noted that the electron thermal relaxation time for gold is $0.04ps$ and for chromium is $0.0068ps$. It can be seen from Figure 4.13a that when $t_p=0.1ps$, there is not much difference in the solution between the hyperbolic model and the parabolic model. A small kink appears around $t=0.35ps$ in the solution obtained by the hyperbolic model. This small kink can be also seen in the papers [Qiu 1993, Chen 2001]. From Figures 4.13b, 4.13c and 4.13d, we see that the kink becomes severe with decrease of the laser pulse duration.



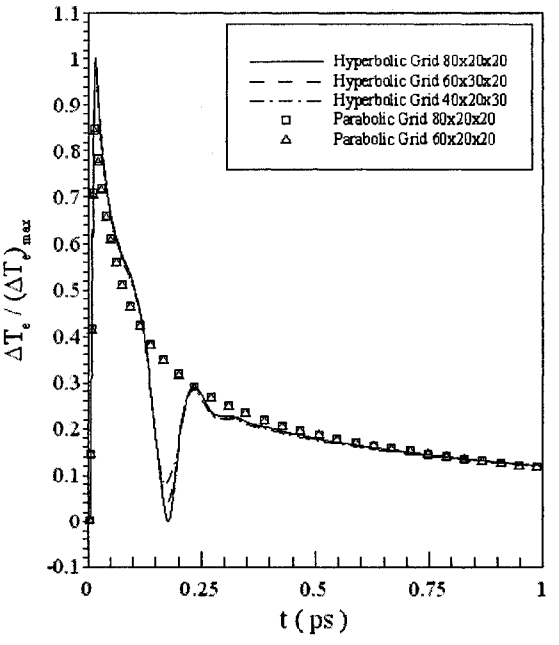
4.13(a)



4.13(b)



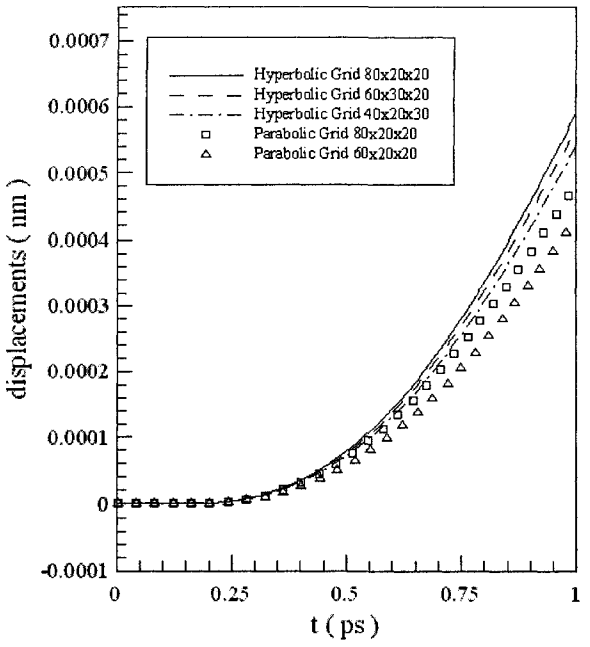
4.13(c)



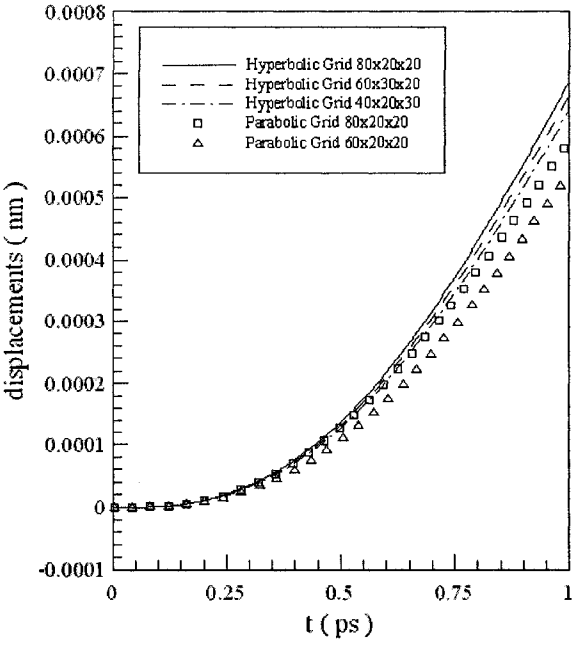
4.13(d)

Figure 4.13 Change in electron temperature at the top point of the sphere versus time with a laser fluence (J) of 500 J/m^2 and four different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, (c) $t_p = 0.01ps$ and (d) $t_p = 0.006ps$.

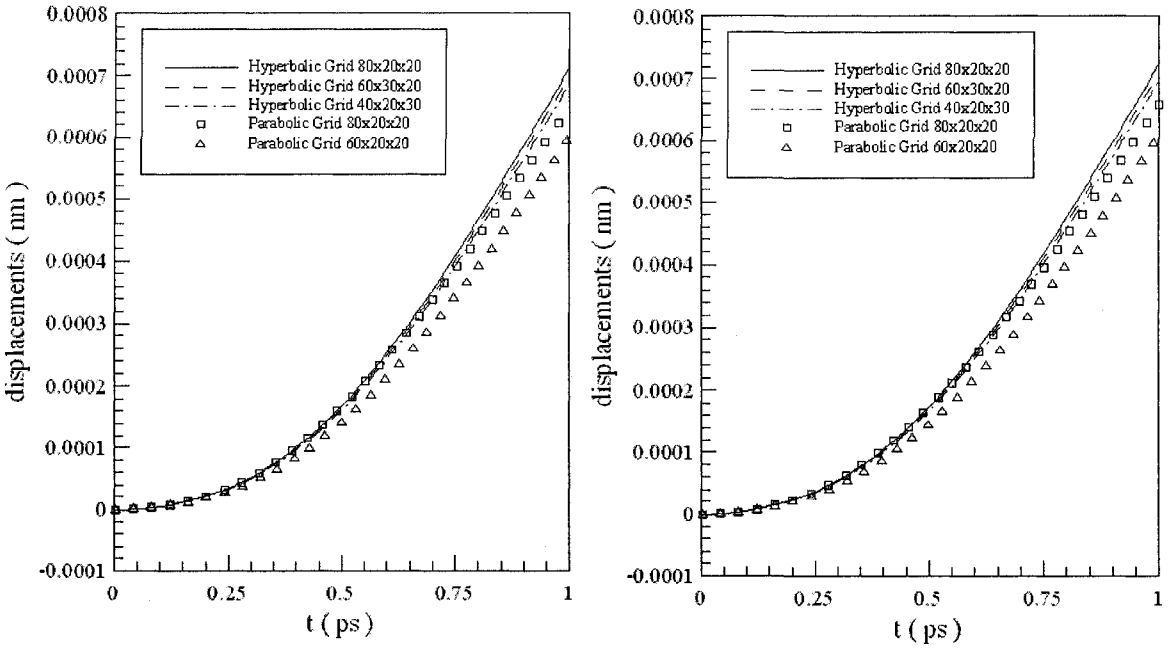
Figure 4.14 shows the displacement (u_r) at the point ($r = L/2, \theta = 0, \varphi = 0$) versus time with four different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, and (c) $t_p = 0.01ps$, and (d) $t_p = 0.006ps$, respectively. It can be seen from both Figures 4.13 and 4.14 that mesh size had no significant effect on the solution and hence the solution is convergent.



4.14(a)



4.14(b)



4.14(c) 4.14(d)

Figure 4.14 Change in displacement (u_r) at the point ($r = L/2, \theta = 0, \varphi = 0$) of the sphere versus time with a laser fluence (J) of 500 J/m^2 and four different laser pulse durations (a) $t_p = 0.1ps$, (b) $t_p = 0.04ps$, (c) $t_p = 0.01ps$ and (d) $t_p = 0.006ps$.

Figures 4.15 and 4.16 show electron temperature and lattice temperature along the diameter at $\varphi = 0$ and $\varphi = \pi$ with three different laser fluences of $J = 200J/m^2$, $500J/m^2$ and $1000J/m^2$ at different times (a) $t = 0.015ps$, (b) $t = 0.03ps$, (c) $t = 5ps$, and (d) $t = 10ps$, respectively when laser pulse duration $t_p = 0.006ps$. It can be seen that the electron temperature rises to its maximum at the beginning and then decreases, while the lattice temperature increases gradually with time in both gold and chromium layers, due to constant heating of acoustic phonons by electrons. Since the heat is transferred from the gold layer to the chromium layer and the conductivity of chromium is smaller than that of gold, the lattice temperature increases drastically across the interface.

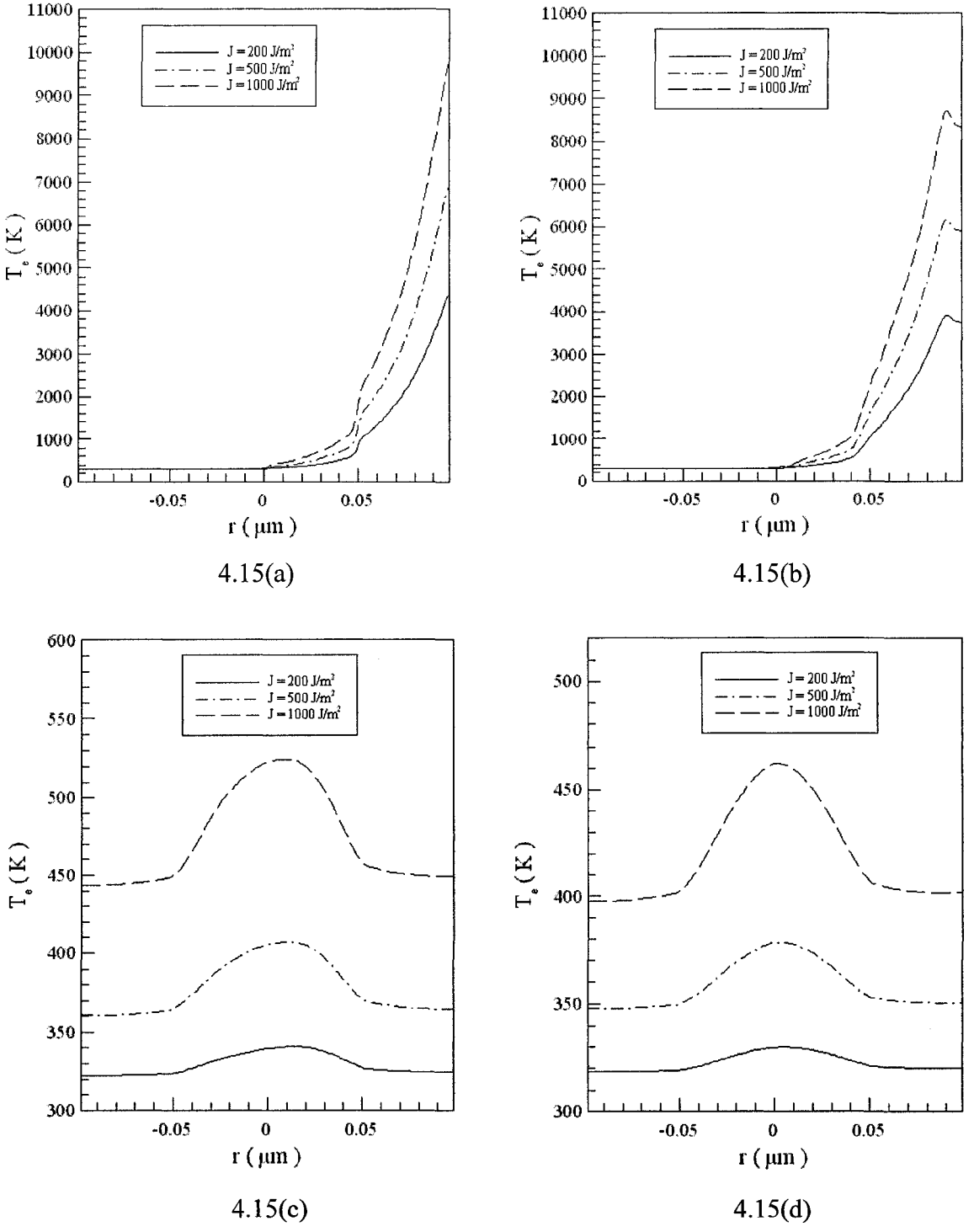


Figure 4.15 Electron temperature profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 0.015ps$, (b) $t = 0.03ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200J/m^2$, $500J/m^2$ and $1000J/m^2$ when $t_p = 0.006ps$.

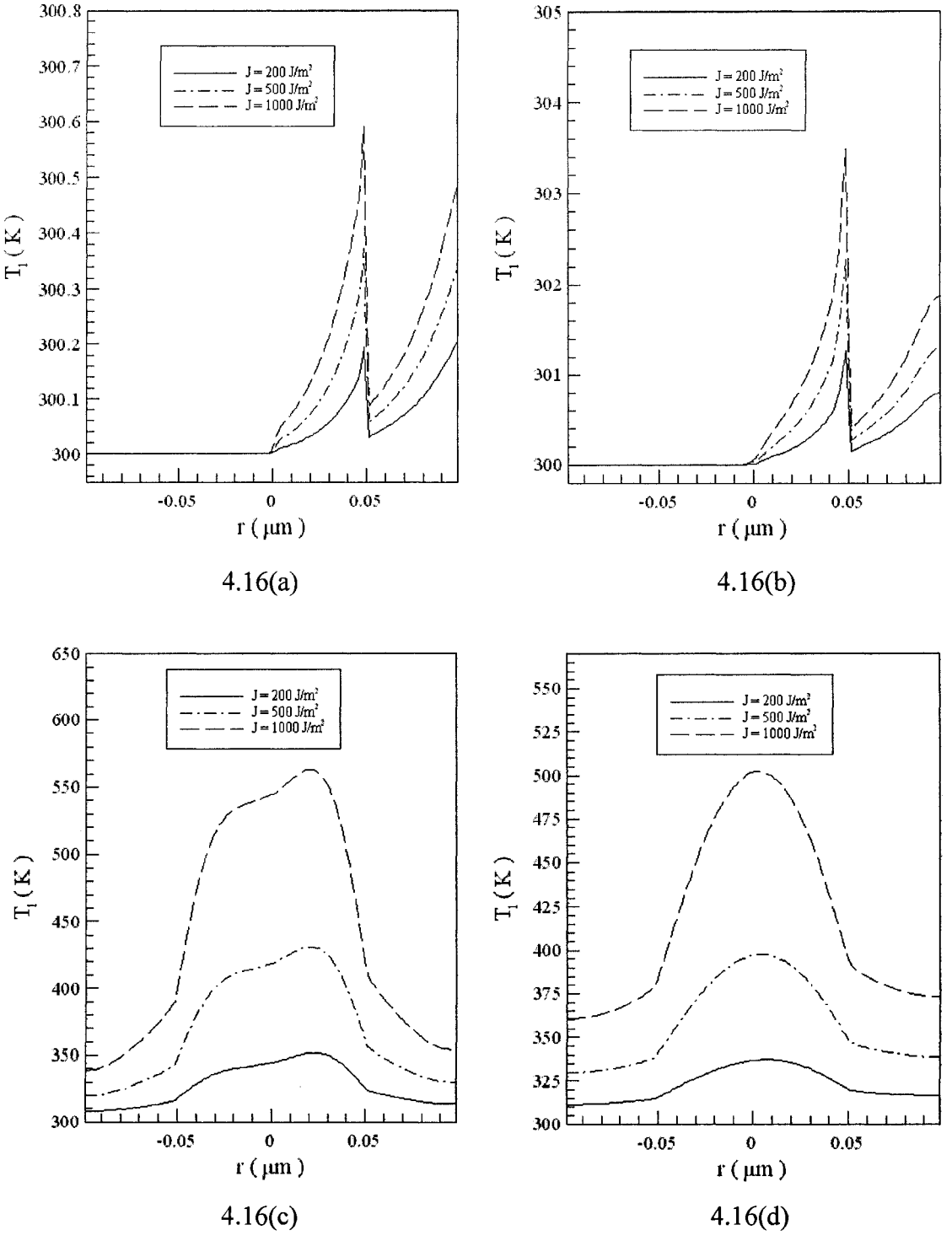


Figure 4.16 Lattice temperature profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 0.015ps$, (b) $t = 0.03ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200J/m^2$, $500J/m^2$ and $1000J/m^2$ when $t_p = 0.006ps$.

Figure 4.17 shows normal stress σ_r along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of 80×20×20 and three different laser fluences ($J = 200J/m^2$, $500J/m^2$, $1000J/m^2$) when laser pulse duration $t_p = 0.006ps$. In our experience, the conventional finite difference method produces local oscillations in the normal stress σ_r. It can be seen from Figure 4.17 that the curve of σ_r is smooth and does not appear to have local oscillations, implying that our method prevents the appearance of non-physical oscillations in the solution.

Figures 4.18-4.23 were plotted based on the results obtained in a mesh of 80×20×20 with a laser fluence of $J = 500J/m^2$. Figures 4.18 and 4.19 show contours of electron temperature distributions and lattice temperature distributions in the cross section of $\theta = 0$ and $\theta = \pi$ at $t_p = 0.006ps$ and four different times (a) $t = 0.015ps$, (b) $t = 0.03ps$, (c) $t = 5ps$, and (d) $t = 10ps$, respectively. Figure 4.18 and 4.19 show that the heat is transferred from the upper hemisphere to the lower hemisphere and also from the gold layer to the chromium layer. Figures 4.20-4.23 show contours of displacement (u_r) in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$, respectively with four different laser pulse durations $t_p = 0.1ps$, $t_p = 0.04ps$, $t_p = 0.01ps$ and $t_p = 0.006ps$. It can be seen from Figures 4.20-4.23 that the displacement u_r moves in the r direction, implying the sphere is expanding. However, the displacement gradient shows a clear difference across the interface, implying that the bond between these two layers could be damaged under high intensity laser irradiation.

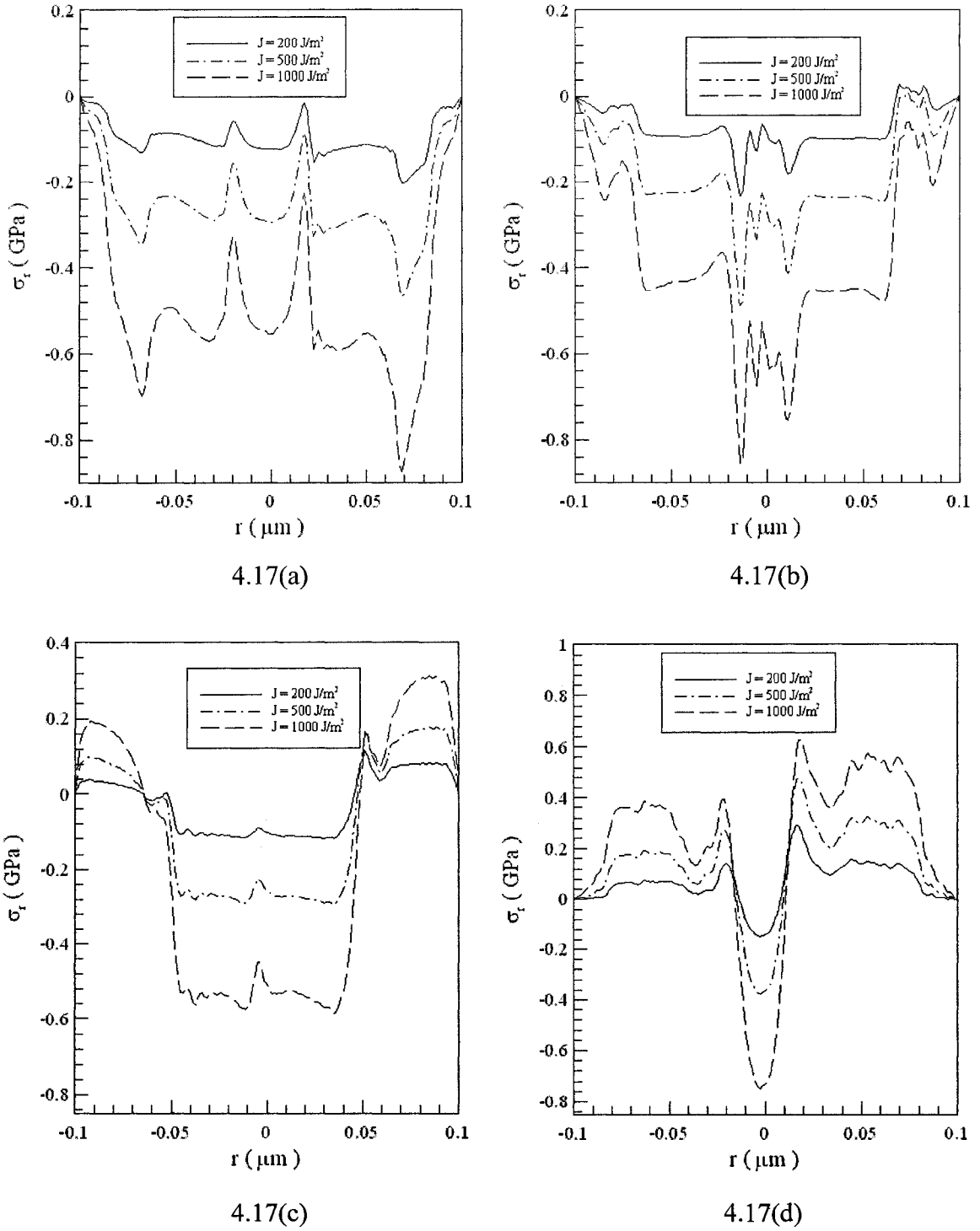
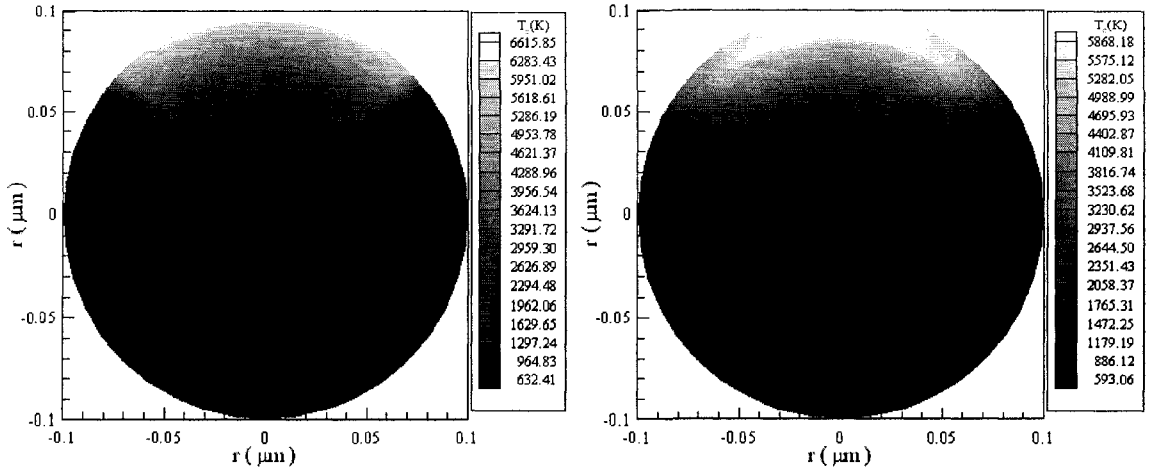
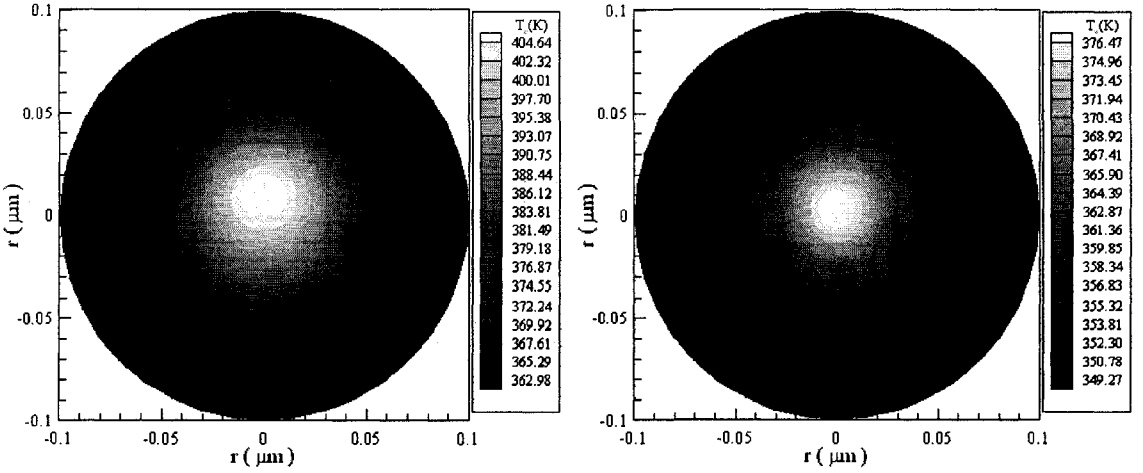


Figure 4.17 Normal stress (σ_r) profiles along the diameter at $\varphi = 0$ and $\varphi = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and three different laser fluences (J) of $200 J/m^2$, $500 J/m^2$ and $1000 J/m^2$ when $t_p = 0.006ps$.

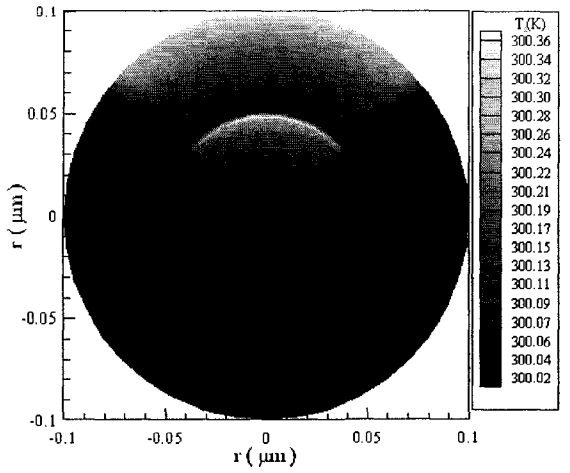


4.18(a) 4.18(b)

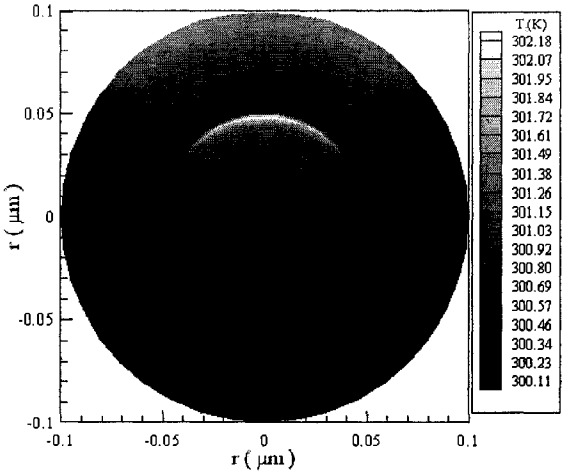


4.18(c) 4.18(d)

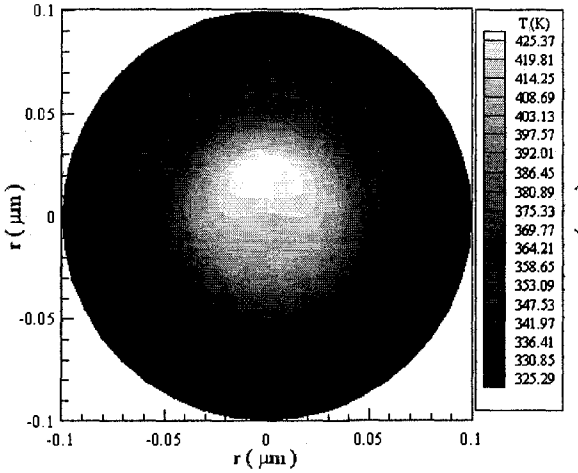
Figure 4.18 Contours of the electron temperature distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 0.015ps$, (b) $t = 0.03ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.006ps$.



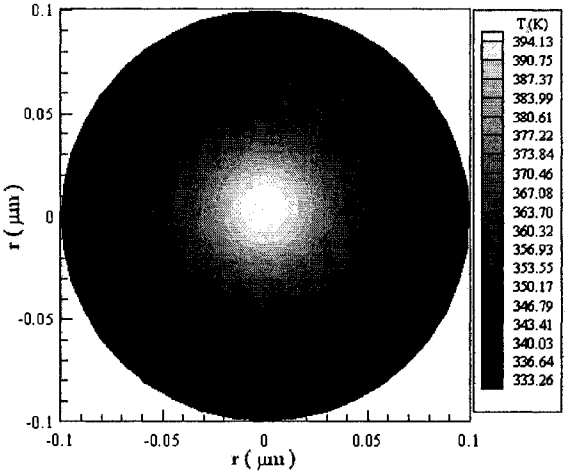
4.19(a)



4.19(b)

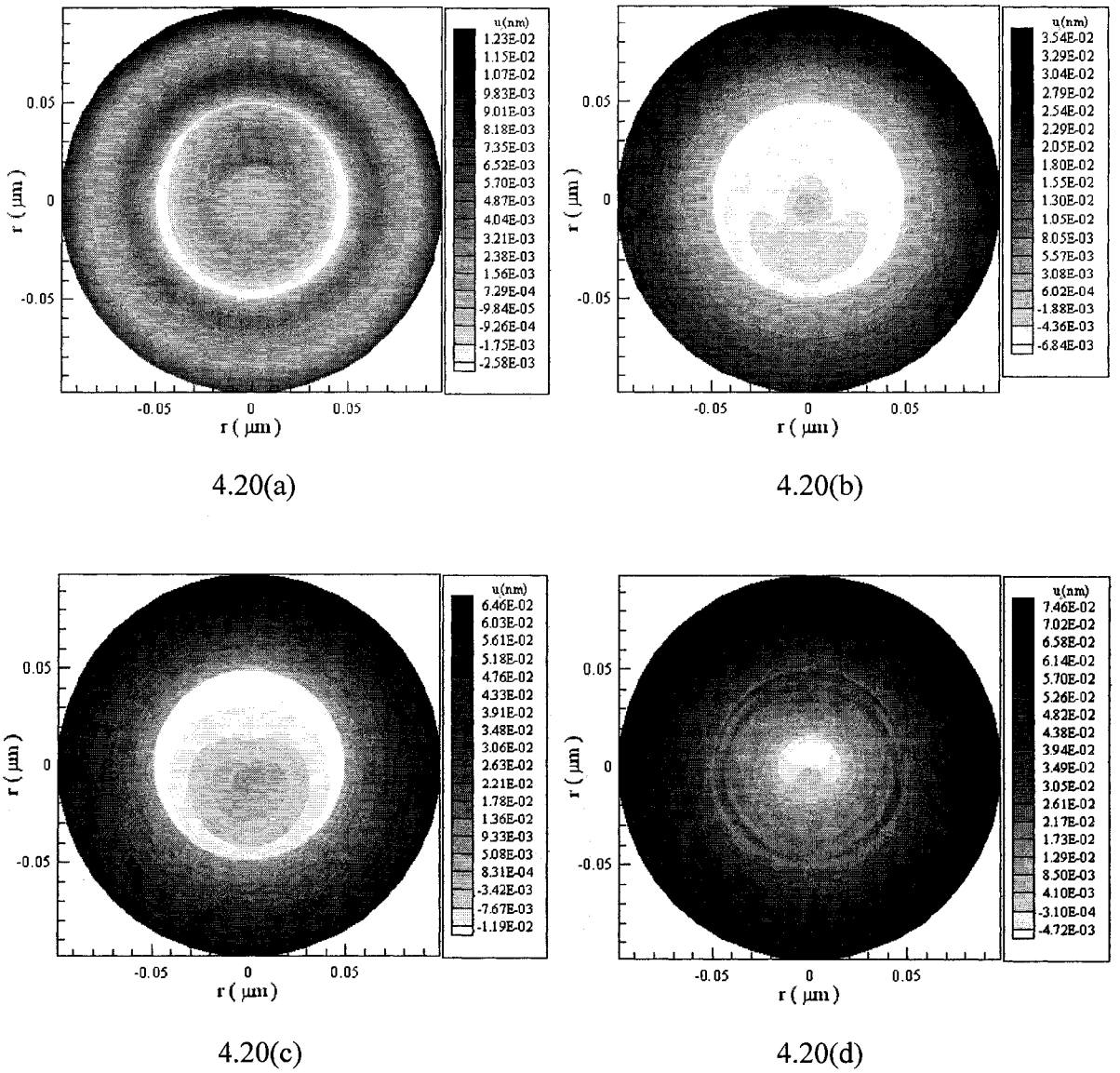


4.19(c)



4.19(d)

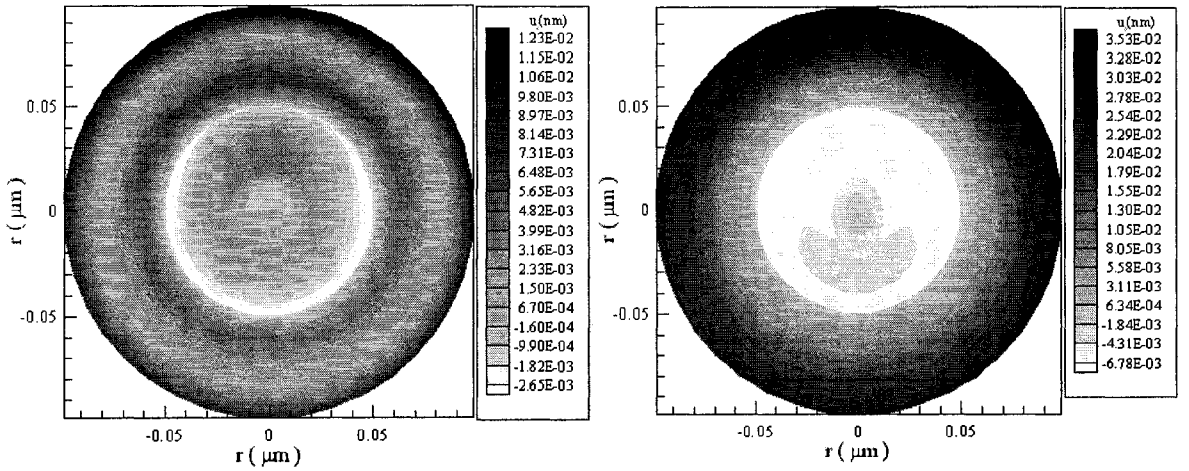
Figure 4.19 Contours of the lattice temperature distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 0.025ps$, (b) $t = 0.05ps$, (c) $t = 5ps$, and (d) $t = 10ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.006ps$.



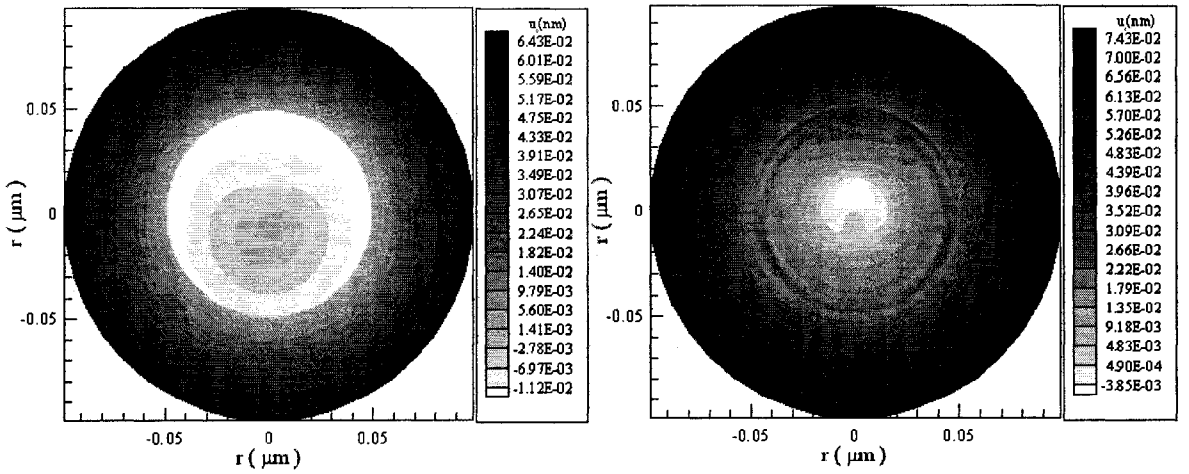
4.20(a) 4.20(b)

4.20(c) 4.20(d)

Figure 4.20 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.1ps$.

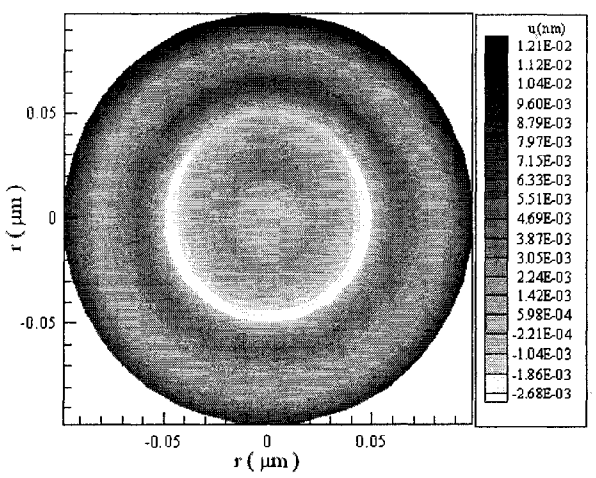


4.21(a) 4.21(b)

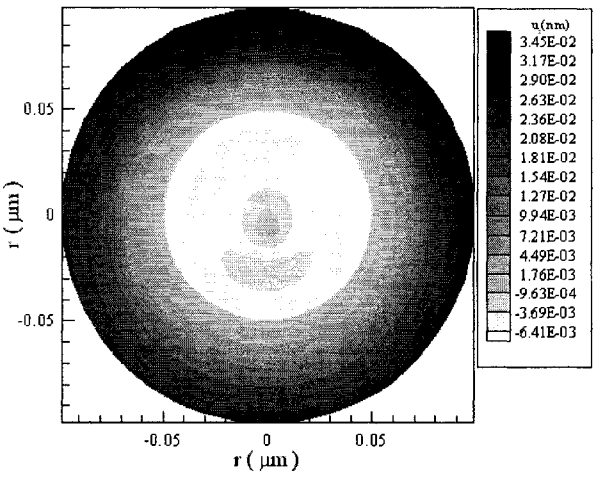


4.21(c) 4.21(d)

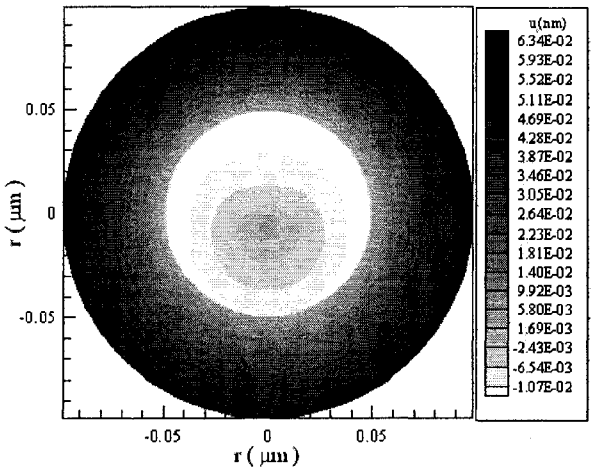
Figure 4.21 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.04ps$.



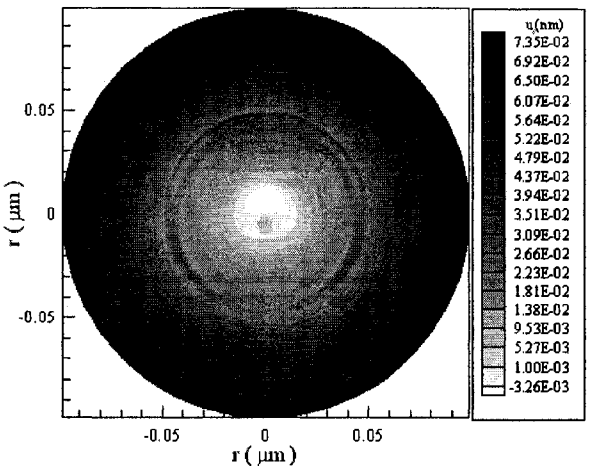
4.22(a)



4.22(b)

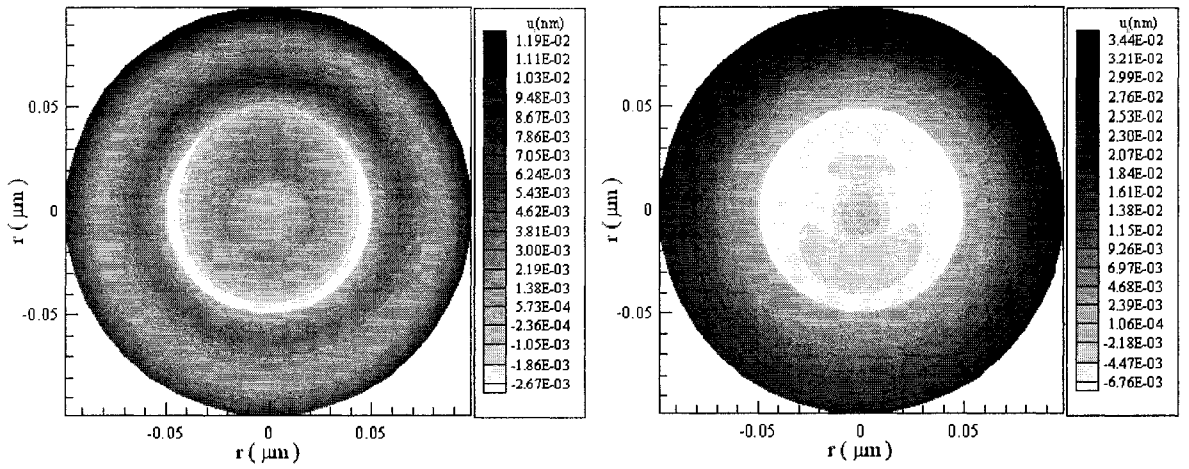


4.22(c)



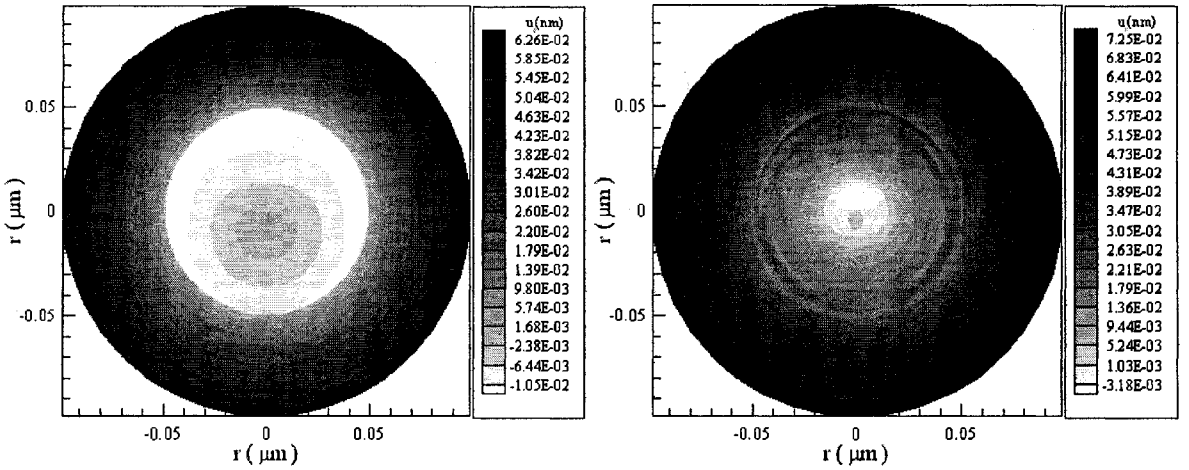
4.22(d)

Figure 4.22 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.01ps$.



4.23(a)

4.23(b)



4.23(c)

4.23(d)

Figure 4.23 Contours of the displacement (u_r) distribution in the cross section of $\theta = 0$ and $\theta = \pi$ at different times (a) $t = 5ps$, (b) $t = 10ps$, (c) $t = 15ps$, and (d) $t = 20ps$ with a mesh of $80 \times 20 \times 20$ and a laser fluence (J) of $500J/m^2$ when $t_p = 0.006ps$.

CHAPTER 5

CONCLUSION AND FUTURE WORK

In this dissertation, we have developed a mathematical model under spherical coordinates for studying thermal deformation in 3D micro spheres exposed to ultrashort-pulsed lasers. The model is based on the hyperbolic two-step heat transport equations, dynamic equations of motion, and the coupling effect between lattice temperature and strain rate, as well as for the hot-electron blast effect in momentum transfer. The hyperbolic model is more dependable as compared with the parabolic model, when the laser pulse duration is shorter than the electron relaxation time.

Based on the developed mathematical model, we have then developed a finite difference method for obtaining temperature, displacement, stresses and strains in micro spheres induced by ultrashort-pulsed lasers. In order to avoid non-physical oscillation in the solution, we have derived this numerical method by introducing velocity components into the dynamic equation of motion and replacing the displacement components. Then, a staggered grid was designed so that the checker-board solution could be eliminated, and a fourth-order compact scheme was further employed to calculate stresses derivatives in the dynamic equations of motion.

The numerical method is tested by two cases: 1) a 3D gold micro sphere, and 2) a 3D double-layered gold and chromium micro sphere exposed to ultrashort-pulsed lasers, respectively. Much shorter laser pulse durations are considered in the numerical examples.

Results indicate that the numerical method is grid independent, implying that the solution is convergent. The method allows avoidance of non-physical oscillations in the solution and there are some differences between the hyperbolic two-step model and the parabolic two-step model. Electron and lattice temperatures, displacements, stresses, and strains are obtained based on the developed finite difference schemes. The displacement along the r-direction reveals that the micro spheres expand during the ultrashort-pulsed irradiation.

The future research may focus on a two-dimensional double-layered micro thin film with nonlinear interfacial conditions, which means the interface is imperfectly thermal contacted, and a three-dimensional double-layered micro thin film, where the model and computation may be much more complicated.

APPENDIX A

SOURCE CODE FOR 3D MICRO SPHERE CASE

```
c main
implicit double precision (a-h,l,o-z)
dimension t(20010),t1(20010),t2(20010),x(150),
$y(50),z(50),
$x1(150),x2(150),y1(50),z1(50),a(150),b(150),
$c(150),beta(150)
dimension eto(150,50,50),etn(150,50,50),
$lto(150,50,50),ltn(150,50,50),
$etm(20010),etm1(20010),ltm(20010),
$ltm1(20010),u1m(20010),u2m(20010),
$u3m(20010),xsao(101,31,31),ysao(101,31,31),
$zsao(101,31,31),ssaoxy(101,31,31),
$ssaoxz(101,31,31),ssaoyz(101,31,31),
$xsaoo(101,31,31),ysaoo(101,31,31),
$zsaoo(101,31,31),ssaooxy(101,31,31),
$ssaooxz(101,31,31),ssaooyz(101,31,31),
$xsan(101,31,31),ysan(101,31,31),zsan(101,31,31),
$ssanxy(101,31,31),ssanxz(101,31,31),
$ssanyz(101,31,31),xseo(101,31,31),
$yseo(101,31,31),zseo(101,31,31),
$sseoxy(101,31,31),sseoxz(101,31,31),
$sseoyz(101,31,31),xsen(101,31,31),
$ysen(101,31,31),zsen(101,31,31),
$ssenxy(101,31,31),ssenxz(101,31,31),
$ssenyz(101,31,31),difx(101,31,31),
$dify(101,31,31),difz(101,31,31),
$difxyx(101,31,31),difxyy(101,31,31),
$difxzx(101,31,31),difxzz(101,31,31),
$difyzy(101,31,31),difyzz(101,31,31),
$v1o(101,31,31),v2o(101,31,31),v3o(101,31,31),
$v1n(101,31,31),v2n(101,31,31),v3n(101,31,31),
$u1o(101,31,31),u2o(101,31,31),u3o(101,31,31),
$u1n(101,31,31),u2n(101,31,31),u3n(101,31,31),
$d(101,31,31),gama(101,31,31),u1(101,31,31),
$u2(101,31,31),u3(101,31,31),u4(101,31,31),
$u5(101,31,31),u6(101,31,31),u7(101,31,31),
$u8(101,31,31),u9(101,31,31),qexo(101,31,31),
$qeyo(101,31,31),qezo(101,31,31),
$qlxo(101,31,31),qlyo(101,31,31),qlzo(101,31,31),
$qex(101,31,31),qey(101,31,31), qez(101,31,31),
$qlx(101,31,31),qly(101,31,31), qlz(101,31,31)

    integer o, counter,l,r

c Lame constant
    lemta=199.0d+9
c Shear modulus
    cmiu=27.0d+9
c Thermal expansion coefficient
    alphat=14.2d-6

    pi=3.14159265359
    lx=1.0D-7
    ly=2.0*pi
    lz=1.0*pi
    n=80
    n2=NINT(n/2.0)+1
    m=20
    r=20
    h1=lx/n
    h2=ly/m
    h3=lz/r

    dt=0.0025d-12
    o=NINT(20d-12/dt)+1
    counter=0
    t(1)=0
    x(1)=0
    y(1)=0
    z(1)=0

c initial condition
    k=1
    do l=1,r+1
    do j=1,m
    do i=1,n+1
    etn(i,j,l)=300.0
    ltn(i,j,l)=300.0
    eto(i,j,l)=300.0
    lto(i,j,l)=300.0


    xsao(i,j,l)=0.0
    ysao(i,j,l)=0.0
    zsao(i,j,l)=0.0
    ssaoxy(i,j,l)=0.0
    ssaoxz(i,j,l)=0.0
    ssaoyz(i,j,l)=0.0

    xsaoo(i,j,l)=0.0
    ysaoo(i,j,l)=0.0
    zsaoo(i,j,l)=0.0
    ssaooxy(i,j,l)=0.0
    ssaooxz(i,j,l)=0.0
    ssaooyz(i,j,l)=0.0

    xseo(i,j,l)=0.0
    yseo(i,j,l)=0.0
    zseo(i,j,l)=0.0
    sseoxy(i,j,l)=0.0
    sseoxz(i,j,l)=0.0
    sseoyz(i,j,l)=0.0
    difxyx(i,j,l)=0.0
    difxyy(i,j,l)=0.0
    difxzx(i,j,l)=0.0
    difxzz(i,j,l)=0.0
    difyzy(i,j,l)=0.0
    difyzz(i,j,l)=0.0
    difx(i,j,l)=0.0
    dify(i,j,l)=0.0
    difz(i,j,l)=0.0
    v1o(i,j,l)=0.0
```

```
v2o(i,j,l)=0.0
v3o(i,j,l)=0.0
u1o(i,j,l)=0.0
u2o(i,j,l)=0.0
u3o(i,j,l)=0.0

qexo(i,j,l)=0.0
qlxo(i,j,l)=0.0
qex(i,j,l)=0.0
qlx(i,j,l)=0.0
qeyo(i,j,l)=0.0
qlyo(i,j,l)=0.0
qey(i,j,l)=0.0
qly(i,j,l)=0.0
qezo(i,j,l)=0.0
qlzo(i,j,l)=0.0
qez(i,j,l)=0.0
qlz(i,j,l)=0.0


end do
end do
end do

etm(k)=300.0
ltm(k)=300.0

do l=2,r+1
z(l)=z(l-1)+h3
z1(l)=z(l-1)+1.0/r
end do
do j=2,m
y(j)=y(j-1)+h2
y1(j)=y1(j-1)+2.0/m
end do
do i=2,n+1
x(i)=x(i-1)+h1
x1(i)=(x(i-1)+h1)*1.0d6
x2(i)=-(x(i-1)+h1)*1.0d6
end do

write(*,*) 'start'

do 1 k=2,o+1

t(k)=t(k-1)+dt
t1(k)=t(k-1)+dt/2.0
t2(k)=(t(k-1)+dt)*1.0d12

do l=1,r+1
do j=1,m
do i=1,n+1
xsan(i,j,l)=xsao(i,j,l)
ysan(i,j,l)=ysao(i,j,l)
zsan(i,j,l)=zsao(i,j,l)
enddo
    enddo
    enddo

call
$temp(n,m,r,lx,h1,h2,h3,x,y,z,t1(k),dt,lto,ltn,eto,
$etn,xsan,ysan,zsan,xsao,ysao,zsao,qexo,qeyo,
$qezo,qlxo,qlyo,
$qlzo,qex,qey,qez,qlx,qly,qlz)


    tol=1d-12
    detuvmax=tol+1d-5
    do while (detuvmax.gt.tol)
    detuvmax=0.0
    det1max=0
    det2max=0
    det3max=0
    det4max=0
    det5max=0
    det6max=0


c Compute stress

do l=2,r
do j=1,m
do i=2,n
xsen(i,j,l)=(lemta+2.0*cmiu)*xsan(i,j,l)
$+lemta*ysan(i,j,l)+lemta*zsan(i,j,l)
$-(3.0*lemta+2.0*cmiu)*alphat*(ltn(i,j,l)-300.0)

ysen(i,j,l)=lemta*xsan(i,j,l)+(lemta+2.0*cmiu)
$*ysan(i,j,l)+lemta*zsan(i,j,l)
$-(3.0*lemta+2.0*cmiu)*alphat*(ltn(i,j,l)-300.0)

zsen(i,j,l)=lemta*xsan(i,j,l)+(lemta+2.0*cmiu)
$*zsan(i,j,l)+lemta*ysan(i,j,l)
$-(3.0*lemta+2.0*cmiu)*alphat*(ltn(i,j,l)-300.0)

end do
end do
end do

do l=1,r+1
do j=1,m
xsen(1,j,l)=xsen(2,j,l)
ysen(1,j,l)=ysen(2,j,l)
zsen(1,j,l)=zsen(2,j,l)
xsen(n+1,j,l)=0.0
ysen(n+1,j,l)=0.0
zsen(n+1,j,l)=0.0
end do
end do

do j=1,m
do i=1,n+1
```

```
      zsen(i,j,1)=zsen(i,j,2)
      ysen(i,j,1)=ysen(i,j,2)
      xsen(i,j,1)=xsen(i,j,2)
      zsen(i,j,r+1)=zsen(i,j,r)
      ysen(i,j,r+1)=ysen(i,j,r)
      xsen(i,j,r+1)=xsen(i,j,r)
      end do
      end do
c
      do l=1,r+1
      do j=1,m
      ssenxy(1,j,l)=0
      ssenxy(n,j,l)=0
      enddo
      enddo
      do i=1,n-1
      do j=1,m
      ssenxy(i,j,1)=0
      ssenxy(i,j,r+1)=0
      enddo
      enddo

      do l=2,r
      do j=1,m
      do i=2,n-1
      ssenxy(i,j,l)=2*cmiu*ssanxy(i,j,l)

      end do
      end do
      end do

      do l=1,r
      do j=1,m
      ssenxz(1,j,l)=0
      ssenxz(n,j,l)=0
      enddo
      enddo

      do i=1,n-1
      do j=1,m
      ssenxz(i,j,1)=0
      ssenxz(i,j,r)=0
      enddo
      enddo
      do l=2,r-1
      do j=1,m
      do i=2,n-1
      ssenxz(i,j,l)=2*cmiu*ssanxz(i,j,l)
      enddo
      enddo
      enddo
      do l=1,r
      do j=1,m
      ssenyz(1,j,l)=0
      ssenyz(n+1,j,l)=0
      enddo

      do i=1,n
      do j=1,m
      ssenyz(i,j,1)=0
      ssenyz(i,j,r)=0
      enddo
      enddo
      do l=2,r-1
      do j=1,m
      do i=2,n
      ssenyz(i,j,l)=2*cmiu*ssanyz(i,j,l)
      enddo
      enddo
      enddo

c     difx----------
      do l=1,r+1
      do j=1,m
      difx(1,j,l)=(xsen(2,j,l)-xsen(1,j,l))/h1
      difx(n,j,l)=(xsen(n+1,j,l)-xsen(n,j,l))/h1
      end do
      end do

      b(2)=0.0
      a(2)=0.916666667
      c(2)=-0.041666667

      do l=1,r+1
      do j=1,m
      d(2,j,l)=(xsen(3,j,l)-xsen(2,j,l))/h1
      $-0.041666667*difx(1,j,l)
      end do
      end do

      do i=3,n-2
      b(i)=-0.041666667
      a(i)=0.916666667
      c(i)=-0.041666667
      do j=1,m
      do l=1,r+1
      d(i,j,l)=(xsen(i+1,j,l)-xsen(i,j,l))/h1
      end do
      end do
      end do

      b(n-1)=-0.041666667
      a(n-1)=0.916666667
      c(n-1)=0

      do l=1,r+1
      do j=1,m
      d(n-1,j,l)=(xsen(n,j,l)-xsen(n-1,j,l))/h1
      $-0.041666667*difx(n,j,l)
      end do
      end do
```

```
      beta(n)=0
      do l=1,r+1
      do j=1,m
      gama(n,j,l)=0
      end do
      end do

      do k1=2,n-1
      i=n-k1+1
      beta(i)=b(i)/(a(i)-c(i)*beta(i+1))
      do j=1,m
      do l=1,r+1
      gama(i,j,l)=(d(i,j,l)+c(i)*gama(i+1,j,l))
     $/(a(i)-c(i)*beta(i+1))
      end do
      end do
      end do

      do j=1,m
      do l=1,r+1
       u1(1,j,l)=0.0
      end do
      end do

      do i=2,n-1
      do j=1,m
      do l=1,r+1
      u1(i,j,l)=beta(i)*u1(i-1,j,l)+gama(i,j,l)
      difx(i,j,l)=u1(i,j,l)
      end do
      end do
      end do

      do i=1,n+1
      a(i)=0.0
      b(i)=0.0
      c(i)=0.0
      beta(i)=0.0
      do j=1,m
      do l=1,r+1
      gama(i,j,l)=0.0
      d(i,j,l)=0.0
      enddo
      enddo
      enddo
c     dify-----------------------------------
      do l=1,r+1
      do i=1,n+1
      dify(i,1,l)=(ysen(i,2,l)-ysen(i,1,l))/h2
      dify(i,m,l)=(ysen(i,1,l)-ysen(i,m,l))/h2
      end do
      end do

      b(2)=0
      a(2)=0.916666667
      c(2)=-0.0416666667

      do l=1,r+1
      do i=1,n+1
      d(i,2,l)=(ysen(i,3,l)-ysen(i,2,l))/h2
     $-0.041666667*dify(i,1,l)
      end do
      end do

      do j=3,m-2
      b(j)=-0.0416666667
      a(j)=0.916666667
      c(j)=-0.0416666667
      do i=1,n+1
      do l=1,r+1
      d(i,j,l)=(ysen(i,j+1,l)-ysen(i,j,l))/h2
      end do
      end do
      end do

      b(m-1)=-0.0416666667
      a(m-1)=0.916666667
      c(m-1)=0

      do l=1,r+1
      do i=1,n+1
      d(i,m-1,l)=(ysen(i,m,l)-ysen(i,m-1,l))/h2
     $-0.041666667*dify(i,m,l)
      end do
      end do

      beta(m)=0
      do l=1,r+1
      do i=1,n+1
      gama(i,m,l)=0
      end do
      end do

      do k1=2,m-1
      j=m-k1+1
      beta(j)=b(j)/(a(j)-c(j)*beta(j+1))
      do i=1,n+1
      do l=1,r+1
      gama(i,j,l)=(d(i,j,l)+c(j)*gama(i,j+1,l))
     $/(a(j)-c(j)*beta(j+1))
      end do
      end do
      end do

      do i=1,n+1
      do l=1,r
      u2(i,1,l)=0.0
      end do
      end do

      do i=1,n+1
```

```
      do j=2,m-1
      do l=1,r
      u2(i,j,l)=beta(j)*u2(i,j-1,l)+gama(i,j,l)
      dify(i,j,l)=u2(i,j,l)
      end do
      end do
      end do
      do i=1,n+1
      a(i)=0.0
      b(i)=0.0
      c(i)=0.0
      beta(i)=0.0
      do j=1,m
      do l=1,r+1
      gama(i,j,l)=0.0
      d(i,j,l)=0.0
      enddo
      enddo
      enddo
c     difz---------------------------------
      do j=1,m
      do i=1,n+1
      difz(i,j,1)=(zsen(i,j,2)-zsen(i,j,1))/h3
      difz(i,j,r)=(zsen(i,j,r+1)-zsen(i,j,r))/h3
      end do
      end do

      b(2)=0.0
      a(2)=0.916666667
      c(2)=-0.0416666667

      do j=1,m
      do i=1,n+1
      d(i,j,2)=(zsen(i,j,3)-zsen(i,j,2))/h3
     $-0.041666667*difz(i,j,1)
      end do
      end do

      do l=3,r-2
      b(l)=-0.0416666667
      a(l)=0.916666667
      c(l)=-0.0416666667
      do i=1,n+1
      do j=1,m
      d(i,j,l)=(zsen(i,j,l+1)-zsen(i,j,l))/h3
      end do
      end do
      end do

      b(r-1)=-0.0416666667
      a(r-1)=0.916666667
      c(r-1)=0

      do j=1,m
      do i=1,n+1
      d(i,j,r-1)=(zsen(i,j,r)-zsen(i,j,r-1))/h3
     $-0.041666667*difz(i,j,r)
      end do
      end do

      beta(r)=0
      do j=1,m
      do i=1,n+1
      gama(i,j,r)=0
      end do
      end do

      do k1=2,r-1
      l=r-k1+1
      beta(l)=b(l)/(a(l)-c(l)*beta(l+1))
      do i=1,n+1
      do j=1,m
      gama(i,j,l)=(d(i,j,l)+c(l)*gama(i,j,l+1))
     $/(a(l)-c(l)*beta(l+1))
      end do
      end do
      end do

      do i=1,n+1
      do j=1,m
      u3(i,j,1)=0.0
      end do
      end do

      do i=1,n+1
      do j=1,m
      do l=2,r-1
      u3(i,j,l)=beta(l)*u3(i,j,l-1)+gama(i,j,l)
      difz(i,j,l)=u3(i,j,l)
      end do
      end do
      end do

      do i=1,n+1
      a(i)=0.0
      b(i)=0.0
      c(i)=0.0
      beta(i)=0.0
      do j=1,m
      do l=1,r+1
      gama(i,j,l)=0.0
      d(i,j,l)=0.0
      enddo
      enddo
      enddo
c     difxyx--------------------------------
      do l=2,r
      do j=1,m
      difxyx(2,j,l)=(ssenxy(2,j,l)-ssenxy(1,j,l))/h1
      difxyx(n,j,l)=(ssenxy(n,j,l)-ssenxy(n-1,j,l))/h1
      end do
      end do
```

```
b(3)=0.0
a(3)=0.916666667
c(3)=-0.0416666667

do l=2,r
do j=1,m
d(3,j,l)=(ssenxy(3,j,l)-ssenxy(2,j,l))/h1
$ -0.041666667*difxyx(2,j,l)
end do
end do

do i=4,n-2
b(i)=-0.0416666667
a(i)=0.916666667
c(i)=-0.0416666667
do l=2,r
do j=1,m
d(i,j,l)=(ssenxy(i,j,l)-ssenxy(i-1,j,l))/h1
end do
end do
end do

b(n-1)=-0.0416666667
a(n-1)=0.916666667
c(n-1)=0.0

do l=2,r
do j=1,m
d(n-1,j,l)=(ssenxy(n-1,j,l)-ssenxy(n-2,j,l))/h1
$ -0.041666667*difxyx(n,j,l)
end do
end do

beta(n)=0

do l=2,r
do j=1,m
gama(n,j,l)=0
end do
end do

do k1=3,n-1
i=n-k1+2
beta(i)=b(i)/(a(i)-c(i)*beta(i+1))
do l=2,r
do j=1,m
gama(i,j,l)=(d(i,j,l)+c(i)*gama(i+1,j,l))
$/(a(i)-c(i)*beta(i+1))
end do
end do
end do

do j=1,m
do l=2,r
u4(2,j,l)=0.0
end do
end do

do i=3,n-1
do j=1,m
do l=2,r
u4(i,j,l)=beta(i)*u4(i-1,j,l)+gama(i,j,l)
difxyx(i,j,l)=u4(i,j,l)
end do
end do
end do

do i=1,n+1
a(i)=0.0
b(i)=0.0
c(i)=0.0
beta(i)=0.0
do j=1,m
do l=1,r+1
gama(i,j,l)=0.0
d(i,j,l)=0.0
enddo
enddo
enddo
c  difxyy---------------------------------
do l=2,r
do i=1,n
difxyy(i,1,l)=(ssenxy(i,1,l)-ssenxy(i,m,l))/h2
difxyy(i,m,l)=(ssenxy(i,m,l)
$-ssenxy(i,m-1,l))/h2
end do
end do

b(2)=0.0
a(2)=0.916666667
c(2)=-0.0416666667

do l=2,r
do i=1,n
d(i,2,l)=(ssenxy(i,2,l)-ssenxy(i,1,l))/h2
$ -0.041666667*difxyy(i,1,l)
end do
end do

do j=3,m-2
b(j)=-0.0416666667
a(j)=0.916666667
c(j)=-0.0416666667
do l=2,r
do i=1,n
d(i,j,l)=(ssenxy(i,j,l)-ssenxy(i,j-1,l))/h2
end do
end do
end do

b(m-1)=-0.0416666667
```

```
a(m-1)=0.916666667
c(m-1)=0.0

do l=2,r
do i=1,n
d(i,m-1,l)=(ssenxy(i,m-1,l)-ssenxy(i,m-2,l))/h2
$ -0.041666667*difxyy(i,m,l)
end do
end do

beta(m)=0

do l=2,r
do i=1,n
gama(i,m,l)=0
end do
end do

do k1=2,m-1
j=m-k1+1
beta(j)=b(j)/(a(j)-c(j)*beta(j+1))
do l=2,r
do i=1,n
gama(i,j,l)=(d(i,j,l)+c(j)*gama(i,j+1,l))
$/(a(j)-c(j)*beta(j+1))
end do
end do
end do

do i=1,n
do l=2,r
u5(i,1,l)=0.0
end do
end do

do i=1,n
do j=2,m-1
do l=2,r
u5(i,j,l)=beta(j)*u5(i,j-1,l)+gama(i,j,l)
difxyy(i,j,l)=u5(i,j,l)
end do
end do
end do

do i=1,n+1
a(i)=0.0
b(i)=0.0
c(i)=0.0
beta(i)=0.0
do j=1,m
do l=1,r+1
gama(i,j,l)=0.0
d(i,j,l)=0.0
enddo
enddo
enddo

c  difxzx--------------------------------
   do l=1,r
   do j=2,m
   difxzx(2,j,l)=(ssenxz(2,j,l)-ssenxz(1,j,l))/h1
   difxzx(n,j,l)=(ssenxz(n,j,l)-ssenxz(n-1,j,l))/h1
   end do
   end do

   b(3)=0.0
   a(3)=0.916666667
   c(3)=-0.0416666667

   do l=1,r
   do j=2,m
   d(3,j,l)=(ssenxz(3,j,l)-ssenxz(2,j,l))/h1
   $ -0.041666667*difxzx(2,j,l)
   end do
   end do

   do i=4,n-2
   b(i)=-0.0416666667
   a(i)=0.916666667
   c(i)=-0.0416666667
   do l=1,r
   do j=2,m
   d(i,j,l)=(ssenxz(i,j,l)-ssenxz(i-1,j,l))/h1
   end do
   end do
   end do

   b(n-1)=-0.0416666667
   a(n-1)=0.916666667
   c(n-1)=0.0

   do l=1,r
   do j=2,m
   d(n-1,j,l)=(ssenxz(n-1,j,l)-ssenxz(n-2,j,l))/h1
   $ -0.041666667*difxzx(n,j,l)
   end do
   end do

   beta(n)=0

   do l=1,r
   do j=2,m
   gama(n,j,l)=0
   end do
   end do

   do k1=3,n-1
   i=n-k1+2
   beta(i)=b(i)/(a(i)-c(i)*beta(i+1))
   do l=1,r
   do j=2,m
   gama(i,j,l)=(d(i,j,l)+c(i)*gama(i+1,j,l))
   $/(a(i)-c(i)*beta(i+1))
```

```
      end do
      end do
      end do

      do j=2,m
      do l=1,r
      u6(2,j,l)=0.0
      end do
      end do

      do i=3,n-1
      do j=2,m
      do l=1,r
      u6(i,j,l)=beta(i)*u6(i-1,j,l)+gama(i,j,l)
      difxzx(i,j,l)=u6(i,j,l)
      end do
      end do
      end do

      do i=1,n+1
      a(i)=0.0
      b(i)=0.0
      c(i)=0.0
      beta(i)=0.0
      do j=1,m
      do l=1,r+1
      gama(i,j,l)=0.0
      d(i,j,l)=0.0
      enddo
      enddo
      enddo
c     difxzz--------------------------------
      do i=1,n
      do j=2,m
      difxzz(i,j,2)=(ssenxz(i,j,2)-ssenxz(i,j,1))/h3
      difxzz(i,j,r)=(ssenxz(i,j,r)-ssenxz(i,j,r-1))/h3
      end do
      end do

      b(3)=0.0
      a(3)=0.916666667
      c(3)=-0.0416666667

      do i=1,n
      do j=2,m
      d(i,j,3)=(ssenxz(i,j,3)-ssenxz(i,j,2))/h3
      $ -0.041666667*difxzz(i,j,2)
      end do
      end do

      do l=4,r-2
      b(l)=-0.0416666667
      a(l)=0.916666667
      c(l)=-0.0416666667
      do i=1,n
      do j=2,m
      d(i,j,l)=(ssenxz(i,j,l)-ssenxz(i,j,l-1))/h3
      end do
      end do
      end do

      b(r-1)=-0.0416666667
      a(r-1)=0.916666667
      c(r-1)=0.0

      do i=1,n
      do j=2,m
      d(i,j,r-1)=(ssenxz(i,j,r-1)-ssenxz(i,j,r-2))/h3
      $ -0.041666667*difxzz(i,j,r)
      end do
      end do

      beta(r)=0.0

      do i=1,n
      do j=2,m
      gama(i,j,r)=0.0
      end do
      end do

      do k1=3,r-1
      l=r-k1+2
      beta(l)=b(l)/(a(l)-c(l)*beta(l+1))
      do i=1,n
      do j=2,m
      gama(i,j,l)=(d(i,j,l)+c(l)*gama(i,j,l+1))
      $/(a(l)-c(l)*beta(l+1))
      end do
      end do
      end do

      do i=1,n
      do j=2,m
      u7(i,j,2)=0.0
      end do
      end do

      do i=1,n
      do j=2,m
      do l=3,r-1
      u7(i,j,l)=beta(l)*u7(i,j,l-1)+gama(i,j,l)
      difxzz(i,j,l)=u7(i,j,l)
      end do
      end do
      end do

      do i=1,n+1
      a(i)=0.0
      b(i)=0.0
      c(i)=0.0
      beta(i)=0.0
      do j=1,m
```

```
      do l=1,r+1
      gama(i,j,l)=0.0
      d(i,j,l)=0.0
      enddo
      enddo
      enddo
c     difyzy--------------------------------
      do l=1,r
      do i=2,n
      difyzy(i,1,l)=(ssenyz(i,1,l)-ssenyz(i,m,l))/h2
      difyzy(i,m,l)=(ssenyz(i,m,l)
     $-ssenyz(i,m-1,l))/h2
      end do
      end do

      b(2)=0.0
      a(2)=0.916666667
      c(2)=-0.0416666667

      do l=1,r
      do i=2,n
      d(i,2,l)=(ssenyz(i,2,l)-ssenyz(i,1,l))/h2
     $ -0.041666667*difyzy(i,1,l)
      end do
      end do

      do j=3,m-2
      b(j)=-0.0416666667
      a(j)=0.916666667
      c(j)=-0.0416666667
      do l=1,r
      do i=2,n
      d(i,j,l)=(ssenyz(i,j,l)-ssenyz(i,j-1,l))/h2
      end do
      end do
      end do

      b(m-1)=-0.0416666667
      a(m-1)=0.916666667
      c(m-1)=0.0

      do l=1,r
      do i=2,n
      d(i,m-1,l)=(ssenyz(i,m-1,l)-ssenyz(i,m-2,l))/h2
     $ -0.041666667*difyzy(i,m,l)
      end do
      end do

      beta(m)=0

      do l=1,r
      do i=2,n
      gama(i,m,l)=0
      end do
      end do

      do k1=2,m-1
      j=m-k1+1
      beta(j)=b(j)/(a(j)-c(j)*beta(j+1))
      do l=1,r
      do i=2,n
      gama(i,j,l)=(d(i,j,l)+c(j)*gama(i,j+1,l))
     $/(a(j)-c(j)*beta(j+1))
      end do
      end do
      end do

      do i=2,n
      do l=1,r
      u8(i,1,l)=0.0
      end do
      end do

      do i=2,n
      do j=2,m-1
      do l=1,r
      u8(i,j,l)=beta(j)*u8(i,j-1,l)+gama(i,j,l)
      difyzy(i,j,l)=u8(i,j,l)
      end do
      end do
      end do

      do i=1,n+1
      a(i)=0.0
      b(i)=0.0
      c(i)=0.0
      beta(i)=0.0
      do j=1,m
      do l=1,r+1
      gama(i,j,l)=0.0
      d(i,j,l)=0.0
      enddo
      enddo
      enddo
c     difyzz--------------------------------
      do i=2,n
      do j=1,m
      difyzz(i,j,2)=(ssenyz(i,j,2)-ssenyz(i,j,1))/h3
      difyzz(i,j,r)=(ssenyz(i,j,r)-ssenyz(i,j,r-1))/h3
      end do
      end do

      b(3)=0.0
      a(3)=0.916666667
      c(3)=-0.0416666667

      do i=2,n
      do j=1,m
      d(i,j,3)=(ssenyz(i,j,3)-ssenyz(i,j,2))/h3
      $ -0.041666667*difyzz(i,j,2)
      end do
      end do
```

```
do l=4,r-2
b(l)=-0.0416666667
a(l)=0.916666667
c(l)=-0.0416666667
do i=2,n
do j=1,m
d(i,j,l)=(ssenyz(i,j,l)-ssenyz(i,j,l-1))/h3
end do
end do
end do

b(r-1)=-0.0416666667
a(r-1)=0.916666667
c(r-1)=0.0

do i=2,n
do j=1,m
d(i,j,r-1)=(ssenyz(i,j,r-1)-ssenyz(i,j,r-2))/h3
$ -0.041666667*difyzz(i,j,r)
end do
end do

beta(r)=0.0

do i=2,n
do j=1,m
gama(i,j,r)=0
end do
end do

do k1=3,r-1
l=r-k1+2
beta(l)=b(l)/(a(l)-c(l)*beta(l+1))
do i=2,n
do j=1,m
gama(i,j,l)=(d(i,j,l)+c(l)*gama(i,j,l+1))
$/(a(l)-c(l)*beta(l+1))
end do
end do
end do

do i=2,n
do j=1,m
u9(i,j,2)=0.0
end do
end do

do i=2,n
do j=1,m
do l=3,r-1
u9(i,j,l)=beta(l)*u9(i,j,l-1)+gama(i,j,l)
difyzz(i,j,l)=u9(i,j,l)
end do
end do
end do
```

```
      do i=1,n+1
      a(i)=0.0
      b(i)=0.0
      c(i)=0.0
      beta(i)=0.0
      do j=1,m
      do l=1,r+1
      gama(i,j,l)=0.0
      d(i,j,l)=0.0
      enddo
      enddo
      enddo

c
c Calculate velocity
c
c
call
$velocity(n,m,r,h1,h2,h3,dt,x,y,z,eto,etn,xseo,yseo,
$zseo,sseoxy,sseoxz,sseoyz,
$xsen,ysen,zsen,ssenxy,ssenxz,ssenyz,
$v1o,v2o,v3o,v1n,v2n,v3n,u1o,u2o,u3o,
$u1n,u2n,u3n,difx,dify,difz,difxyx,difxyy,
$difxzx,difxzz,difyzy,difyzz)

c
c Calculate strain
c
c
      do l=2,r
      do j=1,m
      do i=2,n
      xsan(i,j,l)=((v1n(i,j,l)-v1n(i-1,j,l))/h1)
      $  *dt+xsao(i,j,l)
c
      zsan(i,j,l)=((v1n(i,j,l)+v1n(i-1,j,l))/2+(v3n(i,j,l)
      $  -v3n(i,j,l-1))/h3)*dt/x(i)+zsao(i,j,l)

      if (j.eq.1) then
      ysan(i,j,l)=((v2n(i,j,l)-v2n(i,m,l))/h2+(v1n(i,j,l)
      $+v1n(i-1,j,l))*sin(z(l))/2+(v3n(i,j,l)
      $+v3n(i,j,l-1))*(cos(z(l)))/2)*dt/(x(i)*sin(z(l)))
      $+ysao(i,j,l)
      else
      ysan(i,j,l)=((v2n(i,j,l)-v2n(i,j-1,l))/h2+(v1n(i,j,l)
      $+v1n(i-1,j,l))*sin(z(l))/2+(v3n(i,j,l)
      $+v3n(i,j,l-1)) *(cos(z(l)))/2)*dt
      $/(x(i)*sin(z(l)))+ysao(i,j,l)
      endif
      end do
      end do
      end do
c
c Shear strain
c
```

```
      do l=2,r
      do j=1,m
      do i=2,n-1

      if (j.eq.m) then
      ssanxy(i,j,l)=((v1n(i,1,l)-v1n(i,j,l))/((x(i)+h1/2)
      $*h2*sin(z(l)))+(v2n(i+1,j,l)-v2n(i,j,l))/h1
      $-(v2n(i+1,j,l) +v2n(i,j,l))/(2*(x(i)+h1/2)))
      $*dt/2+ssaoxy(i,j,l)
      else
      ssanxy(i,j,l)=((v1n(i,j+1,l)-1n(i,j,l))/((x(i)+h1/2)
      $*h2*sin(z(l)))+(v2n(i+1,j,l)-v2n(i,j,l))/h1
      $-(v2n(i+1,j,l)  +v2n(i,j,l))/(2*(x(i)+h1/2)))
      $*dt/2+ssaoxy(i,j,l)
      endif
      enddo
      enddo
      enddo

      do l=2,r-1
      do j=1,m
      do i=2,n-1
      ssanxz(i,j,l)=((v1n(i,j,l+1)-1n(i,j,l))/((x(i)+h1/2)
      $*h3)+(v3n(i+1,j,l)-v3n(i,j,l))/h1-(v3n(i+1,j,l)
      $+v3n(i,j,l))/(2*(x(i)+h1/2)))*dt/2+ssaoxz(i,j,l)
      enddo
      enddo
      enddo

      do l=2,r-1
      do j=1,m
      do i=2,n
      if (j.eq.m) then
      ssanyz(i,j,l)=((v3n(i,1,l)-v3n(i,j,l))
      $/(h2*sin(z(l)+h3/2))+(v2n(i,j,l+1)
      $-v2n(i,j,l))/h3-(v2n(i,j,l+1)+v2n(i,j,l))
      $*(cos(z(l)+h3/2))/(2*sin(z(l)+h3/2)))
      $*dt/(2*x(i))+ssaoyz(i,j,l)
      else
      ssanyz(i,j,l)=((v3n(i,j+1,l)-v3n(i,j,l))
      $/(h2*sin(z(l)+h3/2)) +(v2n(i,j,l+1)-v2n(i,j,l))
      $/h3-(v2n(i,j,l+1)+v2n(i,j,l))
      $*(cos(z(l)+h3/2))/(2*sin(z(l)+h3/2)))
      $*dt/(2*x(i))+ssaoyz(i,j,l)
      endif
      end do
      end do
      end do


c
c Completion of calculation of strain
c
      do l=1,r+1
      do j=1,m
      do i=1,n+1

      det1=xsan(i,j,l)-xsaoo(i,j,l)
      det2=ysan(i,j,l)-ysaoo(i,j,l)
      det3=zsan(i,j,l)-zsaoo(i,j,l)
      det4=ssanxy(i,j,l)-ssaooxy(i,j,l)
      det5=ssanxz(i,j,l)-ssaooxz(i,j,l)
      det6=ssanyz(i,j,l)-ssaooyz(i,j,l)

      det=max(abs(det1),abs(det2),abs(det3),
      $abs(det4),abs(det5),abs(det6))
      if( abs(det).gt.detuvmax) detuvmax=abs(det)
      if( abs(det1).gt.det1max) det1max=abs(det1)
      if( abs(det2).gt.det2max) det2max=abs(det2)
      if( abs(det3).gt.det3max) det3max=abs(det3)
      if( abs(det4).gt.det4max) det4max=abs(det4)
      if( abs(det5).gt.det5max) det5max=abs(det5)
      if( abs(det6).gt.det6max) det6max=abs(det6)
      end do
      end do
      end do


c
      do l=1,r+1
      do j=1,m
      do i=1,n+1
      xsaoo(i,j,l)=xsan(i,j,l)
      ysaoo(i,j,l)=ysan(i,j,l)
      zsaoo(i,j,l)=zsan(i,j,l)
      ssaooxy(i,j,l)=ssanxy(i,j,l)
      ssaooxz(i,j,l)=ssanxz(i,j,l)
      ssaooyz(i,j,l)=ssanyz(i,j,l)
      end do
      end do
      end do
c
      write(*,*) 'detuvmax=', detuvmax
c
c  End do with detmax
c
      end do


c-------------------------------------------------------------
      do l=1,r+1
      do j=1,m
      do i=1,n+1
      eto(i,j,l)=etn(i,j,l)
      lto(i,j,l)=ltn(i,j,l)
      xsao(i,j,l)=xsan(i,j,l)
      ysao(i,j,l)=ysan(i,j,l)
      zsao(i,j,l)=zsan(i,j,l)
      ssaoxy(i,j,l)=ssanxy(i,j,l)
      ssaoxz(i,j,l)=ssanxz(i,j,l)
      ssaoyz(i,j,l)=ssanyz(i,j,l)
      xseo(i,j,l)=xsen(i,j,l)
      yseo(i,j,l)=ysen(i,j,l)
```

```
      zseo(i,j,l)=zsen(i,j,l)
      sseoxy(i,j,l)=ssenxy(i,j,l)
      sseoxz(i,j,l)=ssenxz(i,j,l)
      sseoyz(i,j,l)=ssenyz(i,j,l)
      v1o(i,j,l)=v1n(i,j,l)
      v2o(i,j,l)=v2n(i,j,l)
      v3o(i,j,l)=v3n(i,j,l)
      u1o(i,j,l)=u1n(i,j,l)
      u2o(i,j,l)=u2n(i,j,l)
      u3o(i,j,l)=u3n(i,j,l)
      qexo(i,j,l)=qex(i,j,l)
      qeyo(i,j,l)=qey(i,j,l)
      qezo(i,j,l)=qez(i,j,l)
      qlxo(i,j,l)=qlx(i,j,l)
      qlyo(i,j,l)=qly(i,j,l)
      qlzo(i,j,l)=qlz(i,j,l)
      end do
      end do
      end do

      etm(k)=etn(n+1,1,1)
      etm1(k)=etn(n+1,1,r+1)
      ltm(k)=ltn(n+1,1,1)
      ltm1(k)=ltn(n+1,1,r+1)
      u1m(k)=u1n(n2,1,1)
      u2m(k)=u2n(n2,1,1)
      u3m(k)=u3n(n2,1,1)

      write (*,*) 'xsen = ',xsen(n,1,1)
      write (*,*) 'u1m=',u1m(k)
      write (*,*) 'u2m=',u2m(k)
      write (*,*) 'u3m=',u3m(k)



c
      counter=counter+1
      write(*,*) counter
c
c
c     Output intermediate result
c
c
      open(unit=62,file='etm.txt')
      write(62,1020) t2(k), etm(k)
      open(unit=71,file='etm2.txt')
      write(71,1020)  etm(k)
      open(unit=8,file='tk.txt')
      write(8,1020) t2(k)
      open(unit=9,file='ltm2.txt')
      write(9,1020)  ltm(k)
      open(unit=10,file='ltm.txt')
      write(10,1020) t2(k), ltm(k)
      open(unit=11,file='uu1n.txt')
      write (11,1020) t2(k),u1m(k)
      open(unit=12,file='uu2n.txt')
      write (12,1020) t2(k),u2m(k)
      open(unit=13,file='uu3n.txt')
      write (13,1020) t2(k),u3m(k)

c
    1 end do
c
   1010  format(401e15.6)
   1020  format(e15.6,2e15.6)
   end

C subroutines

C calculate temperature

subroutine
$temp(n,m,r,lx,h1,h2,h3,x,y,z,t,dt,lto,ltn,eto,etn,
$xsan,ysan,zsan,xsao,ysao,zsao,qexo,qeyo,
$qezo,qlxo,qlyo,qlzo,qex,qey,qez,
$qlx,qly,qlz)
c
      implicit double precision (a-h,l,o-z)
      dimension x(150),y(50),z(50)
      $eto(150,50,50),etn(150,50,50),lto(150,50,50),
      $ltn(150,50,50),xsan(101,31,31),
      $ysan(101,31,31),zsan(101,31,31),
      $xsao(101,31,31),ysao(101,31,31),
      $zsao(101,31,31),  qexo(101,31,31),
      $qeyo(101,31,31), qezo(101,31,31),
      $qlxo(101,31,31),qlyo(101,31,31),
      $qlzo(101,31,31), qex(101,31,31),
      $qey(101,31,31), qez(101,31,31),
      $qlx(101,31,31),qly(101,31,31), qlz(101,31,31),
      $TEnew(150,50,50),TLnew(150,50,50),
      $TEo(150,50,50),TEold(150,50,50),
      $TLo(150,50,50),TLold(150,50,50),
      $tkex(101,31,31),tkey(101,31,31),
      $tkez(101,31,31)
      integer iteration,l,r,flagE,flagL

C data
C Lame constant
      clemta=199.0d+9
C Shear modulus
      cmiu=27.0d+9
C Thermal expansion coefficient
      alpha=14.2d-6
C Electron heat capacity
      Ae=70.0
C Lattic heat capacity
      cl=2.5d+6
C Electron - lattic coupling factor
      g=2.6d+16
C Electron thermal conducitivity
      tk0=315.0
C Laser fluence
```

```
      flu=500.0
C Laser pulse duration
      tp=0.01d-12
C Optical penetration depth
      zs=15.3d-9
C Surface reflectivity
      sur=0.93
C Spatial profile parameters
      rs=1.0d-6
C Electron relaxation time
      taue=0.04D-12
C Lattice relaxation time
      taul=0.8D-12

      nx=n
      ny=m
      nz=r
      dx=h1
      dy=h2
      dz=h3

      do j=1,ny
      do i=1,nx+1
      do k=1,nz+1
      TEold(i,j,k)=etn(i,j,k)
      TLold(i,j,k)=ltn(i,j,k)
      TEo(i,j,k)=eto(i,j,k)
      TLo(i,j,k)=lto(i,j,k)
      enddo
      enddo
      enddo

      iteration=0


      d=dt*dx*dy*dz
      sse=dt/(2.0*taue+dt)
      ssl=dt/(2.0*taul+dt)
      ee=-2.0*dx*dy*dz*(3.0*clemta+2.0*cmiu)
      $*300.0*alpha
      ffl=2.0*dx*dy*dz*cl

C-------------------------Iteration starts----------------
C flagE and flagL indicate whether TE and TL are
C  precise enough
C keep on iterating as long as flagE or flagL
C equals to 1
2     flagE=1
      flagL=1

      aximumE=0.0

      do j=1,ny
      do i=2,nx+1
      do k=2,nz+1
      temp1=abs((TEold(i,j,k)+TEo(i,j,k))
      $/(TLold(i,j,k)+TLo(i,j,k)))
      temp2=abs((TEold(i-1,j,k)+TEo(i-1,j,k))
      $/(TLold(i-1,j,k)+TLo(i-1,j,k)))

      if (j.eq.1) then
      temp3=abs((TEold(i,ny,k)+TEo(i,ny,k))
      $ /(TLold(i,ny,k)+TLo(i,ny,k)))
      else
      temp3=abs((TEold(i,j-1,k)+TEo(i,j-1,k))
      $/(TLold(i,j-1,k)+TLo(i,j-1,k)))
      endif

      temp4=abs((TEold(i,j,k-1)+TEo(i,j,k-1))
      $/(TLold(i,j,k-1)+TLo(i,j,k-1)))

      tkex(i,j,k)=tk0*(TEo(i,j,k)/TLo(i,j,k))
      tkey(i,j,k)=tk0*(TEo(i,j,k)/TLo(i,j,k))
      tkez(i,j,k)=tk0*(TEo(i,j,k)/TLo(i,j,k))
      enddo
      enddo
      enddo

      do j=1,ny
      do k=1,nz+1
      tkex(1,j,k)=tkex(2,j,k)
      tkex(nx+1,j,k)=tkex(nx,j,k)
       tkey(1,j,k)=tkey(2,j,k)
      tkey(nx+1,j,k)=tkey(nx,j,k)
      tkez(1,j,k)=tkez(2,j,k)
      tkez(nx+1,j,k)=tkez(nx,j,k)
      enddo
      enddo


      do j=1,ny
      do i=1,nx+1
      tkex(i,j,1)=tkex(i,j,2)
      tkex(i,j,nz+1)=tkex(i,j,nz)
      tkey(i,j,1)=tkey(i,j,2)
      tkey(i,j,nz+1)=tkey(i,j,nz)
      tkez(i,j,1)=tkez(i,j,2)
      tkez(i,j,nz+1)=tkez(i,j,nz)
      enddo
      enddo

      do j=1,ny
      do i=2,nx
      do k=2,nz
       S=0.94*flu*(1.0-sur)/(tp*zs)
      $*exp(-(lx-x(i))/zs)
      $*exp(-2.77*(t-2.0*tp)*(t-2.0*tp)/(tp*tp))
      $*abs(cos(z(k)))
      if (abs(S)<1D-200)
      S=0.0
      ffe=4.0*Ae*dx*dy*dz*(TEold(i,j,k)
      $*TEold(i,j,k) +TEold(i,j,k)*TEo(i,j,k)
```

```
$+TEo(i,j,k)*TEo(i,j,k))
$/(3.0*(TEold(i,j,k)+TEo(i,j,k)))
tempx1=dt*dy*dz*tkex(i,j,k)/dx
tempx2=x(i-1)*x(i-1)/(x(i)*x(i))
 $*dt*dy*dz*tkex(i-1,j,k)/dx

tempy1=1.0/(x(i)*x(i)*sin(z(k))*sin(z(k)))
 $*dt*dx*dz*tkey(i,j,k)/dy

if (j.eq.1) then
 tempy2=1.0/(x(i)*x(i)*sin(z(k))*sin(z(k)))
 $*dt*dx*dz*tkey(i,ny,k)/dy
else
 tempy2=1.0/(x(i)*x(i)*sin(z(k))*sin(z(k)))
 $*dt*dx*dz*tkey(i,j-1,k)/dy
 endif

tempz1=1.0/(x(i)*x(i))*dt*dx*dy*tkez(i,j,k)/dz
tempz2=sin(z(k-1))/(x(i)*x(i)*sin(z(k)))*dt
$*dx*dy*tkez(i,j,k-1)/dz

a1=ffe+(tempx1+tempx2+tempy1+tempy2
$+tempz1+tempz2)*sse+g*d

b1=-4.0*dy*dz*taue*sse*(qexo(i,j,k)
$-x(i-1)*x(i-1)/(x(i)*x(i))*qexo(i-1,j,k))

if (j.eq.1) then
b2=-4.0*dx*dz*taue*sse/(x(i)*sin(z(k)))
$*(qeyo(i,j,k)-qeyo(i,ny,k))
else
b2=-4.0*dx*dz*taue*sse/(x(i)*sin(z(k)))
$*(qeyo(i,j,k) -qeyo(i,j-1,k))
endif

b3=-4.0*dx*dy*taue*sse*(1.0/x(i)*qezo(i,j,k)
$-sin(z(k-1))/(x(i)*sin(z(k)))*qezo(i,j,k-1))

c1=tempx1*sse*(TEold(i+1,j,k)+TEo(i+1,j,k)
$-TEo(i,j,k))
c2=-tempx2*sse*(TEo(i,j,k)-TEold(i-1,j,k)
$-TEo(i-1,j,k))

if (j.eq.ny) then
c3=tempy1*sse*(TEold(i,1,k)+TEo(i,1,k)
$-TEo(i,j,k))
else
c3=tempy1*sse*(TEold(i,j+1,k)+TEo(i,j+1,k)
$-TEo(i,j,k))
endif

if (j.eq.1) then
c4=-tempy2*sse*(TEo(i,j,k)-TEold(i,ny,k)
$-TEo(i,ny,k))
else
    c4=-tempy2*sse*(TEo(i,j,k)-TEold(i,j-1,k)-
    $TEo(i,j-1,k))
    endif


    c5=tempz1*sse*(TEold(i,j,k+1)+TEo(i,j,k+1)-
    $TEo(i,j,k))
    c6=-tempz2*sse*(TEo(i,j,k)-TEold(i,j,k-1)-
    $TEo(i,j,k-1))

    d1=-g*d*(TEo(i,j,k)-TLold(i,j,k)-TLo(i,j,k))

    if (k.le.(NINT(nz/4.0)+1)) then
    d2=2.0*d*S
    else
    d2=0.0
    endif

    d3=ffe*TEo(i,j,k)

C TEnew: The new TE at the right now time point
C calculated based on the
C old TEs and TEs at the previous time point
    TEnew(i,j,k)=(b1+b2+b3+c1+c2+c3+c4+c5
    $+c6+d1+d2+d3)/a1

    if (TEnew(i,j,k)<300.0) TEnew(i,j,k)=300.0

C dTE: Difference between the new and the old
C  TEs at now time point
    dTE=abs(TEnew(i,j,k)-TEold(i,j,k))

C aximumE: The maximum difference between all
C  the new and the old TEs
    if (aximumE.le.dTE) then
    aximumE=dTE
    endif
    enddo
    enddo
    enddo

C If aximumE is less than 1D-6, which means that
C  the largest difference
C between the new and old TEs are smaller than
C 1D-6, make flagE=0 to
C stop the iteration for TE
    if(aximumE.lt.1D-2) then
     flagE=0
    endif

C Since the calculation have no boundary TEs, just
C make them equal to
C their neighbours
    do j=1,ny
    do k=1,nz+1
    TEnew(1,j,k)=TEnew(2,j,k)
```

```
      TEnew(nx+1,j,k)=TEnew(nx,j,k)
      enddo
      end do


      do j=1,ny
      do i=1,nx+1
      TEnew(i,j,1)=TEnew(i,j,2)
      TEnew(i,j,nz+1)=TEnew(i,j,nz)
      enddo
      enddo


C-------------TEnew Calculated--------------
C*****************************
C--------------Start calculating TLnew------
      aximumL=0.0

      do j=1,ny
      do i=2,nx
      do k=2,nz
      tempx1=dt*dy*dz*tk0/dx
      tempx2=x(i-1)*x(i-1)/(x(i)*x(i))*dt*dy
     $*dz*tk0/dx
      tempy1=1.0/(x(i)*x(i)*sin(z(k))*sin(z(k)))
     $*dt*dx*dz*tk0/dy
      tempy2=1.0/(x(i)*x(i)*sin(z(k))*sin(z(k)))
     $*dt*dx*dz*tk0/dy
       tempz1=1.0/(x(i)*x(i))*dt*dx*dy*tk0/dz
       tempz2=sin(z(k-1))/(x(i)*x(i)*sin(z(k)))*dt
     $*dx*dy*tk0/dz
      e1=ffl+(tempx1+tempx2+tempy1+tempy2
     $+tempz1+tempz2)*ssl+d*g
      f1=-4.0*dy*dz*taul*ssl*(qlxo(i,j,k)
     $-x(i-1)*x(i-1)/(x(i)*x(i)) *qlxo(i-1,j,k))

      if (j.eq.1) then
      f2=-4.0*dx*dz*taul*ssl/(x(i)*sin(z(k)))
     $*(qlyo(i,j,k)-qlyo(i,ny,k))
      else
      f2=-4.0*dx*dz*taul*ssl/(x(i)*sin(z(k)))
     $*(qlyo(i,j,k)-qlyo(i,j-1,k))
      endif

      f3=-4.0*dx*dy*taul*ssl*(1.0/x(i)
     $*qlzo(i,j,k)-sin(z(k-1))
     $/(x(i)*sin(z(k)))*qlzo(i,j,k-1))

      g1=tempx1*ssl*(TLold(i+1,j,k)+TLo(i+1,j,k)
     $-TLo(i,j,k))
      g2=-tempx2*ssl*(TLo(i,j,k)-TLold(i-1,j,k)
     $-TLo(i-1,j,k))

      if (j.eq.ny) then
      g3=tempy1*ssl*(TLold(i,1,k)+TLo(i,1,k)
     $-TLo(i,j,k)
      else
      g3=tempy1*ssl*(TLold(i,j+1,k)+TLo(i,j+1,k)
     $-TLo(i,j,k))
      endif

      if (j.eq.1) then
      g4=-tempy2*ssl*(TLo(i,j,k)-TLold(i,ny,k)
     $-TLo(i,ny,k))
      else
      g4=-tempy2*ssl*(TLo(i,j,k)-TLold(i,j-1,k)
     $-TLo(i,j-1,k))
      endif

      g5=tempz1*ssl*(TLold(i,j,k+1)+TLo(i,j,k+1)
     $-TLo(i,j,k))
      g6=-tempz2*ssl*(TLo(i,j,k)-TLold(i,j,k-1)
     $-TLo(i,j,k-1))

      hh1=-g*d*(TLo(i,j,k)-TEnew(i,j,k)-TEo(i,j,k))
      hh2=ee*(xsan(i,j,k)+ysan(i,j,k)+zsan(i,j,k)
     $-(xsao(i,j,k)+ysao(i,j,k)+zsao(i,j,k)))
      hh3=ffl*TLo(i,j,k)

      TLnew(i,j,k)=(f1+f2+f3+g1+g2+g3+g4+g5+g6
     $+hh1+hh2+hh3)/e1

      dTL=abs(TLnew(i,j,k)-TLold(i,j,k))

      if(aximumL.lt.dTL) then
      aximumL=dTL
      endif
      enddo
      enddo
      enddo

      if(aximumL.lt.1D-3) then
      flagL=0
      endif

      do j=1,ny
      do k=1,nz+1
      TLnew(1,j,k)=TLnew(2,j,k)
      TLnew(nx+1,j,k)=TLnew(nx,j,k)
      end do
      end do


      do j=1,ny
      do i=1,nx+1
      TLnew(i,j,1)=TLnew(i,j,2)
      TLnew(i,j,nz+1)=TLnew(i,j,nz)
      enddo
      enddo

C Update all the TEold, TLold with TEnew and
C  TLnew
```

```
      do j=1,ny+1
      do i=1,nx+1
      do k=1,nz+1
       TEold(i,j,k)=TEnew(i,j,k)
       TLold(i,j,k)=TLnew(i,j,k)
      enddo
      enddo
      enddo

C If flagE or flagL is still be 1, then we should go
C back to 2
C to do iteration again to calculate new TE and TL
CUse "iteration" to remember the time of iterations

      if((flagE.eq.1).OR.(flagL.eq.1)) then
      iteration=iteration+1
      TEnew(nx+1,1,nz+1)
      goto 2
      else

C-------------Compute qex, qey and qez------------
       do j=1,ny
       do i=1,nx
       do k=2,nz
       qex(i,j,k)=4.0*taue*sse*qexo(i,j,k)
        $/dt-qexo(i,j,k)-tkex(i,j,k)*sse
        $*(TEold(i+1,j,k)+TEo(i+1,j,k)
        $-TEold(i,j,k)-TEo(i,j,k))/dx
       enddo
       enddo
       enddo

      do j=1,ny
      do i=2,nx
      do k=2,nz
      if (j.eq.ny) then
      qey(i,j,k)=4.0*taue*sse*qeyo(i,j,k)/dt
      $-qeyo(i,j,k) -tkey(i,j,k)*sse
      $*(TEold(i,1,k)+TEo(i,1,k)
      $-TEold(i,j,k)-TEo(i,j,k))
      $/(dy*x(i)*sin(z(k)))
      else
      qey(i,j,k)=4.0*taue*sse*qeyo(i,j,k)/dt
      $-qeyo(i,j,k) -tkey(i,j,k)*sse
      $*(TEold(i,j+1,k)+TEo(i,j+1,k)
      $-TEold(i,j,k)-TEo(i,j,k))
      $/(dy*x(i)*sin(z(k)))
      endif
      enddo
      enddo
      enddo

       do j=1,ny
       do i=2,nx
       do k=1,nz
       qez(i,j,k)=4.0*taue*sse*qezo(i,j,k)/dt
      $-qezo(i,j,k) -tkez(i,j,k)*sse
      $(TEold(i,j,k+1)+TEo(i,j,k+1)
      $-TEold(i,j,k)-TEo(i,j,k))
      $/(dz*x(i))
      enddo
      enddo
      enddo

C-------------Compute qlx, qly and qlz----------------
      do j=1,ny
      do i=1,nx
      do k=2,nz
      qlx(i,j,k)=4.0*taul*ssl*qlxo(i,j,k)/dt-qlxo(i,j,k)
      $-tk0*ssl*(TLold(i+1,j,k)+TLo(i+1,j,k)
      $-TLold(i,j,k)-TLo(i,j,k))/dx
      enddo
      enddo
      enddo

      do j=1,ny
      do i=2,nx
      do k=2,nz
      if (j.eq.ny) then
      qly(i,j,k)=4.0*taul*ssl*qlyo(i,j,k)/dt-qlyo(i,j,k)
      $-tk0*ssl*(TLold(i,1,k)+TLo(i,1,k)
      $-TLold(i,j,k)-TLo(i,j,k))/(dy*x(i)*sin(z(k)))
      else
      qly(i,j,k)=4.0*taul*ssl*qlyo(i,j,k)/dt-qlyo(i,j,k)
      $-tk0*ssl*(TLold(i,j+1,k)+TLo(i,j+1,k)
      $-TLold(i,j,k)-TLo(i,j,k))/(dy*x(i)*sin(z(k)))
      endif
      enddo
      enddo
      enddo

      do j=1,ny
      do i=2,nx
      do k=1,nz
      qlz(i,j,k)=4.0*taul*ssl*qlzo(i,j,k)/dt-qlzo(i,j,k)
      $-tk0*ssl*(TLold(i,j,k+1)+TLo(i,j,k+1)
      $-TLold(i,j,k)-TLo(i,j,k))/(dz*x(i))
      enddo
      enddo
      enddo

      endif


      do j=1,ny
      do k=1,nz+1
      qex(1,j,k)=qex(2,j,k)
      qex(nx+1,j,k)=qex(nx,j,k)
      qey(1,j,k)=qey(2,j,k)
      qey(nx+1,j,k)=qey(nx,j,k)
      qez(1,j,k)=qez(2,j,k)
      qez(nx+1,j,k)=qez(nx,j,k)
```

```
      qlx(1,j,k)=qlx(2,j,k)
      qlx(nx+1,j,k)=qlx(nx,j,k)
      qly(1,j,k)=qly(2,j,k)
      qly(nx+1,j,k)=qly(nx,j,k)
      qlz(1,j,k)=qlz(2,j,k)
      qlz(nx+1,j,k)=qlz(nx,j,k)
      enddo
      enddo


      do j=1,ny
      do i=1,nx+1
      qex(i,j,1)=qex(i,j,2)
      qex(i,j,nz+1)=qex(i,j,nz)
      qey(i,j,1)=qey(i,j,2)
      qey(i,j,nz+1)=qey(i,j,nz)
      qez(i,j,1)=qez(i,j,2)
      qez(i,j,nz+1)=qez(i,j,nz)
      qlx(i,j,1)=qlx(i,j,2)
      qlx(i,j,nz+1)=qlx(i,j,nz)
      qly(i,j,1)=qly(i,j,2)
      qly(i,j,nz+1)=qly(i,j,nz)
      qlz(i,j,1)=qlz(i,j,2)
      qlz(i,j,nz+1)=qlz(i,j,nz)
      enddo
      enddo

      do j=1,ny
      do i=1,nx+1
      do k=1,nz+1
      etn(i,j,k)=TEnew(i,j,k)
      ltn(i,j,k)=TLnew(i,j,k)
      enddo
      enddo
      enddo

      write (*,*) 'Te bottom = ', etn(n+1,1,r+1)
      write (*,*) 'Te top = ', etn(n+1,1,1)

C--------------------Iterations Done----------------
      return
      end
C   end of subroutine temp()


subroutine
velocity(n,m,r,h1,h2,h3,dt,x,y,z,eto,etn,xseo,yseo,
$  zseo,sseoxy,sseoxz,sseoyz,
$  xsen,ysen,zsen,ssenxy,ssenxz,ssenyz,
$  v1o,v2o,v3o,v1n,v2n,v3n,u1o,u2o,u3o,
$  u1n,u2n,u3n,difx,dify,difz,difxyx,difxyy,
$  difxzx,difxzz,difyzy,difyzz)

      implicit double precision (a-h,l,o-z)
      dimension x(150),y(50),z(50)
      dimension eto(150,50,50),etn(150,50,50),
     $xseo(101,31,31),yseo(101,31,31),
     $zseo(101,31,31),
     $sseoxy(101,31,31),sseoxz(101,31,31),
     $sseoyz(101,31,31),  xsen(101,31,31),
     $ysen(101,31,31),zsen(101,31,31),
     $ssenxy(101,31,31),ssenxz(101,31,31),
     $ssenyz(101,31,31), v1o(101,31,31),
     $v2o(101,31,31),v3o(101,31,31),
     $ v1n(101,31,31),v2n(101,31,31),
     $v3n(101,31,31), u1o(101,31,31),
     $u2o(101,31,31),u3o(101,31,31),
     $u1n(101,31,31),u2n(101,31,31),
     $u3n(101,31,31),difx(101,31,31),
     $dify(101,31,31),difz(101,31,31),
     $difxyx(101,31,31),difxyy(101,31,31),
     $difxzx(101,31,31),difxzz(101,31,31),
     $difyzy(101,31,31),difyzz(101,31,31)

      integer l,r
c Density
      lou=1.93d+4
c Electron - blast coefficient
      tri=70.0

      do l=2,r
      do j=1,m
      do i=1,n
      if (j.eq.1) then
      v1n(i,j,l)=v1o(i,j,l)+dt*difx(i,j,l)
      $/(lou)+dt*difxzz(i,j,l)
      $/(lou*(x(i)+h1/2))+dt*difxyy(i,j,l)
      $  /(lou*(x(i)+h1/2)*sin(z(l)))+dt
      $*((xsen(i,j,l)+xsen(i+1,j,l))
      $  -(ysen(i,j,l)+ysen(i+1,j,l))/2
      $-(zsen(i,j,l)+zsen(i+1,j,l))/2
      $+(ssenxz(i,j,l)+ssenxz(i,j,l-1))
      $*(cos(z(l)))/(2*sin(z(l)))) /(lou*(x(i)+h1/2))
      $+dt*tri*(etn(i+1,j,l)*etn(i+1,j,l)
      $-etn(i,j,l)*etn(i,j,l))/(lou*h1)
      else
      v1n(i,j,l)=v1o(i,j,l)+dt*difx(i,j,l)
      $  /(lou)+dt*difxzz(i,j,l)
      $  /(lou*(x(i)+h1/2))+dt*difxyy(i,j,l)
      $  /(lou*(x(i)+h1/2)*sin(z(l)))+dt
      $*((xsen(i,j,l)+xsen(i+1,j,l)) -(ysen(i,j,l)
      $+ysen(i+1,j,l))/2-(zsen(i,j,l)+zsen(i+1,j,l))
      $/2+(ssenxz(i,j,l)+ssenxz(i,j,l-1))
      $*(cos(z(l)))/(2*sin(z(l)))) /(lou*(x(i)+h1/2))
      $+dt*tri*(etn(i+1,j,l)*etn(i+1,j,l)
      $-etn(i,j,l)*etn(i,j,l))/(lou*h1)
      endif

      u1n(i,j,l)=v1n(i,j,l)*dt+u1o(i,j,l)
      end do
      end do
      end do
```

```
do j=1,m
do i=1,n
u1n(i,j,1)=u1n(i,j,2)
u1n(i,j,r+1)=u1n(i,j,r)
v1n(i,j,1)=v1n(i,j,2)
v1n(i,j,r+1)=v1n(i,j,r)
end do
end do


do i=2,n
do j=1,m
do l=1,r
if (j.eq.1) then
v3n(i,j,l)=v3o(i,j,l)+dt*difxzx(i,j,l)
$ /(lou)+dt*difz(i,j,l)
$ /(lou*x(i))+dt*difyzy(i,j,l)
$ /(lou*x(i)*sin(z(l)+h3/2))+dt*(((zsen(i,j,l+1)
$ +zsen(i,j,l))/2-(ysen(i,j,l+1)+ysen(i,j,l))/2)
$ *(cos(z(l)+h3/2))/sin(z(l)+h3/2)+3
$*(ssenxz(i,j,l) +ssenxz(i-1,j,l))/2)/(lou*x(i))
$+dt*tri*(etn(i,j,l+1)
$*etn(i,j,l+1)-etn(i,j,l)*etn(i,j,l))/(lou*h3*x(i))
else
v3n(i,j,l)=v3o(i,j,l)+dt*difxzx(i,j,l)
$ /(lou)+dt*difz(i,j,l)
$ /(lou*x(i))+dt*difyzy(i,j,l)
$ /(lou*x(i)*sin(z(l)+h3/2))+dt*(((zsen(i,j,l+1)
$ +zsen(i,j,l))/2-(ysen(i,j,l+1)+ysen(i,j,l))/2)
$ *(cos(z(l)+h3/2))/sin(z(l)+h3/2)+3
$*(ssenxz(i,j,l) +ssenxz(i-1,j,l))/2)/(lou*x(i))
$+dt*tri*(etn(i,j,l+1)
$ *etn(i,j,l+1)-etn(i,j,l)*etn(i,j,l))/(lou*h3*x(i))
endif

u3n(i,j,l)=v3n(i,j,l)*dt+u3o(i,j,l)
end do
end do
end do

do j=1,m
do i=1,n+1
u3n(i,j,r+1)=u3n(i,j,r)
v3n(i,j,r+1)=v3n(i,j,r)
end do
end do


do j=1,m
do l=1,r+1
u3n(1,j,l)=u3n(2,j,l)
u3n(n+1,j,l)=u3n(n,j,l)
v3n(1,j,l)=v3n(2,j,l)
v3n(n+1,j,l)=v3n(n,j,l)
end do
end do



do l=2,r
do i=2,n
do j=1,m
 if (j.eq.m) then
v2n(i,j,l)=v2o(i,j,l)+dt*difxyx(i,j,l)
$ /(lou)+dt*difyzz(i,j,l)
$ /(lou*x(i))+dt*dify(i,j,l)
$ /(lou*x(i)*sin(z(l)))+dt*((ssenyz(i,j,l)
$ +ssenyz(i,j,l-1))*(cos(z(l)))/sin(z(l))+3
$*(ssenxy(i,j,l) +ssenxy(i-1,j,l))/2)/(lou*x(i))
$+dt*tri*(etn(i,1,l)
$*etn(i,1,l)etn(i,j,l)*etn(i,j,l))
$/(lou*h2*x(i)*sin(z(l)))
else
 v2n(i,j,l)=v2o(i,j,l)+dt*difxyx(i,j,l)
 $ /(lou)+dt*difyzz(i,j,l)
 $ /(lou*x(i))+dt*dify(i,j,l)
 $ /(lou*x(i)*sin(z(l)))+dt*((ssenyz(i,j,l)
 $ +ssenyz(i,j,l-1))*(cos(z(l)))/sin(z(l))+3
 $*(ssenxy(i,j,l) +ssenxy(i-1,j,l))/2)
 $/(lou*x(i))+dt*tri*(etn(i,j+1,l)
 $ *etn(i,j+1,l)-etn(i,j,l)*etn(i,j,l))
 $/(lou*h2*x(i)*sin(z(l)))
 endif

u2n(i,j,l)=v2n(i,j,l)*dt+u2o(i,j,l)
end do
end do
end do

do i=1,n
do j=1,m
u2n(i,j,1)=u2n(i,j,2)
u2n(i,j,r+1)=u2n(i,j,r)
v2n(i,j,1)=v2n(i,j,2)
v2n(i,j,r+1)=v2n(i,j,r)
end do
end do

do l=1,r+1
do j=1,n
u2n(1,j,l)=u2n(2,j,l)
u2n(n+1,j,l)=0
v2n(1,j,l)=v2n(2,j,l)
v2n(n+1,j,l)=0
end do
end do

return
end
```

APPENDIX B

SOURCE CODE FOR 3D DOUBLE-LAYERED CASE

```
c main
implicit double precision (a-h,l,o-z)
dimension t(20010),t1(20010),t2(20010),x(150),
$y(50),z(50),
$x1(150),x2(150),y1(50),z1(50),a(150),b(150),
$c(150),beta(150)
dimension eto(150,50,50),etn(150,50,50),
$lto(150,50,50),ltn(150,50,50),
$etm(20010),etm1(20010),ltm(20010),
$ltm1(20010),u1m(20010),u2m(20010),
$u3m(20010),xsao(101,31,31),ysao(101,31,31),
$zsao(101,31,31),ssaoxy(101,31,31),
$ssaoxz(101,31,31),ssaoyz(101,31,31),
$xsaoo(101,31,31),ysaoo(101,31,31),
$zsaoo(101,31,31),ssaooxy(101,31,31),
$ssaooxz(101,31,31),ssaooyz(101,31,31),
$xsan(101,31,31),ysan(101,31,31),zsan(101,31,31),
$ssanxy(101,31,31),ssanxz(101,31,31),
$ssanyz(101,31,31),xseo(101,31,31),
$yseo(101,31,31),zseo(101,31,31),
$sseoxy(101,31,31),sseoxz(101,31,31),
$sseoyz(101,31,31),xsen(101,31,31),
$ysen(101,31,31),zsen(101,31,31),
$ssenxy(101,31,31),ssenxz(101,31,31),
$ssenyz(101,31,31),difx(101,31,31),
$dify(101,31,31),difz(101,31,31),
$difxyx(101,31,31),difxyy(101,31,31),
$difxzx(101,31,31),difxzz(101,31,31),
$difyzy(101,31,31),difyzz(101,31,31),
$v1o(101,31,31),v2o(101,31,31),v3o(101,31,31),
$v1n(101,31,31),v2n(101,31,31),v3n(101,31,31),
$u1o(101,31,31),u2o(101,31,31),u3o(101,31,31),
$u1n(101,31,31),u2n(101,31,31),u3n(101,31,31),
$d(101,31,31),gama(101,31,31),u1(101,31,31),
$u2(101,31,31),u3(101,31,31),u4(101,31,31),
$u5(101,31,31),u6(101,31,31),u7(101,31,31),
$u8(101,31,31),u9(101,31,31),qexo(101,31,31),
$qeyo(101,31,31),qezo(101,31,31),
$qlxo(101,31,31),qlyo(101,31,31),qlzo(101,31,31),
$qex(101,31,31),qey(101,31,31), qez(101,31,31),
$qlx(101,31,31),qly(101,31,31), qlz(101,31,31)

    integer o, counter,l,r

c Lame constant
    lemta=199.0d+9
c Shear modulus
    cmiu=27.0d+9
c Thermal expansion coefficient
    alphat=14.2d-6

    pi=3.14159265359
    lx=1.0D-7
    ly=2.0*pi
    lz=1.0*pi
    n=80
    n2=NINT(n/2.0)+1
    m=20
    r=20
    h1=lx/n
    h2=ly/m
    h3=lz/r

    dt=0.0025d-12
    o=NINT(20d-12/dt)+1
    counter=0
    t(1)=0
    x(1)=0
    y(1)=0
    z(1)=0

c initial condition
    k=1
    do l=1,r+1
    do j=1,m
    do i=1,n+1
    etn(i,j,l)=300.0
    ltn(i,j,l)=300.0
    eto(i,j,l)=300.0
    lto(i,j,l)=300.0


    xsao(i,j,l)=0.0
    ysao(i,j,l)=0.0
    zsao(i,j,l)=0.0
    ssaoxy(i,j,l)=0.0
    ssaoxz(i,j,l)=0.0
    ssaoyz(i,j,l)=0.0

    xsaoo(i,j,l)=0.0
    ysaoo(i,j,l)=0.0
    zsaoo(i,j,l)=0.0
    ssaooxy(i,j,l)=0.0
    ssaooxz(i,j,l)=0.0
    ssaooyz(i,j,l)=0.0

    xseo(i,j,l)=0.0
    yseo(i,j,l)=0.0
    zseo(i,j,l)=0.0
    sseoxy(i,j,l)=0.0
    sseoxz(i,j,l)=0.0
    sseoyz(i,j,l)=0.0
    difxyx(i,j,l)=0.0
    difxyy(i,j,l)=0.0
    difxzx(i,j,l)=0.0
    difxzz(i,j,l)=0.0
    difyzy(i,j,l)=0.0
    difyzz(i,j,l)=0.0
    difx(i,j,l)=0.0
    dify(i,j,l)=0.0
    difz(i,j,l)=0.0
    v1o(i,j,l)=0.0
```

```
v2o(i,j,l)=0.0
v3o(i,j,l)=0.0
u1o(i,j,l)=0.0
u2o(i,j,l)=0.0
u3o(i,j,l)=0.0

qexo(i,j,l)=0.0
qlxo(i,j,l)=0.0
qex(i,j,l)=0.0
qlx(i,j,l)=0.0
qeyo(i,j,l)=0.0
qlyo(i,j,l)=0.0
qey(i,j,l)=0.0
qly(i,j,l)=0.0
qezo(i,j,l)=0.0
qlzo(i,j,l)=0.0
qez(i,j,l)=0.0
qlz(i,j,l)=0.0


end do
end do
end do

etm(k)=300.0
ltm(k)=300.0

do l=2,r+1
z(l)=z(l-1)+h3
z1(l)=z(l-1)+1.0/r
end do
do j=2,m
y(j)=y(j-1)+h2
y1(j)=y1(j-1)+2.0/m
end do
do i=2,n+1
x(i)=x(i-1)+h1
x1(i)=(x(i-1)+h1)*1.0d6
x2(i)=-(x(i-1)+h1)*1.0d6
end do

write(*,*) 'start'

do 1 k=2,o+1

t(k)=t(k-1)+dt
t1(k)=t(k-1)+dt/2.0
t2(k)=(t(k-1)+dt)*1.0d12

do l=1,r+1
do j=1,m
do i=1,n+1
xsan(i,j,l)=xsao(i,j,l)
ysan(i,j,l)=ysao(i,j,l)
zsan(i,j,l)=zsao(i,j,l)
enddo
    enddo
    enddo

call temp(n,m,r,n2,lx,h1,h2,h3,x,y,z,t1(k),dt,
$lto,ltn,eto,etn,xsan,ysan,zsan,xsao,ysao,zsao,
$qexo,qeyo,qezo,qlxo,qlyo,qlzo,qex,qey,qez,qlx,ql
$y,qlz)

    tol=1d-12
    detuvmax=tol+1d-5
    do while (detuvmax.gt.tol)
    detuvmax=0.0
    det1max=0
    det2max=0
    det3max=0
    det4max=0
    det5max=0
    det6max=0

C Compute normal stress

    do j=1,ny+1
    do k=1,nz+1
    saxn(1,j,k)=0.0
    saxn(nx+1,j,k)=0.0
    end do
    end do

    do i=1,nx+1
    do k=1,nz+1
    sayn(i,1,k)=0.0
    sayn(i,ny+1,k)=0.0
    end do
    end do

    do j=1,ny+1
    do i=1,nx+1
    sazn(i,j,1)=0.0
    sazn(i,j,nz+1)=0.0
    end do
    end do

    do i=2,nx
    do j=2,ny

C gold

    clemta=clemta1
    cmiu=cmiu1
    alpha=alpha1
    do k=2,nz2-1
    saxn(i,j,k)=(clemta+2.0*cmiu)*epxn(i,j,k)
$   +clemta*epyn(i,j,k)
$   +clemta*epzn(i,j,k)
$    -(3.0*clemta+2.0*cmiu)*alpha*(TLold(i,j,k)-
$   300.0)
```

```
      sayn(i,j,k)=clemta*epxn(i,j,k)
$     +(clemta+2.0*cmiu)*epyn(i,j,k)
$     +clemta*epzn(i,j,k)
$     -(3.0*clemta+2.0*cmiu)*alpha*(TLold(i,j,k)-
$     300.0)

      sazn(i,j,k)=clemta*epxn(i,j,k)
$     +(clemta+2.0*cmiu)*epzn(i,j,k)
$     +clemta*epyn(i,j,k)
$     -(3.0*clemta+2.0*cmiu)*alpha*(TLold(i,j,k)
$     -300.0)
      end do

C Chromium

      clemta=clemta2
      cmiu=cmiu2
      alpha=alpha2
      do k=nz2+1,nz
       saxn(i,j,k)=(clemta+2.0*cmiu)*epxn(i,j,k)
$     +clemta*epyn(i,j,k)
$     +clemta*epzn(i,j,k)
$     -(3.0*clemta+2.0*cmiu)*alpha*(TLold(i,j,k)
$     -300.0)

       sayn(i,j,k)=clemta*epxn(i,j,k)
$     +(clemta+2.0*cmiu)*epyn(i,j,k)
$     +clemta*epzn(i,j,k)
$     -(3.0*clemta+2.0*cmiu)*alpha*(TLold(i,j,k)
$     -300.0)

      sazn(i,j,k)=clemta*epxn(i,j,k)
$     +(clemta+2.0*cmiu)*epzn(i,j,k)
$     +clemta*epyn(i,j,k)
$     -(3.0*clemta+2.0*cmiu)*alpha*(TLold(i,j,k)
$     -300.0)
      end do

      k=nz2
      saxn(i,j,k)=(saxn(i,j,k+1)+saxn(i,j,k-1))/2
      sayn(i,j,k)=(sayn(i,j,k+1)+sayn(i,j,k-1))/2
      sazn(i,j,k)=(sazn(i,j,k+1)+sazn(i,j,k-1))/2
      end do
      end do

C Calculate shear stress

      do j=1,ny
      do k=2,nz
      saxyn(1,j,k)=0.0
      saxyn(nx,j,k)=0.0
      end do
      end do

      do i=1,nx
      do k=2,nz
      saxyn(i,1,k)=0.0
      saxyn(i,ny,k)=0.0
      end do
      end do

      do j=2,ny-1
      do i=2,nx-1
      clemta=clemta1
      cmiu=cmiu1
      alpha=alpha1
      do k=2,nz2-1
      saxyn(i,j,k)=cmiu*epxyn(i,j,k)
      end do
      clemta=clemta2
      cmiu=cmiu2
      alpha=alpha2
      do k=nz2+1,nz
      saxyn(i,j,k)=cmiu*epxyn(i,j,k)
      end do
      k=nz2
      saxyn(i,j,k)=(saxyn(i,j,k+1)+saxyn(i,j,k-1))/2
      end do
      end do

      do j=2,ny
      do k=1,nz
      saxzn(1,j,k)=0.0
      saxzn(nx,j,k)=0.0
      end do
      end do

      do i=1,nx
      do j=2,ny
      saxzn(i,j,1)=0.0
      saxzn(i,j,nz)=0.0
      end do
      end do

      do j=2,ny
      do i=2,nx-1
      clemta=clemta1
      cmiu=cmiu1
      alpha=alpha1
      do k=2,nz2-1
      saxzn(i,j,k)=cmiu*epxzn(i,j,k)
      end do
      clemta=clemta2
      cmiu=cmiu2
      alpha=alpha2
      do k=nz2,nz-1
      saxzn(i,j,k)=cmiu*epxzn(i,j,k)
      end do
      end do
      end do
```

```
      do i=2,nx
      do k=1,nz
      sayzn(i,1,k)=0.0
      sayzn(i,ny,k)=0.0
      end do
      end do

      do i=2,nx
      do j=1,ny
      sayzn(i,j,1)=0.0
      sayzn(i,j,nz)=0.0
      end do
      end do

      do j=2,ny-1
      do i=2,nx
      clemta=clemta1
      cmiu=cmiu1
      alpha=alpha1
      do k=2,nz2-1
      sayzn(i,j,k)=cmiu*sayzn(i,j,k)
      end do
      clemta=clemta2
      cmiu=cmiu2
      alpha=alpha2
      do k=nz2,nz-1
      sayzn(i,j,k)=cmiu*sayzn(i,j,k)
      end do
      end do
      end do

C Calculate derivative of stress difx

      do k=2,nz
      do j=2,ny
      difx(1,j,k)=(saxn(2,j,k)-saxn(1,j,k))/dx
      difx(nx,j,k)=(saxn(nx+1,j,k)-saxn(nx,j,k))/dx
      end do
      end do

      b(2)=0.0
      a(2)=11.0/12.0
      c(2)=-1.0/24.0

      do k=2,nz
      do j=2,ny
      d(2,j,k)=(saxn(3,j,k)-saxn(2,j,k))/dx
     $ -1.0/24.0*difx(1,j,k)
      end do
      end do

      do i=3,nx-2
      b(i)=-1.0/24.0
      a(i)=11.0/12.0
      c(i)=-1.0/24.0
      do j=2,ny
      do k=2,nz
      d(i,j,k)=(saxn(i+1,j,k)-saxn(i,j,k))/dx
      end do
      end do
      end do

      b(nx-1)=-1.0/24.0
      a(nx-1)=11.0/12.0
      c(nx-1)=0.0

      do k=2,nz
      do j=2,ny
      d(nx-1,j,k)=(saxn(nx,j,k)-saxn(nx-1,j,k))/dx
     $ -1.0/24.0*difx(nx,j,k)
      end do
      end do

      beta(nx)=0.0
      do k=2,nz
      do j=2,ny
      gama(nx,j,k)=0.0
      end do
      end do

      do m=2,nx-1
      i=nx-m+1
      beta(i)=b(i)/(a(i)-c(i)*beta(i+1))
      do j=2,ny
      do k=2,nz
      gama(i,j,k)=(d(i,j,k)+c(i)*gama(i+1,j,k))/(a(i)
     $ -c(i)*beta(i+1))
      end do
      end do
      end do

      do j=2,ny
      do k=2,nz
      u1(1,j,k)=0.0
      end do
      end do

      do i=2,nx-1
      do j=2,ny
      do k=2,nz
      u1(i,j,k)=beta(i)*u1(i-1,j,k)+gama(i,j,k)
      difx(i,j,k)=u1(i,j,k)
      end do
      end do
      end do

      do i=1,nx
      a(i)=0
      b(i)=0
      c(i)=0
      beta(i)=0
      do j=1,ny
```

```
      do k=1,nz
      gama(i,j,k)=0.0
      d(i,j,k)=0.0
      end do
      end do
      end do

C Calculate derivative of stress dify

      do k=2,nz
      do i=2,nx
      dify(i,1,k)=(sayn(i,2,k)-sayn(i,1,k))/dy
      dify(i,ny,k)=(sayn(i,ny+1,k)-sayn(i,ny,k))/dy
      end do
      end do

      b(2)=0.0
      a(2)=11.0/12.0
      c(2)=-1.0/24.0

      do k=2,nz
      do i=2,nx
      d(i,2,k)=(sayn(i,3,k)-sayn(i,2,k))/dy
     $ -1.0/24.0*dify(i,1,k)
      end do
      end do

      do j=3,ny-2
      b(j)=-1.0/24.0
      a(j)=11.0/12.0
      c(j)=-1.0/24.0
      do i=2,nx
      do k=2,nz
      d(i,j,k)=(sayn(i,j+1,k)-sayn(i,j,k))/dy
      end do
      end do
      end do

      b(ny-1)=-1.0/24.0
      a(ny-1)=11.0/12.0
      c(ny-1)=0.0

      do i=2,nx
      do k=2,nz
      d(i,ny-1,k)=(sayn(i,ny,k)-sayn(i,ny-1,k))/dy
     $ -1.0/24.0*dify(i,ny,k)
      end do
      end do

      beta(ny)=0.0
      do i=2,nx
      do k=2,nz
      gama(i,ny,k)=0.0
      end do
      end do

      do m=2,ny-1
      j=ny-m+1
      beta(j)=b(j)/(a(j)-c(j)*beta(j+1))
      do i=2,nx
      do k=2,nz
      gama(i,j,k)=(d(i,j,k)+c(j)*gama(i,j+1,k))/(a(j)
     $ -c(j)*beta(j+1))
      end do
      end do
      end do

      do i=2,nx
      do k=2,nz
      u2(i,1,k)=0.0
      end do
      end do

      do j=2,ny-1
      do i=2,nx
      do k=2,nz
      u2(i,j,k)=beta(j)*u2(i,j-1,k)+gama(i,j,k)
      dify(i,j,k)=u2(i,j,k)
      end do
      end do
      end do

      do i=1,nx
      a(i)=0
      b(i)=0
      c(i)=0
      beta(i)=0
      do j=1,ny
      do k=1,nz
      gama(i,j,k)=0.0
      d(i,j,k)=0.0
      end do
       end do
      end do

C Calculate derivative of stress difz

      do i=2,nx
      do j=2,ny
      difz(i,j,1)=(sazn(i,j,2)-sazn(i,j,1))/dz
      difz(i,j,nz)=(sazn(i,j,nz+1)-sazn(i,j,nz))/dz
      end do
      end do

      b(2)=0.0
      a(2)=11.0/12.0
      c(2)=-1.0/24.0

      do i=2,nx
      do j=2,ny
      d(i,j,2)=(sazn(i,j,3)-sazn(i,j,2))/dz-
     $ 1.0/24.0*difz(i,j,1)
```

```
      end do
      end do

      do k=3,nz-2
      b(k)=-1.0/24.0
      a(k)=11.0/12.0
      c(k)=-1.0/24.0
     do j=2,ny
     do i=2,nx
     d(i,j,k)=(sazn(i,j,k+1)-sazn(i,j,k))/dz
      end do
      end do
      end do

      b(nz-1)=-1.0/24.0
      a(nz-1)=11.0/12.0
      c(nz-1)=0.0

      do i=2,nx
      do j=2,ny
      d(i,j,nz-1)=(sazn(i,j,nz)-sazn(i,j,nz-1))/dz
$     -1.0/24.0*difz(i,j,nz)
      end do
      end do

      beta(nz)=0.0
      do i=2,nx
      do j=2,ny
      gama(i,j,nz)=0.0
      end do
      end do

      do m=2,nz-1
      k=nz-m+1
      beta(k)=b(k)/(a(k)-c(k)*beta(k+1))
      do j=2,ny
      do i=2,nx
     gama(i,j,k)=(d(i,j,k)+c(k)*gama(i,j,k+1))/(a(k)
$     -c(k)*beta(k+1))
      end do
      end do
      end do

      do i=2,nx
      do j=2,ny
      u3(i,j,1)=0.0
      end do
      end do

      do i=2,nx
      do j=2,ny
      do k=2,nz-1
      u3(i,j,k)=beta(k)*u3(i,j,k-1)+gama(i,j,k)
      difz(i,j,k)=u3(i,j,k)
      end do
      end do
      end do

      do i=1,nx
      a(i)=0
      b(i)=0
      c(i)=0
      beta(i)=0
      do j=1,ny
      do k=1,nz
      gama(i,j,k)=0.0
      d(i,j,k)=0.0
      end do
      end do
      end do

C Calculate derivative of stress difxyx

      do k=2,nz
      do j=1,ny
      difxyx(2,j,k)=(saxyn(2,j,k)-saxyn(1,j,k))/dx
      difxyx(nx,j,k)=(saxyn(nx,j,k)-saxyn(nx-
$     1,j,k))/dx
      end do
      end do

      b(3)=0.0
      a(3)=11.0/12.0
      c(3)=-1.0/24.0

      do k=2,nz
      do j=1,ny
      d(3,j,k)=(saxyn(3,j,k)-saxyn(2,j,k))/dx
$     -1.0/24.0*difxyx(2,j,k)
      end do
      end do

      do i=4,nx-2
      b(i)=-1.0/24.0
      a(i)=11.0/12.0
      c(i)=-1.0/24.0
      do j=1,ny
      do k=2,nz
      d(i,j,k)=(saxyn(i,j,k)-saxyn(i-1,j,k))/dx
      end do
      end do
      end do

      b(nx-1)=-1.0/24.0
      a(nx-1)=11.0/12.0
      c(nx-1)=0.0

      do k=2,nz
      do j=1,ny
      d(nx-1,j,k)=(saxyn(nx-1,j,k)-saxyn(nx-
$     2,j,k))/dx
$     -1.0/24.0*difxyx(nx,j,k)
```

```
      end do
      end do

      beta(nx)=0.0
      do k=2,nz
      do j=1,ny
      gama(nx,j,k)=0.0
      end do
      end do

      do m=3,nx-1
      i=nx-m+2
      beta(i)=b(i)/(a(i)-c(i)*beta(i+1))
      do j=1,ny
      do k=2,nz
      gama(i,j,k)=(d(i,j,k)+c(i)*gama(i+1,j,k))/(a(i)
     $ -c(i)*beta(i+1))
      end do
      end do
      end do

      do j=1,ny
      do k=2,nz
      u4(2,j,k)=0.0
      end do
      end do
      do i=3,nx-1
      do j=1,ny
      do k=2,nz
      u4(i,j,k)=beta(i)*u4(i-1,j,k)+gama(i,j,k)
      difxyx(i,j,k)=u4(i,j,k)
      end do
      end do
      end do

      do i=1,nx
      a(i)=0
      b(i)=0
      c(i)=0
      beta(i)=0
      do j=1,ny
      do k=1,nz
      gama(i,j,k)=0.0
      d(i,j,k)=0.0
      end do
      end do
      end do

C Calculate derivative of stress difxyy

      do k=2,nz
      do i=1,nx
      difxyy(i,2,k)=(saxyn(i,2,k)-saxyn(i,1,k))/dy
      difxyy(i,ny,k)=(saxyn(i,ny,k)-saxyn(i,ny-
     $ 1,k))/dy
      end do
      end do

      b(3)=0.0
      a(3)=11.0/12.0
      c(3)=-1.0/24.0

      do k=2,nz
      do i=1,nx
      d(i,3,k)=(saxyn(i,3,k)-saxyn(i,2,k))/dy
     $ -1.0/24.0*difxyy(i,2,k)
      end do
      end do

      do j=4,ny-2
      b(j)=-1.0/24.0
      a(j)=11.0/12.0
      c(j)=-1.0/24.0
      do i=1,nx
      do k=2,nz
      d(i,j,k)=(saxyn(i,j,k)-saxyn(i,j-1,k))/dy
      end do
      end do
      end do

      b(ny-1)=-1.0/24.0
      a(ny-1)=11.0/12.0
      c(ny-1)=0.0

      do k=2,nz
      do i=1,nx
      d(i,ny-1,k)=(saxyn(i,ny-1,k)-saxyn(i,ny-
     $ 2,k))/dy
     $ -1.0/24.0*difxyy(i,ny,k)
      end do
      end do

      beta(ny)=0.0
      do k=2,nz
      do i=1,nx
      gama(i,ny,k)=0.0
      end do
      end do

      do m=3,ny-1
      j=ny-m+2
      beta(j)=b(j)/(a(j)-c(j)*beta(j+1))
      do i=1,nx
      do k=2,nz
      gama(i,j,k)=(d(i,j,k)+c(j)*gama(i,j+1,k))/(a(j)
     $ -c(j)*beta(j+1))
      end do
      end do
      end do

      do i=1,nx
```

```
      do k=2,nz
      u5(i,2,k)=0.0
      end do
      end do
      do i=1,nx
      do j=3,ny-1
      do k=2,nz
      u5(i,j,k)=beta(j)*u5(i,j-1,k)+gama(i,j,k)
      difxyy(i,j,k)=u5(i,j,k)
      end do
      end do
      end do

      do i=1,nx
      a(i)=0
      b(i)=0
      c(i)=0
      beta(i)=0
      do j=1,ny
      do k=1,nz
      gama(i,j,k)=0.0
      d(i,j,k)=0.0
      end do
      end do
      end do

C Calculate derivative of stress difxzx

      do k=1,nz
      do j=2,ny
      difxzx(2,j,k)=(saxzn(2,j,k)-saxzn(1,j,k))/dx
      difxzx(nx,j,k)=(saxzn(nx,j,k)-saxzn(nx-
     $ 1,j,k))/dx
      end do
      end do

      b(3)=0.0
      a(3)=11.0/12.0
      c(3)=-1.0/24.0

      do k=1,nz
      do j=2,ny
      d(3,j,k)=(saxzn(3,j,k)-saxzn(2,j,k))/dx
     $ -1.0/24.0*difxzx(2,j,k)
      end do
      end do

      do i=4,nx-2
      b(i)=-1.0/24.0
      a(i)=11.0/12.0
      c(i)=-1.0/24.0
      do j=2,ny
      do k=1,nz
      d(i,j,k)=(saxzn(i,j,k)-saxzn(i-1,j,k))/dx
      end do
      end do
      end do

      b(nx-1)=-1.0/24.0
      a(nx-1)=11.0/12.0
      c(nx-1)=0.0

      do k=1,nz
      do j=2,ny
      d(nx-1,j,k)=(saxzn(nx-1,j,k)-saxzn(nx-
     $ 2,j,k))/dx
     $ -1.0/24.0 *difxzx(nx,j,k)
      end do
      end do

      beta(nx)=0.0
      do k=1,nz
      do j=2,ny
      gama(nx,j,k)=0.0
      end do
      end do

      do m=3,nx-1
      i=nx-m+2
      beta(i)=b(i)/(a(i)-c(i)*beta(i+1))
      do j=2,ny
      do k=1,nz
      gama(i,j,k)=(d(i,j,k)+c(i)*gama(i+1,j,k))/(a(i)
     $ -c(i)*beta(i+1))
      end do
      end do
      end do

      do j=1,ny
      do k=2,nz
      u6(2,j,k)=0.0
      end do
      end do
      do i=3,nx-1
      do j=2,ny
      do k=1,nz
      u6(i,j,k)=beta(i)*u6(i-1,j,k)+gama(i,j,k)
      difxzx(i,j,k)=u6(i,j,k)
      end do
      end do
      end do

      do i=1,nx
      a(i)=0
      b(i)=0
      c(i)=0
      beta(i)=0
      do j=1,ny
      do k=1,nz
      gama(i,j,k)=0.0
      d(i,j,k)=0.0
      end do
```

```
      end do
      end do

C Calculate derivative of stress difxzz

      do i=1,nx
      do j=2,ny
      difxzz(i,j,2)=(saxzn(i,j,2)-saxzn(i,j,1))/dz
      difxzz(i,j,nz)=(saxzn(i,j,nz)-saxzn(i,j,nz-1))/dz
      end do
      end do

      b(3)=0.0
      a(3)=11.0/12.0
      c(3)=-1.0/24.0

      do i=1,nx
      do j=2,ny
      d(i,j,3)=(saxzn(i,j,3)-saxzn(i,j,2))/dz
$     -1.0/24.0*difxzz(i,j,2)
      end do
      end do

      do k=4,nz-2
      b(k)=-1.0/24.0
      a(k)=11.0/12.0
      c(k)=-1.0/24.0
      do j=2,ny
      do i=1,nx
      d(i,j,k)=(saxzn(i,j,k)-saxzn(i,j,k-1))/dz
      end do
      end do
      end do

      b(nz-1)=-1.0/24.0
      a(nz-1)=11.0/12.0
      c(nz-1)=0.0

      do i=1,nx
      do j=2,ny
      d(i,j,nz-1)=(saxzn(i,j,nz-1)-saxzn(i,j,nz-2))/dz
$     -1.0/24.0*difxzz(i,j,nz)
      end do
      end do

      beta(nz)=0.0
      do i=1,nx
      do j=2,ny
      gama(i,j,nz)=0.0
      end do
      end do

      do m=3,nz-1
      k=nz-m+2
      beta(k)=b(k)/(a(k)-c(k)*beta(k+1))
      do j=2,ny
      do i=1,nx
      gama(i,j,k)=(d(i,j,k)+c(k)*gama(i,j,k+1))/(a(k)
$     -c(k)*beta(k+1))
      end do
      end do
      end do

      do i=1,nx
      do j=2,ny
      u7(i,j,2)=0.0
      end do
      end do
      do i=1,nx
      do j=2,ny
      do k=3,nz-1
      u7(i,j,k)=beta(k)*u7(i,j,k-1)+gama(i,j,k)
      difxzz(i,j,k)=u7(i,j,k)
      end do
      end do
      end do

      do i=1,nx
      a(i)=0
      b(i)=0
      c(i)=0
      beta(i)=0
      do j=1,ny
      do k=1,nz
      gama(i,j,k)=0.0
      d(i,j,k)=0.0
      end do
      end do
      end do

C Calculate derivative of stress difyzy

      do k=1,nz
      do i=2,nx
      difyzy(i,2,k)=(sayzn(i,2,k)-sayzn(i,1,k))/dy
      difyzy(i,ny,k)=(sayzn(i,ny,k)-sayzn(i,ny-
$     1,k))/dy
      end do
      end do

      b(3)=0.0
      a(3)=11.0/12.0
      c(3)=-1.0/24.0

      do k=1,nz
      do i=2,nx
      d(i,3,k)=(sayzn(i,3,k)-sayzn(i,2,k))/dy
$     -1.0/24.0*difyzy(i,2,k)
      end do
      end do

      do j=4,ny-2
```

```
      b(j)=-1.0/24.0
      a(j)=11.0/12.0
      c(j)=-1.0/24.0
      do i=2,nx
      do k=1,nz
      d(i,j,k)=(sayzn(i,j,k)-sayzn(i,j-1,k))/dy
      end do
      end do
      end do

      b(ny-1)=-1.0/24.0
      a(ny-1)=11.0/12.0
      c(ny-1)=0.0


      do k=1,nz
      do i=2,nx
      d(i,ny-1,k)=(sayzn(i,ny-1,k)-sayzn(i,ny-
     $  2,k))/dy
     $ -1.0/24.0*difyzy(i,ny,k)
      end do
      end do

      beta(ny)=0
      do k=1,nz
      do i=2,nx
      gama(i,ny,k)=0
      end do
      end do

      do m=3,ny-1
      j=ny-m+2
      beta(j)=b(j)/(a(j)-c(j)*beta(j+1))
      do i=2,nx
      do k=1,nz
      gama(i,j,k)=(d(i,j,k)+c(j)*gama(i,j+1,k))/(a(j)
     $  -c(j)*beta(j+1))
      end do
      end do
      end do

      do i=2,nx
      do k=1,nz
      u8(i,2,k)=0.0
      end do
      end do
      do i=2,nx
      do j=3,ny-1
      do k=1,nz
      u8(i,j,k)=beta(j)*u8(i,j-1,k)+gama(i,j,k)
      difyzy(i,j,k)=u8(i,j,k)
      end do
      end do
      end do

      do i=1,nx
      a(i)=0
      b(i)=0
      c(i)=0
      beta(i)=0
      do j=1,ny
      do k=1,nz
      gama(i,j,k)=0.0
      d(i,j,k)=0.0
      end do
      end do
      end do

C Calculate derivative of stress difyzz

      do i=2,nx
      do j=1,ny
      difyzz(i,j,2)=(sayzn(i,j,2)-sayzn(i,j,1))/dz
      difyzz(i,j,nz)=(sayzn(i,j,nz)-sayzn(i,j,nz-1))/dz
      end do
      end do

      b(3)=0.0
      a(3)=11.0/12.0
      c(3)=-1.0/24.0

      do i=2,nx
      do j=1,ny
      d(i,j,3)=(sayzn(i,j,3)-sayzn(i,j,2))/dz
     $ -1.0/24.0*difyzz(i,j,2)
      end do
      end do

      do k=4,nz-2
      b(k)=-1.0/24.0
      a(k)=11.0/12.0
      c(k)=-1.0/24.0
      do j=1,ny
      do i=2,nx
      d(i,j,k)=(sayzn(i,j,k)-sayzn(i,j,k-1))/dz
      end do
      end do
      end do

      b(nz-1)=-1.0/24.0
      a(nz-1)=11.0/12.0
      c(nz-1)=0.0

      do i=2,nx
      do j=1,ny
      d(i,j,nz-1)=(sayzn(i,j,nz-1)-sayzn(i,j,nz-2))/dz
     $  -1.0/24.0 *difyzz(i,j,nz)
      end do
      end do

      beta(nz)=0.0
      do i=2,nx
```

```
      do j=1,ny
      gama(i,j,nz)=0.0
      end do
      end do

      do m=3,nz-1
      k=nz-m+2
      beta(k)=b(k)/(a(k)-c(k)*beta(k+1))
      do j=1,ny
     do i=2,nx
     gama(i,j,k)=(d(i,j,k)+c(k)*gama(i,j,k+1))/(a(k)
$    -c(k)*beta(k+1))
      end do
      end do
      end do

      do i=2,nx
      do j=1,ny
      u9(i,j,2)=0.0
      end do
      end do
      do i=2,nx
      do j=1,ny
      do k=3,nz-1
      u9(i,j,k)=beta(k)*u9(i,j,k-1)+gama(i,j,k)
      difyzz(i,j,k)=u9(i,j,k)
      end do
      end do
      end do

      do i=1,nx
      a(i)=0
      b(i)=0
      c(i)=0
      beta(i)=0
      do j=1,ny
      do k=1,nz
      gama(i,j,k)=0.0
      d(i,j,k)=0.0
      end do
      end do
      end do

C Calculate velocity

     call
velocity(nx,ny,nz,nz2,dx,dy,dz,dt,TEo,TEold,
$  saxo,sayo,sazo,saxyo,saxzo,sayzo,
$  saxn,sayn,sazn,saxyn,saxzn,sayzn,vxo,
$  vyo,vzo,vxn,vyn,vzn,
$  uxo,uyo,uzo,uxn,uyn,uzn,difx,dify,
$  difz,difxyy,difxzz,difxyx,difyzz,difxzx,difyzy)

C Calculate strain

      do k=2,nz
      do j=2,ny
      do i=2,nx
      epxn(i,j,k)=(theta*(vxn(i,j,k)-vxn(i-1,j,k))
$  +(1.0-theta)*(vxo(i,j,k)-vxo(i-1,j,k)))*dt/dx
$  +epxo(i,j,k)

      epyn(i,j,k)=(theta*(vyn(i,j,k)-vyn(i,j-1,k))
$  +(1.0-theta)*(vyo(i,j,k)-vyo(i,j-1,k)))*dt/dy
$  +epyo(i,j,k)

      epzn(i,j,k)=(theta*(vzn(i,j,k)-vzn(i,j,k-1))
$  +(1.0-theta)*(vzo(i,j,k)-vzo(i,j,k-1)))*dt/dz
$  +epzo(i,j,k)
      end do
      end do
      end do

C Calculate Shear strain

      do k=2,nz
      do j=2,ny-1
      do i=2,nx-1
      epxyn(i,j,k)=(theta*(vxn(i,j+1,k)-vxn(i,j,k))
$  +(1.0-theta)*(vxo(i,j+1,k)-vxo(i,j,k)))*dt/dy
$  +(theta*(vyn(i+1,j,k)-vyn(i,j,k))
$  +(1.0-theta)*(vyo(i+1,j,k)-vyo(i,j,k)))*dt/dx
$  +epxyo(i,j,k)
      end do
      end do
      end do

      do k=2,nz-1
      do j=2,ny
      do i=2,nx-1
      epxzn(i,j,k)=(theta*(vxn(i,j,k+1)-vxn(i,j,k))
$  +(1.0-theta)*(vxo(i,j,k+1)-vxo(i,j,k)))*dt/dz
$  +(theta*(vzn(i+1,j,k)-vzn(i,j,k))
$  +(1.0-theta)*(vzo(i+1,j,k)-vzo(i,j,k)))*dt/dx
$  +epxzo(i,j,k)
      end do
      end do
      end do

      do k=2,nz-1
      do j=2,ny-1
      do i=2,nx
      epyzn(i,j,k)=(theta*(vyn(i,j,k+1)-vyn(i,j,k))
$  +(1.0-theta)*(vyo(i,j,k+1)-vyo(i,j,k)))*dt/dz
$  +(theta*(vzn(i,j+1,k)-vzn(i,j,k))
$  +(1.0-theta)*(vzo(i,j+1,k)-vzo(i,j,k)))*dt/dy
$  +epyzo(i,j,k)
      end do
      end do
      end do

C Check convergence
```

```
      do k=1,nz+1
      do j=1,ny+1
      do i=1,nx+1
      det1=epxn(i,j,k)-xsaoo(i,j,k)
      det2=epyn(i,j,k)-ysaoo(i,j,k)
      det3=epzn(i,j,k)-zsaoo(i,j,k)
      det4=epxyn(i,j,k)-ssaooxy(i,j,k)
      det5=epxzn(i,j,k)-ssaooxz(i,j,k)
      det6=epyzn(i,j,k)-ssaooyz(i,j,k)

det=max(abs(det1),abs(det2),abs(det3),abs(det4),
 $  abs(det5),abs(det6))

      if( abs(det).gt.detuvmax) detuvmax=abs(det)
      if( abs(det1).gt.det1max) det1max=abs(det1)
      if( abs(det2).gt.det2max) det2max=abs(det2)
      if( abs(det3).gt.det3max) det3max=abs(det3)
      if( abs(det4).gt.det4max) det4max=abs(det4)
      if( abs(det5).gt.det5max) det5max=abs(det5)
      if( abs(det6).gt.det6max) det6max=abs(det6)
      end do
      end do
      end do

      do k=1,nz+1
      do j=1,ny+1
      do i=1,nx+1
      xsaoo(i,j,k)=epxn(i,j,k)
      ysaoo(i,j,k)=epyn(i,j,k)
      zsaoo(i,j,k)=epzn(i,j,k)
      ssaooxy(i,j,k)=epxyn(i,j,k)
      ssaooxz(i,j,k)=epxzn(i,j,k)
      ssaooyz(i,j,k)=epyzn(i,j,k)
      end do
      end do
      end do

      write(*,*) 'detuvmax=', detuvmax

C  End do with detmax

      enddo

C   End the current time step

C-------------------------------------------------------------
      do k=1,nz+1
      do j=1,ny+1
      do i=1,nx+1
      TEo(i,j,k)=TEold(i,j,k)
      TLo(i,j,k)=TLold(i,j,k)
      epxo(i,j,k)=epxn(i,j,k)
      epyo(i,j,k)=epyn(i,j,k)
      epzo(i,j,k)=epzn(i,j,k)
      epxyo(i,j,k)=epxyn(i,j,k)
      epxzo(i,j,k)=epxzn(i,j,k)
      epyzo(i,j,k)=epyzn(i,j,k)
      saxo(i,j,k)=saxn(i,j,k)
      sayo(i,j,k)=sayn(i,j,k)
      sazo(i,j,k)=sazn(i,j,k)
      saxyo(i,j,k)=saxyn(i,j,k)
      saxzo(i,j,k)=saxzn(i,j,k)
      sayzo(i,j,k)=sayzn(i,j,k)
      vxo(i,j,k)=vxn(i,j,k)
      vyo(i,j,k)=vyn(i,j,k)
      vzo(i,j,k)=vzn(i,j,k)
      uxo(i,j,k)=uxn(i,j,k)
      uyo(i,j,k)=uyn(i,j,k)
      uzo(i,j,k)=uzn(i,j,k)
      end do
      end do
      end do


      if (big.lt.(TEold(11,11,1)-300.0)) then
        big=TEold(11,11,1)-300.0
      end if

      TEm(n)=TEold(11,11,1)
      TLm(n)=TLold(11,11,1)
      u1m(n)=uxn(11,11,2)
      u2m(n)=uyn(11,11,2)
      u3m(n)=uzn(11,11,1)
      v1m(n)=vxn(11,11,2)
      v2m(n)=vyn(11,11,2)
      v3m(n)=vzn(11,11,1)

      icounter=icounter+1
      write(*,*) icounter

C Output
      write(8,1020) t(n),TEm(n),TLm(n)
      write(7,1020) t(n),u1m(n),u2m(n),u3m(n)

C Output intermediate result

      if (n.eq.50) then
C The result at time t=0.25ps
C Electron temp

      open(unit=10,file='ctexz025ps.txt')
      do k=1,nz+1
      write(10,1010) (TEold(i,11,k),i=1,nx+1)
      enddo
      open(unit=11,file='te025ps.txt')
      do k=1,nz+1
      write(11,1020) TEold(11,11,k)
      enddo
C Lattice temp
      open(unit=12,file='ctlxz025ps.txt')
      do k=1,nz+1
```

```
      write(12,1010) (TLold(i,11,k),i=1,nx+1)
      enddo
      open(unit=13,file='tl025ps.txt')
      do k=1,nz+1
      write(13,1020) TLold(11,11,k)
      enddo
      end if

      if (n.eq.100) then
C The result at time t=0.5ps
C Electron temp
      open(unit=14,file='ctexz05ps.txt')
      do k=1,nz+1
      write(14,1010) (TEold(i,11,k),i=1,nx+1)
      enddo
      open(unit=15,file='te05ps.txt')
      do k=1,nz+1
      write(15,1020) TEold(11,11,k)
      enddo
C Lattice temp
      open(unit=16,file='ctlxz05ps.txt')
      do k=1,nz+1
      write(16,1010) (TLold(i,11,k),i=1,nx+1)
      enddo
      open(unit=17,file='tl05ps.txt')
      do k=1,nz+1
      write(17,1020) TLold(11,11,k)
      enddo
      end if

 1010    format(401e15.6)
 1020    format(e15.6,3e15.6)
      end

C End main program

C Subroutines

C Calculate temperature

      subroutine temp(nx,ny,nz,nz2,
     $ dx,dy,dz,x,y,z,t,dt,TLo,TLold,TEo,TEold,
     $ epxn,epyn,epzn,epxo,epyo,epzo)

      implicit double precision (a-h,l,o-z)
      dimension x(51),y(51),z(221)
      dimension TEo(41,41,101),TEold(41,41,101),
     $  TLo(41,41,101),TLold(41,41,101),
     $ TEnew(41,41,101),TLnew(41,41,101),
     $  epxn(41,41,101),epyn(41,41,101),epzn(41,41,1
     01),
     $  epxo(41,41,101),epyo(41,41,101),epzo(41,41,1
     01),
     $ dTE(41,41,101),dTL(41,41,101)

      integer iteration,flagE,flagL

C data

C Lame constant
      clemta1=199.0d+9
      clemta2=83.3d+9
C Shear modulus
      cmiu1=27.0d+9
      cmiu2=115.0d+9
C Thermal expansion coefficient
      alpha1=14.2d-6
      alpha2=4.9d-6
C Electron heat capacity
      ce01=2.1d+4
      ce02=5.8d+4
C Lattic heat capacity
      cl1=2.5d+6
      cl2=3.3d+6
C Electron - lattic coupling factor
      g1=2.6d+16
      g2=42.0d+16
C Electron thermal conducitivity
      cke01=315.0
      cke02=94.0
C Laser fluence
      flu=1000.0
C Laser pulse duration
      tp=0.1d-12
C Optical penetration depth
      delta=15.3d-9
C Surface reflectivity
      sur=0.93
C Spatial profile parameters
      zs=1.0d-6

      iteration=0

      rx=dt/(4.0*dx*dx)
      ry=dt/(4.0*dy*dy)
      rz=dt/(4.0*dz*dz)
      deterror=1.0d-3

C-------------------------Iteration starts----------------
C flagE and flagL indicate whether TE and TL are
precise enough
C keep on iterating as long as flagE or flagL
equals to 1

2     do j=2,ny
      do i=2,nx
      clemta=clemta1
      cmiu=cmiu1
      alpha=alpha1
      ce0=ce01
      cl=cl1
      g=g1
```

```
      cke0=cke01
      d0=g*dt/(2.0*cl)
      ee=(3.0*clemta+2.0*cmiu)*alpha*300.0/cl
      do k=2,nz2-1

C Heat source
      aa=-z(k)/delta-((x(i)-10.0*dx)*(x(i)-10.0*dx)
     $   +(y(j)-10.0*dy)*(y(j)-10.0*dy))/(zs*zs)
     $   -2.77*(t-2.0*tp)*(t-2.0*tp)/(tp*tp)

      q=0.94*flu*(1.0-sur)*exp(aa)/(tp*delta)

      a0=ce0*(TEo(i,j,k)+TEold(i,j,k))/(2.0*300.0)
      b1=cke0*(TEold(i+1,j,k)/TLold(i+1,j,k)
     $  +TEold(i,j,k)/TLold(i,j,k))*rx
      b2=cke0*(TEold(i,j,k)/TLold(i,j,k)
     $  +TEold(i-1,j,k)/TLold(i-1,j,k))*rx
      b3=cke0*(TEold(i,j+1,k)/TLold(i,j+1,k)
     $  +TEold(i,j,k)/TLold(i,j,k))*ry
      b4=cke0*(TEold(i,j,k)/TLold(i,j,k)
     $  +TEold(i,j-1,k)/TLold(i,j-1,k))*ry
      b5=cke0*(TEold(i,j,k+1)/TLold(i,j,k+1)
     $  +TEold(i,j,k)/TLold(i,j,k))*rz
      b6=cke0*(TEold(i,j,k)/TLold(i,j,k)
     $  +TEold(i,j,k-1)/TLold(i,j,k-1))*rz

      c1=cke0*(TEo(i+1,j,k)/TLo(i+1,j,k)
     $  +TEo(i,j,k)/TLo(i,j,k))*rx
      c2=cke0*(TEo(i,j,k)/TLo(i,j,k)
     $  +TEo(i-1,j,k)/TLo(i-1,j,k))*rx
      c3=cke0*(TEo(i,j+1,k)/TLo(i,j+1,k)
     $   +TEo(i,j,k)/TLo(i,j,k))*ry
      c4=cke0*(TEo(i,j,k)/TLo(i,j,k)
     $   +TEo(i,j-1,k)/TLo(i,j-1,k))*ry
      c5=cke0*(TEo(i,j,k+1)/TLo(i,j,k+1)
     $   +TEo(i,j,k)/TLo(i,j,k))*rz
      c6=cke0*(TEo(i,j,k)/TLo(i,j,k)
     $   +TEo(i,j,k-1)/TLo(i,j,k-1))*rz


dd=a0+b1+b2+b3+b4+b5+b6+g*dt/(2.0*(1.0+d0))
      TEnew(i,j,k)=(b1*TEold(i+1,j,k)+b2*TEold(i-
1,j,k)
     $   +b3*TEold(i,j+1,k)+b4*TEold(i,j-1,k)
     $   +b5*TEold(i,j,k+1)+b6*TEold(i,j,k-1)
     $   -g*dt*(TEo(i,j,k)-TLo(i,j,k))/(2.0*(1.0+d0))
     $   +g*dt*TLo(i,j,k)/(2.0*(1.0+d0))+a0*TEo(i,j,k)
     $   -g*dt*ee*((epxn(i,j,k)+epyn(i,j,k)+epzn(i,j,k))
     $   -(epxo(i,j,k)+epyo(i,j,k)+epzo(i,j,k)))
     $   /(2.0*(1.0+d0))
     $   +c1*(TEo(i+1,j,k)-TEo(i,j,k))
     $    -c2*(TEo(i,j,k)-TEo(i-1,j,k))
     $    +c3*(TEo(i,j+1,k)-TEo(i,j,k))
     $    -c4*(TEo(i,j,k)-TEo(i,j-1,k))
     $   +c5*(TEo(i,j,k+1)-TEo(i,j,k))
     $    -c6*(TEo(i,j,k)-TEo(i,j,k-1))
     $   +q*dt)/dd

      TLnew(i,j,k)=d0*TEnew(i,j,k)/(1.0+d0)
     $   +d0*(TEo(i,j,k)-TLo(i,j,k))/(1.0+d0)
     $       +TLo(i,j,k)/(1.0+d0)
     $       -ee/(1.0+d0)
     $       *((epxn(i,j,k)+epyn(i,j,k)+epzn(i,j,k))
     $       -(epxo(i,j,k)
     $       +epyo(i,j,k)+epzo(i,j,k)))
      end do
      clemta=clemta2
      cmiu=cmiu2
      alpha=alpha2
      ce0=ce02
      cl=cl2
      g=g2
      cke0=cke02
      d0=g*dt/(2.0*cl)
      ee=(3.0*clemta+2.0*cmiu)*alpha*300.0/cl
      do k=nz2+1,nz
      aa=-z(k)/delta-((x(i)-10.0*dx)*(x(i)-10.0*dx)
     $   +(y(j)-10.0*dy)*(y(j)-10.0*dy))/(zs*zs)
     $   -2.77*(t-2.0*tp)*(t-2.0*tp)/(tp*tp)

      q=0.94*flu*(1.0-sur)*exp(aa)/(tp*delta)

      a0=ce0*(TEo(i,j,k)+TEold(i,j,k))/(2.0*300.0)
      b1=cke0*(TEold(i+1,j,k)/TLold(i+1,j,k)
     $  +TEold(i,j,k)/TLold(i,j,k))*rx
      b2=cke0*(TEold(i,j,k)/TLold(i,j,k)
     $  +TEold(i-1,j,k)/TLold(i-1,j,k))*rx
      b3=cke0*(TEold(i,j+1,k)/TLold(i,j+1,k)
     $  +TEold(i,j,k)/TLold(i,j,k))*ry
      b4=cke0*(TEold(i,j,k)/TLold(i,j,k)
     $  +TEold(i,j-1,k)/TLold(i,j-1,k))*ry
      b5=cke0*(TEold(i,j,k+1)/TLold(i,j,k+1)
     $  +TEold(i,j,k)/TLold(i,j,k))*rz
      b6=cke0*(TEold(i,j,k)/TLold(i,j,k)
     $  +TEold(i,j,k-1)/TLold(i,j,k-1))*rz

      c1=cke0*(TEo(i+1,j,k)/TLo(i+1,j,k)
     $  +TEo(i,j,k)/TLo(i,j,k))*rx
      c2=cke0*(TEo(i,j,k)/TLo(i,j,k)
     $  +TEo(i-1,j,k)/TLo(i-1,j,k))*rx
      c3=cke0*(TEo(i,j+1,k)/TLo(i,j+1,k)
     $  +TEo(i,j,k)/TLo(i,j,k))*ry
      c4=cke0*(TEo(i,j,k)/TLo(i,j,k)
     $   +TEo(i,j-1,k)/TLo(i,j-1,k))*ry
      c5=cke0*(TEo(i,j,k+1)/TLo(i,j,k+1)
     $   +TEo(i,j,k)/TLo(i,j,k))*rz
      c6=cke0*(TEo(i,j,k)/TLo(i,j,k)
     $  +TEo(i,j,k-1)/TLo(i,j,k-1))*rz


dd=a0+b1+b2+b3+b4+b5+b6+g*dt/(2.0*(1.0+d0))
      TEnew(i,j,k)=(b1*TEold(i+1,j,k)+b2*TEold(i-
```

```
$   1,j,k)
$   +b3*TEold(i,j+1,k)+b4*TEold(i,j-1,k)
$   +b5*TEold(i,j,k+1)+b6*TEold(i,j,k-1)
$   -g*dt*(TEo(i,j,k)-TLo(i,j,k))/(2.0*(1.0+d0))
$   +g*dt*TLo(i,j,k)/(2.0*(1.0+d0))+a0*TEo(i,j,k)
$   -g*dt*ee*((epxn(i,j,k)+epyn(i,j,k)+epzn(i,j,k))
$   -(epxo(i,j,k)+epyo(i,j,k)+epzo(i,j,k)))
$   /(2.0*(1.0+d0))
$   +c1*(TEo(i+1,j,k)-TEo(i,j,k))
$   -c2*(TEo(i,j,k)-TEo(i-1,j,k))
$   +c3*(TEo(i,j+1,k)-TEo(i,j,k))
$   -c4*(TEo(i,j,k)-TEo(i,j-1,k))
$   +c5*(TEo(i,j,k+1)-TEo(i,j,k))
$    -c6*(TEo(i,j,k)-TEo(i,j,k-1))
$   +q*dt)/dd

     TLnew(i,j,k)=d0*TEnew(i,j,k)/(1.0+d0)
$    +d0*(TEo(i,j,k)-TLo(i,j,k))/(1.0+d0)
$    +TLo(i,j,k)/(1.0+d0)
$     -ee/(1.0+d0)
$     *((epxn(i,j,k)+epyn(i,j,k)+epzn(i,j,k))
$     -(epxo(i,j,k)
$     +epyo(i,j,k)+epzo(i,j,k)))
     end do
     k=nz2
      TEnew(i,j,k)=(cke02*TEnew(i,j,k+1)
$   +cke01*TEnew(i,j,k-1))
$    /(cke01+cke02)
      TLnew(i,j,k)=(cke02*TLnew(i,j,k+1)
$   +cke01*TLnew(i,j,k-1))
$    /(cke01+cke02)
     end do
     end do
C  Boundary Conditions
     do k=2,nz
     do j=2,ny
     TEnew(1,j,k)=TEnew(2,j,k)
     TEnew(nx+1,j,k)=TEnew(nx,j,k)
     TLnew(1,j,k)=TLnew(2,j,k)
     TLnew(nx+1,j,k)=TLnew(nx,j,k)
     end do
     end do
C
     do k=2,nz
     do i=2,nx
     TEnew(i,1,k)=TEnew(i,2,k)
     TEnew(i,ny+1,k)=TEnew(i,ny,k)
     TLnew(i,1,k)=TLnew(i,2,k)
     TLnew(i,ny+1,k)=TLnew(i,ny,k)
     end do
     end do
C
     do j=2,ny
     do i=2,nx
     TEnew(i,j,1)=TEnew(i,j,2)
     TEnew(i,j,nz+1)=TEnew(i,j,nz)
     TLnew(i,j,1)=TLnew(i,j,2)
     TLnew(i,j,nz+1)=TLnew(i,j,nz)
     end do
     end do

C  Test for convergence
     detmax=0.0
     do i=2,nx
     do j=2,ny
     do k=2,nz
     det1=abs(TEnew(i,j,k)-TEold(i,j,k))
     if (det1.gt.detmax) detmax=det1
     det2=abs(TLnew(i,j,k)-TLold(i,j,k))
     if (det2.gt.detmax) detmax=det2
     enddo
     enddo
     enddo

     if (detmax.le.deterror) goto 3
     do i=1,nx+1
     do j=1,ny+1
     do k=1,nz+1
     TEold(i,j,k)=TEnew(i,j,k)
     TLold(i,j,k)=TLnew(i,j,k)
     enddo
     enddo
     enddo
     iteration=iteration+1
     goto 2

C Update all the TEold, TLold with TEnew and
TLnew

3    do j=1,ny+1
      do i=1,nx+1
      do k=1,nz+1
     TEold(i,j,k)=TEnew(i,j,k)
     TLold(i,j,k)=TLnew(i,j,k)
     enddo
     enddo
     enddo

     write (*,*) "iteration=", iteration
C-----------------------------Iterations Done------------

     END
C  End of subroutine temp()
C  Calculate velocity

 Subroutine
velocity(nx,ny,nz,nz2,dx,dy,dz,dt,TEo,TEold,
$  saxo,sayo,sazo,saxyo,saxzo,sayzo,
$  saxn,sayn,sazn,saxyn,saxzn,sayzn,
$  vxo,vyo,vzo,vxn,vyn,vzn,uxo,uyo,uzo,
$  uxn,uyn,uzn,difx,dify,
$  difz,difxyy,difxzz,difxyx,difyzz,difxzx,difyzy)
```

```
      implicit double precision (a-h,l,o-z)

      dimension TEo(41,41,101),TEold(41,41,101),
     $  saxo(41,41,101),sayo(41,41,101),sazo(41,41,10
     1),
     $  saxyo(41,41,101),saxzo(41,41,101),sayzo(41,41
     ,101),
     $  saxn(41,41,101),sayn(41,41,101),sazn(41,41,10
     1),
     $  saxyn(41,41,101),saxzn(41,41,101),sayzn(41,41
     ,101),
     $ vxo(41,41,101),vyo(41,41,101),vzo(41,41,101),
     $ vxn(41,41,101),vyn(41,41,101),vzn(41,41,101),
     $ uxo(41,41,101),uyo(41,41,101),uzo(41,41,101),
     $ uxn(41,41,101),uyn(41,41,101),uzn(41,41,101),
     $ difx(41,41,101),dify(41,41,101),difz(41,41,101),
     $ difxyy(41,41,101),
     $ difxzz(41,41,101),difxyx(41,41,101),
     $ difyzz(41,41,101),
     $ difxzx(41,41,101),difyzy(41,41,101)

C    Density
      lou1=1.93d+4
      lou2=7190.0
C    Electron - blast coefficient
      tri1=70.0
      tri2=193.3

      theta=0.5

      do j=2,ny
      do i=1,nx
      lou=lou1
      tri=tri1
      do k=2,nz2-1

      vxn(i,j,k)=(difx(i,j,k)
     $  +difxyy(i,j,k)+difxzz(i,j,k)
     $  +tri*theta*(TEold(i+1,j,k)
     $  *TEold(i+1,j,k)-TEold(i,j,k)*TEold(i,j,k))/dx
     $   +tri*(1.0-theta)*(TEo(i+1,j,k)
     $   *TEo(i+1,j,k)-TEo(i,j,k)*TEo(i,j,k))
     $   /dx)*dt/lou+vxo(i,j,k)

      uxn(i,j,k)=(theta*vxn(i,j,k)
     $  +(1.0-theta)*vxo(i,j,k))*dt+uxo(i,j,k)
      end do
      lou=lou2
      tri=tri2
      do k=nz2+1,nz
      vxn(i,j,k)=(difx(i,j,k)
     $  +difxyy(i,j,k)+difxzz(i,j,k)
     $  +tri*theta*(TEold(i+1,j,k)
     $  *TEold(i+1,j,k)-TEold(i,j,k)*TEold(i,j,k))/dx
     $  +tri*(1.0-theta)*(TEo(i+1,j,k)
     $  *TEo(i+1,j,k)-TEo(i,j,k)*TEo(i,j,k))
     $  /dx)*dt/lou+vxo(i,j,k)

      uxn(i,j,k)=(theta*vxn(i,j,k)
     $  +(1.0-theta)*vxo(i,j,k))*dt+uxo(i,j,k)
      end do
      k=nz2
      vxn(i,j,k)=(vxn(i,j,k+1)+vxn(i,j,k-1))/2
      uxn(i,j,k)=(uxn(i,j,k+1)+uxn(i,j,k-1))/2
      end do
      end do

      do i=2,nx
      do j=1,ny

      lou=lou1
      tri=tri1
      do k=2,nz2-1
      vyn(i,j,k)=(difxyx(i,j,k)
     $  +dify(i,j,k)+difyzz(i,j,k)
     $  +tri*theta*(TEold(i,j+1,k)
     $  *TEold(i,j+1,k)-TEold(i,j,k)*TEold(i,j,k))/(dy)
     $  +tri*(1.0-theta)*(TEo(i,j+1,k)
     $   *TEo(i,j+1,k)-TEo(i,j,k)
     $  *TEo(i,j,k)) /(dy))*dt/lou+vyo(i,j,k)

      uyn(i,j,k)=(theta*vyn(i,j,k)
     $   +(1.0-theta)*vyo(i,j,k))*dt+uyo(i,j,k)
      end do
      lou=lou2
      tri=tri2
      do k=nz2+1,nz
      vyn(i,j,k)=(difxyx(i,j,k)
     $  +dify(i,j,k)+difyzz(i,j,k)
     $  +tri*theta*(TEold(i,j+1,k)
     $  *TEold(i,j+1,k)-TEold(i,j,k)*TEold(i,j,k))/(dy)
     $   +tri*(1.0-theta)*(TEo(i,j+1,k)
     $   *TEo(i,j+1,k)-TEo(i,j,k)*TEo(i,j,k))
     $   /(dy))*dt/lou+vyo(i,j,k)

      uyn(i,j,k)=(theta*vyn(i,j,k)
     $   +(1.0-theta)*vyo(i,j,k))*dt+uyo(i,j,k)
      end do
      k=nz2
      vyn(i,j,k)=(vyn(i,j,k+1)+vyn(i,j,k-1))/2
      uyn(i,j,k)=(uyn(i,j,k+1)+uyn(i,j,k-1))/2
      end do
      end do

      do i=2,nx
      do j=2,ny
      lou=lou1
      tri=tri1
      do k=1,nz2-1

      vzn(i,j,k)=(difxzx(i,j,k)
```

```
$  +difyzy(i,j,k)+difz(i,j,k)
$  +tri*theta*(TEold(i,j,k+1)
$  *TEold(i,j,k+1)-TEold(i,j,k)
$  *TEold(i,j,k))/(dz)
$   +tri*(1.0-theta)*(TEo(i,j,k+1)
$   *TEo(i,j,k+1)-TEo(i,j,k)*TEo(i,j,k))
$   /(dz))*dt/lou+vzo(i,j,k)

      uzn(i,j,k)=(theta*vzn(i,j,k)
$     +(1.0-theta)*vzo(i,j,k))*dt+uzo(i,j,k)

      end do
      lou=lou2
      tri=tri2
      do k=nz2,nz
      vzn(i,j,k)=(difxzx(i,j,k)
$     +difyzy(i,j,k)+difz(i,j,k)
$     +tri*theta*(TEold(i,j,k+1)
$     *TEold(i,j,k+1)-TEold(i,j,k)*TEold(i,j,k))/(dz)
$     +tri*(1.0-theta)*(TEo(i,j,k+1)
$     *TEo(i,j,k+1)-TEo(i,j,k)
$     *TEo(i,j,k))/(dz))*dt/lou+vzo(i,j,k)

      uzn(i,j,k)=(theta*vzn(i,j,k)
$     +(1.0-theta)*vzo(i,j,k))*dt+uzo(i,j,k)

      end do
      end do
      end do

      return
      end
```

REFERENCES

Al-Nimr M.A. and Arpaci V.S., "Picosecond Thermal Pulses in Thin Metal Films," *Journal of Applied Physics*, Vol. 85, (1999), pp. 2517-2521.

Al-Nimr M.A. and Arpaci V.S., "The Thermal Behavior of Thin Metal Films in the Hyperbolic Two-Step Model," *International Journal of Heat and Mass Transfer*, Vol. 43, (2000), pp. 2021-2028.

Al-Nimr M.A., Header M., and Naji M., "Use of the Microscopic Parabolic Heat Conduction Model in Place of Macroscopic Model Validation Criterion under Harmonic Boundary Heating," *International Journal of Heat and Mass Transfer*, Vol. 46, (2003), pp. 333-339.

Al-Odat M., Al-Nimr M.A., and Hamdan M., "Thermal Stability of Superconductors under the Effect of a Two-Dimensional Hyperbolic Heat Conduction Model," *International Journal of Numerical Methods for Heat & Fluid Flow*, Vol. 12, (2002), pp. 173-177.

Anisimov S.I., Kapeliovich B.L., and Perel' man T.L., "Electron Emission from Metal Surface Exposed to Ultra-Short Laser Pulse," *Soviet Physics Journal of Experimental and Theoretical Physics*, Vol. 39, (1974), pp. 375-377.

Antaki P.J., "Solution for Non-Fourier Dual Phase Lag Heat Conduction in a Semi-Infinite Slab with Surface Heat Flux," *International Journal of Heat and Mass Transfer*, Vol. 41, (1998), pp. 2253-2258.

Antaki P.J., "Importance of Nonequilibrium Thermal Conductivity during Short-Pulse Laser-Induced Desorption from Metals," *International Journal of Heat and Mass Transfer*, Vol. 45, (2002), pp. 4063-4067.

Atkinson K.E. and Han W., *Theoretical Numerical Analysis: A Function Analysis Framework*, second edition, Texts in Applied Mathematics 39, Springer, (2005).

Brorson S.D., Fujimoto J.G., and Ippen E.P., "Femtosecond Electron Heat Transfer Dynamics in Thin Gold Film," *Physical Review Letters*, Vol. 59, (1987), pp. 1962-1965.

Cattaneo C., "A Form of Heat Conduction Equation which Eliminates the Paradox of Instantaneous Propagation," *Compte Rendus*, Vol. 247, (1958), pp. 431-433.

Chen J.K. and Beraun J.E., "Numerical Study of Ultrashort Laser Pulse Interactions with Metal Films," *Numerical Heat Transfer, Part A,* Vol. 40, (2001), pp. 1-20.

Chen J.K., Beraun J.E., and Tham C.L., "Comparison of One-Dimensional and Two-Dimensional Axisymmetric Approaches to the Thermomechanical Response Caused by Ultrashort Laser Heating," *Journal of Optics A: Pure Applied Optics*, Vol. 4, (2002), pp. 650-661.

Chen J.K., Latham W.P., and Beraun J.E., "Axisymmetric Modeling of Femtosecond-Pulse Laser Heating on Metal Films," *Numerical Heat Transfer, Part B,* Vol. 42, (2002), pp. 1-17.

Chen J.K., Beraun J.E., and Tham C.L., "Investigation of Thermal Response Caused by Pulse Laser Heating," *Numerical Heat Transfer, Part A,* Vol. 44, (2003), pp. 705-722.

Chester M., "Second Sound in Solids," *Physical Review*, Vol. 131, (1963), pp. 2013-2015.

Chiffell R.J., *On the Wave Behavior and Rate Effect of Thermal and Thermomechanical Waves*, M.S. Thesis, University of New Mexico, Albuque, NM, (1994).

Dai W. and Nassar R., "A Finite Difference Method for Solving the Heat Transport Equation at Microscale," *Numerical Methods for Partial Differential Equations*, Vol. 15, (1999), pp. 697-708.

Dai W. and Nassar R., "A Compact Finite Difference Scheme for Solving a Three-Dimensional Heat Transport Equation in a Thin Film," *Numerical Methods for Partial Differential Equations*, Vol. 16, (2000), pp. 441-458.

Dai W. and Nassar R., "A Finite Difference Scheme for Solving a Three-Dimensional Heat Transport Equation in a Thin Film with Micro-Scale Thickness," *International Journal for Numerical Methods in Engineering*, Vol. 50, (2001), pp. 1665-1680.

Dai W. and Nassar R., "A Compact Finite Difference Scheme for Solving a One-Dimensional Heat Transport Equation at the Microscale," *Numerical Heat Transfer, Part A,* Vol. 130, (2001), pp. 431-441.

Dai W. and Zhu T., "A Nonlinear Finite Difference Scheme for Solving the Parabolic Two-Step Model with Temperature Dependent Thermal Properties in a Double-Layered Thin Film Exposed by Ultrashort-Pulsed Lasers," *Computational and Information Science*, Vol. 3314, (2004), pp. 304-309.

Dai W., "Nonstandard Finite Difference Schemes for Solving Nonlinear Micro Heat Transport Equations in Double-Layered Metal Thin Films Exposed to Ultrashort Pulsed Lasers," *Advances in the Applications of Nonstandard Finite Difference Schemes,* Chapter 5, (2005), pp. 191-248.

Du X., Dai W., and Wang P., "A Finite Difference Method for Studying Thermal Deformation in a 3D Micro-Sphere Exposed to Ultrashort Pulsed Lasers," *Numerical Heat Transfer, Part A,* Vol. 53, (2008), pp. 457-484.

Elliot D.J. and Piwczyk B.P., "Single and Multiple Pulse Ablations of Polymetic and High Density Materials with Excimer Laser Radiation at 193 nm and 248 nm," *Mater. Res. Soc. Symp. Proc.* Vol. 129, (1989), pp. 627-636.

Evans L.C., *Patial Differential Equations*, American Mathematical Society, Providence, Phode Island, (1998).

Grigoropoulos C.P., "Heat Transfer in Laser Processing of Thin Films," in: C.L. Tien (Ed), *Annual Review of Heat Transfer*, Vol. 5, Hemisphere, New York, (1994), pp. 77-130.

Jopseph D.D. and Preziosi L., "Heat Waves," *Reviews of Modern Physics*, Vol. 61, (1989), pp. 41-73.

Jopseph D.D. and Preziosi L., "Addendum to the Paper on Heat Waves," *Reviews of Modern Physics*, Vol. 62, (1990), pp. 375-391.

Joshi A.A. and Majumdar A., "Transient Ballistic and Diffusive Phonon Heat Transport in Thin Films," *Journal of Applied Physics*, Vol. 74, (1993), pp. 31-39.

Kaye G.W.C., *Tables of Physical and Chemical Constants and some Mathematical Functions*, Longman, London, UK, (1973), pp. 31.

Kaba I., *A Numerical Method to Solve the Two-Step Parabolic Heat Transport Equations in a Microsphere Subjected to an Ultrafast Laser Pulse,* Ph. D. Dissertation, Louisiana Tech University, LA, (2004).

Kaba I. and Dai W., "A Stable Three-Level Finite Difference Scheme for Solving the Parabolic Two-Step Model in a 3-D Micro-Sphere Heated by Ultrashort-Pulsed Lasers," *Journal of Computational and Applied Mathematics*, Vol. 181, (2005), pp. 125-147.

Kaganov M.L., Lifshitz I. M., and Tanatarov M.V., "Relaxation between Electrons and Crystalline Lattices," *Soviet Physics Journal of Experimental and Theoretical Physics.* Vol. 4, (1957), pp. 173-178.

Knapp J.A., Borgesen P., and Zuhr R.A., "Beam-solid Interactions: Physical Phenomena," *Mater. Res. Soc. Symp. Proc*, Vol. 157 (1990).

Lin C.K., Hwang C.C., and Chang Y.P., "The Unsteady Solution of a Unified Heat Conduction Equation," *International Journal of Heat and Mass Transfer*, Vol. 40, (1997), pp. 1716-1719.

Mills A.F., *Heat and Mass Transfer,* CRC Press, Boca Raton, FL, (1994).

Naji M., Al-Nimr M.A., and Hader M., "The Validity of Using the Microscopic Hyperbolic Heat Conduction Model under as Harmonic Fluctuating Boundary Heating Source," *Inernational Journal of Thermophysics*, Vol. 24, (2003), pp. 545-557.

J. Narayan, V.P. Gosbole, and G.W. White, "Laser Method for Synthesis and Processing of Continuous Diamond Films on Nondiamond Substrates," *Science*, Vol. 52, (1991), pp. 416-418.

Niu T. and Dai W., "A Hyperbolic Two-Step Model Based Finite Difference Scheme for Studying Thermal Deformation in a Double-Layered Thin Film Exposed to Ultrashort Pulsed Lasers," accepted by *International Journal of Thermal Sciences*.

Niu T. and Dai W., "A Hyperbolic Two-Step Model Based Finite Difference Scheme for Studying Thermal Deformation a 3D Thin Film Exposed to Ultrashort Pulsed Lasers," *Numerical Heat Transfer, Part A*, Vol. 53, (2008), pp. 1294-1320.

Opsal J., "The Application of Thermal Wave Technology to Thickness and Grain Size of Aluminum Films, in Metallization," *Performance and Reliability Issues for VLSI and ULSI, SPIE*, Vol. 1596, (1991), pp. 120-131.

Ozisik M.N. and Tzou D.Y., "On the Wave Theory in Heat Conduction," *Journal of Heat Transfer*, Vol. 116, (1994), pp. 526-535.

Patanka S.V., *Numerical Heat Transfer and Fluid Flow*, *Chapter 5*, McGraw-Hill, New York, (1980).

Qiu T.Q. and Tien C.L., "Short-Pulse Laser Heating on Metals," *International Journal of Heat and Mass Transfer*, Vol. 35, (1992), pp. 719-726.

Qiu T.Q. and Tien C.L., "Heat Transfer Mechanisms during Short-Pulse Laser Heating of Metals," *ASME Journal of Heat Transfer*, Vol. 115, (1993), pp. 835-841.

Qiu T.Q., Juhasz T., Suarez C., Bron W.E., and Tien C.L., "Femtosecond Laser Heating of Multi-Layer Metals II – Experiments," *International Journal of Heat and Mass Transfer*, Vol. 37, (1994), pp. 2789-2808.

Tang D.W. and Araki N., "Wavy, Wavelike, Diffusive Thermal Responses of Finite Rigid Slabs to High-Speed Heating of Laser-Pulses," *International Journal of Heat and Mass Transfer*, Vol. 42, (1999), pp. 855-860.

Touloukian Y.S., Powell R.W., Ho C.Y., and Klemens P.G., "Thermal Conductivity," *Thermophysical Properties of Matter*, Vol. 1, IFI/Plenum, New York, (1970), pp. 193.

Touloukian Y.S. and Buyco E.H., "Specific Heat," *Thermophysical Properties of Matter*, Vol. 4, IFI/Plenum, New York, (1970), pp. 204.

Tzou D.Y., "On the Thermal Shock Wave Induced by a Moving and Source," *ASME Journal of Heat Transfer*, Vol. 111, (1989), pp. 232-238.

Tzou D.Y., "Shock Wave Formation around a Moving Heat Source in a Solid With Finite Speed of Heat Propagation," *International Journal of Heat and Mass Transfer*, Vol. 32, (1989), pp. 1979-1987.

Tzou D.Y., "Thermal Shorck Waves Induced by a Moving Crack," *ASME Journal of Heat Transfer*, Vol. 112, (1990), pp. 21-27.

Tzou D.Y., "Thermal Shock Waves Induced by a Moving Crack – A Heat Flux Formulation," *International Journal of Heat and Mass Transfer*, Vol. 33, (1990), pp. 877-885.

Tzou D.Y., "Thermal Shock Phenomena under High-Rate Response in Solids," *Annual Review of Heat Transfer*, Hemisphere Publishing Inc., Washington, D. C., (1992), pp. 111-185.

Tzou D.Y., "A Unified Field Approach for Heat Conduction from Macro-to Micro-Scales," *Journal of Heat Transfer*, Vol. 117, (1995), pp. 1837-1840.

Tzou D.Y., "The Generalized Lagging Response in Small-Scale and High-Rate Heating," *International Journal of Heat and Mass Transfer*, Vol. 38, (1995), pp. 3231-3240.

Tzou D.Y., "Experimental Support for the Lagging Behavior in Heat Propagation," *Journal of Thermophysics and Heat Transfer*, Vol. 6, (1995), pp. 686-693.

Tzou D.Y. and Zhang Y.S., "An Analytic Study on the Fast-Transient Process in Small Scales," *International Journal of Engineering*, Vol. 33, (1995), pp. 1449-1463.

Tzou D.Y., *Macro-To Micro Heat Transfer- The Lagging Behavior*, Taylor & Francis, Washington, DC, (1996).

Tzou D.Y. and Chiu K.S., "Temperature-Dependent Thermal Lagging in Ultrafast Lawer Heating," *International Journal of Heat and Mass Transfer*, Vol. 44, (2001), pp. 1725-1734.

Tzou D.Y., Chen J.K., and Beraun J.E., "Hot-Electron Blast Induced by Ultrashort-Pulsed Lasers in Layered Media," *International Journal of Heat and Mass Transfer*, Vol. 45, (2002), pp. 3369-3382.

Vernotte P., "Les Paradoxes De La Theorie Continue De Lequation De La Chaleur," *Compte Rendus*, Vol. 246, (1958), pp. 3154-3155.

Vernotte P., "Some Possible Complication in the Phenomena of Thermal Conduction", *Compte Rendus*, Vol. 252, (1961), pp. 2190-2191.

Wang L. and Zhou X., "Dual-Phase-Lagging Heat Conduction," *Shandong University Pres*s, Jinan, 2000.

Wang L., Xu M., and Zhou X., "Well-Posedness and Solution Structure of Dual-Phase-Lagging Heat Conduction," *International Journal of Heat and Mass Transfer*, Vol. 44, (2001), pp. 1659-1669.

Wang H., Dai W., Nassar R., and Melnik R., "A Finite Difference Method for Studying Thermal Deformation in a Thin Film Exposed to Ultrashort-Pulsed Lasers," *International Journal of Heat and Mass Transfer*, Vol. 49, (2006), pp. 2712-2723.

Wang H., Dai W., Nassar R., and Melnik R., "A Finite Diffence Method for Studying Thermal Deformation in a Double-Layered Thin Film Exposed to Ultrashort-Pulsed Lasers," *International Journal of Thermal Sciences*, Vol. 45, (2006), pp. 1179-1196.

Wang H., *A Finite Difference Method for Studying Thermal Deformation in Two-Dimensional Micro Scale Metal Thin Films Exposed to Ultrashort Pulsed Lasers,* Ph. D. Dissertation, Louisiana Tech University, LA, (2007).

Wang H., Dai W., and Hewavitharana L. G., "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*, Vol. 47, (2008), pp. 7-24.

Wang P. and Dai W., "A Hyperbolic Two-Step Model Based Finite Difference Method for Studying Thermal Deformation in a 3D Micro Sphere Exposed to Ultrashort-Pulsed Lasers," *Numerical Heat Transfer, Part B,* Vol. 54, (2008), pp. 408-433.

Zhang S., Dai W., Wang H., and Melnik R., "A Finite Difference Method for Studying Thermal Deformation in a 3D Thin Film Exposed to Ultrashort Pulsed Lasers," *International Journal of Heat and Mass Transfer*, Vol. 51, (2008), pp. 1979-1995.