

University of Windsor

Scholarship at UWindor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

2010

Non-destructive investigation of thermal diffusivity in a layered material using finite difference modeling

Ghazal Ghodsi
University of Windsor

Follow this and additional works at: <https://scholar.uwindsor.ca/etd>

Recommended Citation

Ghodsi, Ghazal, "Non-destructive investigation of thermal diffusivity in a layered material using finite difference modeling" (2010). *Electronic Theses and Dissertations*. 8245.
<https://scholar.uwindsor.ca/etd/8245>

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

Non-destructive Investigation of Thermal Diffusivity in a Layered Material Using Finite Difference Modeling

By
Ghazal Ghodsi

A Thesis
Submitted to the Faculty of Graduate Studies
Through the Department of Physics
in Partial Fulfillment of the Requirements for
the Degree of Master of Science at the
University of Windsor

Windsor, Ontario, Canada
2010

© 2010 Ghazal Ghodsi



Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-62713-6
Our file *Notre référence*
ISBN: 978-0-494-62713-6

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

■+■
Canada

Author's Declaration of Originality

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

A finite-difference model for simulations of thermal wave propagation in a layered material is discussed. The method is based on the implicit scheme in solving the partial differential heat equation. According to the conventional implicit methods of solving the heat equation, a large set of equations must be solved to find the temperature distribution of the object at any time. This will cause a serious problem when working with large samples or experiments of long duration as well as generalizing the method to two and three dimensions. To avoid this complication, the concept of sparse matrices is successfully utilized to accelerate the solution of a large system of equations while simulating each time step, as well as reducing the computer memory consumption.

Parker's method of evaluating the thermal diffusivity of a material is tested by this approach. The model proves to give reliable results for thermal diffusivity measurement in 1D and 2D Cartesian systems that show good agreement with experiments conducted on samples of metals and epoxy.

To my one and only: Kiyan

ACKNOWLEDGEMENTS

I want to express my sincere thanks to the people who gave me direct or indirect support during my graduate study and research at the University of Windsor.

My sincere appreciation first goes to Dr. Roman Maev for his constant and continued support which brought me the peace of mind to freely think, study and research. His guidance both with respect to the science behind this project and academic pursuit as a whole has been indispensable.

I would also like to thank Dimitry Gavrilov, who kept me in the right direction during my studies with his great help and cooperation.

Furthermore, the role of Dr. Sokolowski, Dr. Reddish and Dr. Baylis as my committee members is also greatly appreciated.

As well, I would like to thank Sarah Beneteau and Marrie Jeannette for many times they have come to my aid.

This work was supported by the University of Windsor which its role was also of great importance in conducting this study.

And finally, I would like to thank my wonderful husband for his endless support at each and every step of the way and also my parents for their lifetime of selfless love and sacrifices.

TABLE OF CONTENTS

| | |
|--|-----|
| AUTHOR'S DECLARATION OF ORIGINALITY | iii |
| ABSTRACT | iv |
| DEDICATION | v |
| ACKNOWLEDGEMENTS | vi |
| LIST OF TABLES | xi |
| LIST OF FIGURES | xii |
| SYMBOLS AND DESCRIPTIONS | xv |
| | |
| 1 Historical Background..... | 1 |
| 1.1 Discovery of Infrared Radiation..... | 1 |
| 1.2 Thermography as a NDT Tool | 1 |
| 1.3 Advantages and Limitations..... | 2 |
| 1.4 Fundamentals of Infrared Thermography | 3 |
| 1.4.1 Emission..... | 6 |
| 1.4.2 Absorption..... | 7 |
| 1.4.3 Reflection..... | 7 |
| 1.4.4 Transmission..... | 8 |
| 1.4.5 Energy balance and Krichhoff's law..... | 9 |
| 1.5 Different Approaches: active, passive..... | 10 |
| 1.5.1 The <i>passive</i> approach..... | 11 |
| 1.5.2 The <i>active</i> approach | 12 |
| 1.6 Instrumentation for Infrared Thermography: | 20 |
| 1.6.1 Detectors | 20 |
| 1.6.2 Infrared Imaging Instrumentation | 22 |
| 1.7 Summery | 22 |
| | |
| Chapter 2..... | 23 |
| 2 Thermography Techniques | 23 |

| | | |
|-----------|---|----|
| 2.1 | The Heat Equation..... | 23 |
| 2.2 | Pulse Thermography (PT) | 23 |
| 2.2.1 | Data Acquisition in Pulse Thermography..... | 24 |
| 2.2.2 | Defect Characterization | 25 |
| 2.3 | Lock-in Thermography | 27 |
| 2.3.1 | Data Acquisition in LT | 28 |
| 2.3.2 | Establishing Phase and Amplitude Information | 29 |
| 2.4 | Pulse Phase Thermography | 32 |
| 2.4.1 | Theory | 32 |
| 2.4.2 | Data Acquisition in PPT | 34 |
| 2.4.3 | Defect Characterization in PPT Using Depth Inversion..... | 36 |
| 2.5 | Thermal Diffusivity..... | 37 |
| 2.5.1 | Diffusivity Measurement | 37 |
| 2.5.2 | Flash method..... | 39 |
| 2.6 | Summary..... | 41 |
| Chapter 3 | | 42 |
| 3 | Numerical Methods in Heat Transfer | 42 |
| 3.1 | Introduction | 42 |
| 3.2 | Finite Difference Method | 42 |
| 3.3 | Discrete Approximation of Derivatives | 44 |
| 3.3.1 | Finite Difference Approximation of First Derivative | 44 |
| 3.3.2 | Finite Difference Approximation of Second Derivative..... | 47 |
| 3.4 | Employing Finite Difference Approximation in Heat Equation | 48 |
| 3.4.1 | Types of Partial Differential Equations | 48 |
| 3.4.2 | Finite difference schemes in one dimensional parabolic system | 49 |
| 3.5 | How precise are numerical methods?..... | 55 |
| 3.5.1 | Stability | 55 |
| 3.5.2 | Fourier Method of Stability Analysis for Implicit scheme | 56 |
| 3.6 | Summary | 58 |

| | |
|---|-----|
| Chapter 4..... | 59 |
| 4 Implementation of Numerical Methods and Results | 59 |
| 4.1 Employing Explicit Scheme in One Dimension | 59 |
| 4.1.1 Set Temperature at the Boundaries | 60 |
| 4.1.2 Boundaries Subjected to Convection | 62 |
| 4.1.3 Boundaries Subjected to a Prescribed Flux | 65 |
| 4.1.4 Explicit Scheme with Lock-in Thermography..... | 67 |
| 4.2 Simple Explicit Method in 2-D Diffusion..... | 68 |
| 4.3 Implicit Scheme in 1-D | 73 |
| 4.3.1 Implementing the defect in a sample | 79 |
| 4.3.2 Results and conclusion..... | 83 |
| 4.3.3 Error Analysis in Thermal Diffusivity by Flash Method..... | 84 |
| 4.3.4 Experimental evaluation of Flash method | 85 |
| 4.3.4.1 Instrumentation | 85 |
| 4.3.4.1 Results and Error Analysis..... | 88 |
| 4.4 Implicit Method, Two Dimensions | 90 |
| 4.4.1 Theory | 90 |
| 4.4.2 Consideration on boundary condition..... | 92 |
| 4.4.2.1 Heating the surface from the top..... | 92 |
| 4.4.2.2 Heating the surface from the left side | 92 |
| 4.4.2.3 Heating the surface from the right side..... | 92 |
| 4.4.3 Problems with Implementing the Implicit Scheme and the Solution | 94 |
| 4.4.4 Simulation, Implicit Method, 2-D | 97 |
| 4.4.4.1 The coefficient matrix..... | 97 |
| 4.4.4.2 Heater Location..... | 100 |
| 4.4.4.3 Implementing Defects in a Sample | 102 |
| 4.4.4.4 Heat propagation in 3Layers..... | 104 |
| 4.5 Conclusion..... | 106 |
| Chapter 5..... | 107 |
| 5.1 Conclusion and Discussion | 107 |

| | |
|--------------------|-----|
| Appendix A..... | 109 |
| Appendix B..... | 118 |
| References..... | 129 |
| Vita Auctoris..... | 135 |

LIST OF TABLES

| | |
|--|----|
| Table 4-1 Thermal Diffusivity (Flash Method)..... | 83 |
| Table 4-2 Experimental evaluation of thermal diffusivity by Flash method..... | 88 |
| Table 4-3 Thermal diffusivity from the simulated heat propagation using Flash method..... | 98 |

LIST OF FIGURES

| | |
|---|----|
| Figure 1-1 Electromagnetic Spectrum | 3 |
| Figure 1-2 Radiation emitted by a surface (a) Directional distribution (b) Spectral distribution. (Adopted from Maldague, 2001)..... | 4 |
| Figure 1-3 Features of a blackbody cavity (a) absorption, (b) emission from an aperture (Adopted from Maldague, 2001) | 5 |
| Figure 1-4 Comparison of a blackbody and real surface distribution of radiance..... | 6 |
| Figure 1-5 Reflection (a) isotropically diffuse, (b) perfectly specular (Adopted from Maldague, 2001) | 8 |
| Figure 1-6 Transmission process illustrated as a directional irradiance | 9 |
| Figure 1-7 Flux exchange in a semi-transparent medium..... | 10 |
| Figure 1-8 Pulse thermography vs. Lock-in thermography (experimental configuration) | 18 |
| Figure 2-1 Experimental setup for active (optical) thermography by Transmission or Reflection mode, source: http://www.visiooimage.com | 24 |
| Figure 2-2 (a) Temperature on the time domain..... | 25 |
| Figure 2-3 Temperature evolution curve after absorption a rectangular heat pulse: (1) plate made of homogeneous material; (2) same plate containing a subsurface flaw (From Maldague, 1993) | 26 |
| Figure 2-4 Contrast evolution over subsurface defect (From Maldague 1993)..... | 26 |
| Figure 2-5 Input/ output modulated wave in lock-in thermography | 27 |
| Figure 2-7 Computation of phase, amplitude and thermographic images in LT. During each heating cycle, the infrared camera takes four images (Top) local thermal waves are reconstructed for any pixel (Busse, 1994) | 30 |
| Figure 2-8 Reconstructing the thermal wave in LT from 1024 data points (Wu. 1996) . | 31 |
| Figure 2-9 Lock-in thermography apparatus | 31 |
| Figure 2-10 Time frequency duality (a) an ideal temporal pulse of zero duration and infinite amplitude and (b) its frequency spectrum | 32 |
| Figure 2-11 Time frequency duality: (a) square pulse of 1 second duration centered at $t=0$; (b) corresponding frequency spectrum (From Maldague 1993)..... | 33 |

| | |
|--|----|
| Figure 2-12 Temperature evolution of pixel (i,j) through N thermograms..... | 34 |
| Figure 2-13 (a) Temperature (b) Amplitude and (c) Phase profiles for a pixel on the frequency domain..... | 35 |
| Figure 2-14 Depth evaluation with phase contrast and blind frequency | 36 |
| Figure 2-15 Shapes of photothermal excitations useful for thermal diffusivity measurements: (a) large area heating (b) line-heating (c) small spot heating (d) grating heating and (e) circular heating..... | 38 |
| Figure 2-16 Plot of the back surface temperature during time | 40 |
| Figure 3-1 Mesh representing a one dimensional rod through several time steps..... | 43 |
| Figure 3-2 The finite Difference nodes associated with the simple explicit method..... | 51 |
| Figure 3-3 The finite Difference nodes associated with the simple implicit method..... | 54 |
| Figure 4-1 The 2-D finite difference nodes associated with the simple explicit scheme..... | 69 |
| Figure 4-2 The GUI screenshot..... | 78 |
| Figure 4-3 Flash Method. Temperature vs. Time at the rear surface. Heater Temperature: 373 (K), Thermal diffusivity: $10\text{e-}8(\text{m}^2/\text{s})$ | 80 |
| Figure 4-4 Thermal Diffusivity from Flash Method vs. Thickness. (red: α_1 ;black: α_2) | 81 |
| Figure 4-5.a Thermal diffusivity vs. Pulse duration (red: α_1 , black: α_2)..... | 82 |
| Figure 4-5.b Thermal diffusivity vs. Pulse duration for the <i>half time</i> method..... | 82 |
| Figure 4-6 Experimental instrumentation..... | 85 |
| Figure 4-7 The relay system (Left), The Epoxy+Ti ₅₀ samples (Right)..... | 86 |
| Figure 4-8 The snapshot of the infrared camera record..... | 87 |
| Figure 4-9 The spot on the sample defined for temperature analysis and the plot of its temperature evaluation vs. time..... | 87 |
| Figure 4-10 Schematic presentation of converting a 2-D matrix to a 1-D array..... | 91 |

| | |
|--|-----|
| Figure 4-11 Schematic representation of the coefficient matrix (A) in implicit method, heating from the top..... | 92 |
| Figure 4-12 Schematic representation of the initial matrix (B) in implicit method, heating from the left side bar..... | 93 |
| Figure 4-13 Schematic representation of the initial matrix (B) in implicit method, heating from the right side bar..... | 94 |
| Figure 4-14 Implicit method in 2-D, the coefficient matrix for a sample of 5×6 | 96 |
| Figure 4-15 The coefficient matrix representation..... | 97 |
| Figure 4-16 Heat propagation in a sample | 98 |
| Figure 4-17 Finding thermal diffusivity in fig 4-16 using Flash method..... | 99 |
| Figure 4-18 Simulation of heat propagation in a sample with the heater at the middle of the top side..... | 100 |
| Figure 4-19 Simulation of the heat propagation with the heater at the corner..... | 101 |
| Figure 4-20 Heat propagation simulation for a sample with a single defect inside..... | 102 |
| Figure 4-21 (up) Heat propagation simulation for a sample with a multiple defects inside,(down) Flash method of finding thermal diffusivity for the same sample..... | 103 |
| Figure 4-22 Heat propagation in a 3 layered sample, the middle layer is more heat conductive..... | 104 |
| Figure 4-23 Heat propagation in a 3 layered sample, the middle layer is less heat conductive..... | 105 |

Symbols and Descriptions

| <i>Symbol</i> | <i>Page #</i> | <i>Description</i> |
|---|---------------|-------------------------------|
| E | 3 | Electromagnetic energy |
| h | 3 | Planck constant |
| f | 3 | Frequency |
| c | 3 | Speed of light |
| λ | 3 | Wavelength |
| $L_{\lambda,b}$ | 5 | Radiance |
| k | 5 | Boltzmann constant |
| T | 6 | Temperature |
| M_b | 6 | Emissive power |
| σ | 6 | Stefan's constant |
| $\varepsilon(\lambda, T_s)$ | 6 | Emissivity |
| $M(T_s)$ | 6 | Emissive power |
| $M_b(T_s)$ | 6 | Emissive power of a blackbody |
| α | 7 | Absorbance |
| φ_α | 7 | absorbed flux |
| φ_i | 7 | Incident flux |
| $\rho(\lambda, \theta, \phi, \theta', \phi')$ | 7 | Reflection |
| $\phi'_{\lambda,r}$ | 7 | Reflected flux |
| $\phi_{\lambda,i}$ | 7 | Incident flux |

| | | |
|-------------------------------|----|-----------------------------------|
| $\tau(\lambda, \theta, \phi)$ | 8 | Transmittance |
| ϕ_i | 9 | Transmitted flux |
| ρ | 9 | Reflection |
| α | 9 | Absorbance |
| τ | 9 | Transmittance |
| C_T | 13 | Thermal contrast |
| T_s | 13 | Temperature of sound area |
| T_d | 13 | Temperature of defected area |
| D_{c_T} | 13 | Detectability of thermal contrast |
| z | 13 | Depth |
| D^* | 15 | Defect visibility |
| μ | 27 | Diffusion length |
| ω | 27 | Angular frequency |
| t | 27 | time |
| $S(x)$ | 29 | Amplitude of the signal |
| $\varphi(x_i)$ | 29 | Phase |
| $A(x_i)$ | 29 | Amplitude |
| $F(f)$ | 33 | Fourier transform of frequency |
| A_p | 33 | Amplitude of a square pulse |
| Δt_p | 33 | Duration of a square pulse |
| F_n | 34 | Fourier transform |
| f_b | 37 | Blind frequency |
| L | 39 | Sample's thickness |
| D | 39 | Density |
| C_g | 39 | Heat capacity |

| | | |
|---------------|----|-------------------------------|
| $T(L, t)$ | 40 | Temperature at L, t |
| $\phi(x_i)$ | 44 | Temperature at x_i |
| r | 51 | Thermal diffusivity |
| Δt | 51 | Time interval |
| Δx | 51 | Space interval |
| ε | 56 | Error in computational method |

1 Historical Background

1.1 Discovery of Infrared Radiation

Infrared discovery goes back to 1800 when Sir William Herschel (1738–1822), German-born British Astronomer who is famous for discovering Uranus on March 13, 1773. He discovered Infrared or as he phrased it, “Invisible rays” while reproducing Newton’s experience using a prism to pass sunlight through and separate white light into colors. He measured the temperature of each color and noticed that temperature progressively increased from the violet end to the red end of the visible spectrum. But it came to his surprise that temperature increased further when he positioned the thermometer beyond the red end. Further experimentation led to Herschel's conclusion that there must be an invisible form of light beyond the visible spectrum [1].

1.2 Thermography as a NDT Tool

In many applications; from the production lines, where in situ inspection speeds up the process, to the characterization of antique art treasures and cultural heritages, there is an increasing demand for safety[2]. Destructive evaluations can be useful during the design stages, but Non-Destructive Testing and Evaluation techniques (NDT&E) are helpful inspection tools, since they allow to examine materials or components in ways that do not impair future usefulness [3]. NDT inspection techniques though, are required to be reliable, economical, sensitive, user friendly and fast. Moreover, the materials and processes are constantly evolving; hence the inspection technique should be adaptable as well [4].

Infrared Thermography (IT), among the various NDT techniques used nowadays, is known as an attractive tool for non-contact inspections. Different thermal properties of the scene (sample under study) and the object of interest (subsurface defects, cracks,

disbonds, etc.) are detected as thermal contrast assuming emissivity variations, reflections from the environment and atmosphere attenuations are negligible.

Soon after its discovery, Infrared Thermography was used as a NDT tool. From the earliest applications of Infrared NDT, was a patent in 1935 by J. T. Nicholas [5]. He used a radiometer to verify the uniformity at which steel slabs are reheated in a steel-rolling mill. In 1948 R. C. Parker and P. R. Marshall [6] investigated experimentally the temperatures generated at sliding surfaces, especially between railroad wheels and the brake blocks. The current flow in an electric power transmission line was studied by Leslie and J. R. Wait in 1986 [7]. Infrared radiometry also made it possible to detect the overheated components on circuit boards in 1961 [8].

In the mid 1960s when commercial infrared cameras became available, a wide range of applications were developed continually for this non-contact, non-intrusive technique in military, industrial, civil engineering, environmental and medical fields.

1.3 Advantages and Limitations

A thermographic survey is totally non-destructive and unlike X-rays and ultraviolet, infrared is harmless to living bodies. It provides fast inspections (up to a few m² at a time) with minimal access requirements. Measurements in inaccessible areas which are hazardous for other test methods may be done by this method. The system under study in this method needs not to be shut down for inspection, and this not only cuts on the expenses but also real time images enable immediate interpretation by a skilled practitioner [9-11].

It is also important to mention some limitations to this method.

For applying this method to a large surface, a quick and uniform thermal stimulation is difficult to achieve. For outdoor surveys, factors such as high winds, rain, standing water on the roof, etc. may lead to thermal loss and surface cooling, thus affecting the reliability of the interpretation. Cracks and disbonds large enough to result in measurable change of

thermal properties are detectable in this method. Expensive tools and experience are required to obtain and interpret the thermal images [9-11].

1.4 Fundamentals of Infrared Thermography

It is known that all bodies at a temperature above absolute zero (-273°C) emit electromagnetic radiation regardless of their state. The energy and wavelength of the emitted EM wave are related through the well known relation:

$$E = hf = \frac{hc}{\lambda} \quad (1.1)$$

Hence the electromagnetic spectrum would look like figure (1-1).

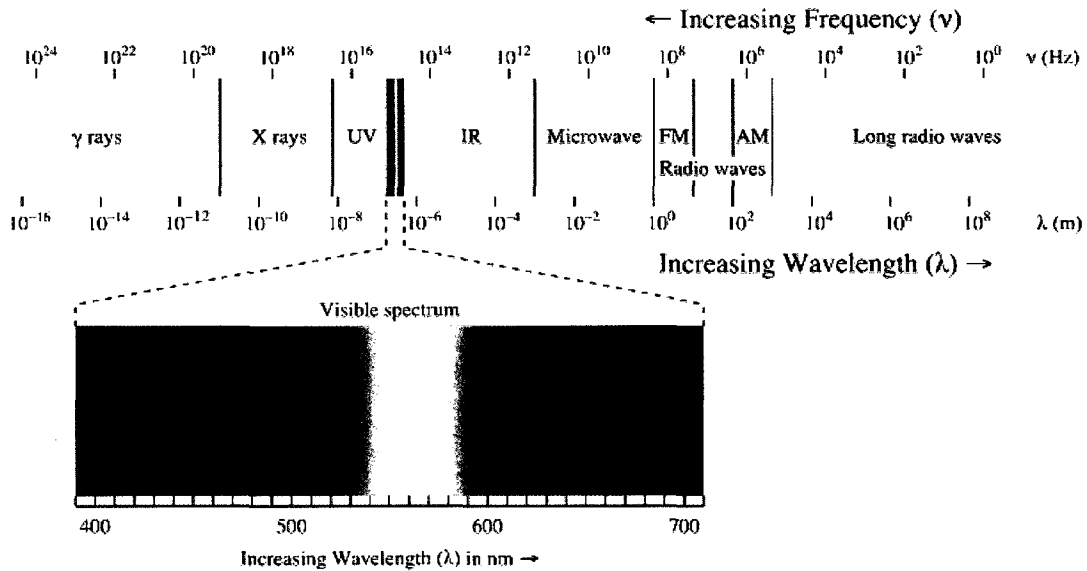


Figure 1-1 Electromagnetic Spectrum [1]

While the visible light wavelength lies between 0.4 to $0.7\ \mu\text{m}$, thermal radiation falls into the infrared section of the electromagnetic spectrum which is enclosed between 0.7 to $1000\ \mu\text{m}$.

This radiation is usually emitted from a small portion of the surface and includes a variety of wavelengths and a distribution of energy. (Figure 1-2)

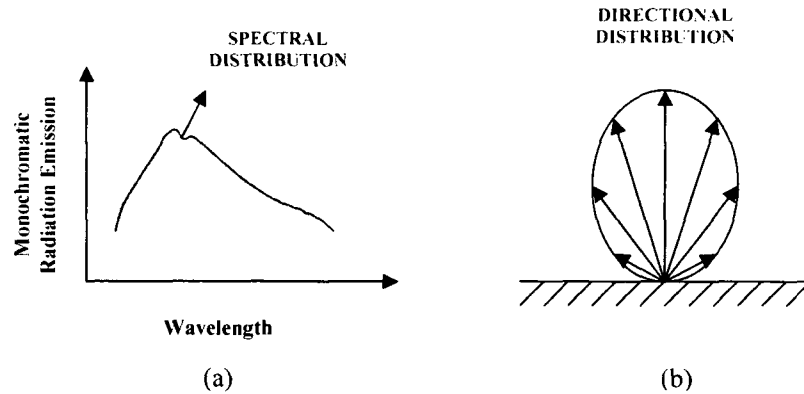


Figure 1-2 Radiation emitted by a surface (a) Spectral distribution (b) Directional distribution. (Adopted from Maldague, 2001) [3]

As an important concept in radiometry, *blackbody* is explained as a perfect radiator. A *blackbody* is defined by Gustav Robert Kirchhoff (1824-1887) to be a surface that absorbs all incident radiation of different wavelengths and directions without reflecting or transmitting it. It also re emits this energy evenly in all directions until it is in equilibrium with the environment. (Figure 1-3)

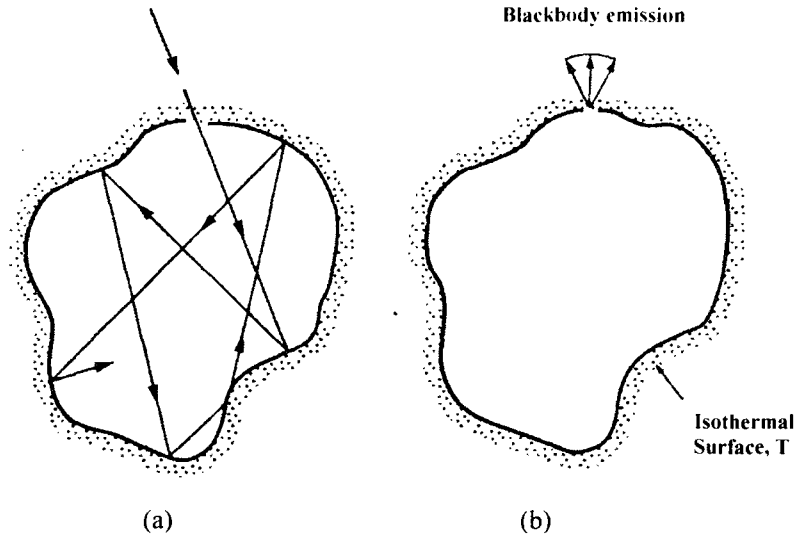


Figure 1-3 Features of a blackbody cavity (a) absorption, (b) emission from an aperture (Adopted from Maldague, 2001) [3]

According to Planck's law, the spectral radiance of a *blackbody* in thermal equilibrium is given by [12].

$$L_{\lambda,b}(\lambda, T) = \frac{2hc_0^2}{\lambda^5 [\exp(hc / \lambda KT) - 1]} \quad \text{W m}^2 \mu\text{m}^{-1} \text{sr}^{-1} \quad (1.2)$$

L would be the power of the *Blackbody* (b) per unit surface area and per unit of solid angel. h and K represent the Planck's constant (6.63×10^{-34} Js) and the Boltzmann's constant (1.381×10^{-23} J/K), c is the light's speed (3×10^8 m/s) and T is the *blackbody* temperature in Kelvin.

This equation predicts the energy radiated as a function of wavelength at temperature T . integrating Planck's law over all wavelengths ranging from 0 to ∞ , the total radiant emission would be:

$$M_b = \sigma T^4 \quad (1.3)$$

known as Stéfan-Boltzmann expression [12]. ($\sigma = 5.6697 \times 10^{-8} \text{ W/m}^2 \text{ K}^4$)

1.4.1 Emission

Unlike a *Blackbody*, a *real* surface does not absorb the incident radiation completely. *Emissivity* is the relative amount of a *real* surface's thermal radiation to that of a *Blackbody* under the same conditions of temperature (T_s) and direction [12].

$$\varepsilon(\lambda, T_s) = \frac{M(T_s)}{M_b(T_s)} \quad (1.4)$$

Emissivity of a surface is a unit less parameter that changes between 0 for a *whitebody* (perfect reflector) and 1 for a *blackbody* (perfect emitter). Figure (1-4) shows the spectral distribution of a *real* surface in comparison with a *blackbody*.

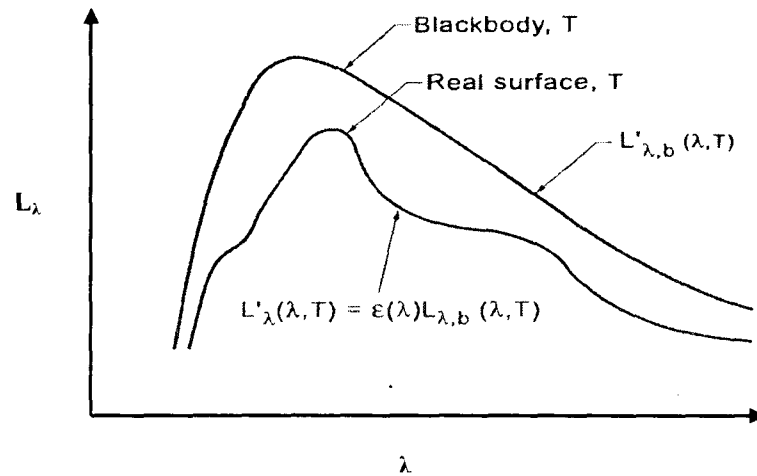


Figure 1-4 Comparison of a blackbody and real surface distribution of radiance [3]

1.4.2 Absorption

The fraction of the incident flux absorbed by the surface is called *absorbance*.

$$\alpha = \frac{d\phi_a}{d\phi_i} \quad (1.5)$$

Same as emissivity, absorbance is also dependent on the incident wavelength and direction; however it is not affected by the temperature of the surface.

1.4.3 Reflection

Reflection is defined as the ratio of the incident radiation reflected by the surface and is affected by the direction of both incident and reflected flux [12].

$$\rho(\lambda, \theta, \phi, \theta', \phi') = \frac{d\phi'_{\lambda, r}}{d\phi_{\lambda, i}} \quad (1.6)$$

This formula is not very convenient to work with, since for a given incident radiation, the reflection in all directions should be considered. Hence in practice surfaces are considered either to be isotropically *diffuse* or perfectly *specular*. As is depicted in figure (1-5), an isotropically *diffuse* surface reflects the incident radiation uniformly in all directions regardless of the incident radiation and a perfectly *specular* surface reflects the incident radiation with the same angle as the incident angle [12].

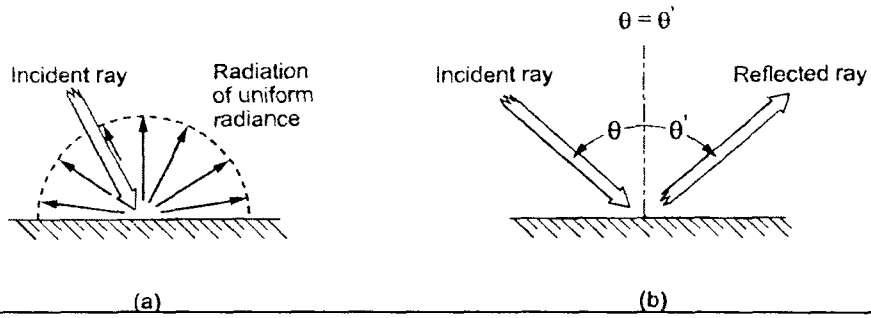


Figure 1-5 Reflection (a) isotropically diffuse, (b) perfectly specular (Adopted from Maldague, 2001) [3]

1.4.4 Transmission

To define *transmission* for a semitransparent object, the possible scattering and absorption by the particles inside should also be taken into account. That's why it is not a simple problem. However on a macroscopic scale, *transmittance* is defined as the ratio of the transmits flux in all directions to the directly incident flux [12].

$$\tau(\lambda, \theta, \phi) = \frac{d\phi_{\lambda, t}}{d\phi_{\lambda, i}} \quad (1.7)$$

It is also possible to integrate *transmittance* over all possible wavelengths ($\lambda: 0 \rightarrow \infty$), and that would be called the total directional hemispheric *transmittance*. (Figure 1-6)

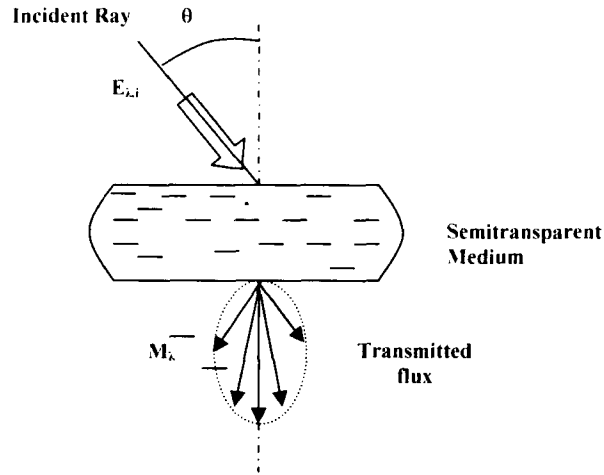


Figure 1-6 Transmission process illustrated as a directional irradiance

1.4.5 Energy Balance and Krichhoff's Law

Incident radiation on a semitransparent material is partly reflected by the surface and the rest is either transmitted through the object or absorbed by it. However, as shown in figure (1-7), the net flux should remain unchanged.

$$\phi_i = \phi_r + \phi_a + \phi_t \quad (1.8)$$

Integrating equation (1.8) over all possible wavelengths and similar directional conditions, properties expressed previously will be related as:

$$\rho + \alpha + \tau = 1 \quad (1.9)$$

For opaque objects ($\tau = 0$), (1.9) becomes:

$$\rho + \alpha = 1 \quad (1.10)$$

Krichhoff's law relates absorption and emission for a *blackbody* [12].

$$\varepsilon(\lambda, \theta', \varphi') = \alpha(\lambda, \theta, \varphi) \quad (1.11)$$

Although equation (1.11) only applies to a *blackbody*, it can still be used for a *diffuse* surface for which emission and absorption are independent of spectral and directional conditions [12]. For thermography inspections black painting is used to increase emissivity and also to prevent reflections from environment.

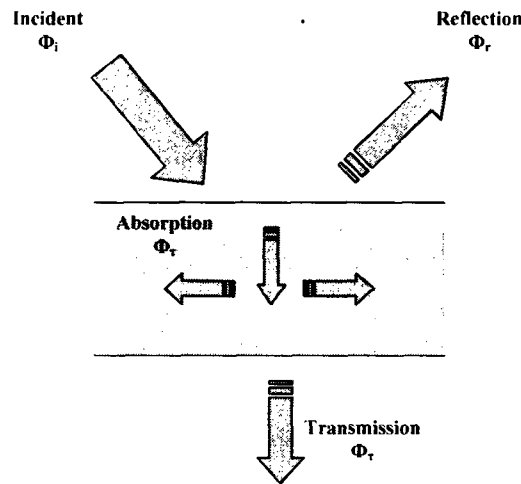


Figure 1-7 Flux exchange in a semi-transparent medium

1.5 Different Approaches: Active, Passive

In order to investigate a structure with Infrared Thermography, there are two approaches that can be used, passive and active. In the passive approach, the specimen and the ambient are naturally at different temperatures (usually the specimen is at higher temperature). While in the active approach, an external excitation source, such as heating or cooling system needs to supply the thermal contrast between different parts of the sample.

1.5.1 The *Passive* Approach

The passive approach is commonly used for civil engineering structures inspection, maintenance, medicine and property evaluation. The material under investigation is usually examined for abnormal temperature profile indicating an irregularity or possible defect, void, etc.

In passive thermography, the key word is the temperature difference with respect to the surrounding, often referred to as the “delta-T” or the “hot spot.” A delta-T of 1 to 2 °C is generally found suspicious while a 4°C value is a strong evidence of abnormal behavior [10]. however there have been more sophisticated applications of this approach rather than only qualitatively pin down the anomalies. These investigations are more complicated and need thermal modeling.

For instance in a reported application [15], a numerical method was developed to simulate the heating process of needles used to sew fabrics in the automobile industry. (Seat cushions and backs, airbags, etc.) The friction between the needle and the fabric (usually synthetic) in high speed sewing generates heat which will cause worn or broken thread and damages the thread as well as burns the fabric and weakens the needle itself. Experiments on the plant floor have shown the temperature raise of up to 100-300°C with the sewing speed of 1000 to 3000 rpm.

The model uses different factors such as needle design, thread fabric properties and sewing speed to predict the temperature distribution in the needle during the sewing process as well as the time needed to reach a steady state. By understanding the heating process, they can reach the optimized operation through changes in needle geometry, cooling system, etc.

1.5.2 The Active Approach

Unlike the passive approach, in the active infrared thermography the characteristics of the stimulus is known and that makes it possible to get quantitative results such as the depth of a possible defect, hence it's widely used in NDT&E. Depending on the different methods of data acquisition and data processing, various active thermography approaches has been developed.

1.5.2.1 Pulsed Thermography (PT)

In Pulse Thermography, the object surface is being stimulated by a heating pulse and the data acquisition is performed by monitoring the temperature evolution of the surface after it starts to cool down. Lamps, flash lamps, scanning lasers or hot air jets are the most popular heating tools and Infrared cameras are used to monitor the signal in the transient area. There are two possible modes of data acquisition in PT. Reflection mode and Transmission mode. In the Reflection mode, the heat source and the Infrared camera are both placed on the same side of the specimen while in the transmission mode, the camera faces the back surface of the sample. Reflection mode is preferred in cases where the defect is closer to the heated surface or the rear surface is not accessible. The signal received by the IR camera (Thermograms) is then being processed to estimate the characteristics of the subsurface defect such as size, depth and thermal resistance.

Cielo in 1984 was the first who came up with the idea of pulsed approach [16]. He applied this technique on the aluminum coating plasma sprayed on aluminum substrates with artificial defects between the coating and the substrate. The parameters of interest were the thickness, adhesive and cohesive strength. In order to interpret the experimental results, he developed a numerical model based on the inversion procedure and got encouraging results.

In 1985, for the first time Balageas et al. developed an analytical model using finite difference method to study the thermal decay in a coated substrate [17]. Given the

thermal properties and the thickness of the substrate, the model could evaluate the thermal properties and the thickness of the coating and the thermal contact resistance of the interface. Cielo and Krapez used this technique to remotely measure the thickness of oil film on water. They could evaluate the oil thickness up to 1 mm with the error range of 0.1 mm [18]. However, there are some practical limitations in adopting this method for open-air environmental applications. The problems could rise by the energy absorption of the laser pulse by the atmospheric aerosols or the oil film immersing into water.

One of the problems with thermal images developed by PT was the blurring. It was assumed that loss of information during the data acquisition was the cause of blurring, however in 1992, Favro et al [19] could remove the blurring by developing an algorithm which performs a mathematical inversion of the scattering process to the experimental data. As proved by Vavilov (1992) [20], the reverse method will predict more than one combination of defect characteristics for a single thermogram. Thus, it was suggested that the iterative calculation should be done knowing some of the defect's parameters, *a priori* beforehand [21]. However, these methods turned out to be time consuming and only applicable to some specific cases.

The difference between the thermal properties of the defected and non-defected (sound) areas causes the thermal contrast which is measured and evaluated in PT.

$$C_T = \frac{T_s - T_d}{T_s} \quad (1.12)$$

The larger the thermal contrast, the more detectable the damaged area. C_T not only changes by different atmospheric conditions and the device precision, but also is affected by the size and depth of the defect. Smaller defects and deeper-located ones will result in lower thermal contrast, hence harder to evaluate. It is understood that the loss in C_T is proportional to the cube of depth:

$$D_{C_T} \approx \frac{1}{z^3} \quad (1.13)$$

Maldague (2002) [22] suggested to use a rule of thumb that says the radius of the smallest detectable defect should be at least one to two times larger than its depth under the surface.

To study the limits in defect evaluation by infrared thermography, Meola *et al* ran a thorough experiment on composite specimens with external material as defects [23].

Results proved the key role of the thickness in defect visibility, meaning that a thin large defect is more difficult to detect than a thick small one at the same depth.

However, there are a few drawbacks in these computational approaches leading to some inaccuracies. One of such was the fact that not always a priori parameter of the defect was available. Also the sound area was assumed to have evenly distributed temperature over the surface.

In 2002, Pilla *et al* [24] solved the one dimensional Fourier equation assuming a Dirac pulse applied to a homogeneous semi-finite plate:

$$\Delta T = \frac{Q}{\sqrt{\pi \rho C k}} \frac{1}{\sqrt{t}} \quad (1.14)$$

In this equation, Q is the total energy absorbed and $\sqrt{\rho C k}$ is called thermal effusivity (e). This equation defines the temperature of sound area locally as a function of time. Hence the variation of effusivity with time is studied to evaluate the depth and thermal resistance of the defect zone.

This approach brought some distinctive advantages to PT by introducing the New Absolute Contrast (NAC) based on the reconstruction of the temperature in the sound area [24].

Since thermal contrast is evaluated based on the total energy (Q) and Q is affected by non-uniform heating and convection, Peng and Almond (1998) [25] proposed to replace it with defect visibility.

$$D^* = \frac{\Delta T_d}{\Delta T_s} \quad (1.15)$$

The subscripts d and s represent the surfaces above the center of the defect and the sound area.

Although pulse thermography is the sufficient choice in many applications, it is affected by the local variation of the emissivity coefficient and non-uniform heating that can mask the defect visibility. The emissivity problem may be solved by painting the surface, but this could be a solution only for parts where this surface finish is allowable [21].

Various methods based on different stimulating or processing techniques are being reviewed in the following sections.

1.5.2.2 Lock-In Thermography (LT)

Heat injection during the short time of a pulse (e.g. typically 10 ms for a powerful flash lamp) may cause initially high temperature on the surface. Another surface heating method is by modulated heat deposition in which energy is distributed over a longer period of time. Therefore it reduces the thermal load of the inspected component to a power density which is typically below the one of sunshine at noon in summer [26].

The basic idea in modulated or Lock-in thermography (LT) is submitting the specimen under study to a periodic (sinusoidal) temperature stimulant. In this case, highly attenuated and dispersive waves will form inside the sample and they are known as “thermal waves” [10]. Like all other waves, thermal waves reflect from the boundaries and interact with any discontinuity in the sample. This reflected modulated temperature contains some useful information of the defect, boundary, etc. The result is the amplitude

and phase of the sinusoidal wave pattern at each point which can be used to evaluate the temperature variation. The phase image can be used to directly specify not only the size but also the depth and thermal resistance of any kind of imperfection in a sample.

Carlomagno and Berardi came up with the idea of modulated thermography in 1976 [27]. Further investigation to utilize NDT of materials with this method was done by Beaudoin *et al* (1985) and Kue *et al* (1992) [28, 29]. In 1994, Busse *et al* used LT for depth profiling of fiber orientation in composite laminates [30]. Since then LT has been used for defect detection in veneered wood (Wu 1994) [31], measure thickness, density and porosity of ceramic coatings (Wu *et al* 1996a, Rantala *et al* 1996) [32, 33] and inspection of aircraft components (Wu *et al* 1996b 1998) [26, 34].

Meola and his group (2002, 2003, 2004) [35-40] at the University of Naples investigated in the capability of this method to deal with other applications such as measurements of material thermal diffusivity, evaluation of assembly ways like adhesive bond joints, as well as control of bonding improvements after surface plasma treatment.

For a thermal wave modulated at frequency ω , the thermal diffusion length is given by this relationship:

$$\mu = \sqrt{\frac{\alpha}{\pi f}} \quad (1.16)$$

Where α is the thermal diffusivity of the material and the wave frequency is $f = \omega/2\pi$.

While μ is the maximum depth range in amplitude image, the maximum thickness that can be inspected in a phase image is proved by Busse and Rosencwaig (1980) and Busse (1982) [41, 42] to be:

$$p = 1.8 \sqrt{\frac{\alpha}{\pi f}} \quad (1.17)$$

Hence the penetration depth of thermal waves in the material depends on the properties of the material (heat conductivity, heat capacity and density) as well as the wave cycle time. The slower the wave, the deeper it can penetrate in the material. To start a test, usually one would start to investigate the surface layer using high frequency waves, then reducing the frequency to observe deeper layers until the whole thickness is examined.

From limitations of this approach, one is the low frequency range needed for investigation of thick materials with low thermal diffusivity which not all the heat flux modulators can provide. Also it is possible to miss a defect because of not choosing a proper frequency. In practice lock-in thermography is rather time consuming for evaluation of thick materials with low thermal diffusivity.

More recently in 1999, Dillenz *et al* used an ultrasonic transducer attached to the sample [44]. Their method has the advantage of selective heating of only the damaged area, and is proved to detect deeper and smaller defects. From the applications of this method are detection of corrosion, vertical cracks and delaminations. In the next section a combination of the previous methods named Pulsed Phase Thermography (PPT) is described.

1.5.2.3 Pulsed Phase Thermography (PPT)

The idea behind Pulse Phase Thermography is to combine two previous methods in a way that enables us to make use of the advantages of PT and LT while avoiding their drawbacks. Data acquisition in Pulse thermography is fast and straight forward while in LT, more specialized equipment is needed plus the experiment is rather time consuming in LT. On the other hand, the data in PT is highly affected by thermal, optical and geometrical surface characterizations and the environmental reflections. while using the phase images to process the data in LT makes it less sensitive to these conditions.

The link between these two methods is established by Fourier transform according to which any pulse can be transformed to a sum of harmonic waves with a mix of

frequencies. (Figure 1-8) Based on this theory, in PPT as in PT, the sample is pulse heated, yet the Fourier transform on the temporal decay wave in the specimen unscrambles the mix of frequencies in each pixel. So that, a phase or magnitude image of each pixel is provided as in Lock-in Thermography.

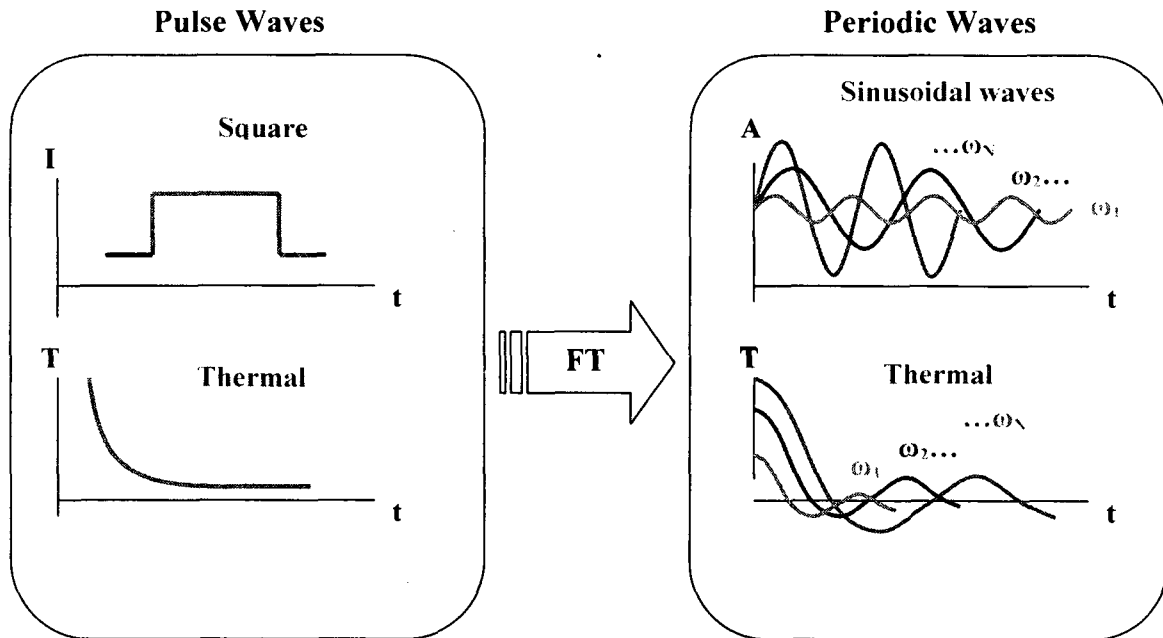


Figure 1-8 Pulse thermography vs. Lock-in thermography (experimental configuration)

Historical Notes on Pulsed Phase Thermography

Pulse Phase approach was introduced by Maldague and Marinetti in 1996 [44]. Maldague and his group kept working on this method and reported their progress and results on its application. Given the non-linear and noisy character of pulsed thermographic data, inverse solutions are difficult to reach, that's why PPT was only used as a qualitative defect detection approach for the first couple of years from its discovery until in 1998. Maldague and his group tried to get quantitative results using neural networks [46]. Neural network approach was chosen due to its ability to work with non linear problems with missing or noisy data. Although this approach opened new windows to quantitative

investigations in infrared NDT, high sampling frequency was needed for the data acquisition system, depending on the material's thermal diffusivity.

A practical drawback of Fourier Transformation is the infinite extensions on time axis of the basis function which causes information loss on the time evolution of the spectral properties of the signal. The time information is needed for defect depth retrieval, so another transform rather than Fourier Transform is needed. In 2000, Maldague *et al* adopted wavelet transform for signal processing [47]. This method provided the frequency analysis without missing on time information. However, the last two approaches proved to be too lengthy in computational routines to be applicable in most NDT processes. Later in 2004, they proposed a new depth retrieval approach based on absolute phase contrast defined as [48].

$$\Delta\varphi = \varphi_d - \varphi_s \quad (1.18)$$

Where the subscripts *d* and *s* stand for the defect and sound areas. They proved that the depth changes with the *blind frequency* (f_b), frequency at which a defect generates enough phase contrast to be invisible, as:

$$z \propto f_b^{1/2} \quad (1.19)$$

They also tested their theory on Plexiglas and Aluminum, two materials with low and high conductivity and the results well agreed with the theory [48]. Following this work, steel plates with flat bottomed holes were studied by this group in 2005 [49]. The defects were located at different depths ranging from 1 to 4.5 mm and the results showed the same correlation between the blind frequency and the defect depth.

PPT has proved to have a great potential for NDT applications and investigations on its applicability and improving the results are still under study.

1.6 Instrumentation for Infrared Thermography:

When it comes to experimental analysis, the set up and instrumentation is of great importance. This is an overview of investigation tools, such as detectors, radiometers, infrared cameras and a brief review of the methods to apply various thermography techniques and data acquisition.

1.6.1 Detectors

Optical receivers are the main part of infrared radiometry systems. As of 1999, the worldwide IR detector market was estimated to be \$10 billion increasing by rate of 30% every year [51]. An infrared detector or sensor is used to convert the radiant energy to some other assessable quantity such as the electrical signal. There are two types of sensors: thermal detectors and photonic detectors.

1.6.1.1 Thermal Detectors

In thermal detectors, the incident radiation raises the surface temperature which changes the electrical conductivity of the heated material and in turn affects the output signal. Thermal detectors response is independent of incident wavelength and this characteristic makes it unique in this way. To discuss the sensitivity of thermal detectors, one should be familiar with effective conductivity G_R which is given by [52]:

$$G_R = 4\sigma T^3 A \quad (1.20)$$

Where T represents the sensor temperature, σ is the Stefan-Boltzmann constant, and A is the detector-sensitive area. Then the detectivity limit is [51]:

$$D^* = \sqrt{\frac{A}{4kT^2G}} \quad (1.21)$$

k is the thermal conductivity and G is the limit of G_R .

Depending on the physical property affected by the surface heating in the thermal detector, there are different types of thermal detectors.

Bolometers, are the ones in which the electrical conductivity is the changing factor. *Thermopiles* are made from many *thermocouples* connected in series or in parallel. Temperature difference generates voltage difference in each thermocouple through Thompson effect. The signal in *Pneumatic* detectors is generated by pressure variation of a definite volume of gas. Another type of thermal detectors is *Pyroelectric* detectors that make use of the Pyroelectric crystals. As introduced in 1984 by Gränicher, temperature change in Pyroelectric crystals produces electric polarity [52]. Temperature variation in the detector's surface Pyroelectric crystals causes the charge variation and thus a transient current to be picked up. *Liquid crystals* are also used as thermal sensors. When illuminated by white light, they reflect color light from red to violet under temperature changes providing the resolution of up to 0.01°C [50].

1.6.1.2 Photonic Detectors

In Photonic detectors, incident photons on a semi conductor material will generate excitation which is directly measured by the detector. The incident photons on photocathode will pull away electrons from the valance to the conduction atomic band. This method prevents unnecessary heating of the sensitive surface. There are two types of photonic detectors: photoemissive and quantum

In *photoemissive photonic detectors*, the current (electron flow) will generate the resulting signal while in *quantum detectors* either the change in conductivity or the generated voltage is being measured. Since there is no heating involved in this process, it is much faster than thermal detectors and the use of solid state material allows these detectors to be compact, robust and dependable [50].

1.6.2 Infrared Imaging Instrumentation

In order to get one or two dimensional images from the detector, there are two main approaches: using a single detector attached to an electromechanical scanning device, or using an array of individual IR detectors with no movement applied (Focal Plain Arrays). Since first introduced in 1973 by Shepherd and Yang [53], large arrays of 512×512 has been produced by different companies and they are the most popular IR imaging tools in today's technology. An infrared camera is made of FPA, the appropriate optics and electronics and cooling system. There are two readout tools for the FPA, Charge Coupled Device (CCD) and Complementary Metal Oxide Semiconductor Device (CMOSD). CMOSD is less expensive than CCD and enables windowing (reading a section of the array rather than the whole data) however; it has higher noise and less sensitivity. In today's applications, CCD leads the scientific and technical field while CMOSD is being employed in the applications such as surveillance, camcorders, snapshots, etc.

1.7 Summery

In this chapter, basic and practical concept of infrared thermography was reviewed. Also a brief review on various NDT applications for IR thermography since its discovery was presented. Pulse thermography and lock-in thermography as two main methods in IR thermography and their advantages and drawbacks were described. As a conclusion, pulse phase thermography, as a combination of both Pulse Thermography and Lock-in Thermography was found to benefit the advantages of both methods without sharing their imperfections.

In the imaging process, thermal and photonic detectors are explained based on their method of converting the thermal variation of the surface to an electric signal. For further signal processing, two types of devises (CCD and CMOSD) are used for different applications. The next step in IR thermography would be image processing methods which will be reviewed in the next chapter.

Chapter 2

2 Thermography Techniques

In this chapter, the main principals of heat propagation in a material are presented as well as different numerical solutions to the heat equation. Mathematics of various image processing methods in IR thermography is discussed to prepare for the logic behind the algorithm developed in the next chapter.

2.1 The Heat Equation

The one dimensional heat equation is [55]:

$$\frac{\partial T(x,t)}{\partial t} = \alpha \frac{\partial^2 T(x,t)}{\partial x^2} \quad 0 \leq x \leq L, \quad t \geq 0 \quad (2.1)$$

In this equation, $\alpha = k/\rho c$ is the thermal diffusivity and is a property of the material with k , ρ and c being heat conductivity ($\text{W.m}^{-1}.\text{K}^{-1}$), density (kg.m^{-3}) and specific heat ($\text{J. kg}^{-1}.\text{K}^{-1}$). T is the transient heat function in a semi-infinite material with thickness L .

This equation can be solved both for a pulse or periodic thermal waves.

2.2 Pulse Thermography (PT)

Pulse thermography is a popular method in IR thermography in which the specimen is pulse heated and the temperature decay is recorded. Assuming the heat pulse to be perfect Dirac delta function (in practice however, a pulse with very short duration is used), the solution to equation (2.1) in a semi-infinite isotropic solid would be [56]:

$$T(z,t) = T_0 + \frac{Q}{\sqrt{k\rho c_p \pi}} \exp\left(-\frac{z^2}{4\alpha t}\right) \quad (2.2)$$

In this equation, z represents the depth and Q is the energy absorbed by the surface [J/m^2] and T_0 is the initial temperature [K].

To study the temperature behavior at the surface ($z=0$), Eq. (2.2) becomes:

$$T(0,t) = T_0 + \frac{Q}{e\sqrt{\pi}} \quad (2.3)$$

e being *effusivity* and defined as $\sqrt{k\rho c_p}$.

2.2.1 Data Acquisition in Pulse Thermography

A typical experimental setup for active thermography is illustrated in figure (2-1). First the thermal pulse is sent to the specimen's surface. Second an infrared camera records the thermal decay of the surface. The heat source and the camera are either placed on the same or opposite sides of the sample depending on the inspection method (Reflection/Transmission) selected.

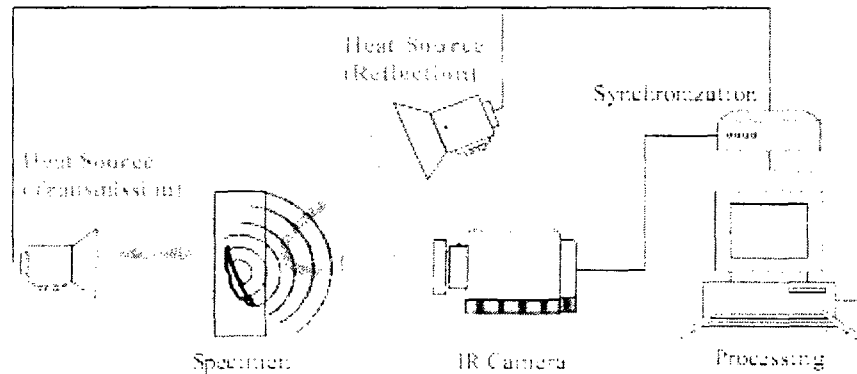


Figure 2-1 Experimental setup for active (optical) thermography by Transmission or Reflection mode, source: <http://www.visioimage.com>

The data is then recorded in 3D matrices, containing both spatial (x - y coordinates) and temporal (z coordinate) information. To process the data there are two options of (1) working at pixel level which is the temperature changes of a fixed point in space through time or (2) at 2D images or *thermograms* which are the temperature profile of the surface on time domain. (Fig. 2-2)

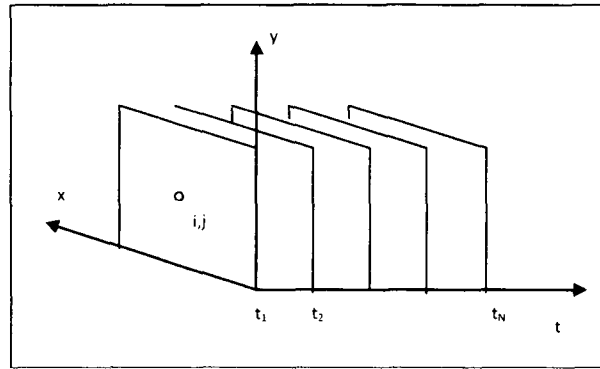


Figure 2-2 Thermograms on the time domain, it is also possible to study the temperature behavior of pixel (i,j) through time.

2.2.2 Defect Characterization

The temperature profile of a sample can be used for subsurface defect detection purposes. After the heat pulse is employed on the sample's surface, it reaches the subsurface layers and drops quickly due to *diffusion* and also radiation and convection losses. However, a defect under the sample's surface will reduce the *diffusion* rate. Figure (2-3) illustrates the temperature decay curves of two samples of the same material with and without a subsurface defect. They behave similarly the first instants after the heat pulse is applied, however as soon as the heat front reached the defect area with different *effusivity*, the heat exchange rate with environment changes hence the difference in temperature above the defect spot.

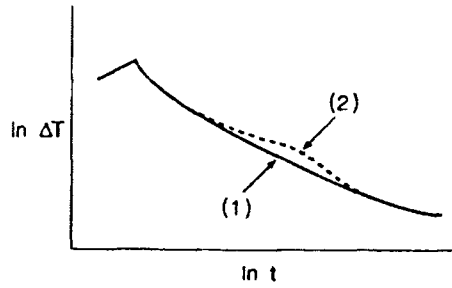


Figure 2-3 Temperature evolution curve after absorption a rectangular heat pulse: (1) plate made of homogeneous material; (2) same plate containing a subsurface flaw (From Maldague, 1993)

This allows finding temperature contrast which is defined as:

$$C(t) = T_{def}(t) - T_{sound}(t) \quad (2.4)$$

Thermal contrast is the base of many defect detection analyses. Some methods employ the time and temperature at which the maximum of the peak is located for each pixel (figure 2-4), to reconstruct Maximum Contrast *timegrams* or thermograms. Also half the time of maximum of thermal contrast curve [57] and the slope of the curve [58] have been used to indicate the size, depth and thermal resistance of the defect.

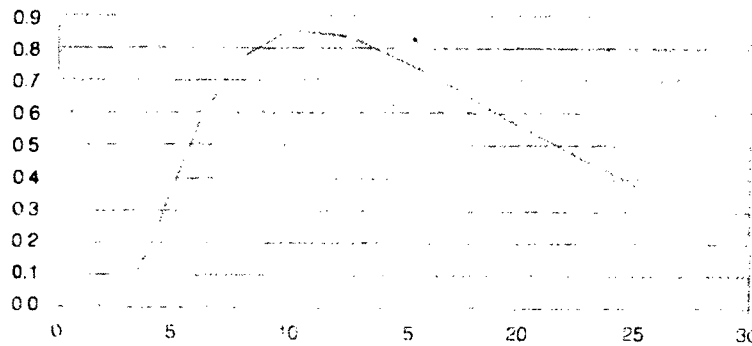


Figure 2-4 Contrast evolution over subsurface defect (From Maldague 1993)

This method provides good results as long as the sample is homogenous with fairly shallow defects. Moreover, non-uniformity in heating process is practically inevitable, which makes the thermal contrast curve difficult to interpret.

2.3 Lock-in Thermography (LT)

In Lock-in thermography (LT), instead of a heat pulse, modulated heat is generated by the heat source and delivered to the surface. The input modulated wave is defined by its angular frequency ω and its magnitude I , the thermal wave generated in the sample will propagate at the same frequency but different amplitude (A) and also a shift in the phase value (ϕ). (Figure 2-5)



Figure 2-5 Input/ output modulated wave in lock-in thermography

The one dimensional solution to Eq. 2.1 for a modulated heat source is expressed as [60]:

$$T(z, t) = T_0 \exp\left(-\frac{z}{\mu}\right) \cos\left(\frac{2\pi z}{\lambda} - \omega t\right) \quad (2.5)$$

It is assumed that the thermal wave is propagating in a homogeneous semi-infinite material where T_0 is the temperature of the surface right after stimulated by the source and μ is the *diffusion length* defined as [60]:

$$\mu = \sqrt{\frac{2\alpha}{\omega}} \quad (2.6)$$

α is *thermal diffusivity* of a material and expressed as [60]:

$$\alpha = \frac{k}{\rho c_p} \quad (2.7)$$

k, ρ, c_p are thermal conductivity, density and specific heat.

2.3.1 Data Acquisition in Lock-in Thermography

Lock-in thermography has the advantage of not being a point by point imaging process. In the conventional thermography methods, a laser beam would stimulate the surface at a certain point and the detector would pick up the locally generated thermal wave from the same point. The Fourier analysis is then to be performed on the read out signal to provide the amplitude and phase information of that point. One drawback to the point by point scanning process is time concerns. The measurement time should be long enough to cover one complete modulation cycle at minimum; for instance, it would take 5 seconds for a thermal wave with frequency of 0.2 Hz to complete its cycle. Now this should be multiplied by the number of pixels (usually as order of 10^5 pixels) to obtain the total data acquisition time. This approach could take so long for a large sample that makes it inapplicable to the industry.

However, this problem would be solved by using an extended heating source instead of the point heating [59]. Also it is possible to employ an infrared camera as a thermal detector array. A Fourier transform should be performed on the signal from each pixel to get the magnitude and phase of the modulation.

One concern is that the process should not take longer than the data acquisition itself. In 1998, Wu and Busse [59] found that the process would be particularly fast and easy using a sinusoidal heat injection.

2.3.2 Establishing Phase and Amplitude Information

It works on the base of stimulating the sample's surface by modulated heat and observing the temperature variation over the surface to reconstruct the responding thermal wave. In case of a sinusoidal heat injection, the temperature field is also a sine wave which can easily be recovered from three or four data points in a modulation cycle. As shown in figure 2-6, the reference signal is reconstructed from 4 equidistant data points, S_i ($i = 1$ to 4) being the detector signals at pixel x_1 . The Amplitude A and phase φ are [59].

$$\varphi(x_1) = \arctan \left[\frac{S_1(x_1) - S_3(x_1)}{S_2(x_1) - S_4(x_1)} \right] \quad (2.8)$$

$$A(x_1) = \sqrt{[S_1(x_1) - S_3(x_1)]^2 + [S_2(x_1) - S_4(x_1)]^2} \quad (2.9)$$

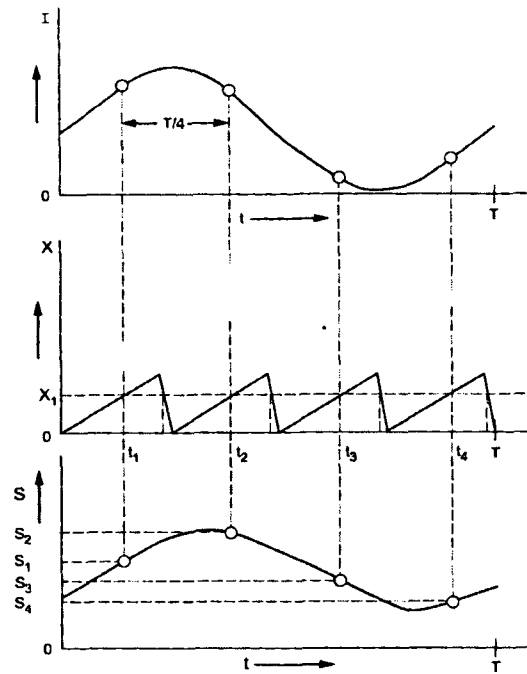


Figure 2-6 Computation of phase, amplitude and thermographic images in LT. During each heating cycle, the infrared camera takes four images (Top) local thermal waves are reconstructed for any pixel (Busse, 1994)

As it is clear from equation (2.8), phase image is less affected by inhomogeneous illumination, surface emissivity variation [59] or the reflection from the environment [61].

Although the image can be recovered by only four data points, in practice one would prefer to work with different modulation frequencies which would fit more than four points in a complete cycle. Moreover, working with more data points will reduce the noise. In that case, as shown by Wu in 1996 [62], all points in one 90° -interval can be averaged to 1 point which is the corresponding S_i in equations (2.8) and (2.9). (Fig. 2-7)

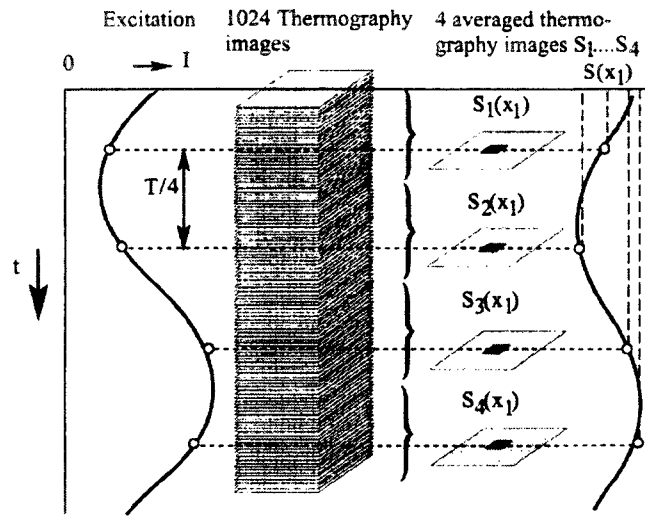


Figure 2-7 Reconstructing the thermal wave in LT from 1024 data points (Wu, 1996)

Finally other than the modulated heat injection, a lock-in apparatus is similar to that of pulse thermography. (Figure 2-8)

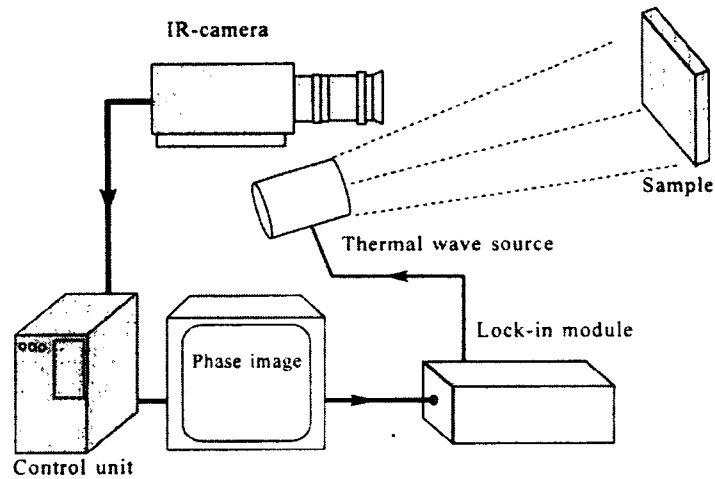


Figure 2-8 Lock-in thermography apparatus

2.4 Pulse Phase Thermography (PPT)

2.4.1 Theory

Pulse phase thermography (PPT) is in fact a combination of the two previously described Lock-in thermography and Pulsed thermography methods. PPT utilizes the time frequency duality to extract phase images from PT results using Fourier transform, the well known mathematical formulation in transform analysis.

In theory, an ideal Dirac pulse of null duration can be transformed to a flat distribution over frequencies ranging from $-\infty$ to $+\infty$ in the frequency domain. (Figure 2-9)

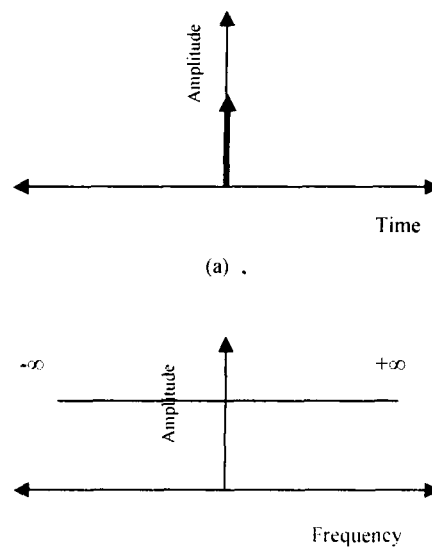


Figure 2-9 Time frequency duality (a) an ideal temporal pulse of zero duration and infinite amplitude and (b) its frequency spectrum

However, in practice the thermal pulse used is different from an ideal Dirac pulse in the sense that it has finite amplitude and duration values. As an example, one can consider a square pulse with amplitude A_p and duration ΔT_p centered on $t = 0$. (Figure 2-10) The corresponding frequency spectrum would be defined by Fourier transform [50].

$$F(f) = \frac{A_p \Delta t_p \sin(\pi f \Delta t_p)}{\pi f \Delta t_p} = A_p \Delta t_p \operatorname{sinc}(\pi f \Delta t_p) \quad (2.10)$$

The shape of F as a function of frequency would look like the graph in figure 2-11, which proves that unlike an ideal Dirac pulse, here a finite frequency range is involved in the frequency domain, each appearing with various amplitudes.

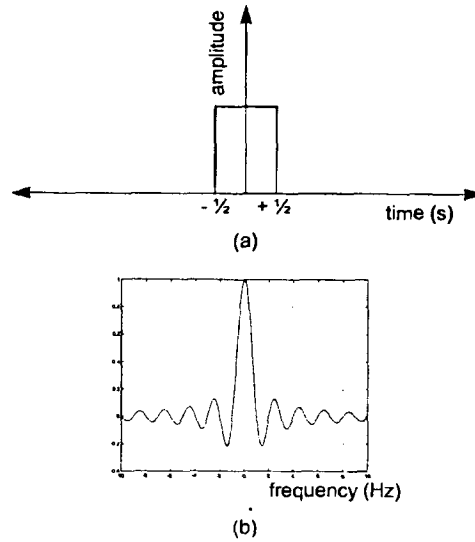


Figure 2-10 Time frequency duality: (a) square pulse of 1 second duration centered at $t=0$; (b) corresponding frequency spectrum (From Maldague 1993)

2.4.2 Data Acquisition in Pulsed Phase Thermography

In Pulsed Phase thermography, after the sample surface is stimulated by a heat pulse, temperature evolution of the surface is recorded in N thermogram sequences. Observing the behavior of a single pixel (i, j) during N recorded thermograms will provide a vector $T(k)$ where k is ranging from 0 to $N-1$. (Figure 2-11) Amplitude and phase images can be extracted from this vector using a discrete one-dimensional Fourier transform [51].

$$F_n = \sum_{k=1}^N T(k) \exp(2\pi i k n / N) = \text{Re}(F_n) + i \text{Im}(F_n) \quad (2.11)$$

In this equation n stands for the frequency increment in the frequency domain.

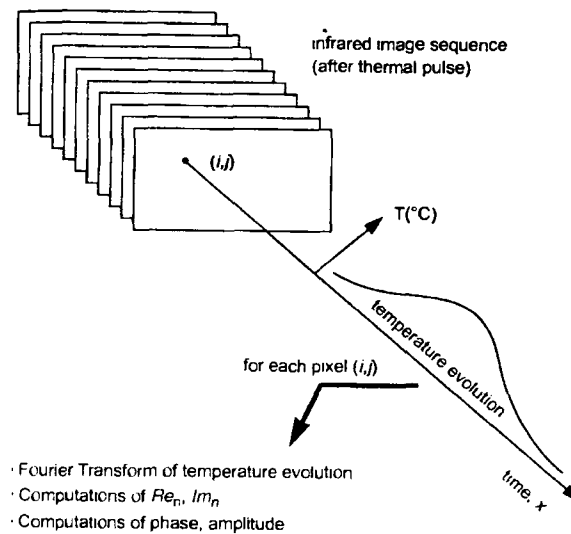


Figure 2-11 Temperature evolution of pixel (i, j) through N thermograms

Establishing Phase and Amplitude in Pulsed Phase Thermography:

The imaginary and real parts of equation (2.11) are then used to extract the amplitude and phase images [51].

$$A_n = \sqrt{\text{Re}_n^2 + \text{Im}_n^2} \quad \text{and} \quad \phi_n = \alpha \tan \frac{\text{Im}_n}{\text{Re}_n} \quad (2.12)$$

This process is repeated for every pixel on the surface until the ampligrams and phasegrams for all the frequencies in the frequency domain is reached [63]. (Figure 2-12)

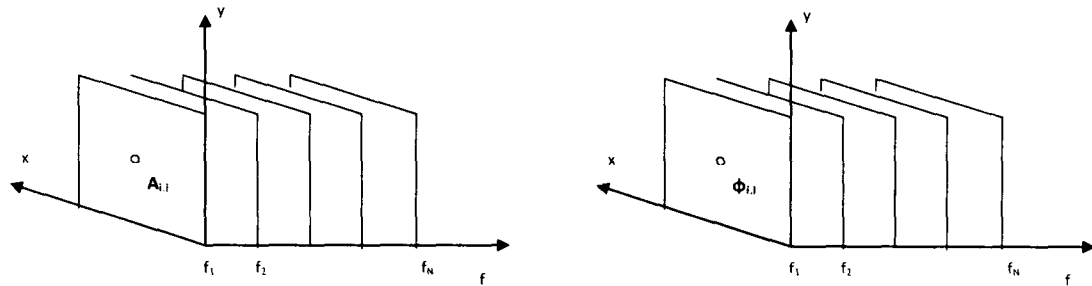


Figure 2-12 (Left) Amoligram and (Right) Phasegram sequence in the frequency domain

For a typical non-defective pixel like (i,j), the amplitude and phase profile would look like figure (2-13) [63].

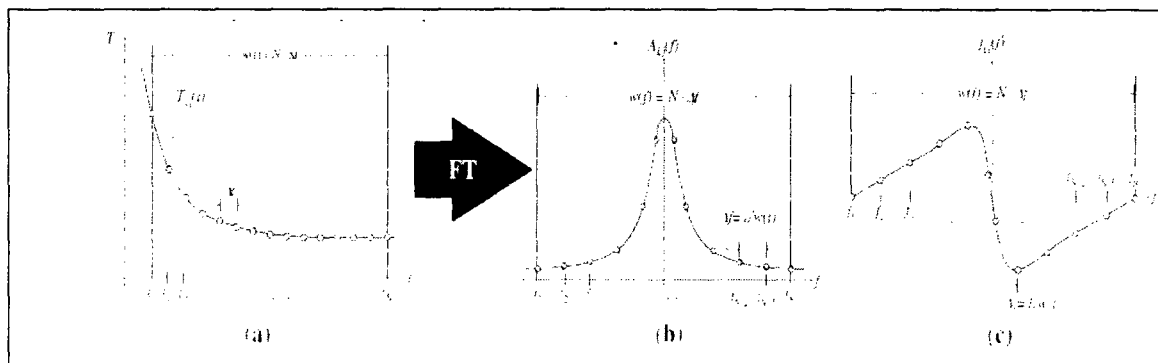


Figure 2-13 (a) Temperature (b) Amplitude and (c) Phase profiles for a pixel on the frequency domain, Adopted from [63]

It can be seen from figure (2-13) that a sequence of N thermograms will provide N/2 useful frequency components and the other half can be predicted due to the odd/even nature of the profiles.

The advantage of working in frequency domain is that the phase images are much less sensitive to problems such as non-uniform heating, reflections from environment, and emissivity variations in the surface. Also as compared to LT, this method has the advantage of getting a complete phase profile from a single run.

2.4.3 Defect Characterization in Pulsed Phase Thermography Using Depth Inversion

Once the data acquisition is done, the phase contrast profiles will be employed to find the defect's depth. Figure (2-14) shows the phase profiles of two flat bottomed holes in a sample located at different depths z_1 and z_2 . Considering z_s as a sound area in the same sample, the phase contrast profiles for each defected pixel is calculated using $\Delta\phi = \phi_d - \phi_s$. As depicted in figure (2-14), phase contrast profiles have non-zero values from $f=0$ to a specific frequency names *blind frequency* (f_b), which has a different value for each pixel. *Blind frequency* is defined as the first point in frequency domain (while moving from higher frequencies to lower ones) at which the phase contrast is visible for a defect. In fact *blind frequency* for a pixel is the point on its phasegram where it merges with the phase profile of a sound area. As a result, deeper defects have lower blind frequencies ($f_{b,z_1} > f_{b,z_2}$).

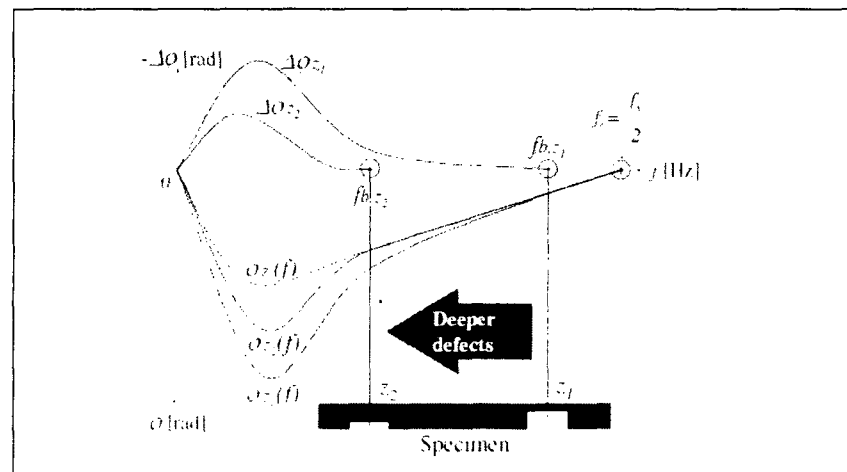


Figure 2-14 Depth evaluation with phase contrast and blind frequency

Once the *blind* frequency is found for a defect zone, the depth z is proved to be related to it as follows [63].

$$z = C_1 \sqrt{\frac{\alpha}{\pi f_b}} + C_2 \quad (2.13)$$

Where $\alpha = k / \rho c_p$ is thermal diffusivity. As this term is used to fit experimental data, C_1 and C_2 are found to be around 1 and 1.5 respectively [63].

2.5 Thermal Diffusivity

Thermal diffusivity is a significant property of a material that establishes the rate of transitory heating and cooling and is defined as:

$$\alpha = \frac{k}{\rho c_p} \quad (2.14)$$

Where k , ρ and c_p are thermal conductivity, density and specific heat.

Measuring α is of great importance to researchers for various reasons. Knowing the thermal diffusivity of a material makes it possible to predict the heat pulse behavior in it, as well as investigating the subsurface defects and composition.

2.5.1 Diffusivity Measurement

Different techniques have been developed over time to measure the thermal diffusivity of a material. *Non-contact* measurements involve a transitory excitation delivered to the surface of the material under investigation and an optical probe to detect α .

Both pulse and modulated heat source can be employed to excite the surface initially as well as various shapes of initial excitations. Here are some schemes for such methods [64]: (Figure (2-15))

- (1) Large area surface heating
- (2) Line heating
- (3) Small spot heating
- (4) Grating heating
- (5) Circular heating

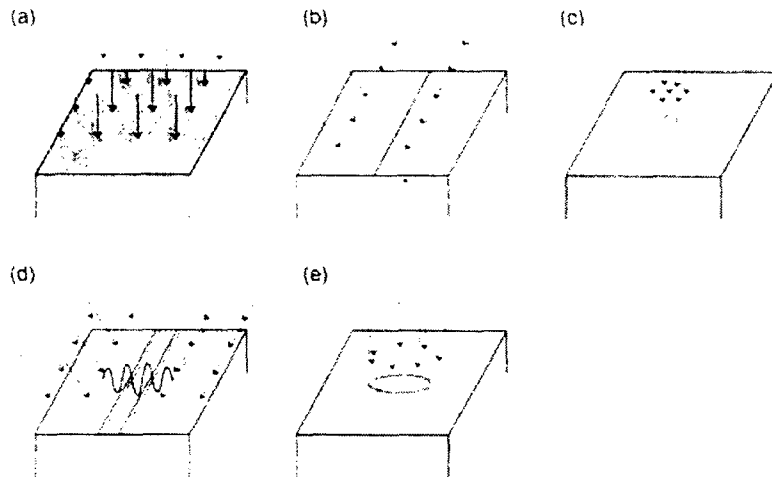


Figure 2-15 Shapes of photothermal excitations useful for thermal diffusivity measurements: (a) large area heating (b) line-heating (c) small spot heating (d) grating heating and (e) circular heating

The optical detection is then completed inside or on the edges of the sample by exploring the changes in one of these properties [64]:

- (1) Transient thermal radiation from the surface
- (2) Refraction of the probe beam
- (3) Reflection of the probe beam
- (4) Diffraction of the probe beam
- (5) Surface deformation

If a heat pulse is used as an initial excitation, the temporal decay of one of these properties is observed to get thermal diffusivity. In case of employing a modulated excitation, the change in its phase is related to the thermal diffusivity.

A comprehensive review of the first three methods can be found in [64] while the transient thermal grating and surface deformation of a metallic film is studied in [65-68].

2.5.2 Flash Method

Advances in material technology and increasing applications of materials at high temperatures have caused a great interest in the methods to determine the thermal diffusivity (α). To measure this parameter in high temperature, implementing old methods are not of great interest due to the time consuming tests required for reliable data or the large sample size needed. However, Parker *et al* [69] introduced Flash method to measure thermal diffusivity.

When a thermally insulated sample of thickness L is being stimulated by a one dimensional thermal pulse ($T(x,0)$), the temperature distribution at a later time will be [69]:

$$T(x,t) = \frac{1}{L} \int_0^L T(x,0) dx + \frac{2}{L} \sum_{n=1}^{\infty} \exp\left(-\frac{n^2 \pi^2 \alpha t}{L^2}\right) \times \cos \frac{n\pi x}{L} \int_0^L T(x,0) \cos \frac{n\pi x}{L} dx \quad (2.15)$$

Assuming that the initial temperature pulse has the radiant energy Q (cal/cm²) and it is uniformly absorbed by the front surface of the sample, $T(x,t)$ will be given by:

$$T(x,0) = Q / DC_g \quad \text{for } 0 < x < g \quad (2.16)$$

$$T(x,0) = 0 \quad \text{for } g < x < L \quad (2.17)$$

g in this equation is a very small depth value close to the surface ($x=0$). Substituting $T(0,t)$ in equation (2.15), the temperature profile at later times is [69]:

$$T(x,t) = \frac{Q}{DCL} \left[1 + 2 \sum_{n=1}^{\infty} \cos \frac{n\pi x}{L} \frac{\sin(n\pi g/L)}{(n\pi g/L)} \times \exp\left(\frac{-n^2 \pi^2}{L^2} \alpha t\right) \right] \quad (2.18)$$

Where D is the density (g/cm³) and C is the heat capacity (cal/g°C). Parker *et al* studied the temperature profile at the rear surface ($L=0$) of such a sample. Since g is a very small value, $\sin(n\pi g/L) \approx n\pi g/L$, and (2.18) becomes:

$$T(L,t) = \frac{Q}{DCL} \left[1 + 2 \sum_{n=1}^{\infty} (-1)^n \exp\left(\frac{-n^2 \pi^2}{L^2} \alpha t\right) \right] \quad (2.19)$$

Figure (2-16) represents the dimensionless plot of the sample's back surface according to equation (2.19). It can be seen from (2.19) that α can be determined in two ways [69].

$$\alpha_1 = (1.38L^2 / \pi^2 t_{\frac{1}{2}}) \quad (2.20)$$

$$\alpha_2 = (0.48L^2 / \pi^2 t_x) \quad (2.21)$$

$t_{1/2}$ is the time required for the back surface temperature to reach half of the maximum temperature and t_x is the time axis intercept of the temperature versus time curve.

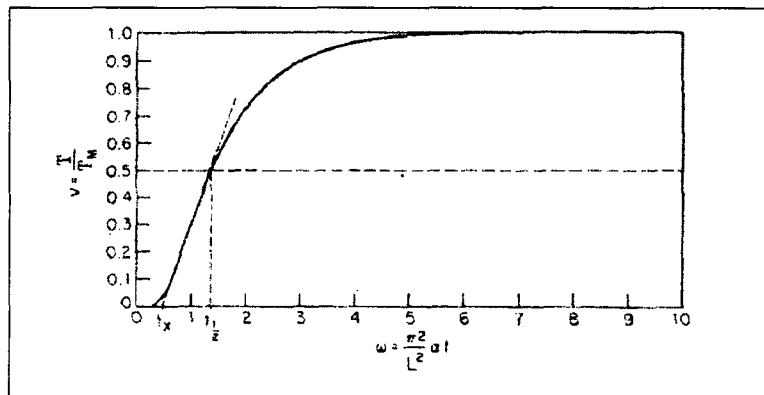


Figure 2-16 Plot of the back surface temperature during time [69]

2.6 Summary

Different methods of data acquisition and data analysis in thermography are discussed in this chapter. Pulse thermography is a popular method where a thermal pulse is introduced to the sample's surface and its temporal decay is studied to investigate surface properties. It provides good results for homogenous samples with fairly shallow defects. However, in practice, the pulse has non-uniformities that make the thermal contrast curve difficult to interpret. Lock-in thermography is another method introduced that uses a modulated thermal wave to stimulate the surface and the change in amplitude and phase of the wave defines thermal properties of the sample. The problem with this method is that there should be a run of experiment for each pixel on the sample, so that it takes a long time. However modulated wave has the advantage of not being sensitive to noises and non-uniformities. As an optimum thermography method PPT (pulse phase thermography) is then introduced that combines the advantages of both methods without their limitations. The surface is pulse heated as in PT while the temporal decay is transformed to phase and amplitude images through Fourier transform.

Also Parker's Flash method of measuring thermal diffusivity is discussed. This method is the most reliable way of measuring thermal diffusivity so far and will be utilized in the next chapters to verify the applicability of the proposed simulation for thermal diffusivity evaluation of materials.

Chapter 3

3 Numerical Methods in Heat Transfer

3.1 Introduction

In the past few years, there has been a great effort to develop various numerical methods for the solution of heat equations. Although there are exact solutions available to the heat equation, they only apply to linear flow of heat or areas of simple shape. Studying thermal wave behavior in the more complicated boundary conditions and sample shapes needs numerical solutions.

Among the various finite difference schemes to solve partial differential equations, a particular scheme should be picked depending on the type of partial differential equation considered.

The equation to be solved is the one-dimensional time dependent heat conduction:

$$\frac{\partial^2 \phi}{\partial x^2} - \frac{1}{\alpha} \frac{\partial \phi}{\partial t} = 0 \quad (3.1)$$

where $\phi(x)$ is the function of temperature distribution along a 1D rod.

The idea behind the numerical solutions is to replace the partial differential equation by a discrete approximation and solve the resulting set of algebraic equations to find the discrete unknowns. The more nodes (discrete points) used in a numerical method, the greater resolution and accuracy of the method.

3.2 Finite Difference Method

The finite difference method (FDM) is one of the most popular numerical solutions to equation (3.1). In this method, a *mesh* is considered as the set of points in location-time

domain for which $\phi(x,t)$ is obtained approximately. Figure (3-1) is a representation of a one dimensional mesh with nodes located equally distanced on position (Δx) and time (Δt) axis.

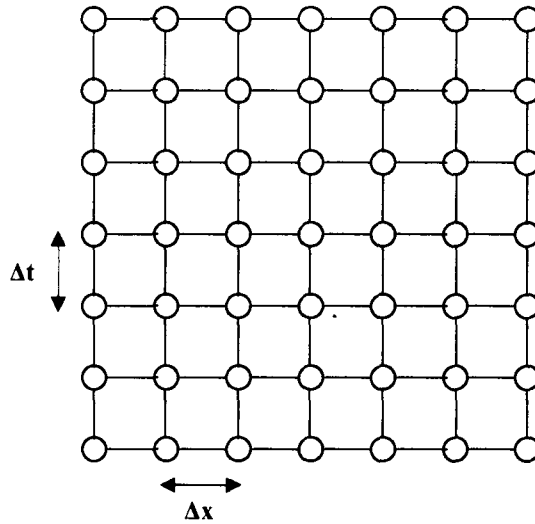


Figure 3-1 Mesh representing a one dimensional rod through several time steps

Applying the finite-difference method to the heat equation is in fact replacing the continuous derivatives with difference formulas. Difference formulas as being explained in the following section use different combinations of the points on the mesh to replace the continuous derivatives. Depending on which combination of the mesh points to use in a difference formula, there are different schemes to employ in a finite difference problem.

After the proper scheme is chosen and applied to the partial differential equation, the numerical solution will approach to the real solution if the time and position spacing go to zero. However, not all the schemes will develop the true solution by the same rate. Hence, choosing a proper finite difference scheme for a particular problem is of great importance. The following sections will be on reviewing some of these schemes and applying them to the heat equation.

3.3 Discrete Approximation of Derivatives

3.3.1 Finite Difference Approximation of First Derivative

In finite difference method to replace the partial differential equations with a proper difference formula, the Taylor series expansion of $\phi(x)$ about the random point x_i is considered [70].

$$\phi(x_i + \delta x) = \phi(x_i) + \delta x \left. \frac{\partial \phi}{\partial x} \right|_{x_i} + \frac{\delta x^2}{2} \left. \frac{\partial^2 \phi}{\partial x^2} \right|_{x_i} + \frac{\delta x^3}{3!} \left. \frac{\partial^3 \phi}{\partial x^3} \right|_{x_i} + \dots \quad (3.2)$$

Where δx represents a small change along the x axis. Solving this equation for $\left(\frac{\partial \phi}{\partial x} \right)_{x_i}$:

$$\left. \frac{\partial \phi}{\partial x} \right|_{x_i} = \frac{\phi(x_i + \delta x) - \phi(x_i)}{\delta x} - \frac{\delta x}{2} \left. \frac{\partial^2 \phi}{\partial x^2} \right|_{x_i} + \frac{\delta x^2}{3!} \left. \frac{\partial^3 \phi}{\partial x^3} \right|_{x_i} + \dots \quad (3.3)$$

When applying this method for a mesh of N nodes along a one dimensional rod of length

L , δx is replaced by Δx ($= \frac{L}{N-1}$) and equation (3.3) would change to:

$$\left. \frac{\partial \phi}{\partial x} \right|_{x_i} \approx \frac{\phi_{i+1} - \phi_i}{\Delta x} - \frac{\Delta x}{2} \left. \frac{\partial^2 \phi}{\partial x^2} \right|_{x_i} + \frac{\Delta x^2}{3!} \left. \frac{\partial^3 \phi}{\partial x^3} \right|_{x_i} + \dots \quad (3.4)$$

Where the exact values of $\phi(x_i)$ and $\phi(x_i + \Delta x)$ are substituted by the approximated solutions ϕ_i and ϕ_{i+1} assuming that i is the mesh point at x_i .

Rewriting equation (3.4) while only keeping the first order derivative gives:

$$\left. \frac{\partial \phi}{\partial x} \right|_{x_i} = \frac{\phi_{i+1} - \phi_i}{\Delta x} + O(\Delta x) \quad (\text{forward}) \quad (3.5)$$

Where the order of “ $O(\Delta x)$ ” determines the *truncation error* associated with the finite difference scheme used. In this case $O(\Delta x)$ is:

$$O(\Delta x) \equiv \frac{\Delta x}{2} \left. \frac{\partial^2 \phi}{\partial x^2} \right|_{x_i} + \frac{(\Delta x)^2}{3!} \left. \frac{\partial^3 \phi}{\partial x^3} \right|_{x_i} + \dots \quad (3.6)$$

Equation (3.5) is called *forward difference formula* to replace the first order derivative of ϕ at x_i . The term *forward* suggests that the nodes involved in this estimate were x_i and x_{i+1} . Δx is under the control of the user, so is the truncation error.

First order backward difference is obtained by involving nodes i and $i-1$, and using $\delta x = -\Delta x$ in equation (3.2):

$$\phi_{i-1} = \phi_i - \Delta x \left. \frac{\partial \phi}{\partial x} \right|_{x_i} + \frac{(\Delta x)^2}{2} \left. \frac{\partial^2 \phi}{\partial x^2} \right|_{x_i} - \frac{(\Delta x)^3}{3!} \left. \frac{\partial^3 \phi}{\partial x^3} \right|_{x_i} + \dots \quad (3.7)$$

Solving for the first order derivative to get *backward difference* equation:

$$\left. \frac{\partial \phi}{\partial x} \right|_{x_i} = \frac{\phi_i - \phi_{i-1}}{\Delta x} + O(\Delta x) \quad (\text{backward}) \quad (3.8)$$

This is an alternative first order difference term used to solve the partial differential equations. However, the truncation error remains of the same order for both *forward* and *backward* formulas.

There is another finite difference approximation known as *central difference* for which the magnitude of the truncation error turns out to be smaller than the previous two methods.

To determine the *central difference* equations, (3.7) is to be subtracted from (3.2):

$$\begin{aligned}
 \phi_{i+1} &= \phi_i + \Delta x \left. \frac{\partial \phi}{\partial x} \right|_{x_i} + \frac{(\Delta x)^2}{2} \left. \frac{\partial^2 \phi}{\partial x^2} \right|_{x_i} + \frac{(\Delta x)^3}{3!} \left. \frac{\partial^3 \phi}{\partial x^3} \right|_{x_i} + \dots \\
 \phi_{i-1} &= \phi_i - \Delta x \left. \frac{\partial \phi}{\partial x} \right|_{x_i} + \frac{(\Delta x)^2}{2} \left. \frac{\partial^2 \phi}{\partial x^2} \right|_{x_i} - \frac{(\Delta x)^3}{3!} \left. \frac{\partial^3 \phi}{\partial x^3} \right|_{x_i} + \dots \\
 \phi_{i+1} - \phi_{i-1} &= 2\Delta x \left. \frac{\partial \phi}{\partial x} \right|_{x_i} + \frac{2(\Delta x)^3}{3!} \left. \frac{\partial^3 \phi}{\partial x^3} \right|_{x_i} + \dots
 \end{aligned} \tag{3.9}$$

Solving for the first order derivative:

$$\left. \frac{\partial \phi}{\partial x} \right|_{x_i} = \frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x} + O(\Delta x)^2 \quad (\text{central}) \tag{3.10}$$

This is the *central difference* method to solve a partial differential equation. As is revealed by equation (3.10), *central difference* is a more accurate approximation since the truncation error associated with it is second order in Δx .

3.3.2 Finite Difference Approximation of Second Derivative

To develop the finite difference scheme for a partial derivative of second order, the same method as in *central difference* is used. However in this scheme, *forward* and *backward* terms are added to get the result:

$$\phi_{i+1} + \phi_{i-1} = 2\phi_i + (\Delta x)^2 \frac{\partial^2 \phi}{\partial x^2} \Big|_{x_i} + \frac{2(\Delta x)^4}{4!} \frac{\partial^4 \phi}{\partial x^4} \Big|_{x_i} + \dots \quad (3.11)$$

Solving for $(\partial^2 \phi / \partial x^2)_{x_i}$, gives:

$$\frac{\partial^2 \phi}{\partial x^2} \Big|_{x_i} = \frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{(\Delta x)^2} + O(\Delta x)^2 \quad (\text{central}) \quad (3.12)$$

This equation is called the *central* scheme since the nodes involved are a central node (x_i) and the neighboring nodes (x_{i-1}) and (x_{i+1}). However the *forward* and *backward* relations are also easy to find [71].

$$\frac{\partial^2 \phi}{\partial x^2} \Big|_{x_i} = \frac{\phi_i - 2\phi_{i+1} + \phi_{i+2}}{(\Delta x)^2} + O(\Delta x) \quad (\text{forward}) \quad (3.13)$$

$$\frac{\partial^2 \phi}{\partial x^2} \Big|_{x_i} = \frac{\phi_{i-2} - 2\phi_{i-1} + \phi_i}{(\Delta x)^2} + O(\Delta x) \quad (\text{backward}) \quad (3.14)$$

Although two/three grid points are involved in the schemes above to find the equivalent approximation for first and second order derivatives, methods using more than three nodes to evaluate the derivatives at node x_i are also possible to develop [71].

As is seen in equation (3.12), the truncation error in the *central* scheme is from the order of magnitude of $(\Delta x)^2$ while it is of order (Δx) for the *forward* and *backward* methods.

3.4 Employing Finite Difference Approximation in Heat Equation

3.4.1 Types of Partial Differential Equations

To solve a second order partial differential equation with finite difference schemes, there are various schemes to choose from. However, selecting the right method depends on the type of the differential equation to be solved.

To study this matter the most general form of the second order partial differential equation presented by Forsythe *et al.* (1967) is considered [72].

$$A \frac{\partial^2 \phi}{\partial x^2} + B \frac{\partial^2 \phi}{\partial x \partial y} + C \frac{\partial^2 \phi}{\partial y^2} + D \frac{\partial \phi}{\partial x} + E \frac{\partial \phi}{\partial y} + F \phi + G(x, y) = 0 \quad (3.15)$$

It is assumed that the coefficients A, B...G are not a function of ϕ , however they may be dependent on x and y . Different values of A, B and C may then result in 3 types of differential equations [72].

$$B^2 - 4AC < 0 \quad \text{Elliptic} \quad (3.16a)$$

$$B^2 - 4AC = 0 \quad \text{Parabolic} \quad (3.16b)$$

$$B^2 - 4AC > 0 \quad \text{Hyperbolic} \quad (3.16c)$$

Here are some examples of the heat conduction equations under each category [71].

The steady state heat conduction equation is in fact *Laplace's* equation and is *elliptic*:

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \quad (3.17)$$

This equation turns to a *Poisson's* equation in case there is an energy generation involved and is also *elliptic*:

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{1}{k} g(x, y) = 0 \quad (3.18)$$

Example of a *parabolic* equation is the one-dimensional time dependent heat equation:

$$\frac{\partial^2 \phi}{\partial x^2} = \frac{1}{\alpha} \frac{\partial \phi}{\partial t} \quad (3.19)$$

The wave equation is of second order in time and is an example of a *hyperbolic* equation:

$$\frac{\partial^2 \phi}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 \phi}{\partial t^2} \quad (3.20)$$

To identify whether an equation is *elliptic* or *parabolic*, the order of the partial differential equations involved should be considered. In equation (3.17), the second order partial derivatives in x and y imply that the condition at any random location on x-y domain is affected by the changes in x and y variables on both sides of that point. Such an equation is called *elliptic*. However, the time derivative in equation (3.19) is of first order, hence any change at a point on the time axis is influenced only by the previous point on it. It is important to know that even one such derivative in an equation is enough for it to be a *parabolic* equation, as in equation (3.19). This explains the reason why *parabolic* equations are easier to work with since they have the advantage of major cutback in data storage.

3.4.2 Finite Difference Schemes in One Dimensional Parabolic System

In this section the finite difference formulation of one-dimensional parabolic systems are studied as well as its stability. Transient heat problem in one-dimension is a one of numerous parabolic systems. A single parabolic problem may have many possible finite

difference solutions. The following is an example of three solutions which were used in this thesis.

In these schemes the partial time and space derivatives are replaced by a discrete approximation. The analysis is based on the assumption that heat is propagating in a finite region of length L .

$$\frac{\partial \phi(x,t)}{\partial t} = \alpha \frac{\partial^2 \phi(x,t)}{\partial x^2} \quad (3.21)$$

To specify discrete values of $\phi(x,t)$ at different time and locations ϕ_i^n is used. The subscript i and the superscript n specify the point at $x=x_i$ at the n th time step.

3.4.2.1 Simple Explicit Scheme

In this method the time derivative in (3.21) is approximated by the *forward* difference and the second order spatial derivative is substituted by the *central* approach.

$$\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} = \alpha \frac{\phi_{i-1}^n - 2\phi_i^n + \phi_{i+1}^n}{(\Delta x)^2} + O(\Delta t) + O(\Delta x)^2 \quad (3.22)$$

Solving for ϕ_i^{n+1} to get:

$$\phi_i^{n+1} = \phi_i^n + \frac{\alpha \Delta t}{(\Delta x)^2} (\phi_{i-1}^n - 2\phi_i^n + \phi_{i+1}^n) \quad (3.23)$$

The truncation error dropped is from the order of $O[\Delta t, (\Delta x)^2]$. A simple rearrangement of the terms in this equation will develop the final answer:

$$\phi_i^{n+1} = r\phi_{i-1}^n + (1-2r)\phi_i^n + r\phi_{i+1}^n \quad (3.24)$$

where

$$r = \frac{\alpha \Delta t}{(\Delta x)^2} \quad (3.25)$$

Equation (3.24) is called the *simple explicit* method in solving the heat diffusion problem. In this scheme, assuming the temperatures at nodes $i-1$, i , $i+1$ are available at time level n , the temperature at i at the next time level ($n+1$) is developed. (Fig 3.2)

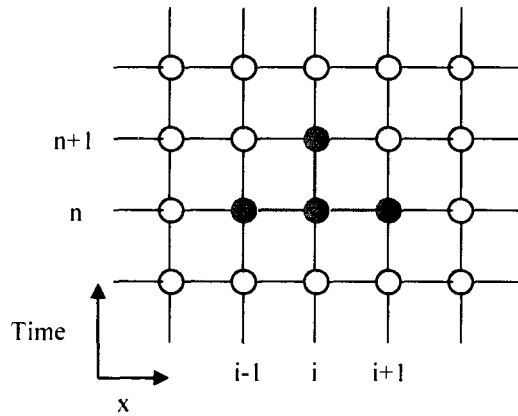


Figure 3-2 The finite Difference nodes associated with the simple explicit method

It is interesting that in this scheme, the value of ϕ_i^{n+1} is independent of the values ϕ_{i-1}^{n+1} and ϕ_{i+1}^{n+1} .

If a rod of length L is divided to M intervals on the mesh, i changes in the range $0 \leq i \leq M$, hence $M+1$ unknown potentials (ϕ_i^n) are to be found. However, there are only $M-1$ algebraic equations available by (3.24) at each time step. The other two equations are found by applying the boundary conditions. The following is a study on heat diffusion in a one dimensional rod with three different boundary conditions [71].

a) If the temperature at the boundaries is known at all times, then the two additional equations would be:

$$\phi(0,t) = \phi_0 \quad (3.26-a)$$

$$\phi(L,t) = \phi_L \quad (3.26-b)$$

b) In case where the edge surfaces are subject to convection due a temperature gradient with the environment:

$$-k \frac{\partial \phi(0,t)}{\partial x} + h_0 \phi(0,t) = h_0 \phi(0,\infty) \quad (3.27-a)$$

$$k \frac{\partial \phi(L,t)}{\partial x} + h_L \phi(L,t) = h_L \phi(L,\infty) \quad (3.27-b)$$

Where h_0 and h_L represent the heat transfer coefficients at $x=0$ and $x=L$, and $\phi(0,\infty)$ and $\phi(L,\infty)$ are the known ambient temperatures at each end into which the heat is transferred. To discretize these two equations, *forward* and *backward difference* schemes are used in equations (3.27-a) and (3.27-b) respectively [71].

c) When there is a prearranged flux (Q_0, Q_L) at the boundaries:

$$-k \frac{\partial \phi}{\partial x} = Q_0 = \text{known} \quad (3.28-a)$$

$$k \frac{\partial \phi}{\partial x} = Q_L = \text{known} \quad (3.28-b)$$

The same approach as in the previous part (b) is applicable to this equation given that h_0 and h_L values are zero here.

3.4.2.2 Simple Implicit Scheme

To derive simple implicit scheme, *backward difference* should be used to discretize the time partial derivative on the left hand side of equation (3.21).

$$\left. \frac{\partial \phi}{\partial t} \right|_{t_n, x_i} = \frac{\phi_i^n - \phi_i^{n-1}}{\Delta t} + O(\Delta t) \quad (3.30)$$

Similar to the explicit method, in this scheme also *central difference* is used to substitute the spatial derivative in equation (3.21). Introducing these discrete values to equation (3.21) results in:

$$\frac{\phi_i^n - \phi_i^{n-1}}{\Delta t} = \alpha \frac{\phi_{i-1}^n - 2\phi_i^n + \phi_{i+1}^n}{(\Delta x)^2} + O(\Delta t) + O(\Delta x)^2 \quad (3.31)$$

Here, same as in Equation (3.22) the accuracy of the approximation is of order of $O[(\Delta x)^2, \Delta t]$. Rearranging this algebraic formula shows that unlike the explicit method, here the potential ϕ_i^n is not independent of its neighboring nodes ϕ_{i-1}^n and ϕ_{i+1}^n .

$$\frac{1}{\Delta t} \phi_i^{n-1} = -\frac{\alpha}{(\Delta x)^2} \phi_{i-1}^n + \left(\frac{1}{\Delta t} + \frac{2\alpha}{(\Delta x)^2} \right) \phi_i^n - \frac{\alpha}{(\Delta x)^2} \phi_{i+1}^n \quad (3.32)$$

Therefore a set of equations should be solved to find the potential of a node at later times. Figure 3-3 illustrates how the nodes associated with simple implicit scheme are different from the explicit method.

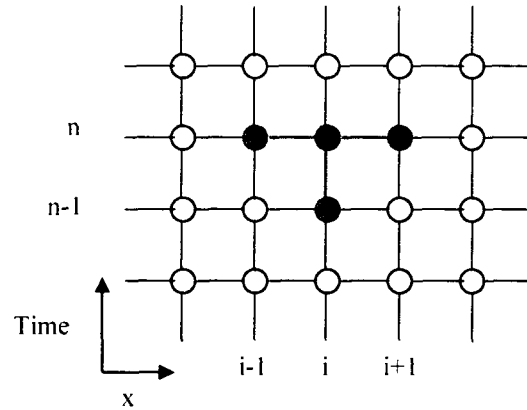


Figure 3-3 The finite Difference nodes associated with the simple implicit method

In this method, at each time step (n), set of N equations should be solved. Although the computational process for the implicit scheme is more complicated than the explicit method, this scheme has a great advantage of being unconditionally stable. Considering the fact that both methods are of the same accuracy, simple implicit method allows an open choice of Δt for a little extra computational effort.

Both implicit and explicit methods are of the first order of accuracy in time domain. ($O(\Delta t)$). In case where more accurate results are demanded, another scheme of importance is the *Crank-Nicolson* scheme with the temporal error of the second order ($O(\Delta t)^2$). This scheme is actually nothing complicated but taking the average of the right hand side in equations (3.22) and (3.31) from explicit and implicit methods, while keeping the left hand side of implicit equation [71]. Similar to the implicit method, this scheme is unconditionally stable.

There are also other schemes to choose from when it comes to discretizing the heat equation for a specific problem. *Combined* method and *three-time-level* method are two examples from many [71]. It is then a matter of choice and priorities in a problem which results in choosing one of the possible methods over the rest.

3.5 How Precise Are Numerical Methods?

When applying the finite difference scheme to discretize the partial differential equations, there is a variety of methods to choose from e.g. implicit, explicit schemes. However, choosing the proper scheme for a problem is not possible unless the question of its stability, consistency, errors and convergence is answered.

The most evident error involved in the finite difference scheme is the *truncation error* as discussed in section 3.3.1, where the terms of higher order of the variable (e.g. Δx) are neglected. The larger the truncation error, the farther the approximation is from the exact solution.

In the limit where the time and space intervals tend to zero, the numerical solution should approach the analytical solution. If this condition is met by a finite difference scheme, it is *convergent*.

One of the common errors in all computational methods is the *round-off error*. As is suggested by its name, this error is caused by rounding off real numbers. This error is less significant nowadays since the modern computers can store real numbers with up to 12 decimal places.

3.5.1 Stability

If the errors involved in a computational scheme are growing at each step with no bounds, it is said to be *unstable*.

Although the simple explicit method is easy to implement computationally, there are some restrictions to it. The solution to equation (3.24) shows divergence and oscillation for some r values. This is referred to as instability, and the proper value range of r to prevent it is:

$$r \leq \frac{1}{2} \tag{3.29}$$

Given that $r = \frac{\alpha \Delta t}{(\Delta x)^2}$ (Eq. 3.25), and α and Δx are specified for a particular heat diffusion problem, Δt can only get limited range of value in order to meet the condition in (3.29). In a problem where the temperature is studied over a long time, either longer time steps or larger number of them should be used. Equation (3.29) restricts the maximum length of the time steps so that the calculations should be performed over a larger number of steps.

This implies the increase in the number of iterations needed which is not favorable.

However, there is not such a restriction in implicit scheme. In order to prove this point there is an analytical way of studying the stability of this technique called the *Fourier method*.

3.5.2 Fourier Method of Stability Analysis for Implicit Scheme

Assuming that ϕ_E is the exact solution to a partial differential equation and ϕ_N is its numerical solution [71], the error of the computational method used is the difference of between these two solutions [71].

$$\varepsilon = \phi_N - \phi_E \quad (3.30)$$

Now let's consider implicit method in which the time and spatial derivatives are replaced by the backward and central differences respectively:

$$\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} = \alpha \frac{\phi_{i-1}^{n+1} - 2\phi_i^{n+1} + \phi_{i+1}^{n+1}}{(\Delta x)^2} \quad (3.31)$$

For small values of Δt and Δx , ϕ_E satisfies this equation already, so that according to equation (3.30), it results in:

$$\frac{\varepsilon_i^{n+1} - \varepsilon_i^n}{\Delta t} = \alpha \frac{\varepsilon_{i-1}^{n+1} - 2\varepsilon_i^{n+1} + \varepsilon_{i+1}^{n+1}}{(\Delta x)^2} \quad (3.32)$$

Assume that the errors introduced to a numerical solution at the beginning of the time, are expressed as complex exponentials

$$\varepsilon(i\Delta x, n\Delta t) \equiv \varepsilon_j^n = e^{jn\Delta t} \cdot e^{j\beta_m i\Delta x} \quad (3.33)$$

Here $j=\sqrt{-1}$. β_m is the Fourier mode, γ is a general complex quantity and n and i represent the time and space step numbers. Let's assume:

$$\xi = e^{\gamma\Delta t} \quad (3.34)$$

Then (3.33) becomes:

$$\varepsilon_i^n = \xi^n \cdot e^{j\beta_m i\Delta x} \quad (3.35)$$

Introducing the Fourier definition of the error in equation (3.32), it becomes:

$$\xi - 1 = \frac{2\alpha\Delta t}{(\Delta x)^2} \xi \left(\frac{e^{j\beta_m\Delta x} + e^{-j\beta_m\Delta x}}{2} - 1 \right) \quad (3.36)$$

Considering that:

$$\cos(\beta_m\Delta x) = \frac{e^{j\beta_m\Delta x} + e^{-j\beta_m\Delta x}}{2} \quad (3.37)$$

And recalling $r = \frac{\alpha\Delta t}{(\Delta x)^2}$, (3.36) is written as:

$$\xi - 1 = 2r\xi(\cos(\beta_m \Delta x) - 1) \quad (3.38)$$

$$\xi - 1 = -4r\xi \sin^2\left(\frac{\beta_m \Delta x}{2}\right) \quad (3.39)$$

$$\xi = \left[1 + 4r \sin^2\left(\frac{\beta_m \Delta x}{2}\right)\right]^{-1} \quad (3.40)$$

Now, looking at equation (3.33) and (3.34), it is evident that the solution will be stable as long as $|\xi| \leq 1$. It is evident that this condition is met by all positive values of r which are the only values that it can get according to its definition. Therefore, implicit method is stable no matter what the choices of Δx and Δt are.

3.6 Summary

In this chapter the numerical solution of the heat equation is reviewed. Since finding the exact solution of this partial differential equation is not straightforward in case of different shapes and boundary conditions, the numerical solutions are considered. Finite difference method (FDM) is introduced to replace the partial derivatives with difference formulas. Among various results of applying FDM to the heat equation, two most popular ones are discussed in this chapter: Explicit and Implicit schemes.

Although explicit scheme is mathematically easier to implement, it is only conditionally stable. On the other hand, this is not the case for the implicit method; however a large set of equations should be solved in this method which is the main focus of this thesis. Next chapter includes the suggested algorithm to overcome this difficulty and the obtained results from both simulations and the experiments.

Chapter 4

4 Implementation of Numerical Methods and Results

In this section, explicit and implicit schemes are tested for various initial and boundary conditions and the results are compared for both methods. All the algorithms are written in MATLAB[®] 7.0.4.

4.1 Employing Explicit Scheme in One Dimension

The explicit finite difference codes are tested for one object under three different types of boundary conditions. The method is then employed to 2 dimensions and the stability of this method is studied as well.

The heat source varies from a square pulse as in pulse thermography to modulated heat as in Lock-in thermography.

The main equation to solve in simple explicit method is: (3.24)

$$\phi_i^{n+1} = r\phi_{i+1}^n + (1-2r)\phi_i^n + r\phi_{i-1}^n$$
$$i = 1, 2, \dots, M-1 \quad \text{and} \quad n = 0, 1, 2, \dots$$

This term will provide a set of M-1 equations while there are M+1 unknowns to find ($i=0, 1, \dots, M$). The other two terms depend on the different boundary conditions in each problem.

The algorithm is as follows:

- (1) Define a two dimensional matrix to store the data, the number of rows in this matrix is defined by the number of time steps and it has M columns to store temperature of the cells.
- (2) Start from $n=0$ and use (3.24) to find the temperature ϕ_i^1 for all values of i (1.....M).

- (3) Go to the next time step ($n=1$) and repeat the same process as in (2) and find ϕ_i^2 and store it in the third ($n=2$) row of the profile matrix.
- (4) Keep performing the calculations until the last time step.

4.1.1 Set Temperature at the Boundaries

It is assumed that the temperatures at $i=1$ and $i=M$ are known during the whole process, i.e.:



Set Temperatures at the Boundaries

$$\phi_1^n = T_a \quad (4.1)$$

$$\phi_M^n = T_b$$

These two equations will provide the two missing equations to solve (3.24). The set of M equations to be solved at each time steps can be written in the matrix form:

$$\begin{bmatrix} \phi_1^{n+1} \\ \phi_2^{n+1} \\ \vdots \\ \phi_{M-1}^{n+1} \\ \phi_M^{n+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ r & (1-2r) & r & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & r & (1-2r) & r \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} T_a \\ \phi_2^n \\ \vdots \\ \phi_{M-1}^n \\ T_b \end{bmatrix} \quad (4.2)$$

The first and last row will maintain the temperature of the boundary cells at T_a and T_b at all times.

```

%Explicit method, 1-D, set boundary temperatures, Dirac Delta pulse

clear all
clc

%Define the test problem

iLnth = 200; % the length of the 1-D rod
iTime = 300000; % the number of the time steps in the time of observation
d_x = 0.0001; % the size of the spatial step
d_t = 0.0001; % the duration of the time step
alf = 10^(-6)*ones(iLnth, 1); % the array of thermal diffusivities
r = alf*d_t/(d_x)^2; % define r : thermal_diffusivity*d_t/(d_x)^2
T = zeros(1, iLnth); % the temperature distribution
T(100) = 50; % the heat sources
T_new = zeros(1, iLnth); % temporarily keeps each T row
T_a=20; % boundary temperatures
T_b=20;
i_save_frame = 1000; % the interval of saving the frames
count = 1;

for i = 1 : iTime % the time loop (steps)

    T_new = zeros(1, iLnth);
    T_new(1,1)=T_a;

    for j = 2 : iLnth-1
        T_new(j) = r(j)*T(j - 1) + (1 - 2*r(j)) * T(j) + r(j) * T(j + 1);
    end

    T_new(1,iLnth) = T_b;

    T(1:iLnth) = T_new(1:iLnth); % assigning the newly calculated temperatures to the real array
end

```

Code Listing 4-1 MATLAB codes to solve 1-D linear heat diffusion problem, explicit method, prescribed boundary temperatures.

4.1.2 Boundaries Subjected to Convection

In case where there is heat transfer in form of convection from the first and last cells on the rod, two additional terms would add to equation (3.24) [71].

$$-k \frac{\partial \phi}{\partial x} + h_0 \phi = h_0 T_{\infty,0} \quad (4.3-a)$$

$$k \frac{\partial \phi}{\partial x} + h_L \phi = h_L T_{\infty,L} \quad (4.3-b)$$

Where h_0 and h_L are heat transfer coefficients at both ends of the rod. These two equations show how heat is transferred from the first and cells on the rod to the neighboring environment with the ambient temperatures $T_{\infty,a}$ and $T_{\infty,b}$ respectively. Although the ambient temperatures are known at both ends, the temperature of the cells is not. Adding these two relations to the main heat equation at the boundaries results a set of M equations to solve at each time step. However, the partial differential equations in (4.3) should also be replaced by the difference methods. Using forward differencing method for (4.3-a) gives:

$$-k \left(\frac{\phi_2^n - \phi_1^n}{\Delta x} \right) + h_0 \phi_1^n = h_0 T_{\infty,0} \quad i=1 \quad (4.4-a)$$

And employing backward difference method to discretize (4.4-b) results in:

$$k \left(\frac{\phi_M^n - \phi_{M-1}^n}{\Delta x} \right) + h_L \phi_M^n = h_L T_{\infty,M} \quad i=M \quad (4.4-b)$$

Rearranging these two equations to get the temperature of the first and last cell in terms of the neighboring cells:

$$\phi_1^n = \frac{\gamma_0}{1 + \gamma_0} T_{\infty,0} + \frac{1}{1 + \gamma_0} \phi_2^n \quad (4.5-a)$$

$$\phi_M^n = \frac{\gamma_L}{1 + \gamma_L} T_{\infty,L} + \frac{1}{1 + \gamma_L} \phi_{M-1}^n \quad (4.5-b)$$

Where

$$\gamma_0 = \frac{h_0 \Delta x}{k} \quad \text{and} \quad \gamma_L = \frac{h_L \Delta x}{k}$$

It should be mentioned that the accuracy of this equation is also from the first order in space, i.e. $O(\Delta x)$, as the forward/backward difference methods used to discretize the partial differential terms.

To implement this method, there should be a minor change in the algorithm mentioned previously.

- (1) Define a two dimensional matrix to store the data, the number of rows in this matrix is defined by the number of time steps and it has M columns to store temperature of the cells.
- (2) The temperature at all points is known at the beginning ($n=0$), for $n=1$ the loop starts from $i=2$ and goes on to $n=M-1$, using equation (3.24) to find the temperature at each node.
- (3) While still on the same time step (i.e. $n=1$), the temperature on the first and last spatial nodes are defined using the set of equations (4.5-a) and (4.5-b).
- (4) Repeat the steps (2) and (3) for the next time step $n=2$ and save the information in the third row of the matrix defined in the first step.
- (5) Repeat this method until the last time step is reached.

Code listing 4-2 is the implementation of this scheme in terms of MATLAB codes.

```

%Explicit method, 1-D, convection at the boundaries, Dirac Delta pulse

clear all
clc

%Define the test problem

iLnth = 200; % the length of the 1-D rod
iTime = 300000; % the number of the time steps in the time of observation
d_x = 0.0001; % the size of the spatial step
d_t = 0.0001; % the duration of the time step
alf = 10^(-6)*ones(iLnth, 1); % the array of thermal diffusivities
r = alf*d_t/(d_x)^2; % define r : thermal_diffusivity*d_t/(d_x)^2
T = zeros(1, iLnth); % the temperature distribution
T(100) = 50; % the heat sources
h0=50; % the heat transfer coefficients at the boundaries
hL=50;
T_amb_0=10; %the ambient temperatures at the boundaries
T_amb_L=30;
Gamma0=h0*d_x/k;
GammaL=hL*d_x/k;
T_new = zeros(1, iLnth); % temporarily keeps each T row
i_save_frame = 1000; % the interval of saving the frames
count = 1;

for i = 1 : iTime % the time loop (steps)

    T_new = zeros(1, iLnth);

    for j = 2 : iLnth-1
        T_new(j) = r(j)*T(j - 1) + (1 - 2*r(j)) * T(j) + r(j) * T(j + 1);
    end

    T_new(1,iLnth)=(GammaL/(1+GammaL))*T_amb_L+(1/(1+GammaL))*T_new(iLnth-1);
    T_new(1,1)=(Gamma0/(1+Gamma0))*T_amb_0+(1/(1+Gamma0))*T_new(2);
    T(1:iLnth) = T_new(1:iLnth); % assigning the newly calculated temperatures to the real array

end

```

Code Listing 4-2 MATLAB Codes to Implement Explicit Scheme in 1D, Boundaries subjected to convection, Pulse Heating.

4.1.3 Boundaries Subjected to a Prescribed Flux

Another possible boundary condition in case of a 1-D rod where a known heat flux is applied at the boundaries at all times [71].

$$-k \frac{\partial \phi}{\partial x} = q_0 = \text{Known} \quad x=0 \quad (4.6-a)$$

$$k \frac{\partial \phi}{\partial x} = q_L = \text{Known} \quad x=L \quad (4.6-b)$$

A comparison between the previously discussed “known convection boundary condition” and this one shows that the latter is obtained by substituting the heat transfer coefficients in equation (4.3) by zero. In order to employ finite difference method to discretize this term, the same method as in previous section can be used with zero values for h_0 and h_L . The two boundary equations then will be obtained as:

$$\phi_1^n = \frac{q_0 \Delta x}{k} + \phi_2^n \quad x=0 \quad (4.7-a)$$

$$\phi_M^n = \frac{q_L \Delta x}{k} + \phi_{M-1}^n \quad x=L \quad (4.7-b)$$

It is worth to be noted that the accuracy of this method is also of the first order in space $O(\Delta x)$. The algorithm to implement the explicit scheme to this problem is precisely the same one used for the prescribed convection boundary condition. The code listing presented below is the MATLAB codes for a 1-D sample with the known heat flux at the boundaries.

```

%Explicit method. 1-D. Known Heat Flux at the boundaries, Dirac Delta pulse

clear all
clc

%Define the test problem

iLnth = 200; % the length of the 1-D rod
iTime = 300000; % the number of the time steps in the time of observation
d_x = 0.0001; % the size of the spatial step
d_t = 0.0001; % the duration of the time step
alf = 10^(-6)*ones(iLnth, 1); % the array of thermal diffusivities
r = alf*d_t/(d_x)^2; % define r : thermal_diffusivity*d_t/(d_x)^2
T = zeros(1, iLnth); % the temperature distribution
T(100) = 50; % the heat sources
q0 = 40; % the heat flux at the boundaries
qL = 60;
k =
T_new = zeros(1, iLnth); % temporarily keeps each T row
i_save_frame = 1000; % the interval of saving the frames
count = 1;

for i = 1 : iTime % the time loop (steps)

    T_new = zeros(1, iLnth);

    for j = 2 : iLnth-1
        T_new(j) = r(j)*T(j - 1) + (1 - 2*r(j)) * T(j) + r(j) * T(j + 1);
    end

    T_new(1,iLnth) = qL*d_x/k+T_new(iLnth-1);
    T_new(1,1) = q0*d_x/k+T_new(2);
    T(1:iLnth) = T_new(1:iLnth); % assigning the newly calculated temperatures to the real array

end

```

Code Listing 4-3 MATLAB codes to solve 1-D linear heat diffusion problem, explicit method, prescribed heat flux at the boundaries.

4.1.4 Explicit Scheme with Lock-in Thermography

In order to benefit from the advantages of lock-in thermography in the explicit method, the process can be repeated utilizing modulated heat source instead of a Dirac delta pulse. In practice the difference in implementing a modulated heat pulse is in defining the temperature distribution at the beginning of the experiment, i.e. $n=0$.

The following code listing (Code Listing 4-4) is an example of a 1-D rod subjected to a sinusoidal pulse.

```
%Explicit Method, 1-D, Sinusoidal Heat Pulse, Known Temperature at the Boundaries.

clear all
clc

% Defining the problem

iLnth = 200; % the number of the cells in the rod
iTime = 30000; % the number of the time steps in the time of observation
d_x = 0.0001; % the size of the spatial step
d_t = 0.0001; % the duration of the time step

i_save_frame = 1000; % the interval of saving the frames
alf = 10^(-6)*ones(iLnth, 1); % the array of thermal diffusivities
r = alf*d_t/(d_x)^2; % define r : thermal_diffusivity*d_t/(d_x)^2
T_a=20; % boundary temperatures
T_b=20;

% the program execution

T = zeros(1, iLnth ); % the temperature distribution
T_new = zeros(1, iLnth+2); % temporarily keeps each T row
f=3; %Frequency of the source
S_Amp=700; %Amplitude at the source
el_No=20; % the element under investigation
for i = 1 : iTime % the time loop (steps)

    T(1,2:6)=S_Amp*sin(f*2*pi*(i-1)/iTime);
    T_new(1,1)=T_a;
    for j = 2 : iLnth-1 % the coordinate loop
        T_new(j) = r(j)*T(j - 1) + (1 - 2*r(j)) * T(j) + r(j) * T(j + 1);
    end
    T_new(1,iLnth) = T_b;
    for k = 1 : iLnth+2
        T(k) = T_new(k); % assigning the newly calculated temperatures to the real array
    end
end
```

Code Listing 4-4 MATLAB codes to implement a sine shape pulse to a 1-D heat diffusion problem.

4.2 Simple Explicit Method in 2-D Diffusion

Although multidimensional linear diffusion problem can be solved for the simple shapes such as a rectangle, cylinder and sphere using analytical methods, for bodies with irregular shapes, complicated boundary conditions or temperature dependent properties (e.g. diffusivity), it is easier to solve the problem with numerical method [71].

In this section the generalization of the finite difference method in solving a 2-D parabolic system is to be studied. Parabolic systems are chosen since they cover a vast range of the heat transfer problems. Heat/mass diffusion in solids, boundary layer type heat transfer problems and transient temperature diffusion problems are a few examples of the parabolic systems.

Two-dimensional linear heat diffusion equation is given by:

$$\frac{\partial \phi}{\partial t} = \alpha \left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} \right) \quad (4.7)$$

Where as in the one dimensional case, α represents the thermal diffusivity. However, the temperature is a function of time and both x and y directions ($\phi(x,y,t)$). As for the explicit method, the time derivative is replaced by the *forward* difference and the spatial derivatives are replaced by *central* differences.

$$\frac{\phi_{i,j}^{n+1} - \phi_{i,j}^n}{\Delta t} = \alpha \frac{\phi_{i-1,j}^n - 2\phi_{i,j}^n + \phi_{i+1,j}^n}{(\Delta x)^2} + \alpha \frac{\phi_{i,j-1}^n - 2\phi_{i,j}^n + \phi_{i,j+1}^n}{(\Delta y)^2} + O(\Delta t) + O(\Delta x)^2 + O(\Delta y)^2 \quad (4.8)$$

Here $\phi_{i,j}^n$ represents the temperature at the spatial coordinates (i,j). Rearranging this equation in terms of the temperature at (i,j) at a later time step (n+1) provides:

$$\phi_{i,j}^{n+1} - \phi_{i,j}^n = r_x (\phi_{i-1,j}^n - 2\phi_{i,j}^n + \phi_{i+1,j}^n) + r_y (\phi_{i,j-1}^n - 2\phi_{i,j}^n + \phi_{i,j+1}^n) \quad (4.9)$$

Where

$$r_x = \frac{\alpha \Delta t}{(\Delta x)^2} \quad \text{and} \quad r_y = \frac{\alpha \Delta t}{(\Delta y)^2} \quad (4.10)$$

The spacing on each x and y axis is a matter of user's choice. To simplify the computational process, the x and y discrete intervals may be chosen the same values without affecting the generality of the solution. Therefore equation (4.9) becomes:

$$\phi_{i,j}^{n+1} = r(\phi_{i-1,j}'' + \phi_{i+1,j}'' + \phi_{i,j-1}'' + \phi_{i,j+1}'') + (1 - 4r)\phi_{i,j}'' \quad (4.11)$$

Where $r=r_x=r_y$.

Figure (4.1) illustrates how the explicit scheme works for 2-D problems. The temperature of a point on the 2-D plane at a later time step may be found knowing the temperature of the same point and its four neighbors at the present time step.

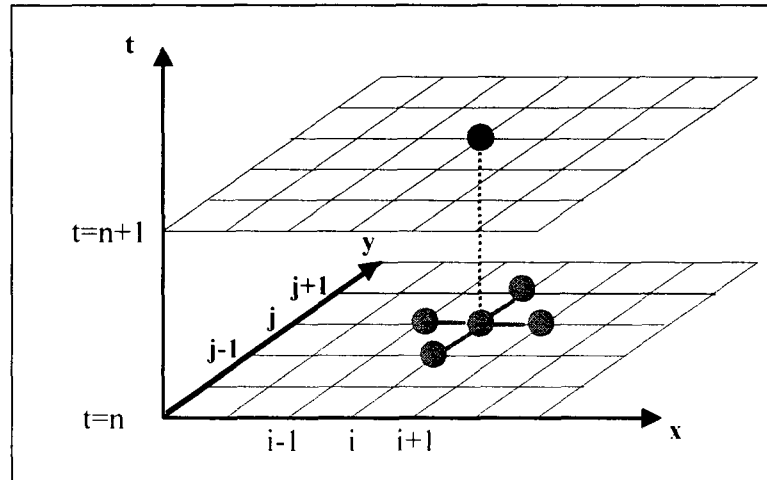


Figure 4-1 The 2-D finite difference nodes associated with the simple explicit scheme.

The MATLAB codes to implement this scheme in case of a flat surface with known temperature at the boundaries subjected to a square heat pulse.

```
% Explicit scheme, 2-D. known temperature at the boundaries, square heat pulse

clear all
clc

% Test Problem

iLnth = 100;           % the number of cells in the rod (x axis)
jLnth = 65;           % the number of cells in the rod (y axis)
iTime = 30000;        % the number of the time steps in the time of observation
► T_a = 40;           % the prescribed temperature at the boundaries (East,West)
► T_b = 50;

% This model works as long as we assume that d_x = d_y, otherwise it'll be more complicated

d_x = 0.001;          % the size of the spatial step (x axis)
d_y = 0.001;          % the size of the spatial step (y axis)
d_t = 0.005;          % the duration of each time step
alf = 10^(-5)*ones(jLnth,iLnth); % the array of thermal diffusivities
r = alf*d_t/(d_x)^2; % define r : thermal_diffusivity*d_t/(d_x)^2 , d_x = d_y;

% the program execution

T = zeros(jLnth , iLnth ); % the temperature distribution
► T(10:20 , :) = 10000; % the heat source
T_new = zeros(jLnth , iLnth); % temporarily keeps each T distribution

for t = 1 : iTime % the time loop (steps)
    for j = 2 : jLnth-1 % the coordinate loop
        for i = 2 : iLnth-1
            T_new(j,i) = r(j,i)*(T(j - 1,i) + T(j + 1,i) + T(j,i - 1) + T(j,i + 1)) + (1 - 4*r(j,i)) * T(j,i);
        end

        ► T_new(j,iLnth) = T_a; % known temperature at East
        ► T_new(j,1) = T_b;

    end

    for j = 1 : jLnth % assigning the newly calculated temperatures to the real array
        for i = 1 : iLnth
            T(j,i) = T_new(j,i);
        end
    end
end
```

Code Listing 4-5 MATLAB codes to implement the explicit scheme to a 2-D surface.

A modulated heat wave may replace the square pulse by defining the heat source as a sinusoidal wave initially generated on x or y axis. In the previous example, the ▲ line should be replaced by:

```
T(:,2:4) = 700*sin(t*20*pi/iTime);
```

This change will represent a sinusoidal heat wave with the initial amplitude of magnitude 700° and the frequency of $(1/iTime)$ second⁻¹ and excites the entire second to forth rows of the surface at the beginning.

Also there is no restriction on the boundary conditions in this 2-D generalization since to apply a different boundary condition to this problem, one may only replace the lines ► by the type of boundary conditions discussed in sections 4.1.2 and 4.1.3.

To generalize the heat diffusion problem to a layered sample, the only change in this method is to consider different thermal diffusivities on it. Code listing 4-6 represents MATLAB codes for a 2 layer sample.

```

%%2 layers- 2 dimension
%% parameters

iLnth = 100;    % the number of cells in the rod (x axis)
jLnth1 = 30;    % the number of cells in the rod (y axis)----first layer
jLnth2 = 30;    % the number of cells in the rod (y axis)----second layer
jLnth = jLnth1 + jLnth2;
iTime = 50000;  % the number of the time steps in the time of observation
% This model works as long as we assume that d_x = d_y, otherwise it'll be more complicated
d_x = 0.001;    % the size of the spatial step (x axis)
d_y = 0.001;    % the size of the spatial step (y axis)
d_t = 0.0003;   % the duration of each time step

defect1_x1 = 30; % the coordinates of the point where the defect starts
defect1_y1 = 20;
defect1_x2 = 33; % the coordinates of the point where the defect ends
defect1_y2 = 40;
defect2_x1 = 40;
defect2_y1 = 36;
defect2_x2 = 45;
defect2_y2 = 55;

alfa_1 = 10^(-6); % alfa1 = first layer
alfa_2 = 10^(-4); % alfa2 = second layer
defect1_alf = 1*10^(-10);
defect2_alf = 1*10^(-10);

alf = ones(jLnth,iLnth); % the array of thermal diffusivities
alf ( 1 : jLnth1 , :) = alfa_1;
alf ( jLnth1+1 : jLnth , :) = alfa_2;
% alf(defect1_y1 : defect1_y2 , defect1_x1 : defect1_x2) = defect1_alf; % there is a change of diffusivity
% where the defect is located
% alf(defect2_y1 : defect2_y2 , defect2_x1 : defect2_x2) = defect2_alf; % there is a change of diffusivity
% where the defect is located

r = alfa*d_t/(d_x)^2; % define r : thermal_diffusivity*d_t/(d_x)^2 , d_x = d_y;
i_save_frame = 100; % the interval of saving the frames

%% the program execution
T = zeros(jLnth , iLnth); % the temperature distribution (+ 2 'virtual' rows + 2 'virtual' columns on the
sides)

T(30,:) = 10000; % the heat source
T_new = zeros(jLnth, iLnth); % temporarily keeps each T distribution
count = 1;

for t = 1 : iTime % the time loop (steps)

    for j = 2 : jLnth % the coordinate loop (includes the 'real' cells only - i.e.. from the 2nd to the prelast
        for i = 2 : iLnth
            T_new(j,i) = r(j,i)*(T(j - 1,i) + T(j + 1,i)+ T(j,i - 1)+T(j,i + 1)) + (1 - 4*r(j,i)) * T(j,i);
        end
    end

    for j = 1 : jLnth % assigning the newly calculated temperatures to the real array

        for i = 1 : iLnth
            T(j,i) = T_new(j,i);
        end
    end
end

```

Code Listing 4-6 Explicit method for a 2 layers sample in 2-D

4.3 Implicit Scheme in 1-D

Mathematical implementation of the implicit scheme is more complicated than the explicit scheme since here a set of equations should be solved at each time step in order to get the temperature of the nodes on it.

To get a clear image of the system of equations to be solved, it's better to rearrange equation (3.32) into the matrix form.

$$\frac{1}{\Delta t} \phi_i^{n+1} = -\frac{\alpha}{(\Delta x)^2} \phi_{i-1}^n + \left(\frac{1}{\Delta t} + \frac{2\alpha}{(\Delta x)^2} \right) \phi_i^n - \frac{\alpha}{(\Delta x)^2} \phi_{i+1}^n$$

$$\phi_i^n = \phi_i^{n+1} (1 + 2r) - r\phi_{i-1}^{n+1} - r\phi_{i+1}^{n+1} \quad (4.12)$$

Before arranging the matrix form of the equation above, the boundary conditions should be resolved. It is assumed that there is not a temperature gradient at the boundaries, meaning zero heat flux:

$$\left. \frac{\partial \phi}{\partial x} \right|_{x=L} = 0 \quad (4.13)$$

From the *forward difference* we have:

$$\frac{\phi_{M+1} - \phi_M}{\Delta x} = 0 \quad \phi_{M+1} = \phi_M \quad (4.14)$$

where M is the last spatial step. It is assumed that there is an "invented" temperature point at the extended region after the last point. In order for the equation (4.12) to satisfy the boundary condition:

$$\phi_M^n = \phi_M^{n+1} (1 + 2r) - r\phi_{M-1}^{n+1} - r\phi_{M+1}^{n+1} \quad (4.15)$$

This becomes:

$$\phi_M^n = \phi_M^{n+1} (1 + r) - r\phi_{M-1}^{n+1} \quad (4.16)$$

The same process for the $x=0$ boundary condition results in:

$$\phi_1^n = \phi_1^{n+1}(1+r) - r\phi_2^{n+1} \quad (4.17)$$

The matrix form of the implicit method is then defined:

$$\begin{bmatrix} 1+r & -r & 0 & 0 & 0 & 0 \\ -r & 1+2r & -r & 0 & 0 & 0 \\ 0 & -r & 1+2r & -r & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & -r & 1+2r & -r \\ 0 & 0 & 0 & 0 & -r & 1+r \end{bmatrix} \begin{bmatrix} \phi_1^{n+1} \\ \phi_2^{n+1} \\ \phi_3^{n+1} \\ \vdots \\ \vdots \\ \phi_{M1}^{n+1} \end{bmatrix} = \begin{bmatrix} \phi_1^n \\ \phi_2^n \\ \phi_3^n \\ \vdots \\ \vdots \\ \phi_{M1}^n \end{bmatrix} \quad (4.18)$$

The algorithm that should be used is as the following:

(1) Starting from $n=0$, ϕ_i^0 is known for all i values. Therefore the matrix problem to be solved can be simplified as:

$$AX = B \quad (4.19)$$

(2) Once X (the temperature distribution at $n=1$) is found, it acts as the input data (B) for the next time step.

(3) Going to $n=1$, another matrix equation of type (4.19) is solved for known values of ϕ_i^1 , and this way set of ϕ_i^2 is found.

(4) Repeat the same process and it is completed when $n=M-1$.

The matrix of the coefficients (A) in the equation (4.18) is called a *tridiagonal* matrix in linear algebra since only the main diagonal and the first immediate diagonals above and below the main one have nonzero values.

In order to implement this scheme in MATLAB codes, separate functions are defined for each purpose from constructing the sample and all its properties to solving the tridiagonal matrix equation and drawing the plots.

```

function Tobj=make_obj(a,b,c,e,f,g)

Tobj.iLnth=a;
Tobj.d_x=b;
Tobj.d_t=c;

Tobj.alpha=10^(-6)*ones(a,1);

Tobj.T_H=e;
Tobj.T_R=f;
Tobj.r=Tobj.alpha*Tobj.d_t/(Tobj.d_x)^2;
Tobj.pulse_duration=g;

```

Code Listing 4-7 MATLAB function to make the sample under study

The properties involved in making a computational object are its thermal diffusivity, the duration of a time step and the length and number of the spatial steps. The next step is to build the coefficient matrix for the defined object.

```

function M=make_matrix(Tobj)

M.diag=zeros(Tobj.iLnth,1);           %Diag
M.lower=zeros(Tobj.iLnth-1,1);        %Lower diag
M.upper=zeros(Tobj.iLnth-1,1);        %Upper diag

for i=1:Tobj.iLnth-1

    M.upper(i)=-Tobj.r(i+1);
    M.diag(i)=1+Tobj.r(i)+Tobj.r(i+1);
    M.lower(i)=-Tobj.r(i+1);

end

M.diag(1)=1+Tobj.r(2);
M.diag(Tobj.iLnth)=1+Tobj.r(Tobj.iLnth);

```

Code Listing 4-8 MATLAB function to build the matrix of coefficients.

Now to solve the matrix equation in (4.18), the function *solve-tridiag* is introduced. The input to this function is the four arrays that are known to the problem, which are the main, upper, lower diagonals of the coefficients matrix (A) plus the temperature distribution of the sample at the previous time step (B). The *Gaussian elimination* is used to solve this matrix equation.

```
function x = solve_tridiag(a,b,c,d)

% x = solve_tridiag(a,b,c,d) solves A*x = d for x, where a is the diagonal, b is the upper diagonal, and c is
% the lower diagonal of A.

% NOTE:
% size(a)=size(d)=n
% size(b)=size(c)=n-1

% Initialization
m = zeros(size(a));
l = zeros(size(c));
y = zeros(size(d));
n = size(a,1);

%1. LU decomposition
m(1,:) = a(1,:);

y(1,:) = d(1,:);

%2. Forward substitution (L*y=d, for y)

for i = 2 : n
    i_1 = i-1;
    l(i_1,:) = c(i_1,:)./m(i_1,:);
    m(i,:) = a(i,:) - l(i_1,:).*b(i_1,:);

    y(i,:) = d(i,:) - l(i_1,:).*y(i_1,:); %2. Forward substitution (L*y=d, for y)
end

%3. Backward substitutions (U*x=y, for x)

x(n,:) = y(n,:)./m(n,:);

for i = n-1 : -1 : 1
    x(i,:) = (y(i,:) - b(i,:).*x(i+1,:))./m(i,:);
end
```

Code Listing 4-9 MATLAB code of a function to solve the matrix equation of type (4.18).

```

function Temp_el=T_Evolve(Tobj,M,iTime,el_no)
% function Temp=T_Evolve(Tobj,M,iTime)

T = zeros(Tobj.iLnth,1);           % the temperature distribution
T_new = zeros(Tobj.iLnth,1);       % temporarily keeps each T row

T_new=ones(Tobj.iLnth,1)*Tobj.T_R;
T_new(1)=Tobj.T_H;

d=T_new;

% Temp=zeros(iTime,Tobj.iLnth);    % since keeping temp of the rod at every time step takes lot of memory
% Temp is a matrix that keeps the info at
% some time intervals defined in the code.

Temp_el=zeros(iTime,1);           % holds the temperature of el_no'th element for at all the moments

for i = 1 : iTime                  % the time loop (steps)

    % Temp(i,:)=T_new;

    if i*Tobj.d_t<Tobj.pulse_duration
        d=T_new;
    else
        T_new=solve_tridiag(M.diag,M.upper,M.lower,d);
        d=T_new;
        Temp_el(i)=T_new(el_no);
    end

end
end

```

Code Listing 4-10 MATLAB codes for implicit method in solving the 1-D heat diffusion problem

It is to be noted that this simulation system suggests that the matrix equation $Ax=B$ should be solved for each time step separately. With the number of time steps from the order of 100000, it would take a lot of memory to keep the temperature distribution of the 1-D rod after each round. To avoid this problem, it is suggested that instead of saving the data from all the time steps, it should be done at larger time intervals, for example every 100 time steps.

In order to have an easier access to the code and get the results, this scheme is set up on the GUI (Graphic User Interface) format that makes it easier to change the parameters in a problem and see the resulting plot.

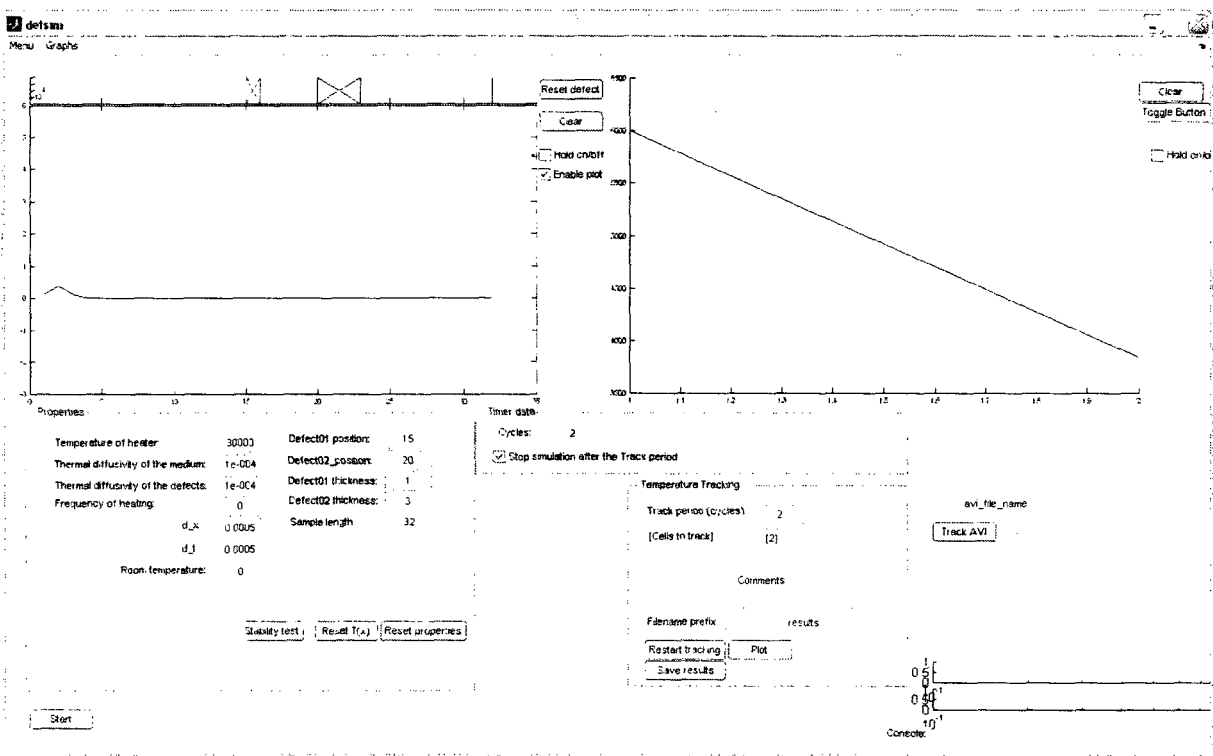


Figure 4-2 The GUI screenshot

The program is designed in away that the user can define if the heat source is a simple square pulse heat or a modulated one. In case of the square heat pulse, the duration of the pulse is adjustable as well. If the heat source used is a modulated wave, the frequency of the heat pulse can be defined in this format. Other parameters to determine are the time and space intervals (d_t , d_x).

4.3.1 Implementing the Defect in a Sample

In case of dealing with a defective sample, the thermal diffusivity of the sample is not constant through out the sample but varies within the sample from point to point. The idea for the simulation to include the imperfections in the sample is to consider the thermal diffusivity of the material as a function of the location in the sample (i.e. $\alpha(i)$), hence results in $r(i)$ in Eq. (4.18). This program (Fig. 4-2) allows simulating the heat diffusion in a homogeneous material with up to two imperfections inside. The location and size of the defects may be determined by the user as well as their thermal diffusivity.

Generally two types of results are sought from this scheme:

- (1) The temperature evolution of a specific element.
- (2) The temperature distribution on the 1-D rod at a specific time step.

The plot on the left hand side of Fig 4-2 is the real time representation of the temperature distribution on the 1-D rod. Once a number of time cycles (defined by the user) are finished, on the left hand side of Fig 4-2, the temperature variation of the defined cells versus time is shown. The user can choose to track up to several cells for their temperature variation during the time of the experiment.

Once the plot is obtained, the Flash method (section 2.5.2) is used to determine the thermal diffusivity of the sample. Parker's Flash method of thermal diffusivity determination either uses the half time, $t_{1/2}$ which is the time required for the back surface temperature to reach half of the maximum temperature or t_x that is the time axis intercept of the temperature versus time curve. In order to determine a reliable method, both these methods are utilized to obtain the thermal diffusivity for a sample and the result is presented. (Fig 4-3)

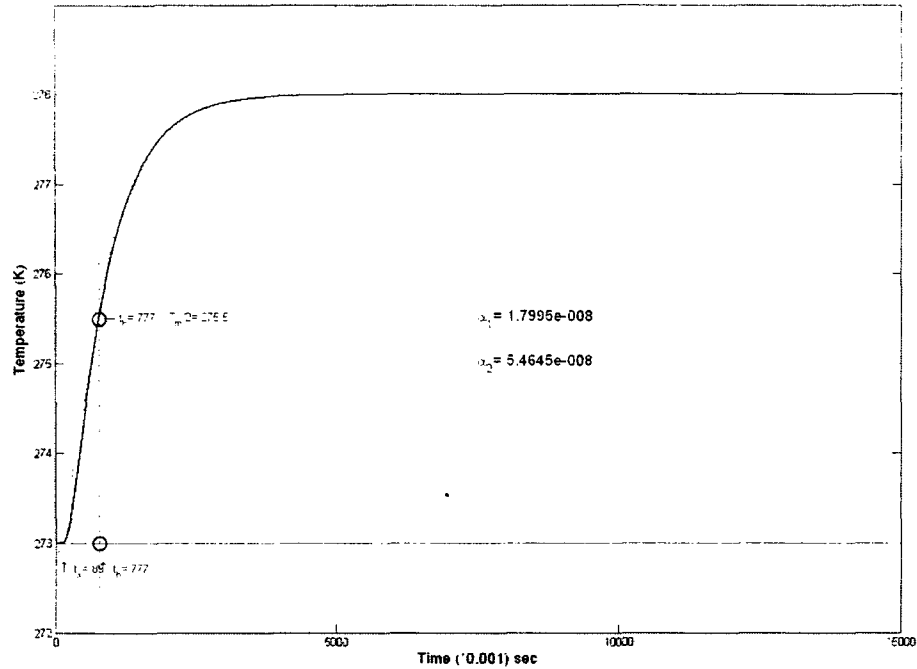


Figure 4-3 Flash Method. Temperature vs. Time at the rear surface. Heater Temperature: 373 (K), Thermal diffusivity: $10e-8(m^2/s)$.

Figure 4-3 represents a typical measurement of the thermal diffusivity. α_1 and α_2 result from using *half-time* and *intercept* methods respectively.

In order to find the optimum conditions for using the Flash method, a series of experiments are simulated.

In figure 4-4, the thermal diffusivity is found for samples of different thicknesses. It is evident from this figure that 1-1.5 mm thick samples should be chosen if *half time* method is used (α_1) and 2.5-3 mm thick samples are best to use if the *intercept* method is employed (α_2).

It is to be noted that the real thermal diffusivity value for this sample has been chosen to be $10^{-4} (m^2/s)$.

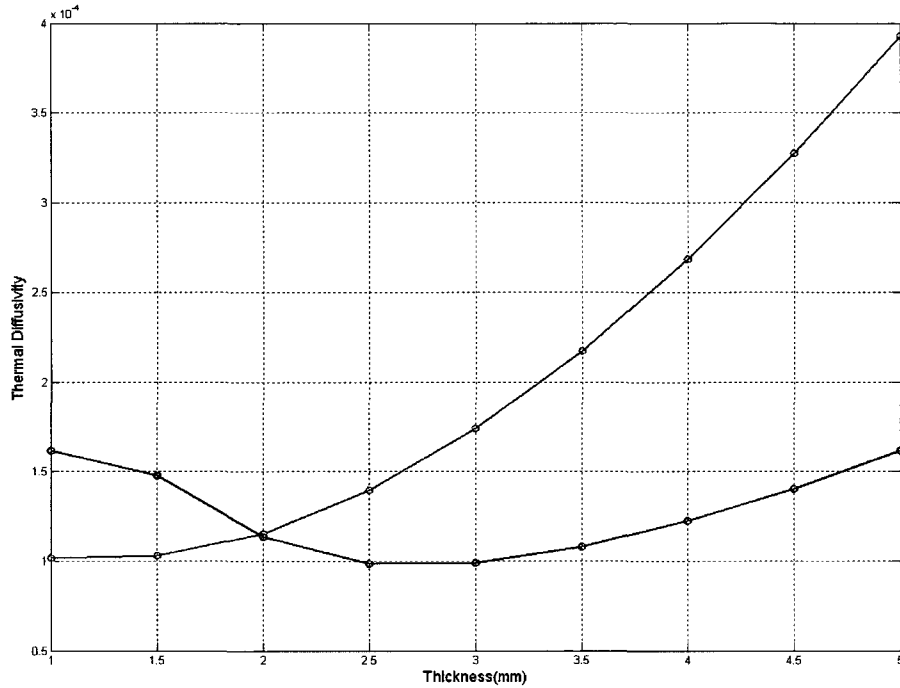


Figure 4-4 Thermal Diffusivity from Flash Method vs. Thickness. (red: α_1 ; black: α_2)

Another significant factor in the experiment is the heat pulse duration. The square thermal pulse generated by the flash heater is available in 5 different pulse durations: [1/1425, 1/900, 1/500, 1/175 (sec)]. The simulation is done for each thermal pulse and the resulting α_1 and α_2 are presented in red and black in figure (4-5.a). Since the real thermal diffusivity employed in this experiment is 10^{-4} (m^2/s), it is evident in this figure that the *half time* method's results are more appealing. The next figure is a closer look at this plot (Fig (4-5.b)). It can be concluded that the longer the pulse duration, the more accurate results are obtained.

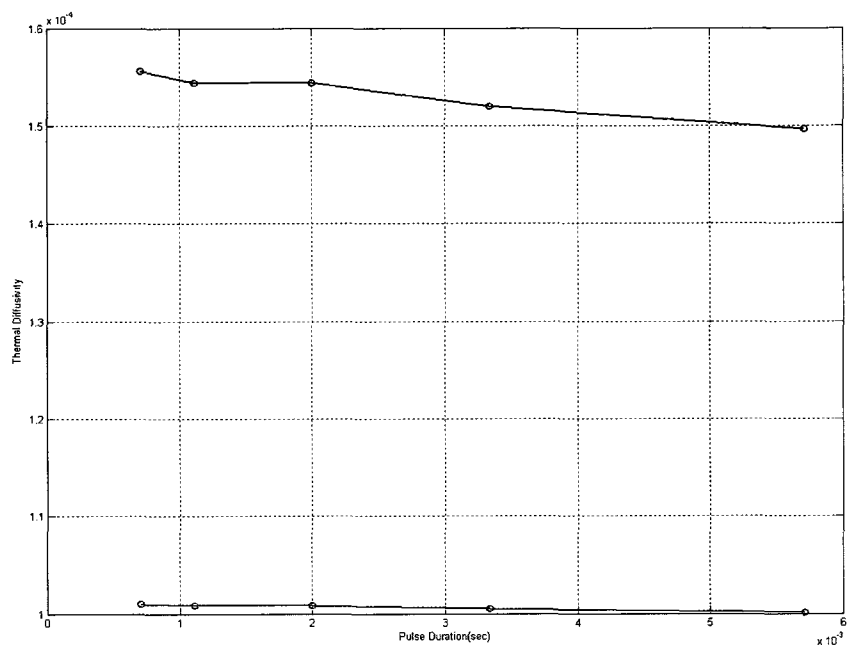


Figure 4-5.a Thermal diffusivity vs. Pulse duration (red: α_1 , black: α_2)

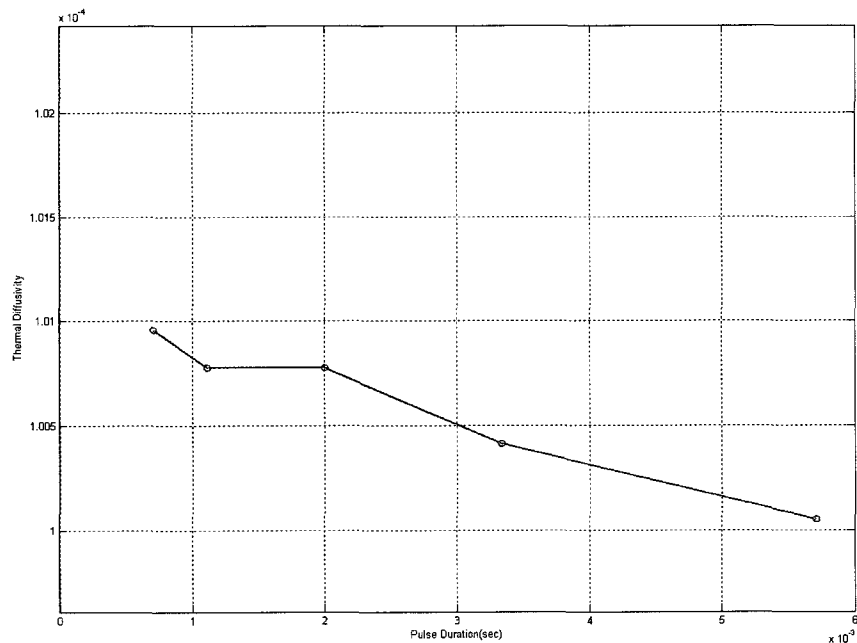


Figure 4-5.b Thermal diffusivity vs. Pulse duration for the *half time* method

4.3.2 Results and Conclusion

With the information from the simulations above, the best possible conditions to run the experiment are as follows:

- employing the *half time* method for finding thermal diffusivity.
- working with longer duration square pulse. (1/175 sec)
- studying on the samples with thickness of 1 to 1.5 mm.

The experiment is done for several types of materials and the results are compared to the real thermal diffusivity values from the text books.

Table 4-1 Thermal Diffusivity (Flash Method)

| Material | Thickness (mm) | Estimated thermal diffusivity (m ² /s) | Thermal diffusivity (textbook) (m ² /s) | Percentage of deviation from the textbook values of α (m ² /s) |
|----------------|-------------------|--|---|---|
| Aluminum | 1.5 | 91.1×10^{-6} | 84.2×10^{-6} | %7.6 |
| Chromium | 1 | 29.6×10^{-6} | 29.2×10^{-6} | %1.3 |
| Copper | 1.5 | 112.8×10^{-6} | 112.3×10^{-6} | %0.4 |
| Lead | 1.5 | 23.6×10^{-6} | 24.1×10^{-6} | %2.1 |
| Silicon Carbon | 1 | 229.6×10^{-6} | 229.7×10^{-6} | %0.04 |
| Wood | 1.5 | 8.5×10^{-8} | 8.2×10^{-8} | %3.5 |

4.3.3 Error Analysis in Thermal Diffusivity by Flash Method

The uncertainty involved in the Flash method is to be found by analyzing the formulas:

$$\alpha_1 = (1.38L^2 / \pi^2 t_{1/2}) \quad (4.20-a)$$

$$\alpha_2 = (0.48L^2 / \pi^2 t_x) \quad (4.20-b)$$

Employing error propagation formula to Eq. (4.20):

$$\frac{\delta\alpha}{\alpha} = \frac{1.38}{\pi^2} \sqrt{\left(\frac{\delta(L^2)}{L^2}\right)^2 + \left(\frac{\delta(t_{1/2})}{t_{1/2}}\right)^2} \quad (4.21)$$

$$\frac{\delta\alpha}{\alpha} = \frac{1.38}{\pi^2} \sqrt{\left(\frac{2\delta(L)}{L}\right)^2 + \left(\frac{\delta(t_{1/2})}{t_{1/2}}\right)^2} \quad (4.22)$$

To find the uncertainty in Eq. (4.20-b), the same process should be repeated:

$$\frac{\delta\alpha}{\alpha} = \frac{0.48}{\pi^2} \sqrt{\left(\frac{2\delta(L)}{L}\right)^2 + \left(\frac{\delta(t_x)}{t_x}\right)^2} \quad (4.23)$$

It is to be noted that the uncertainty in length and time are in fact equal to the smallest intervals chosen in each problem, i.e. d_x and d_t . In the process of finding thermal diffusivity for the materials in table (4-1):

$$d_x = 10^{-3}(\text{m}) \quad \text{and} \quad d_t = 10^{-3}(\text{sec})$$

It can be concluded from equation (4.23) that the relative error in thermal diffusivity is inversely dependent on the sample's thickness. The thicker the specimen and the longer the time duration, the more precise is the estimated thermal diffusivity. Meaning that in practice, a pulse of greater power is needed to stimulate the surface and the detector should take images for a longer period of time.

4.3.4 Experimental Evaluation of Flash Method

4.3.4.1 Instrumentation

The experimental setup includes a flash heater that generates the heat pulse, the sample, the infrared camera to record the thermal evolution of the surface and a computer to analyze the data. Figure (4.6) illustrates the flash heater at the bottom of the picture and the infrared camera above. A single Speedotron 206 VF flash lamp is used connected to a 4803CX power supply with maximum power of 4800W*s. The flash duration was set to 1/175 second. The infrared camera used in this apparatus is a FLIR SC4000 with InSb detector, 320x256 resolution, multiple detector/ ROIC modes and 14-bit digital data. A flat surface with small circular opening acts as a diaphragm to generate a point source to apply on the sample's surface. While the sample faces the flash, the camera records the temperature changes on the rear surface of the sample. Since every millisecond counts in this process, it is essential for the camera and flash heater to start precisely at the same time. A relay system is devised for this purpose. (Fig 4.7)

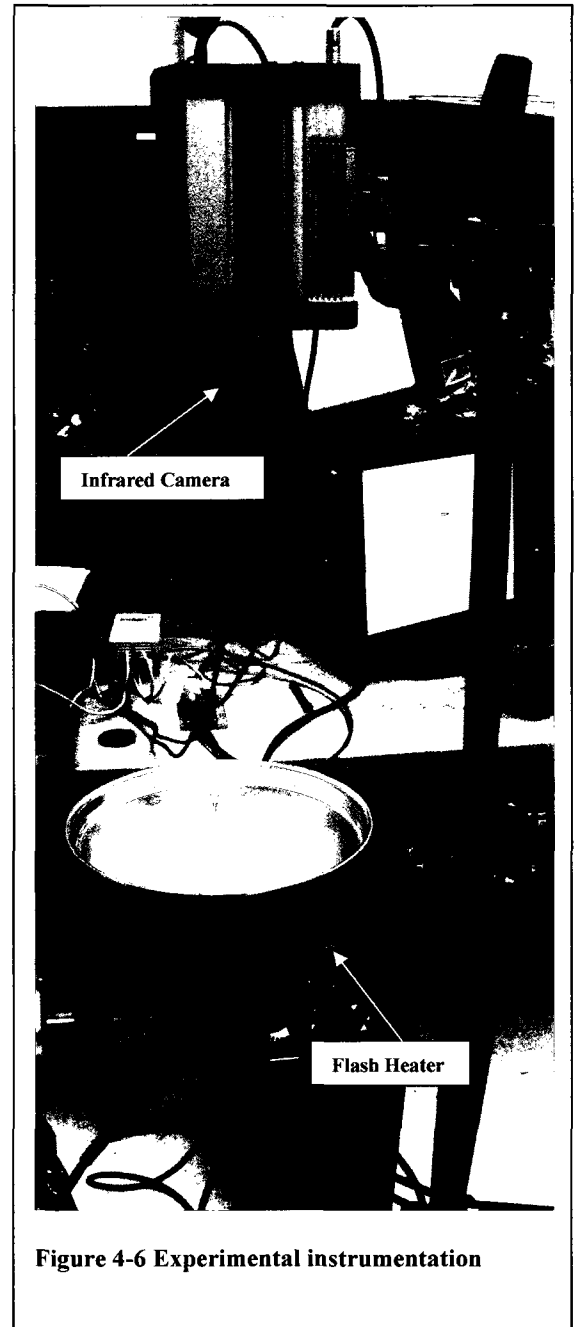


Figure 4-6 Experimental instrumentation

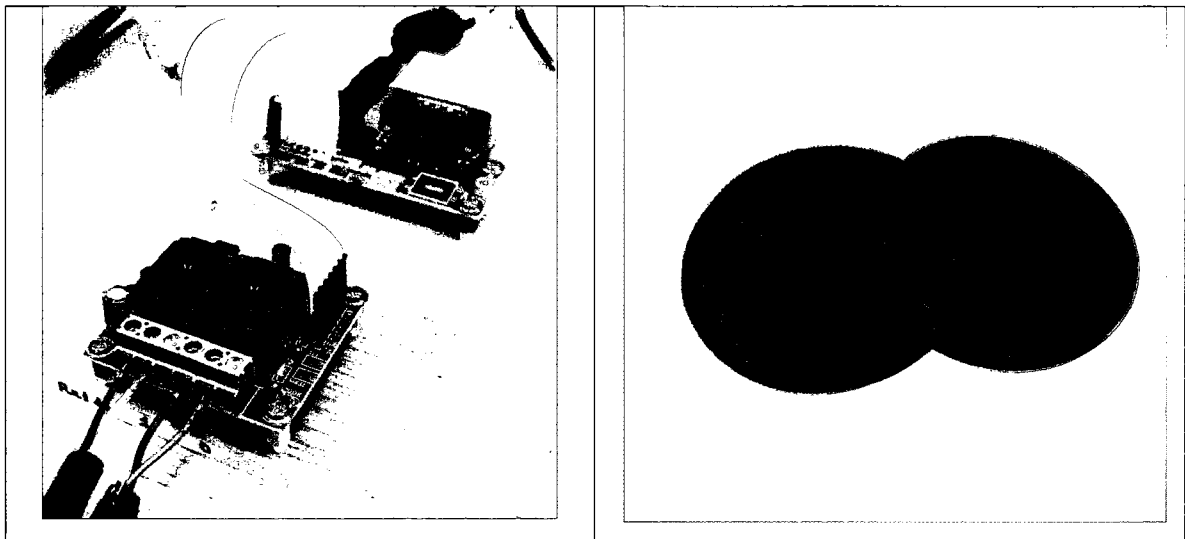


Figure 4-7 The relay system (Left), The Epoxy+Ti₅₀ samples (Right)

There are two relays on this chip where the first one switches the camera on and the second one activates the flash. In practice, to prevent losing the first received data from the sample, it is suggested that the camera starts recording before the heater. The software is designed to make it possible by defining a time delay between the two relays.

The camera then starts to collect images. The user is to determine the number of the images to be taken and the frequency of taking them. The program stores the data and the user can study the behavior of any spot on the sample. Figure (4.9) illustrates the spot under study and the plot at the bottom is its temperature variation over the time of the experiment.

The data is then converted to a two dimensional array of temperature versus time using MATLAB. The maximum temperature of the back surface is then found, and the time when the surface temperature reaches half its maximum value is employed to find thermal diffusivity.

The next section describes the experimental results and the error analysis for this method.

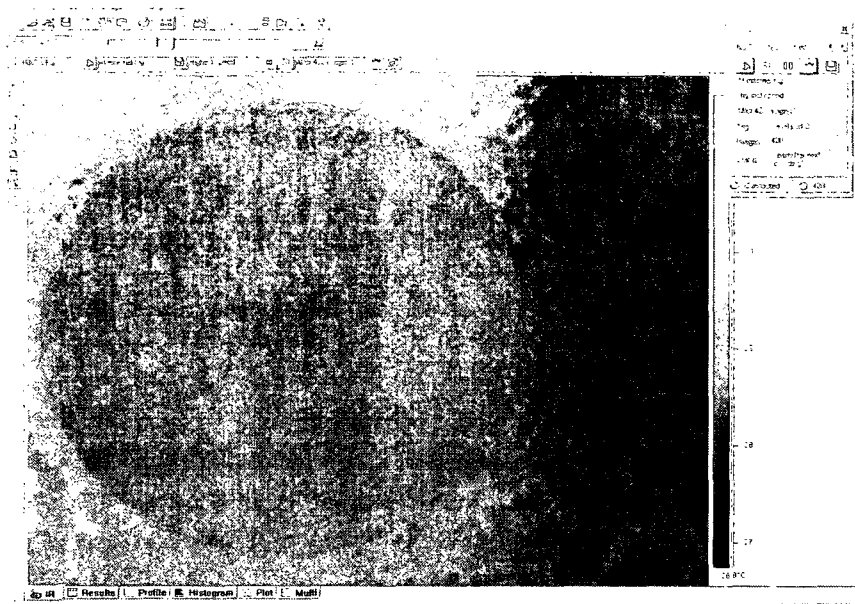


Figure 4-8 The snapshot of the infrared camera record

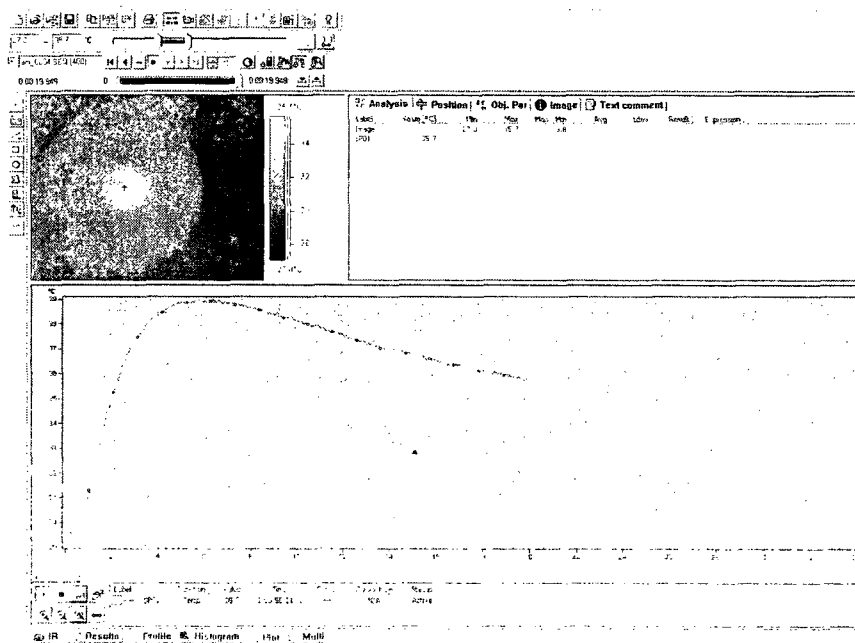


Figure 4-9 The spot on the sample defined for temperature analysis and the plot of its temperature evaluation vs. time

4.3.4.2 Results and Error Analysis

The experiment is done for samples made of Titanium₅₀ powder embedded in Epoxy. The thickness of the samples is selected to be 1mm, 2mm to match the optimum conditions for Flash method discussed in the previous section. The infrared camera is set to take 400 images with the frequency of 10 shots per second. The process is repeated for each sample 20 times, stimulating a different part of the sample with the heat pulse each time. The sample's dimension is large relative to the flash diaphragm, so that it can be assumed that the heat is propagating only in one dimension. The material is homogeneous, hence the total thermal diffusivity is found from the average of thermal diffusivities of each part. The result is presented in table (4.2) and the error analysis is discussed in the following.

Table 4-2 Experimental evaluation of thermal diffusivity by Flash method

| $\alpha(\text{m}^2/\text{s})$ Epoxy+Ti ₅₀ (thickness:1.00mm) | $\alpha(\text{m}^2/\text{s})$ Epoxy+Ti ₅₀ (thickness:2.02mm) |
|--|--|
| 2.5448×10^{-7} | 3.0263×10^{-7} |
| 2.5448×10^{-7} | 3.1103×10^{-7} |
| 2.7993×10^{-7} | 2.9466×10^{-7} |
| 2.5448×10^{-7} | 2.8711×10^{-7} |
| 2.5448×10^{-7} | 2.9466×10^{-7} |
| 2.5448×10^{-7} | 3.0263×10^{-7} |
| 2.5448×10^{-7} | 3.9466×10^{-7} |
| 3.1103×10^{-7} | 3.9466×10^{-7} |
| 3.1103×10^{-7} | 3.9466×10^{-7} |
| 2.7993×10^{-7} | 3.0263×10^{-7} |
| Avg: 2.7088×10^{-7} | Avg: 3.4030×10^{-7} |

To find the error involved in this experiment equation (4.22) should be employed. The reading error in length and time are:

$$\delta L = 0.01 \text{ mm}$$

$$\delta t = 0.05 \text{ sec}$$

Which result from the accuracy of the equipment in the lab. Substituting these values in Eq.(4.22) the error in experimental measurement of thermal diffusivity is:

| Sample Thickness | 1.00 mm | 2.02mm |
|-------------------------|-------------------------|-------------------------|
| $\delta\alpha(m^2 / s)$ | 0.0387×10^{-7} | 0.0486×10^{-7} |
| Relative error | %1.43 | %1.42 |

4.4 Implicit Method, Two Dimensions

4.4.1 Theory

Considering a system with thermal propagation in two directions, the temperature is a function of (x,y). The heat equation is then:

$$\frac{\partial \phi}{\partial t} = \alpha \left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} \right) \quad (4.24)$$

The same as in case of 1-D, the *backward difference* and *central difference* are used to replace the time and spatial partial derivatives respectively:

$$\frac{\phi_{i,j}^{n+1} - \phi_{i,j}^n}{\Delta t} = \alpha \left(\frac{\phi_{i,j-1}^{n+1} - 2\phi_{i,j}^{n+1} + \phi_{i,j+1}^{n+1}}{(\Delta x)^2} + \frac{\phi_{i-1,j}^{n+1} - 2\phi_{i,j}^{n+1} + \phi_{i+1,j}^{n+1}}{(\Delta y)^2} \right) \quad (4.25)$$

Assuming that the space intervals along x and y directions are chosen to be the same size and rearranging equation (4.25) results:

$$\phi_{i,j}^{n+1} - \phi_{i,j}^n = r(\phi_{i,j-1}^{n+1} + \phi_{i,j+1}^{n+1} + \phi_{i-1,j}^{n+1} + \phi_{i+1,j}^{n+1} - 4\phi_{i,j}^{n+1}) \quad (4.26)$$

Where

$$r = \frac{\alpha \Delta t}{(\Delta x)^2} = \frac{\alpha \Delta t}{(\Delta y)^2}$$

$$\phi_{i,j}^n = -r(\phi_{i,j-1}^{n+1} + \phi_{i,j+1}^{n+1} + \phi_{i-1,j}^{n+1} + \phi_{i+1,j}^{n+1}) + (1 + 4r)\phi_{i,j}^{n+1} \quad (4.27)$$

Same as in case of a 1-D rod, here the temperature of a cell on the surface at a time step is related to the temperature of the same cell with its four neighbors at a later time step.

When dealing with a 1-D wire in implicit scheme, x and B involved in the matrix equation (Ax=B) are one dimensional arrays. However, in case of a flat surface, they both change to 2 dimensional matrices. This would make matrix A to have 3

dimensions. Since complicated mathematics is needed to solve such a problem, a new scheme is being proposed.

In this scheme it is suggested that instead of a 2-D matrix representing the surface temperature, we can change it to a one dimensional array by changing each row's position at the end of the previous row. (Fig. 4.10)

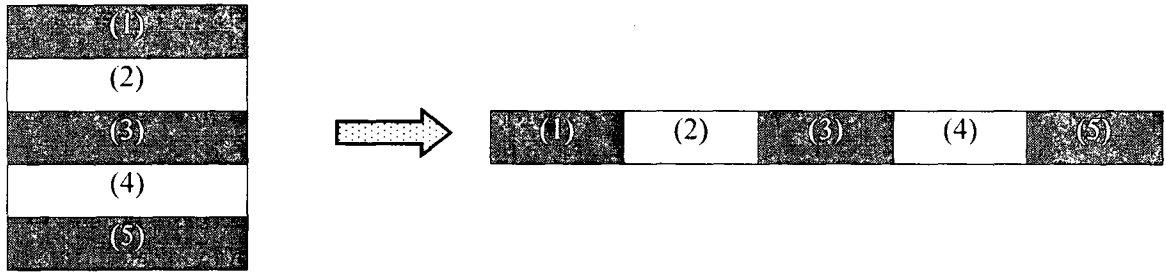


Fig 4.10 Schematic presentation of converting a 2-D matrix to a 1-D array

The 1-D array of a mesh of N by M is arranged in the following way: for the temperature $\phi''_{i,j}$, i and j both start from 1, the next temperature on the array has the same i value but j=2. Repeat the same manner until j reaches its maximum value (j=M), then i is replaced by i+1 and j starts from zero. While i remains fixed, j increases until its maximum. The process is repeated until i=N. On the first approximation, the general form of the equation would look like equation (4.28).

$$\begin{bmatrix}
 \ddots & \ddots & 0 & \ddots & 0 & 0 & 0 \\
 \ddots & \ddots & \ddots & 0 & \ddots & 0 & 0 \\
 0 & \ddots & \ddots & \ddots & 0 & \ddots & 0 \\
 -r & \cdots & -r & (1+4r) & -r & \cdots & -r \\
 0 & \ddots & 0 & \ddots & \ddots & \ddots & 0 \\
 0 & 0 & \ddots & 0 & \ddots & \ddots & \ddots \\
 0 & 0 & 0 & \ddots & 0 & \ddots & \ddots
 \end{bmatrix}
 \begin{bmatrix}
 \phi_{1,1}^{n+1} \\
 \vdots \\
 \phi_{i,j-1}^{n+1} \\
 \phi_{i,j}^{n+1} \\
 \phi_{i,j+1}^{n+1} \\
 \vdots \\
 \phi_{N,M}^{n+1}
 \end{bmatrix}
 =
 \begin{bmatrix}
 \phi_{1,1}^n \\
 \vdots \\
 \phi_{i,j-1}^n \\
 \phi_{i,j}^n \\
 \phi_{i,j+1}^n \\
 \vdots \\
 \phi_{N,M}^n
 \end{bmatrix}
 \quad (4.28)$$

It is evident that the one dimensional arrays in the above equation is a column matrix of $(N \times M)$ elements or the more convenient form is:

$$[A]_{NM \times NM} [x]_{NM \times 1} = [B]_{NM \times 1} \quad (4.29)$$

Similar to the implicit method in 1-D, x is to be found at each time step and is replaced as B for the next time. Up to this point, it is known that A is a five diagonal matrix with non zero elements on the main and the first upper and below main diagonals, and two other ones which are each located at M cells away from the main diagonal.

From the three diagonal matrix in 1-d and five diagonal matrix in 2-D, it can be understood that in a 3-D study, a seven diagonal matrix will be formed.

To get the precise information on A , study of boundary conditions is necessary.

4.4.2 Consideration on Boundary Condition

To start with, let's consider the initial conditions applied on the sample. Depending on the location of the heater at the beginning of the experiment, B at $t=0$ may be written as the following:

4.4.2.1 Heating the Surface from the Top

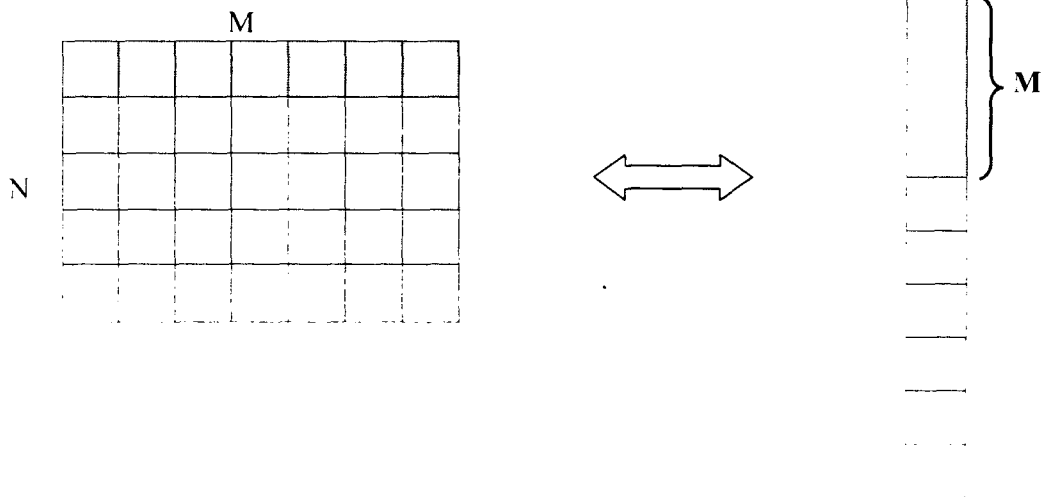


Figure 4-11 Schematic representation of the coefficient matrix (A) in implicit method, heating from the top.

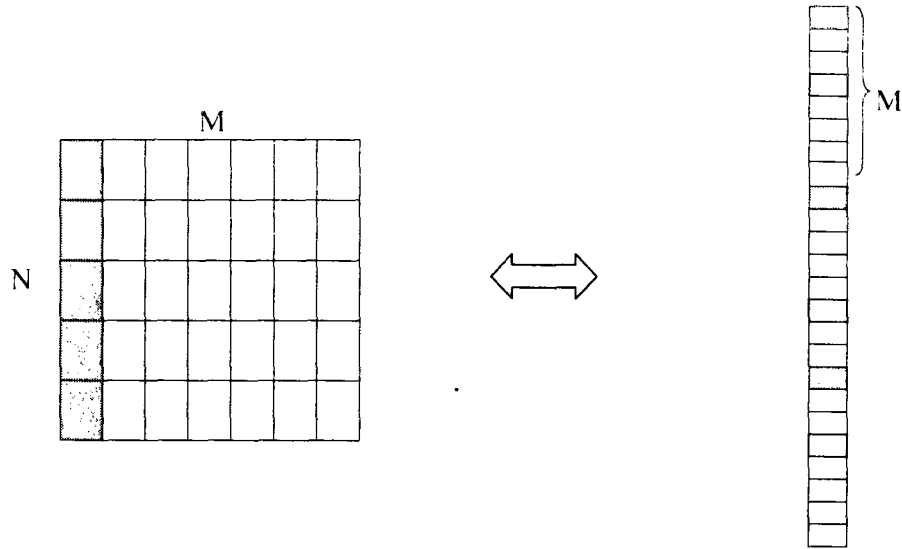


Figure 4-12 Schematic representation of the initial matrix (B) in implicit method, heating from the left side bar.

The N by M mesh on the left hand side of Fig. (4.11) represents the surface under study and it is evident that in the process of converting this 2-D matrix to a 1-D array, the first row will be located at the first M cells of B .

4.4.2.2 Heating from the Left Side

Figure (4.12) represents the location of the cells on the left side of the surface on the 1-D array. The first cell is located at the top of the array and the next one is M cells away.

4.4.2.3 Heating from the Right Side

When the source is located at the right hand side of the surface, the initial 1-D array (B) looks like Fig. (4.13). Here again the interval between each two heated cells is M .

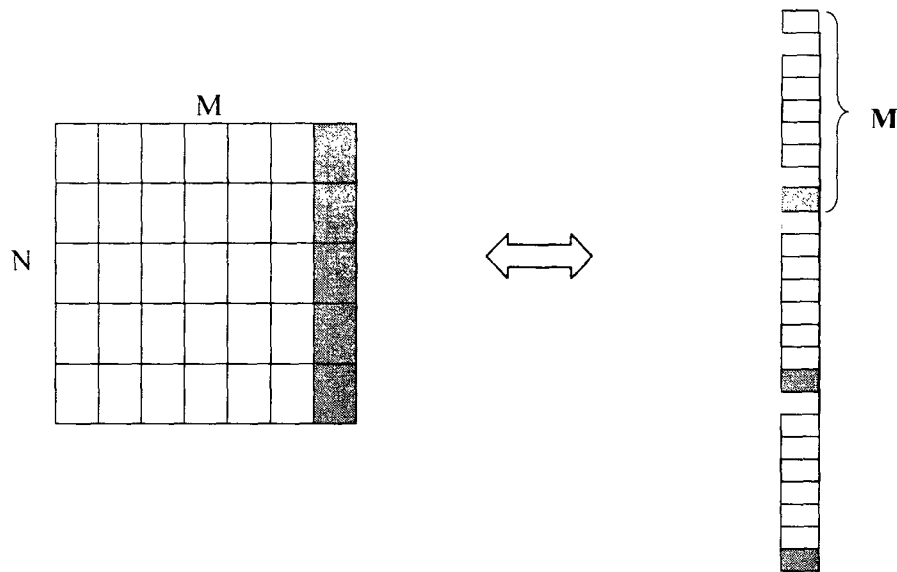


Figure 4-13 Schematic representation of the initial matrix (B) in implicit method, heating from the right side bar.

Thus the initial condition may be set and maintained for as long as desired in the sense that the initially heated cells on the column vector may remain heated for several time steps.

4.4.3 Problems with Implementing the Implicit Scheme and the Proposed Solution

Now considering the boundary conditions of the second order, it can be assumed that the simulation is done under the conditions where there is no heat flux between the edges and the environment. This is the case when the environment is not a good heat conductor with small coefficient of heat transfer (air in this case), and the temperature difference on the edges and the environment are small relative to the source temperature.

Putting it in mathematics, it's known that the heat flux is proportional to the first order spatial derivative of temperature:

$$flux \propto \frac{\partial \phi}{\partial x}$$

In the case where the flux is zero on the boundaries:

$$\frac{\phi''_{M+1} - \phi''_M}{\Delta x} = 0$$

Although the mathematical interpretation of this scheme is not complicated, finding the precise values and locations of the elements in the coefficients matrix (A) is challenging.

The coefficient matrix from the 1-D implicit method is helpful here. Same as in 1-D case, here there is a factor $(-r)$ on the cells corresponding to the “neighbors” and a factor $(1+\lambda r)$ for the main cell where λ is the number of the “neighbors”. It is shown in figure (4.14) that there are cells with 1, 2 and 4 neighbors in the matrix A.

To figure out the general lay out of A, the case of a 5 by 6 sample matrix is studied. (Fig 4.14) It is obvious that the matrix should be 30 by 30. It is also found that there are five block of 6 by 6 matrices on the main diagonal with similar lay outs. The elements on the main diagonal colored in dark grey are $(1+\lambda r)$, and other elements colored in light blue are $-r$ factors. λ for each dark grey cell may be determined by counting the number of blue cells at the same row. All the other elements in white hold zero.

Although the lay out of A is now found, from this example it is understood that for a typical sample with the number of spatial divisions of order 100 by 100, A would be $10^4 \times 10^4$. Storing such huge amount of data is almost impossible and solving the matrix equation from the order of 10^8 needs sufficient computational powers and time.

To overcome this problem, another scheme is devised [73]. This scheme takes advantage of the fact that a major portion of the elements in A are zero. Such a matrix is called “sparse” matrix. This method takes into account only the nonzero elements and all the mathematical procedures of matrix algebra are still available. The elaborate explanation of this scheme is attached in Appendix A.

Once the data is stored on the computer, the matrix equation is to be solved. The MATLAB code for storing A and solving the matrix equation is attached in Appendix B.

The following are the results of the simulations and discussion.

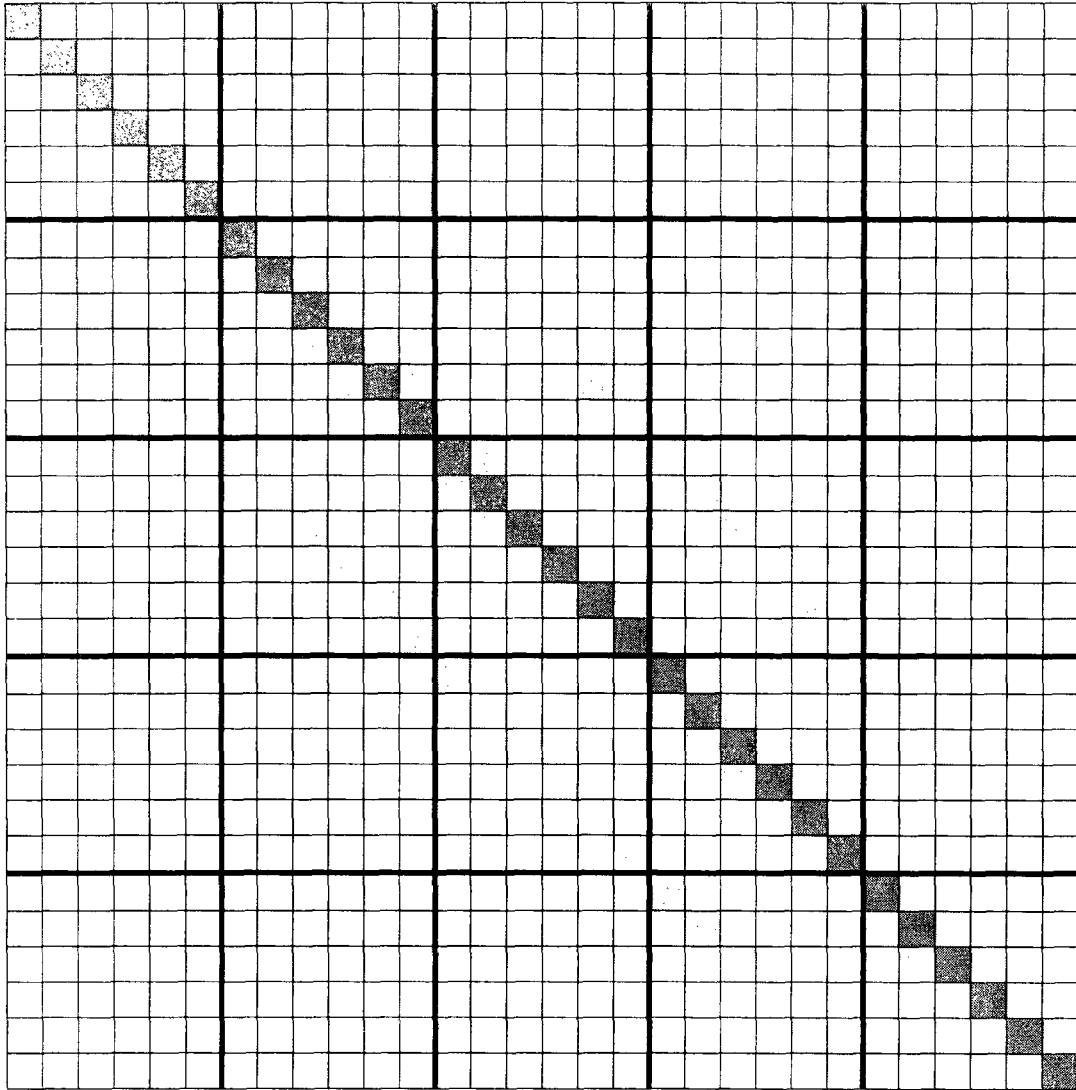


Fig 4.14 Implicit method in 2-D, the coefficient matrix for a sample of 5×6 .

4.4.4 Simulation, Implicit Method, 2-D

With the theory developed in previous sections (4.4.1 and 4.4.3) in hand, the simulation is put into practice [73]. To make sure that the results are reliable, the program is to be tested for different initial conditions before the main run is completed.

4.4.4.1 The Coefficient Matrix

To start with the simulation, it is necessary to make sure that the coefficient matrix (A) built by the sparse matrix method looks the same as the one in figure (3.17). So that the program was tested for a 5 by 6 sample of Aluminum.

In all the following simulations the time and spatial intervals are fixed:

$$d-t = 0.01 \text{ sec}$$

and

$$d-x = d-y = 0.1 \text{ mm}$$

Temperature of the heater is set to 1000 K and the environment is at 300 K.

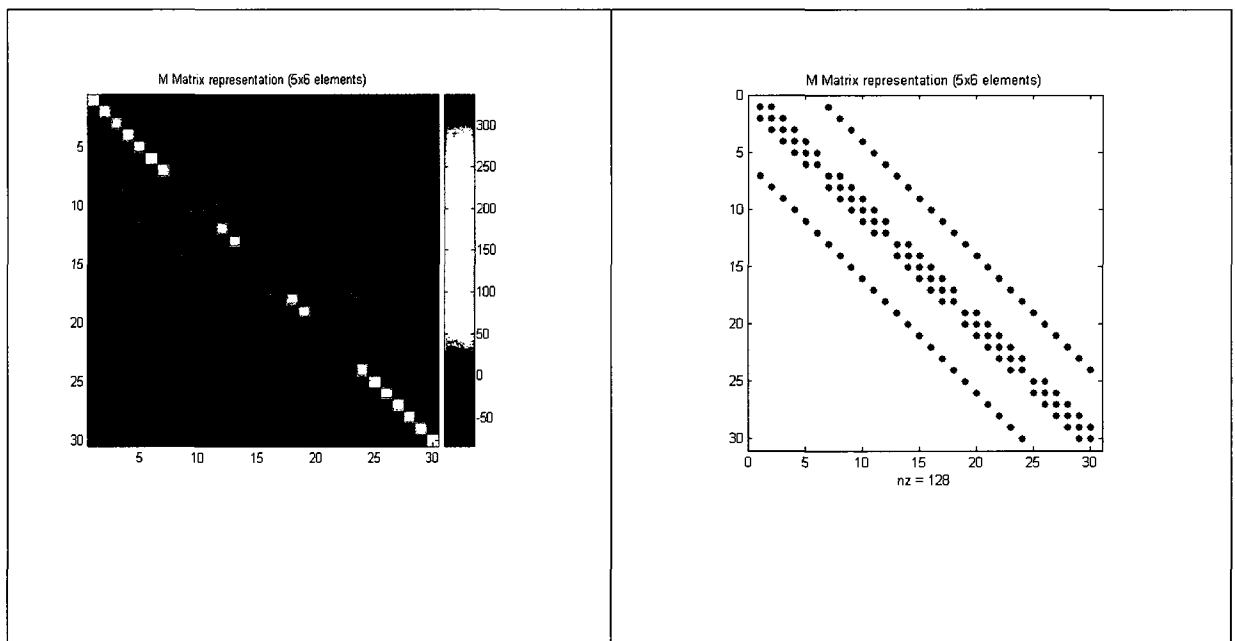


Figure 4-15 The coefficient matrix representation

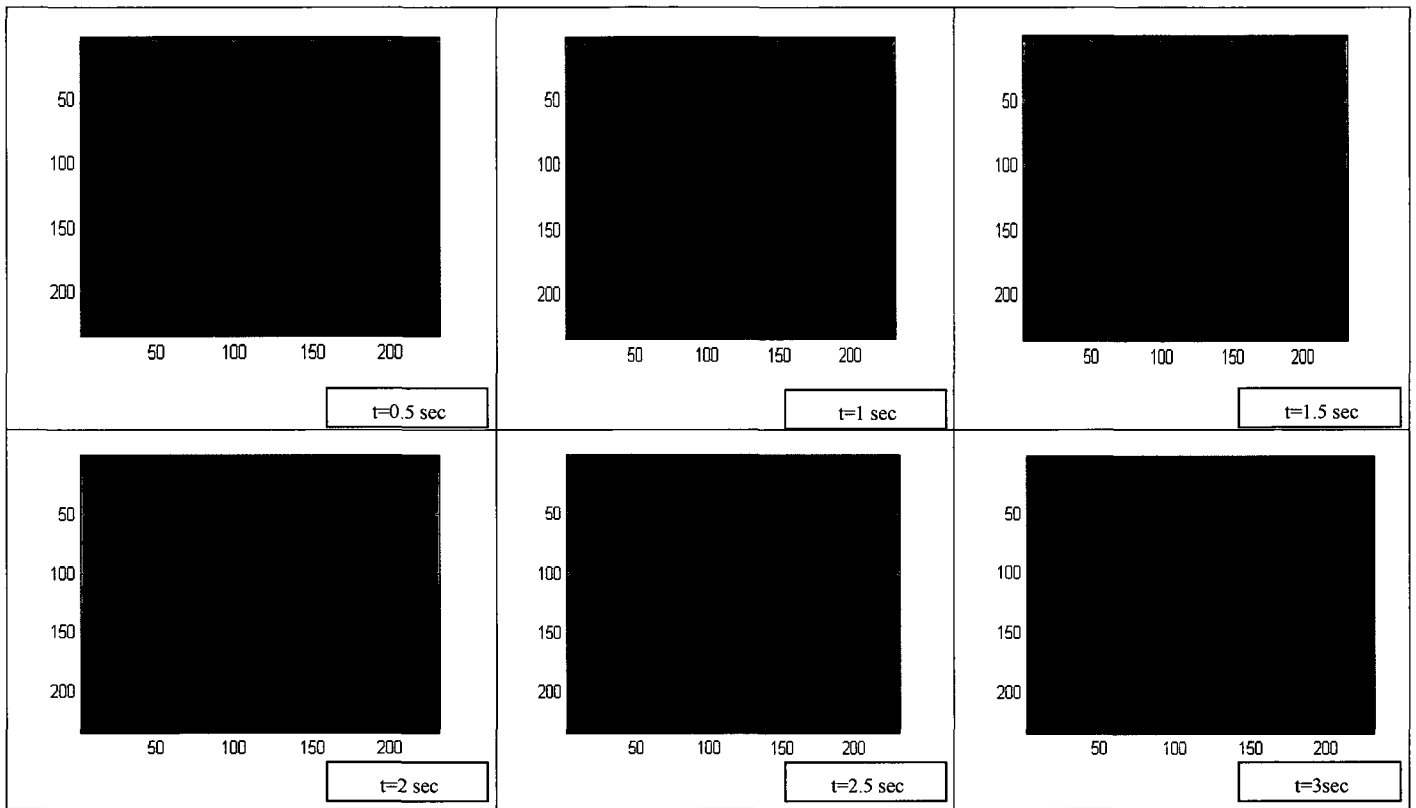


Figure 4-16 Heat propagation in a sample

The figure above represents the heat propagation in a sample while the heat is generated from the top side the sample and is maintained for 100 time steps (1 sec). The program is set to simulate the first 3 seconds of the propagation. In the last slide, 3 spots are defined and thermal diffusivity for them is found using Flash method. Both *half time* method and *intercept* method are used to find thermal diffusivity (α_1 and α_2). (figure 4-17) The results are then compared to the textbook values of α for the same sample. (Table 4-3)

Table 4-3 Thermal diffusivity from the simulated heat propagation using Flash method

| α \ coordinate | (19,24) | (19,25) | (19,26) | Aluminum |
|--------------------------------|-------------------------|-------------------------|-------------------------|-----------------------|
| α_1 (m ² /s) | 7.174×10^{-7} | 7.6451×10^{-7} | 7.0778×10^{-7} | 8.42×10^{-7} |
| α_2 (m ² /s) | 8.5713×10^{-7} | 8.0812×10^{-7} | 7.6498×10^{-7} | |
| Relative error α_1 | %14.8 | %9.2 | %15.9 | |
| Relative error α_2 | %1.8 | %4.1 | %9.1 | |

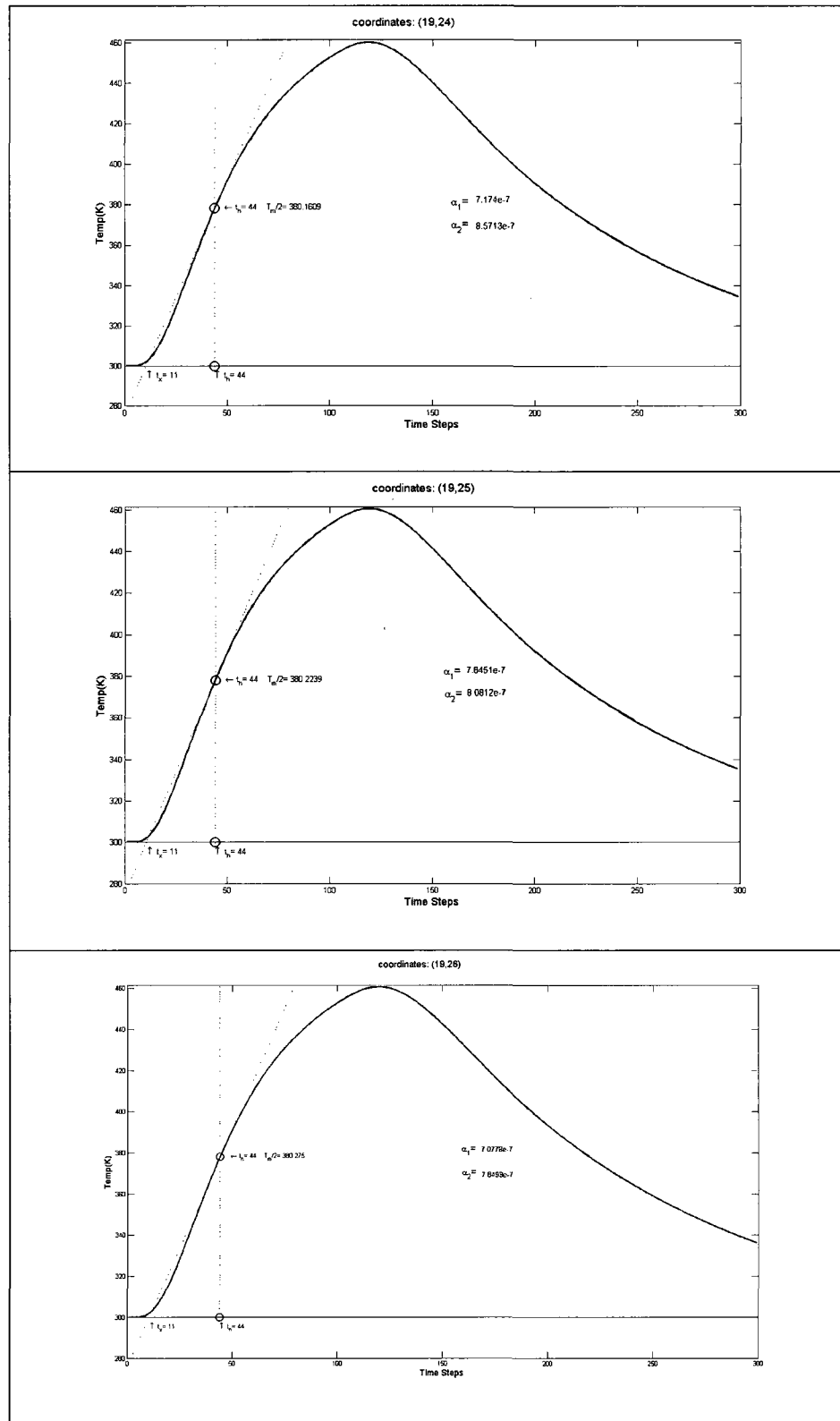


Figure 4-17 Finding thermal diffusivity in fig 4-16 using Flash method.

4.4.4.2 Heater Location

The program proves to work with the heater located at the top middle or top corner of the sample.(figures 4-18,4-19) The sample is assumed to be 3 by 4 cm. the spatial divisions are 0.1mm, the temperature of the heater is 1000K while the environment is at room temperature.

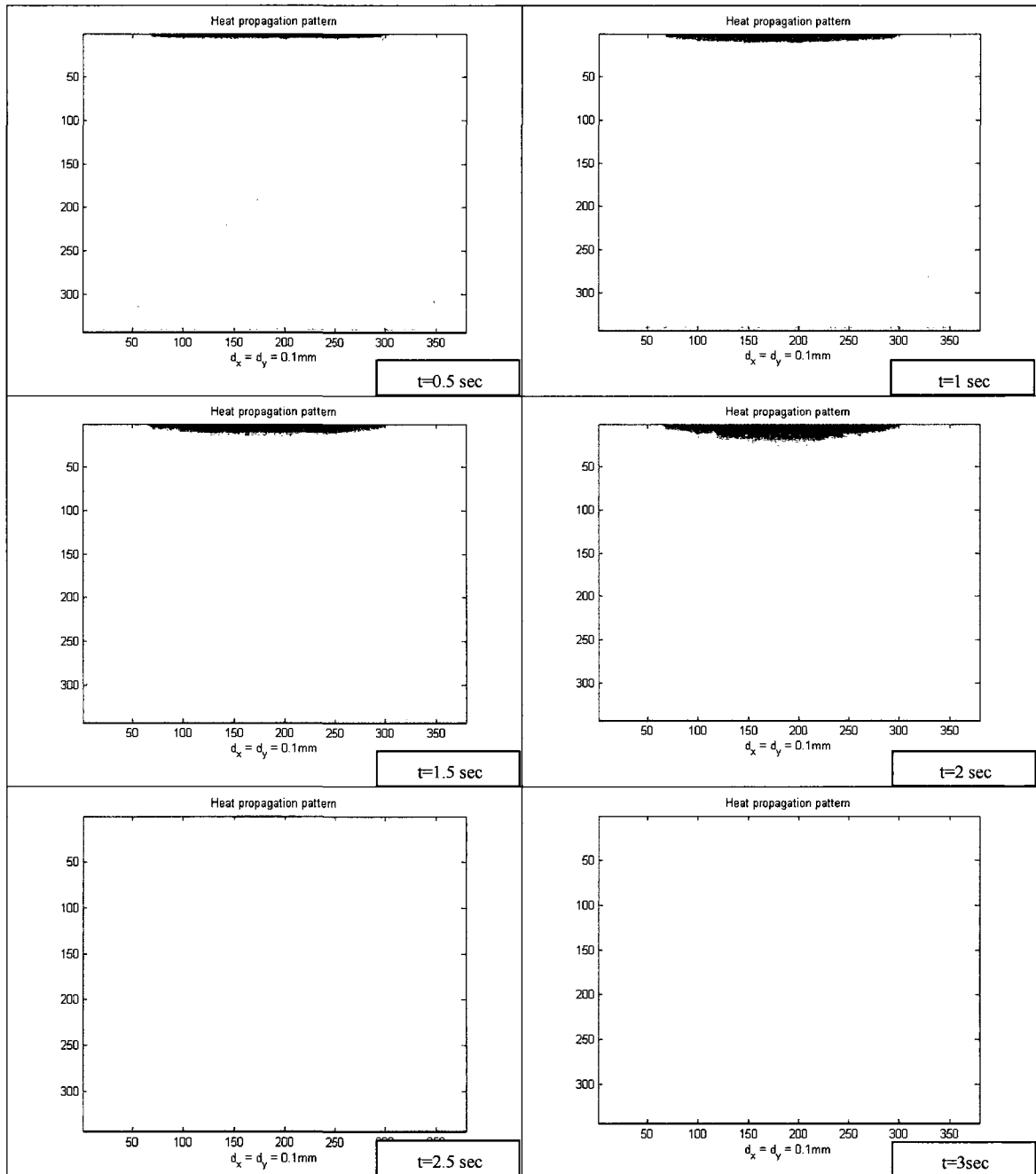


Figure 4-18 Simulation of heat propagation in a sample with the heater at the middle of the top side

The same initial conditions are applied to the same sample except for the location of the heater which at the top left corner of the sample this time.

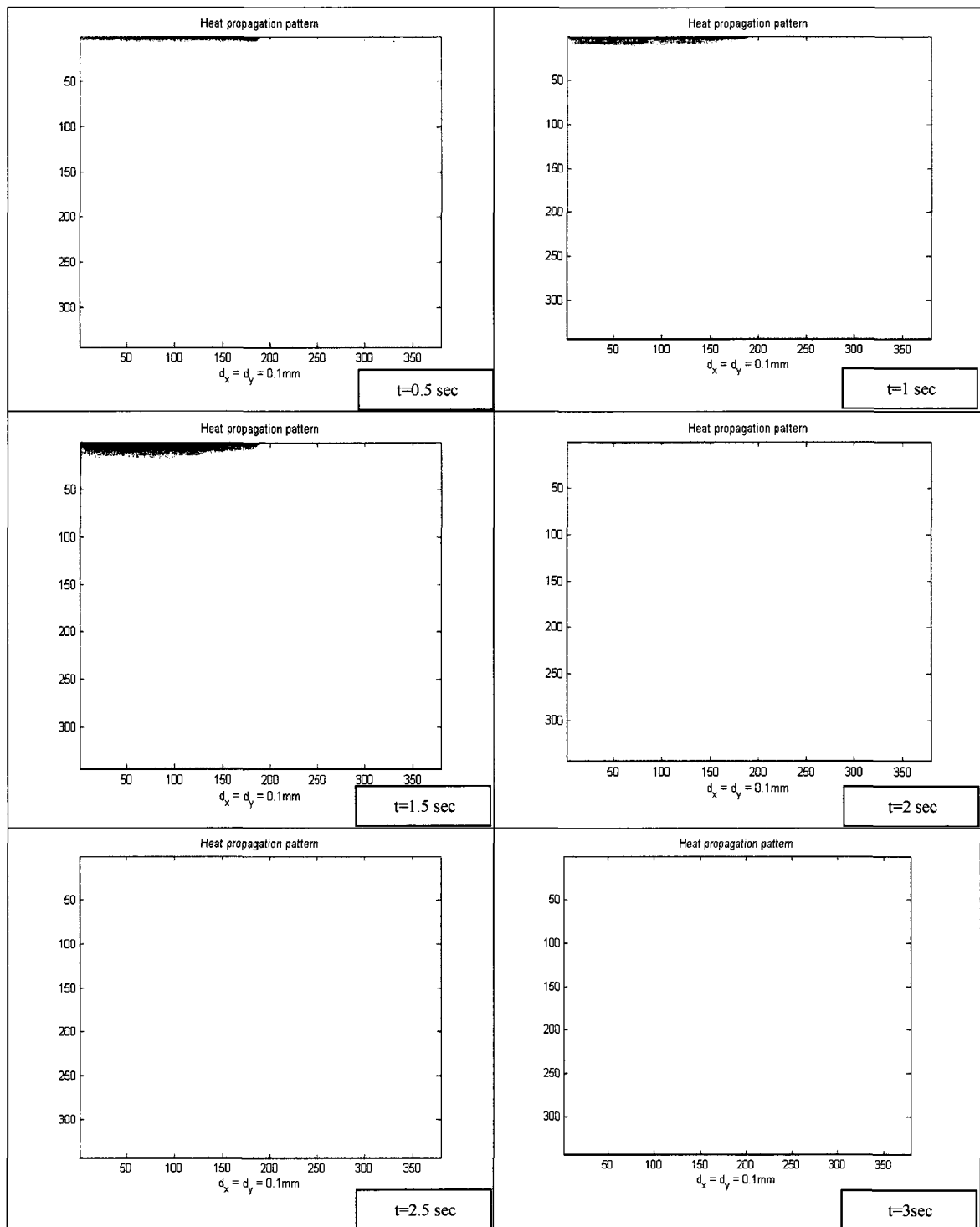


Figure 4-19 Simulation of the heat propagation with the heater at the corner

4.4.4.3 Implementing Defects in a Sample

A single defect can be implemented into a sample at random coordinates. In this example the sample of dimensions 2by 3.5 cm has a defect of a rectangular shape. The defect is laid at cells (10,145:175) with a different thermal diffusivity as the one from the sample:

$$\alpha_{\text{sample}} = 8.1e-7 \quad \alpha_{\text{defects}} = 2.2e-7$$

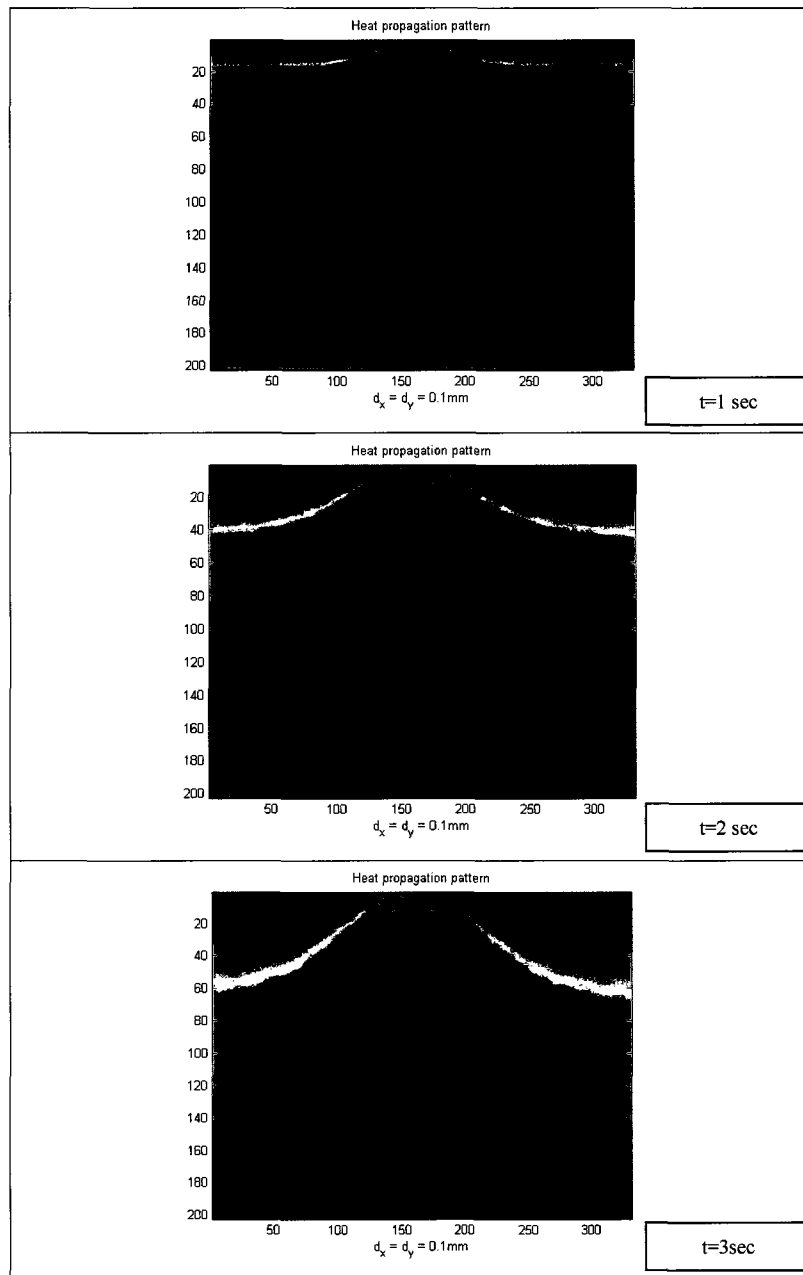


Figure 4-20 Heat propagation simulation for a sample with a single defect inside

Figure (4-20) is a demonstration of the heat propagation through time in this sample which shows a proper agreement with the real defected samples in which the thermal waves change direction and are refracted by the imperfections.

Another test is done on a sample with several defects of different shapes. The thermal diffusivity of the sample and the defects are as follows:

$$\alpha_{\text{sample}} = 8.1\text{e-}7$$

$$\alpha_{\text{defects}} = 2.2\text{e-}7$$

The thermal diffusivity of the sample at is then found using Parker method.

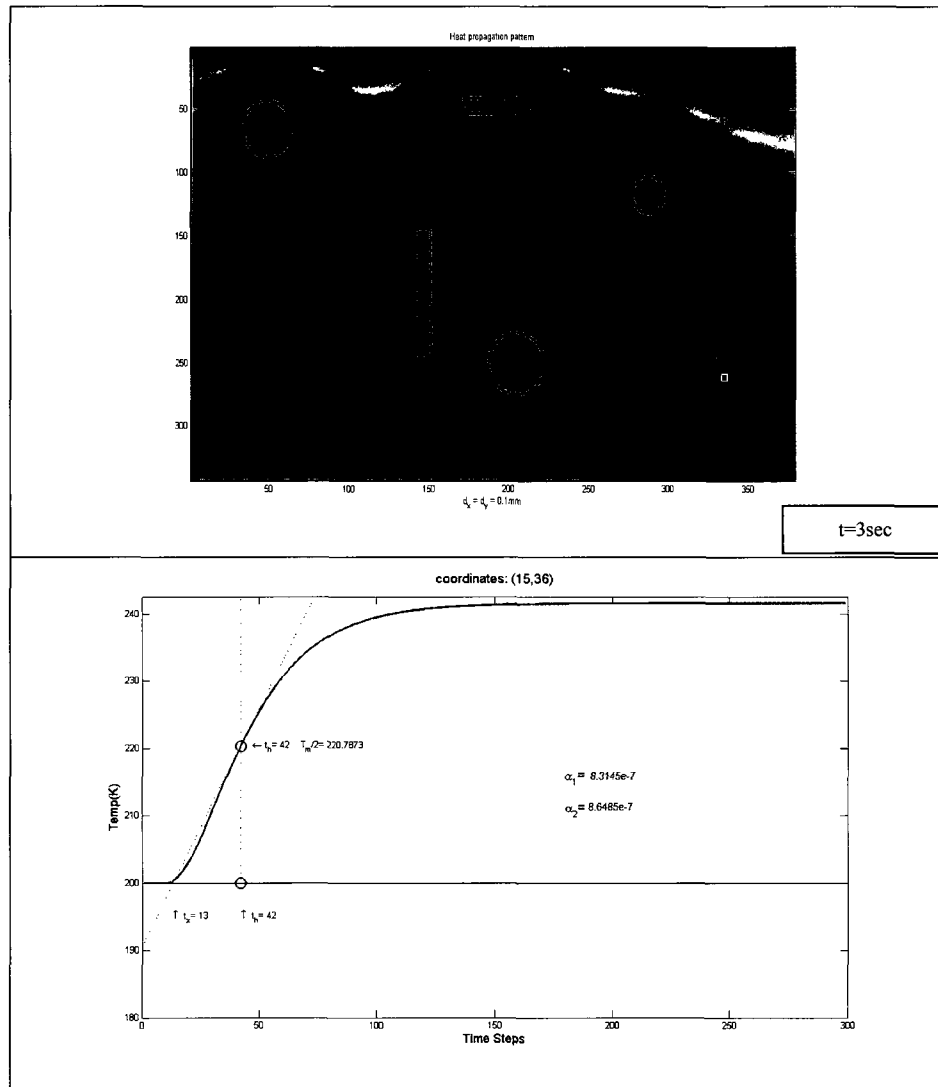


Figure 4-21 (up) Heat propagation simulation for a sample with a multiple defects inside,(down) Flash method of finding thermal diffusivity for the same sample

4.4.4.4 Heat propagation in 3 Layers:

The simulation is taken further to a sample of three layers. In this case two conditions are considered. If the middle layer is more thermally conductive than the other two layers the simulated heat propagation looks like figure (4-22).

Here the following values for thermal diffusivities are adopted:

$$\alpha_1 = \alpha_3 = 2.50e-7$$

$$\alpha_2 = 2.75e-7$$

Size of the sample: 30 by 40 cells.

Thickness of each layer: 10 rows.

Heat source in the middle: from cell #10 to 30.

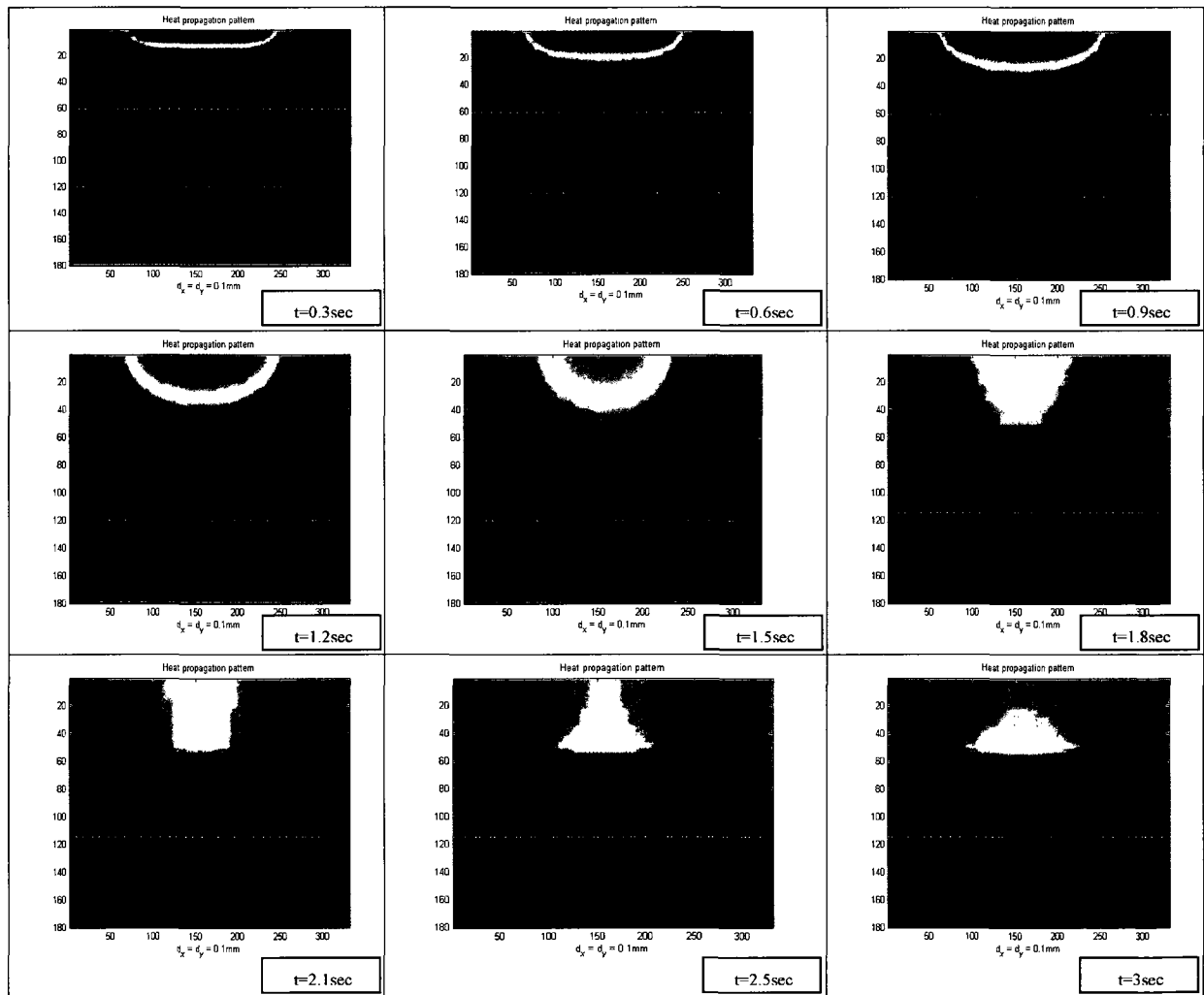


Figure 4-22 Heat propagation in a 3 layered sample, the middle layer is more heat conductive

Figure (4-23) is the representation of the heat propagation in a three layered sample with the middle layer more thermally conductive than the other two.

The thermal diffusivities of the layers are:

$$\alpha_1 = \alpha_3 = 2.80e-7$$

$$\alpha_2 = 2.50e-7$$

Size of the sample: 30 by 40 cells.

Thickness of each layer: 10 rows.

Heat source in the middle: from cell #10 to 30.

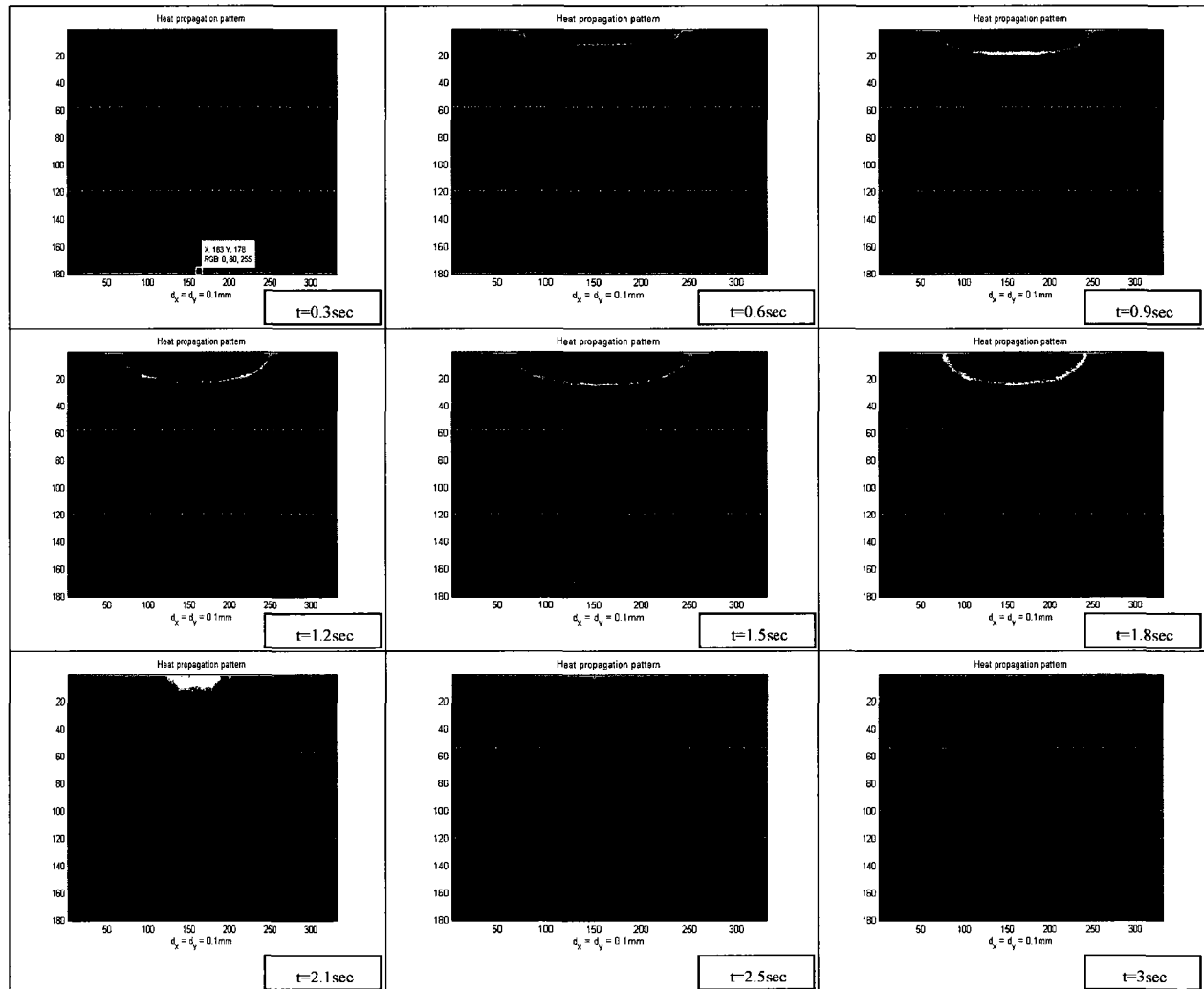


Figure 4-23 Heat propagation in a 3 layered sample, the middle layer is less heat conductive

It is interesting to compare the heat propagation process in figures 3-22 and 3-23. It seems from figure 3-22 that the heat does not go much further than the second layer and is stuck there until it cools down. However in figure 3-23 the heat is transferred through the middle layer smoothly and reaches the third layer which is expected from theory [73].

4.5 Conclusions

In this chapter, an algorithm is prepared and tested to simulate the thermal wave propagation in a homogeneous material. The simple implicit scheme is adapted to discretize the partial differential heat equation. To implement the scheme in practice, a large set of equations is to be solved which makes it complicated in terms of computer memory consumption and the duration of the test. Applying the concept of sparse matrices is a proper choice to overcome this problem.

Although the results are satisfactory, there are some limitations that should be mentioned. The results presented here are from simulating samples with maximum dimensions of 3 cm by 4 cm. It will take about 15 minutes to run an experiment for a sample of 0.4 cm by 0.4 cm, however the duration of the simulation does not increase linearly with respect to dimensions. For a sample of 3x4 cm, it will take half a day to get the results. If time was critical concern, writing the same code in C++ rather than MATLAB is supposed to decrease the duration of a test run.

Another source of error may be the boundary condition considered in this chapter. The simulation is done assuming that there is no heat flux from the environment. It is suggested to try other boundary conditions to see if the results are more accurate.

Chapter 5

5 Conclusions and Discussion

A finite-difference model for simulating the propagation of thermal waves in a layered material is developed. The method is based on the implicit scheme in solving the partial differential heat equation.

The fundamentals of infrared thermography were reviewed. Also a brief review on various NDT applications for IR thermography since its discovery was presented. Pulse thermography and lock-in thermography as two main methods in IR thermography and their advantages and drawbacks were described. Pulse phase thermography, as a combination of both PT and LT was found to benefit the advantages of both methods without sharing their imperfections. In the imaging process, thermal and photonic detectors are explained based on their methods of converting the thermal variation of the surface to an electric signal.

In the second chapter, different methods of data acquisition and data analysis in thermography were presented. Pulse thermography and Lock-in thermography are studies as well as the advantages and limitations of each method. Finally Pulse Phase thermography as a combination of these two schemes proves to benefit from the advantages of both methods without sharing their limitations.

Parker's Flash method of measuring thermal diffusivity was also introduced. This method is utilized in the next chapters to verify the applicability of the proposed simulation for thermal diffusivity evaluation of materials.

Chapter 3 involves the mathematics to solve the heat equation. It turns out that the exact solution to this partial differential equation is not easy to get hence the numerical solution are sought. Finite Difference method as an efficient tool in solving a partial differential equation is described and employed to solving the heat equation. Simple Explicit and Implicit schemes as the most common schemes resulting from FDM are investigated.

Both explicit and implicit schemes are implemented through simulations for 1D and 2D objects. The implicit scheme is chosen for further evaluations since it has the advantage of being unconditionally stable however, more complicated math is involved in it.

A large set of equations is to be solved in the implicit method at each time step to find the temperature distribution of the surface. For a typical sample of 3cm by 3 cm, the coefficient matrix is from the order of (10^5 by 10^5) which is almost impossible to store on the computer memory plus the fact that this process is to be repeated as many times as the time steps (usually 10^3 times).

The fact that the coefficient matrix in this problem contains a large number of zero elements suggests utilizing the concept of sparse matrices. RR(U)O as a fast method of storing such a matrix in the program is discussed in Appendix A.

As a result, the algorithm to implement the implicit method is defined (Appendix B) and the results for an object of two dimensions with or without defects is presented. The simulation is also taken to the next level by considering a sample of three layers. In the experiments, Parker's Flash method is used to measure thermal diffusivity and the results are compared to those from the simulations and textbooks. It can be seen from the results that the *intercept* method of thermal diffusivity measurement from Flash method leads to more reliable results in case of a 2D sample than the *halftime* scheme. The maximum relative error involved is found to be %10.

The error could be the result of assuming no heat flux on the boundary cells. More complicated boundary conditions may result in more accurate thermal diffusivities. Another concern of this simulation is also the time period. A typical sample of 3by 3 cm would take the simulation 12 hours to run which can be enhanced by working in C++ environment instead of MATLAB used here.

The result of this study has been accepted by the 10th QUIRT (The Quantitative Infrared Thermography) conference at University of Laval, Quebec, Canada on July 2010 [73].

Appendix A:

Row-wise Representation Upper Ordered (RR(U)O) Format.

When dealing with modeling of systems with large numbers of elementary cells, one usually faces the necessity to solve matrix equations of type $A \cdot x = B$, where A is a large sparse matrix. Depending on the type of simulated situation, A may be *tridiagonal* (temperature is function of time and one coordinate, $T(x, t)$), *five-diagonal* (case of $T = T(x, y, t)$) or *seven-diagonal* matrix (case of $T = T(x, y, z, t)$). In all these situations the number of nonzero elements is small relative to the total number of values, which requires vast amounts of memory to store and sufficient computational powers to process such matrices.

To decrease the load on computing system and storage, the way the sparse matrices are handled should differ from traditional way. An alternative format of matrix should take into account the sparsity of matrix, avoid storing of zero elements but still allow for basic mathematical procedures of matrix algebra.

One of such alternative ways of handling sparse matrices is so called *row-wise representation, complete, ordered* (RR(C)O).

Suppose we have following sparse matrix:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 6 & 0 & 8 & 5 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 9 & 0 & 0 & 0 \end{pmatrix}$$

In RR(C)O this matrix can be represented by three lists:

$$AN = 1 \ 2 \ 6 \ 8 \ 5 \ 3 \ 9 \quad (\text{nonzero values})$$

$$IA = 1 \ 3 \ 2 \ 4 \ 5 \ 3 \ 2 \quad (\text{column indices for each of nonzero values stored})$$

$$JA = 1 \ 2 \ 3 \ 6 \ 7 \ 8 \quad (\text{pointers of the first elements in each row})$$

In this case the storage of matrix A requires space allocation for 25 values if stored in usual way, while the three lists (AN , IA and JA) take only 20. In general, if we assume there are M nonzero values in $N \times N$ sparse matrix, AN uses M cells, IA uses M cells and $JA = N+1$ cells. Thus, the matrix A will require $2 \cdot M + N - 1$ units of memory, while in common way it would take N^2 units.

Solving FDM simulation problems one has to deal with large sparse matrices which have another useful property – their symmetry. Taking this property into account, one can save just upper half of the matrix and its diagonal elements, which allows to reduce the amount of memory space about twice relative to RR(C)O. This improved way of matrix storage is known as *row-wise representation, upper, ordered* (RR(U)O).

For example, suppose we have following matrix:

$$A = \begin{pmatrix} 1 & 0 & 1 & 3 & 0 \\ 0 & 3 & 0 & 4 & 5 \\ 1 & 0 & 7 & 0 & 0 \\ 3 & 4 & 0 & 2 & 0 \\ 0 & 5 & 0 & 0 & 5 \end{pmatrix}$$

By its upper triangle, in RR(U)O this matrix would be represented by four lists:

$$DA = 1 \ 3 \ 7 \ 2 \ 5 \quad (\text{D-list})$$

$$AN = 1 \ 3 \ 4 \ 5 \quad (\text{N-list})$$

$$IA = 3 \ 4 \ 4 \ 5 \quad (\text{I-list})$$

$$JA = 1 \ 3 \ 5 \ 5 \ 5 \quad (\text{J-list})$$

As can be seen, the matrix description now contains only 14 values. In general case, if the number of nonzero elements in $N \times N$ matrix is M , then diagonal contains M elements, the list of nonzero elements - $\frac{1}{2}(M - N)$, the list of column indices - $\frac{1}{2}(M - N)$, and the list of the rows pointers - N . In total, RR(U)O description of a matrix takes $N+M$ units of memory, which makes this format of matrix description to be the one of choice.

Though RR(U)O is very different from conventional way of matrix description, it is still very convenient for mathematical operations. Before we proceed to solution of matrix equations, we have to describe some of the basic considerations.

Merging of lists. By *merging* we will mean combination of two or more lists of numbers keeping only unique values in the resulting array. For example, two lists $A = [1 \ 6 \ 3 \ 5]$ and $B = [2 \ 2 \ 8 \ 9 \ 5]$ will give $C = [1 \ 2 \ 3 \ 5 \ 6 \ 8 \ 9]$.

One can see that C keeps only unique values of both A and B. This operation will play a significant role in mathematics related to RR(U)O format (*merge_list.m*).

Solving of matrix equations. One of the most common ways to solve a system of equations, expressed as a matrix equation $A \cdot x = B$, is expressing the matrix A as a product of three matrices:

$$A = L \cdot D \cdot U,$$

where L is lower triangular matrix with ones on the main diagonal, U is upper triangular matrix with ones on the main diagonal, and D is a diagonal matrix. After this kind of decomposition, one can easily solve equation

$$A \cdot x = L \cdot D \cdot U \cdot x = B$$

by consequent solution of three matrix equations:

$$L \cdot \omega = B \text{ gives } \omega, \text{ then}$$

$$D \cdot \nu = \omega \text{ gives } \nu, \text{ and, finally,}$$

$$U \cdot x = \nu \text{ allows one to find } x.$$

The solution can be made shorter. During triangulization of matrix A one actually constructs a set of other equations, which are the combinations of the initial ones. Keeping track of the permutations, one can get a new set of right-hand sides B' and solve following system:

$$D \cdot U \cdot x = B'$$

Later we will see this approach allows for reducing the amount of calculations.

Solution of a system of type $D \cdot \nu = \omega$ is simple. The matrix D^{-1} , inverse to diagonal matrix D is a diagonal matrix with diagonal elements, inverse to corresponding elements in D :

$$\text{If, for example, } D = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix}, \text{ then } D^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/3 & 0 \\ 0 & 0 & 1/5 \end{pmatrix}.$$

Thus the solution of $D \cdot \nu = \omega$ comes to multiplication of each element of column vector ω by diagonal elements of matrix D^{-1} . Provided that D^{-1} is given in RR(U)O format, these values are stored in designated list and can easily be accessed in D-list.

The next part of the solution is solution of the system $U \cdot x = \omega$, where U is an upper diagonal matrix with ones on the main diagonal. It is easy to check that U can be written as a product of a number of elementary column matrices:

$$U = \begin{pmatrix} 1 & a & b & d \\ & 1 & c & e \\ & & 1 & f \\ & & & 1 \end{pmatrix} = \begin{pmatrix} 1 & & d \\ & 1 & e \\ & & 1 & f \\ & & & 1 \end{pmatrix} \begin{pmatrix} 1 & b \\ & 1 & c \\ & & 1 \\ & & & 1 \end{pmatrix} \begin{pmatrix} 1 & a \\ & 1 \\ & & 1 \\ & & & 1 \end{pmatrix}$$

Thus, the solution of $U \cdot x = \omega$ is $x = U^{-1} \cdot \omega$, where U^{-1} is a reverse product of the elementary column matrices:

$$U^{-1} = \left(\begin{pmatrix} 1 & a & b & d \\ & 1 & c & e \\ & & 1 & f \\ & & & 1 \end{pmatrix} \right)^{-1} = \begin{pmatrix} 1 & -a & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{pmatrix} \begin{pmatrix} 1 & -b \\ & 1 & -c \\ & & 1 \\ & & & 1 \end{pmatrix} \begin{pmatrix} 1 & -d \\ & 1 & -e \\ & & 1 & -f \\ & & & 1 \end{pmatrix}$$

Working with U expressed in RR(U)O format, it should be easy to extract elements with certain column indexes and reverse their sign, forming elementary column matrices. These elements are extracted from N-list using column numbers from I-list.

Multiplication of a matrix and a column vector. Suppose one has to multiply an $N \times M$ matrix A stored in RR(U)O format by a column vector B of length M . The multiplication of A by B is then a column vector C of length M . Each component of the vector C is a product of a corresponding row of matrix A and vector B .

The algorithm for this multiplication is simple. For acquiring i^{th} component of C one first finds i^{th} value in the list JA , which contains the numbers of the first elements of each row. Then, looking over the elements in AN from $JA(i)$ to $JA(i+1)-1$, one takes $IA(JA(k))$ and multiplies by $B(JA(k))$. The component of C is acquired as a sum of these products.

In case of multiplication of a column vector by an elementary column matrix L_n^C , where n is the number of column containing nonzero elements atop of the main diagonal. Consider an example:

$$L_3^C = \begin{pmatrix} 1 & & a_1 \\ & 1 & a_2 \\ & & 1 \\ & & & 1 \end{pmatrix}, h = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix}$$

$$L_3^C \cdot b = \begin{pmatrix} 1 & a_1 & & \\ & 1 & a_2 & \\ & & 1 & \\ & & & 1 \end{pmatrix} \cdot \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix} = \begin{pmatrix} b_1 + a_1 b_3 \\ b_2 + a_2 b_3 \\ b_3 \\ b_4 \end{pmatrix}$$

Thus, during the multiplication $L_n^C \cdot b$ only those components of b become altered, which indices are less than n . The altered components are the sums of initial components (b_1, b_2, \dots) and the products of n^{th} component (here – b_3) and the elements of n^{th} column of L_n^C (a_1, a_2, \dots).

The code for this kind of multiplication is extremely simple:

```
1   col_ind = L.I(L.J(1));
2   for i = 1:1:length(L.N)
3       b(i) = b(i) + L.N(i)*b(col_ind);
4   end
```

Now, after reviewing the way to solve matrix equation with upper triangle matrix, we need to find the way to convert an arbitrary matrix to upper triangle configuration.

Note: the matrix equation containing large sparse symmetric matrix A and right-hand side B can be solved in MATLAB using *solve AXB* function

Matrix triangulization.

Triangle matrix is one which has nonzero elements only on its diagonal and only on one of the sides of the diagonal. Depending on the side where nondiagonal nonzero elements are located, the matrix can be called *upper-* or *lower-* diagonal.

Triangulization of a matrix is a general name of a procedure transforming an arbitrary matrix to triangular configuration.

One of the most convenient methods of matrix triangulization is Gauss elimination. Further we will use this method for conversion of a matrix to upper diagonal configuration. In Gauss elimination one finds linear configurations of the matrix rows in order to make some of the matrix elements to become zero. To convert a matrix to upper diagonal, one should eliminate all the elements under the diagonal by adding the upper rows multiplied by certain numbers.

Suppose one has a matrix:

$$A = \begin{pmatrix} 1 & 5 & 3 \\ 2 & 6 & 5 \\ 3 & 4 & 9 \end{pmatrix}$$

One can eliminate the first column under the diagonal by multiplying the first row by -2 and -3 , and adding the row to the second and the third correspondingly:

$$A = \begin{pmatrix} 1 & 5 & 3 \\ 0 & -4 & -1 \\ 0 & -11 & 0 \end{pmatrix}$$

To eliminate the second row one multiplies the second row by $-11/4$ and adds it to the third row:

$$A = \begin{pmatrix} 1 & 5 & 3 \\ 0 & -4 & -1 \\ 0 & 0 & 11/4 \end{pmatrix}$$

In case of larger matrices, the procedure is a cycle over the rows. At each step i the i^{th} row is added to all the lower rows multiplied by a negative ratio of the first elements of the destination row and i^{th} row.

The procedure stays absolutely the same in case of large symmetric sparse matrices. But as it was discussed before, it is problematic to store them in usual manner. Instead we agreed to use special format for storing sparse matrices. In this format we keep separately diagonal elements (D-list), the elements of the upper triangle (N-list), and the portrait of the upper triangle (in lists I and J). The power of this format is in the fact it does not store zero elements. But at the same time this creates additional difficulties for using RR(U)O for triangulization of matrix. The origin of these difficulties is in the fact we do not know a priori where in the matrix a nonzero element can appear after some row is added to another. For example, suppose we have a matrix

$$A = \begin{pmatrix} 1 & 5 & 3 \\ 5 & 6 & 0 \\ 3 & 0 & 9 \end{pmatrix},$$

which can be written in RR(U)O as:

D: 1 6 9

N: 5 3

I: 2 3

J: 1 3 3

After triangulization this matrix will have a new element in position (2, 3):

$$U = \begin{pmatrix} 1 & 5 & 3 \\ & -19 & -15 \\ & & 11.84 \end{pmatrix}.$$

In other words, we do not know the portrait of the final upper triangular matrix in advance. For this reason, the process of triangulization of an arbitrary sparse symmetric matrix stored in RR(U)O format is subdivided into two parts: *symbolic* stage, where the portrait of a new matrix is to be found, and *numeric* one, where calculations are to be made and the numbers to fill the cells in the portrait found.

On the symbolic stage one should remember that one row is added to another, so their portraits are merged. In the most recent example we had to merge the portraits of the first and the second rows, making the element (2, 3) to appear in the second row.

Thus, knowing that i^{th} row will be added to the j^{th} one, we should first merge the I-lists of the two rows to get the new portrait of new row, then find how much the row grew (increased its length). This increase of the length of the row will shift the values of J-array. As we remember, this list holds the positions of the first elements of each row of the upper triangle. Thus, if, say, the new matrix became larger by 2 elements, all the subsequent rows will be shifted 2 elements further.

The procedure of the symbolic part is as follows:

Now, when the portrait is ready, and the old matrix is expanded, one can be sure there will be no need to change the I- and J-lists. One can perform the rows additions.

During the numeric part one checks the presence of nonzero elements in each row of the upper triangle (N-list) and, since the matrix is symmetric, finds out to which destination row the current row will be added (the number of the destination row is equal to the column index of the current nonzero element in the current row), as well as the coefficient to multiply the row by (it is equal to the negative ratio of the element in current row with the number equal to the destination row number divided by the current element).

The code for the numeric procedure is as follows:

```

. --- Numeric part of gaussian elimination
count_k = 0;
for i = 1 : 1 : Nrows
    ST_i = A.J(i);
    ST_N = A.J(i + 1) - ST_i;

    for j = 0 : 1 : ST_N - 1
        count_k = count_k + 1;
        k(count_k, 1) = - single(A.N(A.J(i) + j) / A.D(i));
        k(count_k, 2) = single(i);
        k(count_k, 3) = single(A.I(A.J(i) + j));

        for l = j : 1 : ST_N - 1
            if l == j
                A.D(A.I(ST_i + 1)) = A.D(A.I(ST_i + 1)) + k(count_k, 1) * A.N(ST_i + 1);
            else
                Col_in = A.I(ST_i + 1);
                Row_in = A.I(ST_i + j);
                x = find_pos(A.I, A.J, Row_in, Col_in);
                A.N(x) = A.N(x) + k(count_k, 1) * A.N(ST_i + 1);
            end
        end
    end
end

```

It should be noted that the list k is stored intentionally. After the procedure is done, one will have the scheme of all the permutations performed with the rows. This will allow to rearrange the right-hand side when solving the matrix equation of type $Axx = B$.

After triangulization one should normalize each row in respect to the diagonal element, so the matrix is upper triangular with ones on the diagonal. This would allow representing the matrix as a product of a number of elementary column matrices.

Appendix B:

Sample codes (main body and some functions in RR(U)O) operating the simulation in case of a 2D sample

The following code represents the main body of the developed program:

```
clear all
clc
tic

%Initial inputs-----

iTime=300;
N_rows = 400;           % the number of rows in the sample
N_cols = 400;           % the number of columns in the sample
d_x = 0.0001;           % the size of the spatial step
d_t = 0.01;             % the duration of the time step
t_length=N_rows*N_cols;

x0=[199 200 201];       % Inspection points coordinates
y0=[399 399 399];       % Inspection points coordinates


T_H=1000;               %Source temperature
T_R=300;                %Room Temp
source_width=20;
source_depth=1;
source_beg=10;          %begining location of the source
pulse_dur=1*iTime/3;

alpha=2.5e-7;
r=alpha*d_t/d_x^2

r_mtx=zeros(N_rows, N_cols);
r_mtx(:,:)=r;

% alpha_defect=2.2e-3;
% r_prime=alpha_defect*d_t/d_x^2
% r_mtx(4,5,6,7)=r_prime;
% r_mtx(10,15,1,7)=r_prime;
% r_mtx(4,19,2,3)=r_prime;
% r_mtx(8,39,3,1)=r_prime;

% imagesc(r_mtx)

alpha_2=2.75e-7;
r_2=alpha_2*d_t/d_x^2
```



```

r_mtx(10:20,:)=r_2;

% alpha_3=2.6e-7;
% r_3=alpha_3*d_t/d_x^2
% r_mtx(20:end,:)=r_3;

%Transformation Matrix Generation-----

M = make_fivediag_RRUO(N_rows,N_cols,r_mtx);

figure(2)
A=rruo2norm(M);
pcolor(A)
view(0,270)
% spy(A)
title('M Matrix representation (5x6 elements)')

%%
%Diagonalization on The Matrix in RR(U)O-----

[DA, k]= diagonalize_mtxA(M);

% return

for i = 1:size(k, 1)
    if k(i,2) == 0
        disp(['Here"s the number: ', num2str(i)])
        break
    end
end

k = k(1:i-1, :);

%%
% clc
% N_rows = 100;      % the number of the cells in the rod
% N_cols = 100;
% t_length=N_rows*N_cols;
%
%
% load DAK_01
% for i = 1:size(k, 1)
% if k(i,2) == 0
% disp(['Here"s the number: ', num2str(i)])
% break
% end
% end
%
% k = k(1:i-1, :);

%Initial Boundary Conditions-----

T = zeros(t_length,1);      % the temperature distribution
T_new=ones(t_length,1)*T_R;
% T(1:source_depth*N_cols,1)=T_H;
T_new(source_beg:source_beg+source_width-1,1)=T_H;

Temp_el=zeros(length(x0),iTime-1);

```

```

%Solving and frame saving-----

for i = 1 : iTime-1          % the time loop

    if i<pulse_dur

        T=solveDAXB(DA, k, T_new);
        %      T=M\T_new;
        %      T(1:source_depth*N_cols,1)=T_H;
        T(source_beg:source_beg+source_width-1,1)=T_H;

    else

        T=solveDAXB(DA, k, T_new);
        %      T=M\T_new;
    end

    for nn=1:t_length
        if T(nn)<T_R
            T(nn)=T_R;
        end
    end

    T_2D=(reshape(T,N_cols,N_rows))';

    for m = N_rows-2:N_rows
        T_2D(m,:)=max(T_2D(m,:));
    end

    for ii=1:length(x0)

        Temp_el(ii,i)=T_2D(x0(ii),y0(ii));

    end

    pcolor(T_2D);
    axis([1 N_cols 1 N_rows 0 T_H 0 T_H])
    colorbar
    view(0,270)
    shading interp

    F(i)=getframe;

    %   pause(0.03)

    T_new=T;

    disp(['step:', num2str(i)])
end
toc

%Alpha calculation and plotting-----

for i=1:length(x0)

    POBJ.distance=y0(i);
    POBJ.d_x=d_x;
    POBJ.d_t=d_t;
    POBJ.iTime=iTime;

```

```

PObj.x_0=x0(i);
PObj.y_0=y0(i);
PObj.T_R=T_R;

figure(i+3)
plot_temp(Temp_el(i,:),PObj)

end

% movie(F)

```

And followings are code description of two of the main functions used in main body of the program:

```

function A=make_fivediag_RRUO(N_rows,N_cols,r)

% this function makes the five diagonal symetric matrix in RRUO format. It accepts
% the number of rows and columns on the specimen along with r(.....) and makes the heat
% transform matrix in RRUO format. The output is structure "A" composing of:
%
% A.D - diagonal elements
% A.N - nonzero non-diagonal elements of the upper part of matrix A
% A.I - column numbers of elements in A.N
% A.J - pointers of the first elements in each row

% Uncomment to check the code from inside:-----

N_rows=5;
N_cols=6;
r=ones(N_rows,N_cols);
r=1;

% -----

N_total=N_rows*N_cols;
A.D=ones(1, N_total);
A.N=ones(1, [(N_rows-1)*N_cols+N_rows*(N_cols-1)]);
A.I=zeros(1, length(A.N));

A.J=zeros(1, N_total);

% A.D calculation-----

kk=1;

% for i=1:N_rows
%   for j=1:N_cols
%
%       if i==1 || i==N_rows
%
%           if j==1 || j==N_cols
%               A.D(kk)=1+2*r(i,j);
%           else
%               A.D(kk)=1+3*r(i,j);
%           end
%
%       else
%           if j==1 || j==N_cols
%               A.D(kk)=1+3*r(i,j);
%           else
%               A.D(kk)=1+4*r(i,j);

```

```

%         end
%     end
%
%         kk=kk+1;
%     end
% end

for i=1:N_rows
    for j=1:N_cols

        if i==1

            if j==1 || j==N_cols
                A.D(kk)=1+2*r(i,j);
            else
                A.D(kk)=1+3*r(i,j);
            end

            elseif i==N_rows

                if j==1 || j==N_cols
                    A.D(kk)=1+3*r(i,j);
                else
                    A.D(kk)=1+4*r(i,j);
                end

            else
                if j==1 || j==N_cols
                    A.D(kk)=1+3*r(i,j);
                else
                    A.D(kk)=1+4*r(i,j);
                end
            end

            kk=kk+1;
        end
    end

% for i=1:N_rows
%     for j=1:N_cols
%
%         if j==1 || j==N_cols
%             A.D(kk)=1+3*r(i,j);
%         else
%             A.D(kk)=1+4*r(i,j);
%         end
%
%         kk=kk+1;
%     end
% end

% A.D(1)=1+4*r(i,j);
% A.D(N_total)=1+2*r(i,j);

% A.N calculation-----

mm=1;
i=1;
j=1;

for mm=1:length(A.N)-(N_cols-1)

    if rem(floor(mm/(2*N_cols-1)),2)==0

        if rem(mm,2*N_cols-1)==0
            A.N(mm)=-r(i+1,j);
            i=i+1;
            j=1;
        end
    end
end

```

```

elseif rem(mm,2)==1
    j=j+1;
    A.N(mm)= -r(i,j);

else

    A.N(mm)=-r(i+1,j-1);

end

else

    if rem(mm, 2*N_cols-1)==0
        A.N(mm)=-r(i+1,j);
        i=i+1;
        j=1;
    elseif rem(mm,2)==1

        A.N(mm)=-r(i+1,j-1);

    else
        j=j+1;
        A.N(mm)= -r(i,j);

    end

end

end

for mm=length(A.N)-(N_cols-1)+1:length(A.N)

    A.N(mm)= -r(i,j);
    j=j+1;

end

% A.l calculation-----

a2=N_cols-1;
a3=N_cols-2;
a1=2+a3;

for i=1:length(A.N)-(N_cols-1)

    if rem(floor(i/(2*N_cols-1)),2)==0

        if rem(i,2*N_cols-1)==0
            A.l(i)=a1+1;

        elseif rem(i,2)==1
            A.l(i)= a1 - a3;

        else
            A.l(i)=a1+a2;

        end

    else

        if rem(i, 2*N_cols-1)==0
            A.l(i)=a1+1;

```

```

elseif rem(i,2)==1
    A.I(i)= a1 +a2;

else
    A.I(i)=a1 -a3;

end

end

a1=A.I(i);

end

A.I(length(A.N)-(N_cols-1)+1)=A.I(length(A.N)-(N_cols-1)-2)+2;

for i=length(A.N)-(N_cols-1)+2:length(A.N)

    A.I(i)=A.I(i-1)+1;

end

% A.J calculation-----

A.J(1)=1;

for i=2:length(A.J)-N_cols

    A.J(i)= A.J(i-1)+2;

    if rem(i,N_cols)==1

        A.J(i)=A.J(i-1)+1;

    end

end

for i=length(A.J)-N_cols+1:length(A.J)

    A.J(i)=A.J(i-1)+1;

end

% A.D=A.D';
% A.N=A.N';
% A.I=A.I';
% A.J=A.J';

% Checking the results for small numbers (N_rows & N_cols <= 50)-----

% M=ruo2norm(A);
%
% pcolor(M)
% colormap(jet)

```

```

function plot_temp(Final_Temp,PObj)

[T_max,t]=max(Final_Temp); % finds the maximum of the graph (temperature of the
chosen element vs. time)

for i=1:PObj.iTime-2 %finds the time at which the temperature is half of the
maximum
    if(Final_Temp(i)<=PObj.T_R+(T_max-PObj.T_R)/2 && Final_Temp(i+1)>=PObj.T_R+(T_max-PObj.T_R)/2)
        t_half=i; %the time at which the temperature is half of the maximum
    end
end

% t_x calculation -----
X1=[t_half ; t_half+1];
Y1=[Final_Temp(t_half) ; Final_Temp(t_half+1)];

f=fit(Y1,X1,'linear');
t_x=round(feval(f,PObj.T_R));

% alpha calculation -----

L=PObj.distance*PObj.d_x;
alpha1=1.38*L^2/(pi^2*t_half*PObj.d_t) %finds alpha coef. from t_half
alpha2=0.48*L^2/(pi^2*t_x*PObj.d_t) %finds alpha coef. from t_x

% plotting commands -----

fff=fit(X1,Y1,'linear');
fplot(fff,[0,PObj.iTime/2],':r')
hold on

X2=[0 PObj.iTime];
Y2=[PObj.T_R PObj.T_R];
t=1:PObj.iTime-1;

plot(t,Final_Temp,'b','linewidth',2)
plot(t_half,Final_Temp(t_half),'ok','linewidth',2,'MarkerSize',12)
plot(t_x,24,'ok','linewidth',2,'MarkerSize',12)
plot(t_half,PObj.T_R,'ok','linewidth',2,'MarkerSize',12)
plot(X2,Y2,'k')
plot([t_half t_half],[PObj.T_R-0.5 T_max+0.5],':r')
text(t_half+5,Final_Temp(t_half),['\leftarrow t_h= ',num2str(t_half), ' T_m/2= ',
num2str(PObj.T_R+(T_max-PObj.T_R)/2)])
text(t_half,PObj.T_R-5, ['\uparrow t_h= ',num2str(t_half)])
text(t_x,PObj.T_R-5, ['\uparrow t_x= ',num2str(t_x)])
text(3*PObj.iTime/5,PObj.T_R+(T_max-PObj.T_R)/2,['\alpha_1= ',num2str(alpha1)],'fontSize',13)
text(3*PObj.iTime/5,PObj.T_R+(T_max-PObj.T_R)/2-10,['\alpha_2= ',num2str(alpha2)],'fontSize',13)
axis([0 PObj.iTime PObj.T_R-20 T_max+1])
xlabel('Time Steps', 'fontSize',13)
ylabel('Temp(K)', 'fontSize',13)

hold off

```

References

- [1] X. Maldague, "Infrared methodology and technology" , Yverdon, Switzerland, Gordon and Breach Science Publishers, 1994, pp. 137-149.
- [2] C. Meola, G. M. Carlomagno and G. Giorleo, "The Use of Infrared Thermography for Materials Characterization", J. Mater Process Technol., 2004, pp. 155-156 and 1132–1137.
- [3] X. Maldague, "Infrared and Thermal Testing", Nondestructive Handbook on Infrared Technology, ASNT Handbook Series, 3rd edition, Volume 3, ASNT Press, Columbus, Ohio, 2001, p. 718.
- [4] G. Giorleo, C. Meola and A. Squillace, "Analysis of Defective Carbon-Epoxy by Means of Lock-in Thermography", Res. Nondestr. Eval., 2000, pp. 241-250,.
- [5] N. P. Avdelidis, A. Moropoulou, "Applications of infrared thermography for the investigation of historic structures", Journal of Cultural Heritage, Volume 5, Issue 1, 2004, pp. 119-127,
- [[7] Maldague X. "Introduction to NDT by active infrared thermography", Mater. Eval., 2002, pp. 1060–1073.
- [8] C. Meola, G. Giorleo, L. Nele, A. Squillace and G. M. Carlomagno, "Infrared thermography in the quality assurance of manufacturing systems", Nondestructive Testing and Evaluation, Volume 18, Number 2, 2002 , pp. 83-90.
- [9] V. Pašagić , "Infrared Thermography in Marine Applications", Brodogradnja Journal of Naval Architecture and Ship Building Industry, vol.59, iss.2, 2008.
- [10] X. Maldague, "Introduction to NDT by active infrared thermography". Mater Eval, 2002. pp. 1060-1063.

- [11] D. Titman, "Applications of thermography in non-destructive testing of structures", Elsevier, International Conference on Structural Faults and Repairs, London, UK, 2001, pp. 149-154.
- [12] F. P. Incropera and D. De Witt, "Fundamentals of Heat and Mass Transfer", 3rd edition, John Wiley & Sons, USA, 1990.
- [13] N. P. Avdelidis, A. Moropoulou, "Applications of infrared thermography for the investigation of historic structures", Journal of Cultural Heritage, Volume 5, Issue 1, 2004, pp. 119-127.
- [14] X. Maldague, "Application of infrared Thermography in nondestructive Evaluation", Trends in Optical Nondestructive Testing, Citeseer, 2000.
- [15] Q. Li, E. Liasi, D. L. Simon, R. Du, J. Bujas-Dimitrijevic, A. Chen, "Heating of industrial sewing machine needles: FEA model and verification using IR radiometry", Proc. SPIE, Vol. 3700, 1999.
- [16] P. Cielo, "Pulsed photothermal evaluation of layered materials", Journal of Applied Physics, Volume 56, Issue 1, 1984.
- [17] D. L. Balageas, J. C. Krapez, P. Cielo, "Pulsed photothermal modeling of layered materials", Journal of Applied Physics, 1986.
- [18] J. C. Krapez, P. Cielo, "Optothermal evaluation of oil film thickness on water", Journal of Applied Physics, Vol. 72, Issue 4, p. 1255.
- [19] L. D. Favro, D. J. Kuo, R.L. Thomas, "Inversion of pulsed thermal-wave images for defect sizing and shape recovery", Proceedings of the International Conference on Thermal Sensing and Imaging Diagnostic Applications, Orlando, FL, 1992, pp. 178-181.

- [20] Vavilov V P, D Balageas, G Busse and G M Carlomagno "Thermal non destructive testing: short history and state-of-art", Proc. QIRT 92 (Paris), EETI editions), 1992, pp. 179–193.
- [21] C. Meola, G. M. Carlomagno, "Recent advances in the use of infrared thermography", Measurement Science and Technology, Volume 15, Number 9, 2004 , pp. 27-58.
- [22] X. Maldague, "Introduction to NDT by active infrared thermography", Mater. Eval., 2002, pp. 1060–1073.
- [23] C. Meola, G. M. Carlomagno, L. Giorleo, "Geometrical limitation to detection of defects in composites by means of infrared thermography", J. Nondestruct Eval., 2004.
- [24] Pilla M., Klein M., Maldague X. and Salerno A., "New absolute contrast for pulsed thermography", Proc. QIRT 02 (Dubrovnik), 2002 , pp. 53–58.
- [25] Peng W. and Almond D. "NDE of CFRP composites by transient thermography", Proc. QIRT 98 (Paris), Balageas, 1998.
- [26] D. Wu, Th. Zweschper, A. Salerno, and G. Busse, "Lock-in Thermography for Nondestructive Evaluation of Aerospace Structures", D-70569 Stuttgart-Vaihingen Germany, 1998.
- [27] G. M. Carlomagno and P. G. Berardi, "Unsteady thermotopography in non-destructive testing" Proc. III Infrared Information Exchange (St Louis), 1976, pp. 33–40.
- [28] J. L. Beaudoin, E. Merienne. R. Danjoux and M. Egee, "Numerical system for infrared scanners and application to the subsurface control of materials by photo thermal radiometry" Infrared Technology and Applications. Proc. SPIE , 1985, pp. 287–292.

- [29] P. K. Kuo, Z. J. Feng, T. Ahmed, L. D. Favro, R. L. Thomas and J. Hartikainen, "Parallel thermal wave imaging using a vector lock-in video technique", *Photoacoustic and Photothermal Phenomena*, 1987, pp. 415–418.
- [30] W. Karpen, D. Wu, R. Steegmüller, G. Busse, "Depth profiling of orientation in laminates with local lock-in thermography", Elsevier Science B.V., 2002.
- [31] D. Wu, "Lock-in thermography for defect characterization in veneered wood", *Proc. QIRT 94* ed D. Balageas, 1994, pp. 298–302.
- [32] D. Wu, J. Rantala, W. Karpen, G. Zenzinger, B. Schönbach, W. Rippel, R. Steegmüller, L. Diener and G. Busse, "Applications of lock-in-thermography methods", *Proc. QNDE: Review of Progress in Quantitative Nondestructive Evaluation*, vol. 15, New York, 1996, pp. 511–519.
- [33] J. Rantala, D. Wu and G. Busse, "Amplitude modulated lock-in vibrothermography for NDE of polymers and composites", *Res. Nondestruct. Eval.*, 1996, pp. 215–28.
- [34] D. Wu, A. Salerno, U. Malter, R. Aoki, R. Kochendorfer, P. K. Kachele, K. Woithe, K. Pfister and G. Busse, "Inspection of aircraft structural components using lock-in thermography" *Proc. QIRT 96 (Paris)* ed D. Balageas, 1996, pp. 251–256.
- [35] C. Meola, G. M. Carlomagno, A. Squillace and G. Giorleo, "Non-destructive control of industrial materials by means of lock-in thermography", *Meas. Sci. Technol.* 13, 2002, pp. 1583–90.
- [36] C. Meola, G. Giorleo, L. Nele, A. Squillace and G. M. Carlomagno, "Infrared thermography in the quality assurance of manufacturing systems", *Nondestruct. Test. Eval.* 18, 2002, pp. 83–90.

- [37] C. Meola, G. M. Carlomagno and L. Giorleo, "Non-destructive evaluation of bonded structures with lock-in thermography", *J. Adhesion Sci. Technol.* 17, 2003, pp. 1207–1222.
- [38] C. Meola, G. Giorleo, L. Nele, A. Squillace, "Experimental characterization of an innovative Glare fibre reinforced metal laminate in pin bearing" *J. Compos. Mater.* 37, 2003, pp. 1543–1545.
- [39] C. Meola, L. Nele and L. Giorleo, "Technological characterization of thermoshinking cross-linked polyethylene by destructive and non-destructive techniques" *J. Mater. Proc. Technol.* 133, 2003, pp. 353–358.
- [40] C. Meola, G.M. Carlomagno, U. Prisco and A. Vitiello, "Non-destructive control of polyethylene blanket insulation" *Res. Nondestruct. Eval.* 15, 2004, pp. 55–63.
- [41] G. Busse and A. Rosencwaig, "Subsurface imaging with photoacoustics", *Appl. Phys. Lett.* 36, 1980, pp. 815–816.
- [42] G. Busse, "Optoacoustic and photothermal material inspection techniques" *Appl. Opt.* 21, 1982, pp. 107–110.
- [43] A. Dillenz, G. Busse and D. Wu, "Ultrasound lock-in thermography: feasibility and limitations", *Diagnostic Imaging Technologies and Industrial Applications, Proc. SPIE 3827*, 1999, pp. 10–15.
- [44] X. Maldague, "Pulse phase infrared thermography", *Journal of applied physics*, vol.79, iss. 5, 1996, p. 2694.
- [45] F. Weritz, R. Arndt, M. Röllig, C. Maierhofer and H. Wiggensauser, "Investigation of concrete structures with pulse phase thermography". *Journal of Materials and Structures*, Vol. 38, Number 9, 2006, pp. 843-849.

- [46] X. Maldague, Y. Largouet and J. P. Couturier, "A Study of Defect Depth Using Neural Networks in Pulsed Phase Thermography: Modeling, Noise, Experiments", *Rev. Gén. Therm.* 37, 1998, pp. 704-717.
- [47] F. Galamiche, X. Maldague, "Depth defect retrieval using the wavelet pulsed phased thermography," QIRT 2000 (Quantitative Infrared Thermography), Eurotherm Seminar 64, Reims, France, 2002, pp. 194-199.
- [48] C. Ibarra-Castanedo and X. Maldague, "Defect Depth Retrieval from Pulsed Phase Thermographic Data in Plexiglas and Aluminum Samples", Proceedings Vol. 5405, Thermosense XXVI, 2004, pp. 348-356.
- [49] C. Ibarra Castanedo, N. P. Avdelidis, x. Maldague, "Qualitative and quantitative assessment of steel plates using pulsed phase thermography", *Materials evaluation* ISSN 0025-5327, vol. 63, 2005, pp. 1128-1133.
- [50] X. Maldague, "Theory and Practice of Infrared Technology for Nondestructive Testing", J. Willy publication, 2001, pp. 610-653.
- [51] R. A. Ballingall, "Review of infrared focal plane arrays", *SPIE proc. Infrared Technology and Applications* 1320, 1990, pp. 70-87.
- [52] H. Granicher, "Concise encyclopedia of science and technology", McGraw-Hill, NY, 1984.
- [53] F. D. Shepherd, AC. Yang, "Silicon schottky retains for infrared imaging", *IEDM Technology Digest*, 1973, pp. 310-313.
- [53] F. D. Shepherd, AC. Yang, "Silicon schottky retains for infrared imaging", *IEDM Technology Digest*, 1973, pp. 310-313.
- [54] I. Castanedo. K. Clémente, "Quantitative subsurface defect evaluation by pulsed phase thermography: depth retrieval with the phase". *Society of Photo-Optical Instrumentation Engineers*, 2005.

- [55] G. W. Recktenwald, "Finite-Difference Approximations to the Heat Equation", Class Notes, 2004.
- [56] H. S. Carslaw and J. C. Jaeger, "Conduction of Heat in Solids", 2nd edition, Clarendon Press, Oxford, 1986.
- [57] J.C. Krapez, D. Balageas, A. Deom and F. Lepoutre, "Early Detection by Stimulated Infrared Thermography. Comparison with Ultrasonics and Holo/Shearography", Advances in signal processing for Nondestructive Evaluation of Materials, Kluwer Academic Publishers, Netherlands, 1994, pp. 303-321.
- [58] R. E. Martin, A. L. Gyekenyesia and S. M. Shepard, "Interpreting the Results of Pulsed Thermography Data," Mater. Eval., 61, 2003, pp.611-616.
- [59] D. Wu, G. Busse, "Lock-in thermography for nondestructive evaluation of materials" Revue Generale de Thermique 37, 1998, pp. 693-703.
- [60] L. D. Flavrol and X. Han, "Thermal Wave Materials Characterization and Thermal Wave Imaging", Sensing for Materials Characterization- Processing and Manufacuring, ASNT TONES1, 1998, pp. 399-415.
- [61]G. Busse, D. Wu and W. Karpen, "Thermal Wave Imaging with Phase Sensitive Modulated Thermography," J. Appl. Phys. 71, 1992, pp. 3962-3965.
- [62] D. Wu, "Lockin-Thermographie fur die zerstörungsfreie Werkstoffpriifung und Werkstoffcharakterisierung", Thesis, University of Stuttgart, Germany, 1996.
- [63] C. Ibarra-Castanedo, D. Gonzalez, "Discrete signal transforms as a tool for processing and analyzing pulsed thermographic data", Thermosense XXVIII, Thermosense. Conference. Kissimmee. Florida, USA , 2006..

- [64] H. K. Park, I C. P, Grigoropoulos and A. C. Tam, "Optical Measurements of Thermal Diffusivity of a Material", international Journal of Thermophysics, Vol. 16, No. 4, 1995.
- [65] H. K. Park, C. P. Grigoropoulos, A. C. Tam, «Optical measurements of thermal diffusivity of a material», International Journal of Thermophysics, volume 16, 2005, pp. 973-995.
- [66]. A. Harata, H. Nishimura, and T. Sawada. Appl. Phys. Lett. 57:132 (1990).
- [67]. C. D. Marshall, I. M. Fishman, R. C. Dorfman, C. B. Eom, and M. D. Fayer, "Thermal diffusion, interfacial thermal barrier, and ultrasonic propagation in $\text{YBa}_2\text{Cu}_3\text{O}_{7-x}$ thin films: Surface-selective transient-grating experiments ", phys. Rev. B, volume 45, issue 17, 1992.
- [68] S. V. Vintsents and V. B. Sandomirskii, Phys. Stat. SoL 133:K7 (1992).
- [69] W. J. Parker, "Flash method of determining thermal diffusivity, heat capacity, and thermal conductivity", Journal of applied physics, Vol. 32, Iss.9, 1961, p.1679.
- [70] G. W. Recktenwald, "Finite difference approximations to the heat equation", Class notes, cecs.pdx.ed, 2004.
- [71] M. Necati Ozisik, "Finite difference methods in heat transfer", CRC Press, 1994.
- [72] G. E. Forsythe and W. R. Wasov. "Finite-difference methods for partial differential equations", John Wiley, New York, 1967.
- [73] G. Ghodsi, D. Gavrilov, E. Maeva, R. G. Maev. "An Implicit Finite-Difference model for Thermal Processes Simulation and Surveys", accepted in the 10th QUIRT (Quantitative Infrared Thermography), University of Laval, Quebec, CA, to be held on July 2010.

Vita Auctoris

Ghazal Ghodsi was born in 1983 in Tehran, Iran. She received her Bachelor of Science degree in Physics from the University of Tehran in 2006. After graduation she immigrated to Canada and is currently a candidate for the degree of Master of Science in Physics at the University of Windsor.