A finite element solution of thermal wave propagation in elastic media

Dominic N. Dalo

Follow this and additional works at: http://scholarworks.rit.edu/theses

Recommended Citation
A FINITE ELEMENT SOLUTION OF THERMAL WAVE PROPAGATION IN ELASTIC MEDIA

by

Dominic N. Dalo

A Thesis Submitted

in

Partial Fulfillment

of the

Requirements for the Degree of

MASTER OF SCIENCE

in

Mechanical Engineering

Approved by:

Prof.

(Trial Advisor)

Prof.

Prof.

Prof.

Prof.

(Department Head)

DEPARTMENT OF MECHANICAL ENGINEERING
COLLEGE OF ENGINEERING
ROCHESTER INSTITUTE OF TECHNOLOGY
ROCHESTER, NEW YORK
APRIL, 1987
A Finite Element Solution of Thermal Wave Propagation in Elastic Media.

I Dominic N. Dalo hereby grant permission to the Wallace Memorial Library, of R.I.T., to reproduce my thesis in whole or in part. Any reproduction will not be for commercial use or profit.

3 April 1987
ACKNOWLEDGEMENT

To my Mother and Father, who have provided the support and encouragement to allow me to achieve, I dedicate this work.

To Dr. Hany Ghoneim, my thesis advisor, I wish to express my deepest gratitude and bestow the accolade of being the professor who has most stimulated my desire to learn.

To Drs. William F. Halbleib and Robert A. Ellson, who taught me the fundamentals of mechanics and thermodynamics, my thanks and admiration.

To Drs. Charles W. Haines and Bhalchandra V. Karlekar, my appreciation for providing me with invaluable guidance during my college career.

And finally to the members of my thesis defense committee, I am much obliged for the time they spent reading and evaluating my work.
ABSTRACT

The rational theory of thermodynamics is used to develop equations governing the propagation of thermal waves. A new vector quantity is introduced to account for the thermal waves. The phenomenon of thermal wave propagation is illustrated by considering the problem of one-dimensional heat conduction in a finite slab subjected to a heat flux pulse. The solution is obtained via the finite element method. A discussion of the results and their significance is also presented.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>List of Figures</td>
<td>vi</td>
</tr>
<tr>
<td></td>
<td>Nomenclature</td>
<td>vii</td>
</tr>
<tr>
<td>1.0</td>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1</td>
<td>Historical Review</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>Literature Review</td>
<td>1</td>
</tr>
<tr>
<td>1.3</td>
<td>Approach Taken</td>
<td>3</td>
</tr>
<tr>
<td>1.4</td>
<td>Notation</td>
<td>3</td>
</tr>
<tr>
<td>2.0</td>
<td>THEORY</td>
<td>5</td>
</tr>
<tr>
<td>2.1</td>
<td>List of Fundamental Quantities</td>
<td>5</td>
</tr>
<tr>
<td>2.2</td>
<td>Fundamental Equations of Continuum Mechanics</td>
<td>6</td>
</tr>
<tr>
<td>2.3</td>
<td>Constitutive Relations</td>
<td>7</td>
</tr>
<tr>
<td>2.4</td>
<td>Derivation of Equations</td>
<td>8</td>
</tr>
<tr>
<td>3.0</td>
<td>PROBLEM DESCRIPTION</td>
<td>13</td>
</tr>
<tr>
<td>3.1</td>
<td>Definition of Problem</td>
<td>13</td>
</tr>
<tr>
<td>3.2</td>
<td>Dimensional Analysis</td>
<td>15</td>
</tr>
<tr>
<td>4.0</td>
<td>FINITE ELEMENT IMPLEMENTATION</td>
<td>17</td>
</tr>
<tr>
<td>4.1</td>
<td>Function Approximation</td>
<td>17</td>
</tr>
<tr>
<td>4.2</td>
<td>Weighted Residual</td>
<td>19</td>
</tr>
<tr>
<td>4.3</td>
<td>Temporal Approximation</td>
<td>22</td>
</tr>
<tr>
<td>5.0</td>
<td>RESULTS AND DISCUSSION</td>
<td>25</td>
</tr>
<tr>
<td>5.1</td>
<td>Temperature Response</td>
<td>25</td>
</tr>
<tr>
<td>5.2</td>
<td>Thermal Flux</td>
<td>27</td>
</tr>
<tr>
<td>5.3</td>
<td>Thermal Waves</td>
<td>27</td>
</tr>
<tr>
<td>5.4</td>
<td>Finite Element Program</td>
<td>36</td>
</tr>
<tr>
<td>6.0</td>
<td>CONCLUSIONS</td>
<td>39</td>
</tr>
<tr>
<td>References</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>Appendixes</td>
<td>43</td>
<td></td>
</tr>
<tr>
<td>A.</td>
<td>Formation of Finite Element Matrices</td>
<td>43</td>
</tr>
<tr>
<td>B.</td>
<td>Finite Element Program</td>
<td>53</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Caption</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Geometry and Thermal Conditions</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td>Shape Functions for Linear Elements</td>
<td>18</td>
</tr>
<tr>
<td>3</td>
<td>Thermal Response for Various Values of Alpha</td>
<td>26</td>
</tr>
<tr>
<td>4</td>
<td>Heat Flux Profiles for $\alpha = 10.0, \tau = 0.15$</td>
<td>28</td>
</tr>
<tr>
<td>5</td>
<td>Heat Flux Profiles for $\alpha = 100.0, \tau = 0.05$</td>
<td>29</td>
</tr>
<tr>
<td>6</td>
<td>Heat Flux Profiles for $\alpha = 1000.0, \tau = 0.025$</td>
<td>30</td>
</tr>
<tr>
<td>7</td>
<td>Temperature Profiles for $\alpha = 10.0$</td>
<td>31</td>
</tr>
<tr>
<td>8</td>
<td>Temperature Profiles for $\alpha = 100.0$</td>
<td>32</td>
</tr>
<tr>
<td>9</td>
<td>Temperature Profiles for $\alpha = 1000.0$</td>
<td>33</td>
</tr>
<tr>
<td>10</td>
<td>Temperature Profiles for $\alpha = 1000.0$ after reflection of thermal wave</td>
<td>35</td>
</tr>
</tbody>
</table>
NOMENCLATURE

Zeroth Order Tensors
A  material constant associated with elastic heat flux
a  dimensionless material parameter
b  dimensionless elastic heat flux
C_v  specific heat at constant volume
e  specific internal energy
θ  weighting parameter
κ  thermal diffusivity
k  thermal conductivity
h  length of finite element
ρ  mass density
r  internal heat supply per unit mass
s  specific entropy
τ  dimensionless time
t  time
T  temperature
u  dimensionless temperature
ψ  Helmholtz free energy
ω  frequency of heat pulse
X  dimensionless position
x  position
Q_0  amplitude of heat pulse

First Order Tensors
β_i  elastic heat flux
b_i  dimensionless elastic heat flux
f_i  body force per unit mass
g_i  constitutive relation associated with elastic heat flux
q_i  total heat flux per unit area
φ_i  shape function associated with b
ψᵢ shape function associated with u
xᵢ position
Xᵢ reference position
Bᵢ nodal values of dimensionless elastic heat flux
Uᵢ nodal values of dimensionless temperature

**Higher Order Tensors**

Yᵢⱼ coefficient of thermal expansion tensor
εᵢⱼ total strain tensor
eᵦᵢⱼ elastic strain tensor
σᵢⱼ total stress tensor
kᵢⱼ thermal conductivity tensor
Dᵢⱼⱼⱼ elastic modulus tensor
1.0 INTRODUCTION

1.1 HISTORICAL BACKGROUND

The conduction of heat in materials has long been established as a diffusion process that is directly proportional to the temperature gradient and a material property defined as the thermal conductivity. A series of experiments, the first of which was conducted in the mid-1940's have indicated that thermal disturbances can propagate as waves. Peshkov [1], the earliest of the investigators to observe these thermal waves, introduced the term "second sound" to describe the waves' analogous behavior to ordinary sound waves. Peshkov's experiments were conducted using liquid helium at a temperature of approximately 2 K. Nearly twenty years later, Ackerman and his associates [2] were able to repeat Peshov's findings using solid helium crystals, again at cryogenic temperatures. Similar experiments have been conducted using halogen alkali crystals with no definitive results. Hence, to date, second sound effects have only been observed in non-metallic materials and under very special conditions.

1.2 LITERATURE REVIEW

Due to the nature of the materials and conditions in which the second sound waves have been observed, modern physicists believe that thermal wave propagation is a microscopic phenomenon that falls into the realm of quantum mechanics. Bertman and Sandiford [3] give a non-mathematical, but otherwise cogent argument defending this school of thought. Modern physicists have hypothesized a particle representation of the thermal waves. This description
is very similar to the photon representation of light waves to the extent that thermal particles have been called phonons. Extremely homogeneous materials, such as liquid and solid helium which have had all impurities "frozen out" of them, are thought to promote the propagation of these phonons. Impure or "dirty" materials, which includes all metals, contain so many imperfections that the phonons are scattered in a random fashion resulting in classical heat diffusion. In keeping with quantum theory, second sound effects are proposed to occur only at discrete energy levels i.e., only phonons of specific frequencies can propagate as a wave.

There are those of another school, mainly classical physicists, mathematicians and engineers, who believe that thermal wave propagation can be explained using a continuum approach. Lord and Shulman [4] were among the earliest researchers known to present a generalized theory of thermoelasticity which accounted for the propagation of thermal waves. Their approach is based upon modifying the classical Fourier law of heat conduction to include an additional term which involves the rate of change of the heat flux. This modification generally results in a hyperbolic heat conduction equation as opposed to the standard parabolic form. Ozisik and Vick [5] present a closed form solution to an uncoupled one-dimensional hyperbolic heat conduction equation. Their results indicate that this type of modification to the law of heat conduction does result in a solution that exhibits oscillatory behavior.

Researchers have sought to develop theories which validate the modification of the Fourier's law in this manner. Others have taken different approaches to the development of the equations governing the propagation of thermal waves. Bogy and Naghdi [6] show how equations may be obtained by introducing the time rate of change of temperature as a constitutive variable.
Gurtin and Pipkin [7] develop a theory based on nonlinear materials with memory. Green and Naghdi [8] use a modified form of the Clausius-Duhem statement of the second law of thermodynamics as the basis of their theory. A vector field is introduced to represent the flow of a microscopic excitation thought to originate the thermal waves in the work of Atkin and his associates [9]. While some of preceding works have obtained possible field equations, none have presented any quantitative results for comparative purposes.

1.3 APPROACH TAKEN

In this work, the rational theory of thermodynamics was employed to develop coupled thermoelastic field equations which account for the thermal waves. An additional constitutive relation was introduced to define a new vector quantity associated with the second sound waves. The ensuing partial differential equations along with the fundamental equations of continuum mechanics are sufficient to completely define the state of the body. The propagation of thermal waves is demonstrated by solving an example of one-dimensional heat conduction in a finite slab. Due to the presence of nonlinearities in the governing equations, the solution was obtained via the finite element method. A computer code to implement the finite element algorithm was developed and key results are presented.

1.4 NOTATION

Prior to deriving the governing equations, the notation to be used throughout the remainder of the work will be defined. Since a continuum approach is being used, Cartesian tensors are used to describe various quantities. Among the methods of designating tensorial quantities, the index notation is felt to be the most convenient and concise. A zeroth order tensor, or
scalar, is a variable with no index and possesses only a magnitude. A first order tensor, or vector, is a quantity with a magnitude and a direction and is denoted by a Roman or Greek letter with a lower case Roman subscript. An example of a first order tensor is position, \( x_i \), where the subscript "i" is understood to take on the values \( 1,2,3 \) in that order. A second order tensor is denoted by a variable with two distinct subscripts e.g., the stress tensor, \( \sigma_{ij} \). Third, fourth and higher order tensors are represented by three, four or more distinct subscripts. Note that if a subscript is repeated, the rank or order of the tensor decreases by two i.e., \( \sigma_{ii} \) and \( A_{ijjk} \) are zero and second order tensors respectively.

In order for tensors to be added and subtracted, they must be of the same order. The resultant tensor is of the same order as the original tensors. Multiplication can be performed between tensors of any order. The rank of the product is the sum of any nonrepeating subscripts e.g., the products \( \sigma_{ij} \varepsilon_{ij} \) and \( D_{ijk} \varepsilon_{ij} \) are of ranks zero and two, respectively. Division by a tensor of order one or greater is undefined.

Partial differentiation with respect to time will be denoted by a superimposed dot (\( \cdot \)). Partial differentiation with respect to position will be denoted by indices. Indices used to represent the rank of the tensor before differentiation and the order of differentiation are separated by a comma. Subscripts after the variable but before the comma signify the rank before differentiation and subscripts after the comma denote the order of the differentiation. The rank of a tensor often changes upon differentiation. The rank is again given by the sum of the nonrepeating subscripts, but now includes the subscripts after the comma as well. Examples of this are \( T_{\cdot i} \) and \( \sigma_{ji\cdot j} \) which are both tensors of order one.
2.0 THEORY

The development of the thermoelastic equations proposed to govern the propagation of thermal waves is based on the rational theory of thermodynamics first introduced by Coleman and Noll [10]. The theory requires three basic postulates [11]:

1. A list of the fundamental thermomechanical quantities needed for a complete description of the thermodynamic process in a continuum
2. The basic equations of continuum mechanics i.e., balance of linear momentum, conservation of energy, and the second law of thermodynamics expressed in terms of the quantities in (1)
3. Constitutive assumptions which express some of the quantities in (1) with others in the list.

2.1 LIST OF FUNDAMENTAL QUANTITIES

In compiling a list of quantities, which completely describes a physical phenomenon in a continuum, assumptions must be made as to what must be included and what can be neglected. The scope of the theory will be used to narrow down the selection of the quantities needed in the analysis. Keeping in mind that the proposed theory is to embrace mechanical and thermal behavior of a continuum, the following quantities are proposed. For all temporal field problems the list of independent variables includes a reference position and time. Mechanical variables include the displacement vector, the strain tensor, the stress tensor, and the body force vector. Thermodynamic variables include
absolute temperature, specific entropy, specific internal energy, internal heat supply per unit mass, and the heat flux vector per unit area. In addition to the above, a vector which is associated with the thermal waves is also introduced. This vector will be further quantified later in the analysis. To recapitulate, the list of thermomechanical quantities along with their respective symbols is

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Displacement vector</td>
<td>( x_i )</td>
</tr>
<tr>
<td>Elastic strain tensor</td>
<td>( \varepsilon_{ij} )</td>
</tr>
<tr>
<td>Stress Tensor</td>
<td>( \sigma_{ij} )</td>
</tr>
<tr>
<td>Body force vector</td>
<td>( f_i )</td>
</tr>
<tr>
<td>Absolute temperature</td>
<td>( T )</td>
</tr>
<tr>
<td>Specific Entropy</td>
<td>( s )</td>
</tr>
<tr>
<td>Specific internal energy</td>
<td>( e )</td>
</tr>
<tr>
<td>Internal heat supply</td>
<td>( r )</td>
</tr>
<tr>
<td>Heat flux vector</td>
<td>( q_i )</td>
</tr>
<tr>
<td>&quot;Thermal&quot; vector</td>
<td>( \beta_i )</td>
</tr>
</tbody>
</table>

which are all functions of reference position vector \( X_i \) and time, \( t \).

2.2 FUNDAMENTAL EQUATIONS OF CONTINUUM MECHANICS

The derivation of the governing equations is based upon the fundamental equations of continuum mechanics. The balance of linear momentum is

\[
\sigma_{ji,j} + \rho(f_i - \dddot{x}_i) = 0
\]  

where \( \rho \) is the mass density of the continuum in the in the reference state. The conservation of energy (or first law of thermodynamics) is

\[
\rho \dot{e} = \sigma_{ij} \dot{\varepsilon}_{ij} - q_{i,j} + \rho r.
\]
The form of the second law of thermodynamics that will be used here is that of an entropy production inequality, also known as the Clausius-Duhem inequality [12]

\[ \rho \dot{s} - \rho \frac{r}{T} + \left( \frac{q_i}{T} \right) \geq 0. \]  \hspace{1cm} (2.3)

By introducing the Helmholtz free energy, defined as [13]

\[ \psi = \varepsilon - Ts \]  \hspace{1cm} (2.4)

the specific internal energy, \( \varepsilon \), can be replaced in the list of fundamental quantities by the free energy, \( \psi \). Substituting (2.4) into (2.2), the first law of thermodynamics can be written as

\[ \rho (\psi + Ts + Ts) = a_{ij} \varepsilon_{ij} - q_{i,i} + r. \]  \hspace{1cm} (2.5)

By substituting for \( \rho r \) from (2.5) into (2.3) the second law of thermodynamics becomes

\[ -\rho (\psi + sT) + a_{ij} \varepsilon_{ij} - q_i \frac{T_{i}}{T} \geq 0. \]  \hspace{1cm} (2.6)

Equations (2.5) and (2.6) are the fundamental equations which will be used in the derivation.

### 2.3 Constitutive Relations

Of the list of fundamental quantities defined, it is considered that the absolute temperature, the temperature gradient, the strain tensor, and the internal state variable introduced to account for the thermal wave must be known to completely determine the state of the body. Hence, the remaining quantities in the list form the dependent variables. From this assumption, the
following constitutive relations are proposed

\[ \psi = \psi(T, T', e_{ij}^e, \beta_i) \] (2.7)

\[ s = s(T, T', e_{ij}^e, \beta_i) \] (2.8)

\[ q_i = q_i(T, T', e_{ij}^e, \beta_i) \] (2.9)

\[ \sigma_{ij} = \sigma_{ij}(T, T', e_{ij}^e, \beta_i) \] (2.10)

\[ \dot{\beta}_i = g_i(T, T', e_{ij}^e, \beta_i). \] (2.11)

The modification of standard thermoelasticity is accomplished through (2.11).

At this point it is necessary to further define the nature of the total strain tensor, \( e_{ij} \). The total strain can be separated into two parts i.e., strain due to mechanical loading and strain due to thermal deformation. The mechanical or elastic strain will be denoted by \( e_{ij}^e \) and the thermal strain by \( \gamma_{ij}(T - T_r) \) where \( \gamma_{ij} \) is the coefficient of thermal expansion tensor and \( T_r \) is the reference temperature in the state where no strain is present. The total strain tensor is therefore given by

\[ e_{ij} = e_{ij}^e + \gamma_{ij}(T - T_r). \] (2.12)

2.4 DERIVATION OF EQUATIONS

From the rational theory of thermodynamics, the second law will be combined with the constitutive relations to see if any restrictions can be placed on the dependent variables. Substituting the time derivative of the free energy, \( \psi \) and (2.12) into (2.6) yields
\[ -\rho \left( \frac{\partial \psi}{\partial \dot{T}} + \frac{\partial \psi}{\partial T_{r_i}} \dot{T}_{r_i} + \frac{\partial \psi}{\partial \epsilon_{ij}^e} \epsilon_{ij}^e + \frac{\partial \psi}{\partial \beta_i} \beta_i + s \right) \]

\[ + \sigma_{ij} \dot{\epsilon}_{ij} + \sigma_{ij} \dot{\gamma}_{ij} (T - T_r) + \sigma_{ij} \dot{\gamma}_{ij} T - q_i \frac{T_{r_i}}{T} \geq 0. \]

Rearranging terms, equation (2.13) can be rewritten as

\[ -\rho \left( \frac{\partial \psi}{\partial \dot{T}} + s - \frac{1}{\rho} \sigma_{ij} \gamma_{ij} \right) \dot{T} = \rho \frac{\partial \psi}{\partial T_{r_i}} \dot{T}_{r_i} - \rho \left( \frac{\partial \psi}{\partial \epsilon_{ij}^e} - \frac{1}{\rho} \sigma_{ij} \epsilon_{ij}^e \right) \]

\[ -\rho \frac{\partial \psi}{\partial \beta_i} \beta_i + \sigma_{ij} \dot{\gamma}_{ij} (T - T_r) - q_i \frac{T_{r_i}}{T} \geq 0. \]

Since \( \dot{T}, \dot{T}_{r_i} \) and \( \dot{\epsilon}_{ij}^e \) are unconstrained, the coefficients of these variables must vanish in order for the above inequality to be satisfied. This yields the following relationships

\[ s = -\frac{\partial \psi}{\partial \dot{T}} + \frac{1}{\rho} \sigma_{ij} \gamma_{ij} = s(T, \epsilon_{ij}^e) \]

\[ \sigma_{ij} = \rho \frac{\partial \psi}{\partial \epsilon_{ij}^e} = \sigma_{ij} (T, \epsilon_{ij}^e) \]

\[ \frac{\partial \psi}{\partial T_{r_i}} = 0 \quad \therefore \quad \psi = \psi(T, \epsilon_{ij}^e, \beta_i). \]

Note that the quantities \( s, \sigma_{ij} \) and \( \psi \) are independent of the temperature gradient. Substituting the Helmholtz free energy into the conservation of energy equation yields
\[ \rho \left( \frac{\partial \psi}{\partial T} \frac{\partial \psi}{\partial T} + \frac{\partial \varphi}{\partial e} \varepsilon_{ij}^{e} + \frac{\partial \varphi}{\partial \beta} \beta_{i}^{e} + T_{s} + T_{s}^{*} \right) \]

\[ = \sigma_{ij} \dot{\varepsilon}_{ij}^{e} + \sigma_{ij} \dot{\gamma}_{ij}^{*} (T - T_{r}) + \sigma_{ij} \dot{\gamma}_{ij}^{*} - q_{i,i} + \rho r. \]

or by rearranging

\[ \left( \rho s + \rho \frac{\partial \psi}{\partial T} - \sigma_{ij} \dot{\gamma}_{ij} \right) \frac{\partial \psi}{\partial T} + \left( \rho \frac{\partial \varphi}{\partial e} - \sigma_{ij} \right) \dot{\varepsilon}_{ij}^{e} + \rho \frac{\partial \varphi}{\partial \beta} \dot{g}_{i} \]

\[ + \rho T_{s} = \sigma_{ij} \dot{\gamma}_{ij}^{*} (T - T_{r}) - q_{i,i} + \rho r. \]

Employing the relations (2.15) (2.17) obtained from the second law and assuming that the coefficient of thermal expansion tensor is invariant with respect to time, (2.18) reduces to

\[ \rho \frac{\partial \varphi}{\partial \beta} \dot{g}_{i} + \rho T_{s} = -q_{i,i} + \rho r. \]

Substituting the constitutive relation (2.15) into (2.19) yields

\[ \rho \frac{\partial \varphi}{\partial \beta} \dot{g}_{i} - \rho T \frac{\partial \varphi}{\partial t} \frac{\partial \varphi}{\partial T} + T \dot{\sigma}_{ij} \dot{\gamma}_{ij} = -q_{i,i} + \rho r. \]

Expanding the time derivative of the Helmholtz free energy, (2.20) becomes

\[ \rho \frac{\partial \varphi}{\partial \beta} \dot{g}_{i} + \rho T \left( \frac{\partial \varphi}{\partial t} \frac{\partial \varphi}{\partial t} \frac{\partial \varphi}{\partial T} + \frac{\partial \varphi}{\partial T} \frac{\partial \varphi}{\partial e} \dot{\varepsilon}_{ij}^{e} + \frac{\partial \varphi}{\partial \beta} \dot{g}_{i} \right) \]

\[ + T \dot{\sigma}_{ij} \dot{\gamma}_{ij} = -q_{i,i} + \rho r. \]

Before proceeding any further in the analysis, closed forms of \( q_{i}, \psi_{i} \) and \( g_{i} \) must be defined. From the classical theory of heat conduction, the heat flux is given by Fourier's law [14]
\[ q_i = -k_{ij} T_j \] (2.22)

where \( k_{ij} \) is the thermal conductivity tensor. Chester [15] modified the heat flux law using a Maxwell viscoelastic model applied to heat transfer. Here a Kelvin heat flux model of the form

\[ q_i = -k_{ij} T_j + \beta_i \] (2.23)

is proposed, where \(-k_{ij} T_j\) is the viscous heat flux and \( \beta_i \) is the "elastic" heat flux.

For the Helmholtz free energy function, Green and Naghdi [16] propose that a function in \( \epsilon \epsilon_{ij} \) and \( T \) of the form

\[ \psi = \psi_0 + \frac{1}{2\rho} D_{ijkl} \epsilon_{ij} \epsilon_{kl} - C_v \frac{T}{T_r} \left( \ln \frac{T}{T_r} - 1 \right) \] (2.24)

is a sufficient representation of \( \psi \) where \( \psi_0 \) is the free energy in the reference state, \( D_{ijkl} \) are the elastic moduli and \( C_v \) is the specific heat at constant volume. All the aforementioned material quantities are assumed to be constant. Following the same logic, (2.24) can be extended to include an elastic heat flux term in the form

\[ \psi = \psi_0 + \frac{1}{2\rho} D_{ijkl} \epsilon_{ij} \epsilon_{kl} - C_v \frac{T}{T_r} \left( \ln \frac{T}{T_r} - 1 \right) - \frac{1}{2} m \beta_i \beta_i \] (2.25)

where \( m \) is a material constant.

In order to define \( \beta_i \) (expressed as \( g_i \)), the second law of thermodynamics must once again be invoked. Substituting (2.23) and (2.25) into the second law results in

\[ -\rho m \beta_i \beta_i + k_{ij} \frac{(T_{ij})^2}{T} - \beta_i \frac{T_{ij}}{T} \geq 0 \]
\[-\left(\rho m \dot{\beta}_i + \frac{T_{,i}}{T}\right) \beta_i + k_{ij} \frac{(T_{,i})^2}{T} \geq 0.\] 

(2.26)

or

In order for the inequality to be satisfied the coefficient of $\beta_i$ must vanish. Therefore

\[\dot{\beta}_i = g_i = -\frac{1}{\rho m} \frac{T_{,i}}{T} \] 

(2.27)

For simplicity, let $A = 1/\rho m$. Using (2.16), (2.23), (2.25) and (2.27) equation (2.21) can be expressed as

\[-(k_{ij} T_{,i})_{,j} + \rho C_v \dot{T} + \beta_{i,j} + \frac{T_{,i}}{T} \]

\[= \rho r - TD_{ijkl} c_{ij}^e \gamma_{kl} \] 

(2.28)

Equations (2.1), (2.27) and (2.28) form the coupled balance of momentum and conservation of energy relations which constitute the modified thermoelasticity equations that are proposed to govern the propagation of thermal waves.
3.0 PROBLEM DESCRIPTION

3.1 DEFINITION OF PROBLEM

The intent of this work is to obtain a suitable theory which describes the propagation of thermal waves. In order to simplify the solution of the governing equations, the temperature and strain are assumed to be uncoupled. This assumption is valid for many materials when the rate of strain is small. Furthermore, the problem considered herein is that of one dimensional heat conduction in a finite slab composed of a material with constant thermal properties. The slab is initially at a uniform temperature throughout and no internal heat generation is present. One boundary of the slab is considered to be insulated for all time and the other boundary is subjected to a pulse of heat flux for a short period of time and is then insulated for all time thereafter. While these boundary conditions may seem unrealistic, they have been shown to yield meaningful results by Ozisik and Vick [5]. The solutions for insulated and convective boundary conditions differ only in magnitude and duration i.e., the general trends of the solutions are identical for both. These conditions are illustrated in Figure 1. For these conditions, equations (2.27) and (2.28) proposed to govern heat conduction which includes second sound effects become
$q(0, t) = 0 \quad (t > t_{\text{pulse}})$

$g(x, t) = 0$

$q(L, t) = 0$

$T(x, 0) = T_0$

Figure 1: Geometry and Thermal Conditions
\[ -k \frac{\partial^2 T}{\partial x^2} + \rho C_v \frac{\partial T}{\partial t} + \frac{\partial \beta}{\partial x} + \frac{\beta}{T} \frac{\partial T}{\partial x} = 0 \]  
(3.1)

and

\[ \frac{\partial \beta}{\partial t} = -\frac{A}{T} \frac{\partial T}{\partial x} \text{ in } 0 < x < L \text{ for } t > 0 \]  
(3.2)

the initial and boundary conditions are

\[ t = 0, T(x,0) = T_0 \quad 0 \leq x \leq L \]

and

\[ x = 0, q(0,t) = Q_0 \sin(\omega t) \quad 0 \leq t \leq \frac{\pi}{\omega} \]

\[ q(0,t) = 0 \quad t > \frac{\pi}{\omega} \]

\[ x = L, q(L,t) = 0 \quad t > 0 \]

respectively.

**3.2 DIMENSIONAL ANALYSIS**

In order to simplify the subsequent analysis the following dimensionless quantities are introduced

\[ u = \frac{T}{T_0} \quad ; \text{dimensionless temperature} \]

\[ X = \frac{x}{L} \quad ; \text{dimensionless position} \]

\[ \tau = \frac{\kappa t}{L^2} \quad ; \text{dimensionless time} \]

\[ b = \frac{\beta L}{kT_0} \quad ; \text{dimensionless elastic heat flux} \]

\[ Q = \frac{qL}{kT_0} \quad ; \text{dimensionless (total) heat flux} \]

and \( \kappa = \frac{k}{\rho C_v} \), is the thermal diffusivity of the material. Upon substitution of these quantities, the governing equations become

\[ -\frac{\partial^2 u}{\partial X^2} + \frac{\partial u}{\partial \tau} + \frac{\partial b}{\partial X} + \frac{b}{u} \frac{\partial u}{\partial X} = 0 \]  
(3.3)

and
\[
\frac{\partial b}{\partial t} = -a \frac{1}{u} \frac{\partial u}{\partial X}
\]  

(3.4)

where \(a = AL^2 / \kappa kT_0\). Similarly the equation for the total heat flux becomes

\[
Q = -\frac{\partial u}{\partial X} + b
\]  

(3.5)

Equations (3.3) and (3.4) will be solved by employing the finite element method.
4.0 FINITE ELEMENT IMPLEMENTATION

4.1 FUNCTION APPROXIMATION

The non-dimensional form of the governing equations, as developed in the previous section, are

\[-\frac{\partial^2 u}{\partial X^2} + \frac{\partial u}{\partial t} + \frac{\partial b}{\partial X} + \frac{b}{u} \frac{\partial u}{\partial X} = 0\]  \hfill (3.3)

and

\[\frac{\partial b}{\partial t} = -a \frac{1}{u} \frac{\partial u}{\partial X}\]  \hfill (3.4)

The solution to the above nonlinear, coupled, partial differential equations is accomplished via the Galerkin finite element method [17]. For this analysis linear elements were used for simplicity. The dependent variables \(u\) and \(b\) are approximated by

\[u(X,t) = \sum_{i=1}^{n} \psi_i(X) U_i(t)\]  \hfill (4.1)

and

\[b(X,t) = \sum_{i=1}^{n} \phi_i(X) B_i(t)\]  \hfill (4.2)

where \(\psi_i(X)\) and \(\phi_i(X)\) are the shape functions, \(U_i\) and \(B_i\) are the nodal values of the dimensionless temperature and heat flux respectively and \(n\) is the total number of nodes contained in the model. Figure 2 illustrates the form of typical shape functions used for one-dimensional linear elements.

Substituting (4.1) and (4.2) into (3.3) and (3.4), respectively yields the following expressions, (where the prime and the superimposed dot denote the
Figure 2: Shape Functions for Linear Elements
derivatives with respect to position and time, respectively)

\[- \sum_{i=1}^{n} \psi_i''(X) U_i(t) + \sum_{i=1}^{n} \psi_i'(X) U_i'(t) + \sum_{i=1}^{n} \phi_i'(X) B_i(t) \]

\[+ V^i \sum_{i=1}^{n} \sum_{j=1}^{n} \psi_i'(X) U_i'(t) \phi_j(X) B_j(t) \neq 0 \]  \hspace{1cm} (4.3)

and

\[\sum_{i=1}^{n} \phi_i'(X) B_i(t) + \alpha V^i \sum_{i=1}^{n} \psi_i'(X) U_i(t) \neq 0 \]  \hspace{1cm} (4.4)

Note that expressions (4.3) and (4.4) are no longer exactly equal to zero since approximate representation for u and b are used. Hence, (4.3) and (4.4) are called residuals of their original equations. The variable \( \psi^i \) present in both equations is a function which takes into account the non-linearity i.e., \( l/u \) and is dependent upon the finite element node in question. It is discussed further in Appendix A.

4.2 WEIGHTED RESIDUAL

The next step in the finite element formulation of a problem by function approximation, is to weight the residual so that it may be set equal to zero again. This accomplished by selecting a weighting function, \( W_k \) and multiplying each term in the residual by it. The weighted residual is then integrated over the domain of the continuum in question and set equal to zero. The selection of the weighting function determines the particular subcategory of the finite element method used. When the weighting function is chosen as \( \psi_k(X) \), the procedure is known as the Galerkin method. The Galerkin method is highly popular since the selection of the weighting function is greatly simplified and the resultant coefficient matrices are often symmetric. In
general, each equation is multiplied by a different shape function. Different functions will be shown in this analysis, but both are of the form shown in Figure 2.

Upon weighting and integrating over the domain, expressions (4.3) and (4.4) become

$$
-\int_0^1 \psi_k \sum_{i=1}^n \psi_i U_i \, dX + \int_0^1 \psi_k \sum_{i=1}^n \psi_i U_i' \, dX + \int_0^1 \psi_k \sum_{i=1}^n \phi_i B_i \, dX
$$

(4.5)

$$
+ \int_0^1 \psi_k V^i \sum_{i=1}^n \sum_{j=1}^n \psi_j \phi_i U_j \, dX = 0
$$

and

$$
\int_0^1 \phi_k \sum_{i=1}^n \phi_i B_i \, dX + a \int_0^1 \phi_k V^i \sum_{i=1}^n \psi_i U_i \, dX = 0.
$$

(4.6)

The order of the derivative in the first term of equation (4.5) can be reduced by trading one of the differentiations between the weighting function, $\psi_k$, and the shape function, $\psi_i$. This is accomplished by integrating the first term by parts. Equation (4.5) then becomes

$$
\int_0^1 \psi_k \sum_{i=1}^n \psi_i U_i \, dX - \left[ \psi_k \sum_{i=1}^n \psi_i U_i \right]_0^1 + \int_0^1 \psi_k \sum_{i=1}^n \psi_i U_i' \, dX
$$

(4.7)

$$
+ \int_0^1 \psi_k \sum_{i=1}^n \phi_i B_i \, dX + \int_0^1 \psi_k V^i \sum_{i=1}^n \sum_{j=1}^n \psi_j \phi_i U_j B_j \, dX = 0.
$$

By reversing the order of the integration and summation signs (4.7) and (4.6) can be expressed as
\[
\sum_{i=1}^{n} \left[ \int_{0}^{1} \psi_{k} \psi_{i} \, dX \right] U_{i} + \sum_{i=1}^{n} \left[ \int_{0}^{1} \psi_{k} \psi_{i} \, dX \right] = \sum_{i=1}^{n} \left[ \int_{0}^{1} \psi_{k} \psi_{i} \, dX \right] B_{i} (4.8)
\]

and
\[
\sum_{i=1}^{n} \left[ \int_{0}^{1} \psi_{k} \psi_{i} \, dX \right] B_{i} + a \sum_{i=1}^{n} \left[ \int_{0}^{1} V^{i} \psi_{k} \psi_{i} \, dX \right] U_{i} = 0. (4.9)
\]

A more compact form for (4.8) and (4.9) is
\[
K_{ik} U_{i} + C_{ik} U_{i} + L_{ik} B_{i} + W_{ijk} B_{j} U_{i} = Q_{k} (4.10)
\]

and
\[
D_{ik} B_{i} + aE_{ik} U_{i} = 0 (4.11)
\]

where
\[
C_{ik} = \int_{0}^{1} \psi_{k} \psi_{i} \, dX (4.12a)
\]
\[
D_{ik} = \int_{0}^{1} \psi_{k} \psi_{i} \, dX (4.12b)
\]
\[
E_{ik} = \int_{0}^{1} V^{i} \psi_{k} \psi_{i} \, dX (4.12c)
\]
\[
K_{ik} = \int_{0}^{1} \psi_{k} \psi_{i} \, dX (4.12d)
\]
\[
L_{ik} = \int_{0}^{1} \psi_{k} \phi_{i} \, dX (4.12e)
\]
\[
W_{ijk} = \int_{0}^{1} V^{i} \psi_{k} \psi_{i} \phi_{j} \, dX (4.12f)
\]
and

\[ Q_k = \left[ \psi_k \sum_{i=1}^{n} \psi_i^I U_i \right]_0^1 \quad (4.12g) \]

By multiplying \( W_{ijk} \) by \( B_j \) in (4.10) the equation can be expressed as

\[ K_{ik} U_i + C_{ik} \dot{U}_i + L_{ik} B_i + N_{ik} U_i = Q_k \quad (4.13) \]

where

\[ N_{ik} = W_{ijk} B_j. \]

This form facilitates the implementation of (4.11) and (4.13) into a computer algorithm. The details of the integration necessary to form the coefficient matrices (4.12) are found in Appendix A.

4.3 TEMPORAL APPROXIMATION

With (4.11) and (4.13) the original governing equations, which contained partial derivatives with respect to space and time, have been reduced to two sets of ordinary differential in time. With a further approximation, the temporal derivative can be eliminated to obtain two sets of algebraic equations. For first order time derivatives, the theta method is often employed [17]. This method uses a finite difference scheme to represent the time derivative variable and a weighting technique to obtain an average of the dependent variable at two consecutive time steps. For a single ordinary differential equation of the form

\[ \dot{x} = f(x,t) \]

the theta method takes on the form

\[ x^{i+1} - x^i = \Delta t[\theta f^{i+1} + (1-\theta)f^{i}] \]
where, \( \Delta t \) is the time marching increment, \( \theta \), the weighting parameter and the superscripts \( j \) and \( j+1 \) indicate the values of the variables at a time, \( t \) and a time \( t + \Delta t \) later, respectively. Applying the theta method to (4.13) and (4.11), the expressions become

\[
C_{ik} \left[ U^j_{i} - U^j_{i} \right] + \Delta t M_{ik} \left[ \theta U^{j+1}_{i} + (1 - \theta) U^j_{i} \right] \\
+ \Delta t L_{ik} \left[ \theta B^j_{i} + (1 - \theta) B^j_{i} \right] = \Delta t \left[ \theta Q^{j+1}_{k} + (1 - \theta) Q^j_{k} \right] 
\]

and

\[
D_{ik} \left[ B^j_{i} - B^j_{i} \right] + \alpha \Delta t E_{ik} \left[ \theta U^{j+1}_{i} + (1 - \theta) U^j_{i} \right] = 0
\]

where \( M_{ik} = K_{ik} + N_{ik} \).

Solving (4.14) and (4.15) for \( U^{j+1} \) and \( B^{j+1} \) respectively, yields

\[
\left[ C_{ik} + \Delta t \theta M_{ik} \right] U^{j+1}_{i} = \left[ C_{ik} - \Delta t (1 - \theta) M_{ik} \right] U^j_{i} \\
- \Delta t \theta L_{ik} B^{j+1}_{i} - \Delta t (1 - \theta) L_{ik} B^j_{i} + \Delta t \left[ \theta Q^{j+1}_{k} + (1 - \theta) Q^j_{k} \right]
\]

and

\[
D_{ik} B^{j+1}_{i} = D_{ik} B^j_{i} - \alpha \Delta t \theta E_{ik} U^{j+1}_{i} - \alpha \Delta t (1 - \theta) E_{ik} U^j_{i}
\]

Since the preceding systems of algebraic equations are coupled, they must be solved simultaneously. Solution is obtained by assuming a value of one of the dependent variables and successively iterating between equations until they are satisfied within some error tolerance. A FORTRAN program which performs all of the numerical manipulations necessary to construct and solve the sets of equations given by (4.16) and (4.17) is found in Appendix B.

When employing the theta method for solution of first order differential equations, the analyst must choose a value for the weighting parameter, \( \theta \). Values of \( \theta \) between zero and one inclusive are allowed by the method.
With theta equal to zero and one the familiar forward the backward difference schemes are recovered. Comparisons of solutions to differential equations using various values of theta are performed in Reddy [17] and Burden and Faires [18]. Both references indicate that a value of $\theta = \frac{1}{2}$ generally produces the most accurate solution. When theta equals one-half the "$\theta$"-method is generally known as the Crank-Nicolson finite difference scheme. In addition to greater accuracy, The Crank-Nicolson scheme has the additional benefit of being unconditionally stable. For these reasons the Crank-Nicolson scheme is employed in the FORTRAN code in Appendix B.
5.0 RESULTS

The problem of one-dimensional heat conduction in a finite slab was solved using the finite element method. From the dimensional analysis performed in Section 3.0, all of the material properties have been lumped into the dimensionless parameter alpha, $\alpha$. Solutions were obtained for different values of alpha to demonstrate its effect on heat transfer. Note that for alpha equal to zero the problem reduces to that of classical heat conduction.

5.1 TEMPERATURE RESPONSE

Figure 3 depicts the temperature response with respect to time at a point in the slab located at $X = 1.0$. The response is given for alpha spanning three orders of magnitude for illustrative purposes. For the lowest value of alpha, the temperature response exhibits a behavior that is similar to that predicted by the classical theory of heat conduction. However, the presence of the overshoot in the solution indicates that elastic heat flux is present. Utilizing the terminology associated with vibratory mechanical systems, it can be said that heat conduction with small values of alpha is highly damped and has a low natural frequency. For the intermediate value of alpha shown i.e., $\alpha = 100$, the oscillatory behavior of the temperature becomes more pronounced. This trend continues as alpha is increased.

Increasing alpha in the governing differential equations has the effect of increasing the thermal “stiffness” of the slab. This is analogous to adding a stiffer spring in a mechanical system. The additional stiffness causes the natural frequency of the thermal response to increase and it reduces the total percent damping in the system. In the range of alpha considered, there exists a
Figure 3: Thermal Response for Various Values of Alpha
shift in the temperature response from a dissipative or viscous nature to a conservative or elastic nature as alpha increases.

5.2 THERMAL FLUX

To further illustrate the shift from viscous to elastic behavior in the thermal response, Figures 4-6 give the heat flux profiles within the slab for the three values of alpha considered above. Note that for each value of alpha, the heat fluxes are shown at different instances of time. This was done because the thermal disturbances travel at higher speeds as alpha increases. As defined in Section 2.0, the total heat flux is given by the classical Fourier or viscous term and the newly introduced elastic heat flux. For small alpha, as depicted in Figure 4, both the viscous and elastic heat flux contribute significantly to the total flux. In the median ranges of alpha, the total heat flow is comprised primarily of the elastic heat flux as illustrated in Figure 5. Figure 6 displays the heat flux for large alpha. In this figure, the curves for the total and elastic heat flux are nearly identical indicating that the elastic heat flow is the primary mode of heat conduction. For all of the flux plots shown, the total flux at the boundaries is zero or nearly zero. This agrees with the fact that insulated boundaries were imposed upon the slab. Since both the viscous and elastic heat fluxes were computed using numerical approximations, they do not always add to yield exactly zero at the boundaries.

5.3 THERMAL WAVES

To illustrate the propagation of thermal waves in the slab, temperature profiles at various instants of time are given in Figures 7-9 for the same three values of alpha. The profiles in all three figures show the propagation of the
Figure 4: Heat Flux Profiles for $\alpha = 10.0$ and $\tau = 0.15$
Figure 5: Heat Flux Profiles for $\alpha = 100.0$ and $\tau = 0.05$
Figure 6: Heat Flux Profiles for $\alpha = 1000.0$ and $\tau = 0.025$
Figure 7: Temperature Profiles for $\alpha = 10.0$
Figure 8: Temperature Profiles for $\alpha = 100.0$
Figure 9: Temperature Profiles for $\alpha = 1000.0$
applied pulse through the slab prior to being reflected by the right surface of the slab. A previous remark regarding the increase in wave propagation speed with increasing alpha is clearly shown in this series of figures. For each value of alpha considered, the thermal wave reaches the right wall in a progressively shorter period of time. In all three figures, the temperature profile is given for $\tau = 0.01$. For this instant of time, the thermal pulse is observed to penetrate deeper into the slab for higher values of alpha.

These two facts verify that the thermal wave propagation speed increases as alpha does. This ties into the observation that the thermal "stiffness" increases with alpha. The propagation speed of sound waves in a solid increases as the stiffness or more precisely the elastic modulus, of the material increases. The results of this analysis indicate that the propagation speed of the thermal waves increases as the thermal "stiffness" of the slab material increases. Since these results concur with those for first, or ordinary, sound waves in a solid, the analogy between thermal and mechanical wave propagation becomes more apt.

Figures 7-9 also provide some information on the viscous/elastic nature of the heat flow within the slab. For low alpha, the thermal wave is wide and small in amplitude indicating that the applied pulse has largely diffused out in the slab. As the alpha increases the thermal wave becomes more well defined i.e., it reduces in width and increases in amplitude. This is due to the presence of a larger elastic heat flux which requires a longer time to decay. Figure 10 shows the thermal wave after it has reflected off the right boundary of the slab. Note that the maximum amplitude of the wave has decreased and the minimum temperature in the slab has increased. As the thermal wave travels through the slab, it increases the energy, in the form of increased temperature,
Figure 10: Temperature Profiles for \( \alpha = 1000.0 \) after reflection of thermal wave
of the material in its wake. This phenomenon continues until the wave has completely dissipated and the slab reaches a uniform temperature.

5.4 **FINITE ELEMENT PROGRAM**

Before making any final conclusions, some comments on the finite element model and the FORTRAN code used to solve the problem of thermal wave propagation are in order. Since one-dimensional heat transfer is assumed, the finite element model simply consists of a string of one-dimensional “line” elements. The program allows the slab to be divided into any number of uniform size elements. The number of elements used in the analysis was varied to observe its effect on the solution. For small models of ten to twenty elements, convergence was very rapid i.e., within five iterations. The small models also allowed a large time marching increment to be used while still maintaining a fast convergence rate. Unfortunately, these models do not yield enough information within the slab to produce acceptable figures without curve fitting.

A medium size model of fifty elements was used exclusively to produce all the figures within this work. The convergence rate for this size model is also acceptable but is highly dependent on the time marching interval. For time intervals on the order of 0.01 dimensionless time units, convergence could not be obtained within the maximum number of iterations allowed. The maximum number of iterations built into the code serves as a safety to prevent an infinite loop situation. Due to the large number of mathematical manipulations required for each iteration, the iterations are limited to a maximum of 100. A time interval of 0.001 dimensionless time units produced convergence within the iteration limit specified. It should be pointed out that the higher ranges of
alpha required the largest number of iterations and therefore were used to judge convergence rates. Time intervals on the order of 0.0001 units yielded rapid convergence. In order to keep the total number of iterations to a minimum (total iterations is given by the number of iterations at each time step multiplied by the number of time steps) a time marching increment of 0.005 non-dimensional time units was utilized. Convergence with this increment was within five iterations at each time step.

Large models of 100 elements were experimented with briefly. The results from these models were not any different from an accuracy point-of-view, than the medium sized model. However, when using this number of elements, a much smaller time marching increment, than those discussed above, had to be used. This, coupled with the fact that the number of mathematical operations performed at each iteration is a cubic function of the number of elements used, requires large amounts of computer time to obtain the entire solution. For this reason the major portion of the analysis was performed using a medium sized model.

Mention was made in section 4.0 that the solution was obtained by successively iterating between the two systems of equations until a convergence criteria was satisfied. The criteria used for convergence was that the difference between temperatures obtained at subsequent iterations be less than a specified tolerance. Since the temperature varies at each node, and infinite norm of the temperature difference at respective nodes is obtained to determine the maximum change in temperature between iterations. When this value of the infinite norm of the temperature "vector" becomes less than the tolerance requested, the solution is considered to have been found. The
convergence tolerance used in the FORTRAN code was $5 \times (10)^{-6}$. Note that is the smallest value that can be used with single precision computer arithmetic.

The final portion of the discussion deals with the applied temperature flux at the left boundary. Two general types of functions were experimented with, namely a step input and a half-sine pulse input. The amplitude and duration of the pulse input were varied to observe their effect on the thermal response. The results were again analogous to those of a mechanical system i.e., the amplitude and frequency of the thermal oscillations varied as the input pulse was modified. For all of the figures presented in this section a pulse with an amplitude of 10.0 dimensionless heat flux units was used. The step input was used to verify that the computer code was functioning properly and hence no results with that input are presented here.
6.0 CONCLUSIONS

From the foregoing analysis, it has been shown that it is possible to modify the classical diffusion equation to account for thermal wave propagation. This is accomplished by including a second term in Fourier's law of heat conduction. This second term has been called the "elastic" heat flux vector due to the oscillatory behavior of the temperature within the media. The nature of the solution to the proposed governing equations intimate that the methodology used herein is plausible. However, no experimental evidence exists to verify or dispute this.

In order to perform such a comparison, the temperature field within an insulated slab at consecutive instances of time is needed. For a quasi-static process occurring in a conventional media such as a metal, this data is easily obtainable. Acquisition of this same data in a material that promotes thermal wave propagation is precluded for the following reason. Thermal waves have been observed only in materials with highly ordered microstructures devoid of impurities or discontinuities. The presence of a temperature sensing probe within such a material would introduce a discontinuity thereby disrupting the flow of any thermal waves. Some data could be acquired by using temperature probes at the boundaries of the media to observe temperature fluctuation with time at a single point. With this in formation, the dimensionless parameter, $\alpha$, could be changed by varying the individual quantities which constitute it. Temperature data with the new alpha should be taken at the boundary. These
two sets of data should follow the trends presented in the previous section if the proposed governing equations are correct.

Possible extensions to the work done in this thesis is the solution of the governing equations in two dimensions. Such a solution could provide information on whether thermal waves exhibit constructive/destructive interference like other more familiar waves. This could further the analogy between first (ordinary) and second (thermal) waves. Another obvious extension would be performing experiments while varying the parameter alpha as described in the previous paragraph.
REFERENCES


APPENDIX A

Formation of Finite Element Matrices
In this section, the details of the formation of the system matrices will be expanded upon. The final form of the weighted residuals given in Section 4.0 were

\[
\sum_{i=1}^{n} \left[ \int_{0}^{l} \psi_{k} \phi_i \, dX \right] U_i + \sum_{i=1}^{n} \left[ \int_{0}^{l} \psi_{k} \psi_i \, dX \right] \dot{U}_i + \sum_{i=1}^{n} \left[ \int_{0}^{l} \psi_{k} \phi_i \, dX \right] B_i
\]

(A1)

\[
+ \sum_{i=1}^{n} \sum_{j=1}^{n} \left[ \int_{0}^{l} V^i \psi_{k} \psi_j \, dX \right] B_j U_i = \left[ \psi_{k} \sum_{i=1}^{n} \psi_i U_i \right]_{0}^{l}
\]

and

\[
\sum_{i=1}^{n} \left[ \int_{0}^{l} \psi_{k} \phi_i \, dX \right] \dot{B}_i + \alpha \sum_{i=1}^{n} \left[ \int_{0}^{l} V^i \psi_{k} \psi_i \, dX \right] U_i = 0
\]

(A2)

Note that in equations (A1) and (A2), the limits of integration encompass the entire region occupied by the slab. For a given node, however, the value of the integral is nonzero only for a region “2h” wide centered about the particular node (except for the boundary nodes, where the integral is nonzero only for a region “h” wide to the right or left of the node). This is due to the nature of the shape functions used in the analysis (see Figure 2). Also for a given weighting function \( \psi_k \), the summation over the index “i” contains only three nonzero terms (again except at the boundaries) since the weighting functions used are identical to the shape functions.

As an example of how the matrices are generated, the second “term” in equation (A1) will be expanded in “i” and “k” (i,k = 1,2,3,...,n). Using summation notation, the second matrix term in (A1) is

\[
\sum_{i=1}^{n} \left[ \int_{0}^{l} \psi_{k} \phi_i \, dX \right] U_i
\]

(A3)
where it is understood that the range of "k" is (1,2,3,...,n). For a given value of "k", expression (A3) can be expanded as

\[ \sum_{i=1}^{n} \int_{0}^{1} \psi_k \psi_i d\hat{U}_i = \int_{-h}^{h} \psi_k \psi_{k-1} d\hat{U}_{k-1} \]

\[ + \int_{-h}^{h} \psi_k \psi_{k} d\hat{U}_k + \int_{-h}^{h} \psi_k \psi_{k+1} d\hat{U}_{k+1}. \]  \[ \text{(A4)} \]

Note that when "k" takes on the value 1 (or n), the first (third) term i (A4) is eliminated. Since the shape and weighting functions are discontinuous at the nodes, the integration must be further broken up into two intervals, namely -h to zero and zero to +h. Expression (A4) then becomes

\[ \sum_{i=1}^{n} \int_{0}^{1} \psi_k \psi_i d\hat{U}_i = \int_{-h}^{0} \psi_k \psi_{k-1} dX + \int_{0}^{h} \psi_k \psi_{k-1} dX \left| \hat{U}_{k-1} \right. \]

\[ + \left. \int_{-h}^{0} \psi_k \psi_{k} dX + \int_{0}^{h} \psi_k \psi_{k+1} dX \right| \hat{U}_{k} + \left. \int_{-h}^{0} \psi_k \psi_{k+1} dX \right| \hat{U}_{k+1}. \]  \[ \text{(A5)} \]

**Interior Nodes**

For a general interior node, expression (A5) becomes

\[ \sum_{i=1}^{n} \int_{0}^{1} \psi_k \psi_i d\hat{U}_i = \int_{-h}^{0} \left( 1 + \frac{X}{h} \right) \left( - \frac{X}{h} \right) dX + \int_{0}^{h} \left( 1 - \frac{X}{h} \right) \left( 0 \right) dX \left| \hat{U}_{k-1} \right. \]

\[ + \left. \int_{-h}^{0} \left( 1 + \frac{X}{h} \right)^2 dX + \int_{0}^{h} \left( 1 - \frac{X}{h} \right)^2 dX \right| \hat{U}_{k} \]

\[ + \left. \int_{-h}^{0} \left( 1 + \frac{X}{h} \right) \left( 0 \right) dX + \int_{0}^{h} \left( 1 - \frac{X}{h} \right) \left( \frac{X}{h} \right) dX \right| \hat{U}_{k+1}. \]  \[ \text{(A6)} \]

upon substituting the mathematical equations for the shape and weighting functions. Note that two of the integrals are zero because one of the functions is
zero in the interval of integration. Performing the integrations, expression (A6) takes the form

\[ \sum_{i=1}^{n} \int_{0}^{1} \psi_i \psi_i \, dX \, U_i = \frac{h}{6} \left( 2U_{k-1} + 4U_k + U_{k+1} \right) \]  

(A7)

for the weighting function, \( \psi_k \), associated with an interior node.

**Boundary Nodes**

Expression (A5) takes on a different form for the two nodes located at the boundaries. For the first node (A5) becomes

\[ \sum_{i=1}^{n} \int_{0}^{1} \psi_i \psi_i \, dX \, U_i = \left[ \int_{0}^{h} \psi_i \psi_i \, dX \right] U_1 + \left[ \int_{0}^{h} \psi_1 \psi_1 \, dX \right] U_2 \]

or in mathematical form

\[ \sum_{i=1}^{n} \int_{0}^{1} \psi_i \psi_i \, dX \, U_i = \left[ \int_{0}^{h} \left( 1 - \frac{X}{h} \right)^2 \, dX \right] U_1 \]

(A8)

\[ + \left[ \int_{0}^{h} \left( 1 - \frac{X}{h} \right) \left( \frac{X}{h} \right) \, dX \right] U_2. \]

Similarly, the expression for the last node is

\[ \sum_{i=1}^{n} \int_{0}^{1} \psi_i \psi_i \, dX \, U_i = \left[ \int_{-h}^{0} \psi_i \psi_{i-1} \, dX \right] U_{n-1} + \left[ \int_{-h}^{0} \psi_n \psi_{n-1} \, dX \right] U_n \]

(A9)

\[ = \left[ \int_{-h}^{0} \left( 1 + \frac{X}{h} \right) \left( -\frac{X}{h} \right) \, dX \right] U_{n-1} + \left[ \int_{-h}^{0} \left( 1 + \frac{X}{h} \right)^2 \, dX \right] U_n \]

Upon integrating, (A8) and (A9) become
\[
\sum_{i=1}^{n} \int_{0}^{1} \psi_{i} \psi_{i} \, dX \, U_{i} = \frac{h}{6} \left( 2U_{1} + U_{2} \right) \quad (A10)
\]

and

\[
\sum_{i=1}^{n} \int_{0}^{1} \psi_{i} \psi_{i} \, dX \, U_{i} = \frac{h}{6} \left( U_{n-1} + 2U_{n} \right) \quad (A11)
\]

respectively. Rewriting expressions (A7), (A10) and (A11) in matrix notation yields

\[
\begin{bmatrix}
2 & 1 \\
1 & 4 & 1 \\
\vdots & \ddots & \ddots \\
1 & 4 & 1 \\
1 & 4 & 1 \\
1 & 2
\end{bmatrix}
\begin{bmatrix}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots
\end{bmatrix}
\begin{bmatrix}
\dot{U}_{1} \\
\dot{U}_{2} \\
\dot{U}_{3} \\
\vdots \\
\dot{U}_{n-1} \\
\dot{U}_{n} \\
\dot{U}_{n+1}
\end{bmatrix}
\]

The variable \( V_{i} \), mentioned in Section 4.0, takes into account the nonlinear temperature term in both governing equations. For a given shape function, \( \psi_{i} \), the variable \( V_{i} \) is defined as
\[ V^i = \begin{cases} \frac{2}{U_{i-1} + U_i} & -h \leq X < 0 \\ \frac{2}{U_i + U_{i+1}} & 0 \leq X \leq h \end{cases} \]  \hspace{1cm} (A12)

where the two sets of limits represent the two intervals of the shape function. This form of the variable, \( V^i \), is an average of the temperature at the "i"th node and the two nodes directly adjacent to it.

The single term on the right hand side of equation (A1) represents the boundary conditions of the problem. Utilizing equation (4.1), this term can also be written as

\[ \left[ \psi_k \sum_{i=1}^{n} \psi_i U_i \right]_0^1 = \left[ \psi_k \frac{\partial u}{\partial X} \right]_0^1 \]  \hspace{1cm} (A13)

Note that (A13) is a vector quantity with all elements equal to zero except at the boundary nodes. For the first node (A13) becomes

\[ \left[ \psi_1 \frac{\partial u}{\partial X} \right]_0^1 = \left[ \psi_1 \frac{\partial u}{\partial X} \right]_{X=1} - \psi_1 \frac{\partial u}{\partial X} \bigg|_{X=0} \]  \hspace{1cm} (A14)

The first term in (A14) is zero because the shape function, \( \psi_1 \), is zero at \( X = 1 \) and the second term is simply \( -\partial u/\partial X \big|_{X=0} \) since \( \psi_1 = 1 \) at \( X = 0 \). Substituting the definition of the total heat flux (equation 3.5) into (A14) yields

\[ \left[ \psi_1 \frac{\partial u}{\partial X} \right]_0^1 = Q \bigg|_{X=0} - b \bigg|_{X=0} = Q_1 - B_1 \]  \hspace{1cm} (A15)

where \( Q_1 \) is the heat flux applied to the left boundary of the slab as defined in Section 3.0 and \( B_1 \) is the nodal value of the "elastic" heat flux at the first node.
In the same manner the conditions at the right boundary are

\[
\left[ \psi_n \frac{du}{dx} \right]_{0}^{1} = Q \left|_{x=1} \right. - b \left|_{x=1} \right.
= Q_n - B_n
\tag{A16}
\]

Since the total heat flux at the right wall is zero for all time as defined by the boundary conditions in Section 3.0, (A16) is simply

\[
\left[ \psi_n \frac{du}{dx} \right]_{0}^{1} = -B_n
\tag{A17}
\]

Expanding the remaining terms in equations (A1) and (A2) as outlined above results in the two systems of matrix equations presented on the following pages. The presence of \( U_i \) and \( B_i \) in both the matrix and vector elements of these equations results in a nonlinear system.
\[
C_{ik} \dot{U}_i + K_{ik} U_i + L_{ik} B_i + N_{ik} \dot{U}_i = (f_k)_b
\]

\[
\begin{bmatrix}
2 & 1 \\
1 & 4 & 1 \\
& 1 & 4 & 1
\end{bmatrix}
\begin{bmatrix}
\dot{U}_1 \\
\dot{U}_2 \\
\dot{U}_3 \\
\vdots \\
\dot{U}_{n-1} \\
\dot{U}_n \\
\dot{U}_{n+1}
\end{bmatrix}
= \frac{h}{6}
\]

\[
\begin{bmatrix}
1 & -1 \\
-1 & -2 & -1 \\
\ddots & \ddots & \ddots & \ddots \\
& \ddots & \ddots & \ddots & \ddots \\
& & \ddots & \ddots & \ddots & \ddots \\
1 & -1 & -1 & -1 & -1 & -1
\end{bmatrix}
\begin{bmatrix}
B_1 \\
B_2 \\
B_3 \\
\vdots \\
B_{n-1} \\
B_n \\
B_{n+1}
\end{bmatrix}
+ \frac{1}{h}
\]

\[
\begin{bmatrix}
-1 & 1 \\
-1 & 0 & 1 \\
\ddots & \ddots & \ddots & \ddots \\
& \ddots & \ddots & \ddots & \ddots \\
& & \ddots & \ddots & \ddots & \ddots \\
-1 & 1 & 1 & 1 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
U_1 \\
U_2 \\
U_3 \\
\vdots \\
U_{n-1} \\
U_n \\
U_{n+1}
\end{bmatrix}
+ \frac{1}{2}
\]
\[ D_{ik} \dot{B}_i + a E_{ik} U_i = 0 \]

\[
\begin{bmatrix}
A \\
\vdots \\
U_{n+1}
\end{bmatrix}
\begin{bmatrix}
\dot{B}_1 \\
\dot{B}_2 \\
\dot{B}_3 \\
\vdots \\
\dot{B}_n \\
\dot{B}_{n+1}
\end{bmatrix}
+ a
\begin{bmatrix}
\frac{-1}{U_1 + U_2} & \frac{1}{U_1 + U_2} \\
\frac{-1}{U_1 + U_2} & \frac{1}{U_1 + U_2} & \frac{-1}{U_2 + U_3} & \frac{1}{U_1 + U_2} \\
\vdots & \vdots & \ddots & \ddots \\
\frac{-1}{U_{n-1} + U_n} & \frac{1}{U_{n-1} + U_n} & \frac{-1}{U_{n-1} + U_n} & \frac{1}{U_{n-1} + U_n} & \frac{-1}{U_n + U_{n+1}} & \frac{1}{U_n + U_{n+1}} \\
\frac{-1}{U_n + U_{n+1}} & \frac{1}{U_n + U_{n+1}} & \frac{-1}{U_n + U_{n+1}} & \frac{1}{U_n + U_{n+1}} & \frac{-1}{U_n + U_{n+1}} & \frac{1}{U_n + U_{n+1}} \\
\end{bmatrix}
\begin{bmatrix}
U_1 \\
U_2 \\
U_3 \\
\vdots \\
U_n \\
U_{n+1}
\end{bmatrix}
= 0
APPENDIX B

Finite Element Program
PROGRAM NAME: NONLINFE

PROGRAMMER: DOMINIC N. DALO

DATE WRITTEN: APRIL 1986

OBJECTIVE: CALCULATE THE TEMPORAL SOLUTION TO THE PROBLEM OF THERMAL WAVE PROPAGATION IN SOLIDS USING THE FINITE ELEMENT METHOD.

DESCRIPTION OF VARIABLES:

SUCCESS - FLAG TO INDICATE CONVERGENCE HAS BEEN ACHIEVED
I - DO LOOP COUNTER
J - DO LOOP COUNTER
IT - NUMBER OF ITERATIONS
ITMAX - MAXIMUM NUMBER OF ITERATIONS
KTEST - VARIABLE USED TO CONTROL PRINTING SEQUENCE
KSTOR - VARIABLE USED TO CONTROL PRINTING SEQUENCE
N - NUMBER OF NODES USED (SIZE OF MATRICES AND VECTORS)
MAXN - MAXIMUM NUMBER OF NODES TO BE USED
ALPH - DIMENSIONLESS MATERIAL CONSTANT
DELT - DIMENSIONLESS TIME MARCHING INTERVAL
THET - VALUE OF THETA USED IN THE "THETA" METHOD
H - DIMENSIONLESS LENGTH OF FINITE ELEMENTS
EPS - CONVERGENCE CRITERIA
UI - INITIAL VALUE OF DIMENSIONLESS TEMPERATURE
BI - INITIAL VALUE OF DIMENSIONLESS "ELASTIC" HEAT FLUX
T - DIMENSIONLESS TIME
TMAX - MAXIMUM DIMENSIONLESS TIME FOR WHICH THE SOLUTION IS TO BE COMPUTED
TAU - DURATION OF FORCING FUNCTION IN DIMENSIONLESS TIME UNITS
TPRINT - VARIABLE USED TO CONTROL PRINTING SEQUENCE
BJ - DIMENSIONLESS "ELASTIC" HEAT FLUX VECTOR
BJ1 - DIMENSIONLESS "ELASTIC" HEAT FLUX VECTOR AT A TIME, DELT LATER
UJ - DIMENSIONLESS TEMPERATURE VECTOR
UJ1 - DIMENSIONLESS TEMPERATURE VECTOR AT A TIME, DELT LATER
FT - FORCING FUNCTION VECTOR
QV - VISCOUS HEAT FLUX VECTOR
QT - TOTAL HEAT FLUX VECTOR
X - POSITION VECTOR
RHSB -
RHSU -
BTEST - TEMPORARY WORK VECTORS USED
UTEST - WITHIN THE PROGRAM
V1 -
V2 -
V3 -
C - COEFFICIENT MATRIX ASSOCIATED WITH THE PARTIAL DERIVATIVE OF TEMPERATURE WITH RESPECT TO TIME
COEFFICIENT MATRIX ASSOCIATED WITH THE
PARTIAL DERIVATIVE OF THE DIMENSIONLESS
"ELASTIC" HEAT FLUX WITH RESPECT TO TIME

- COEFFICIENT MATRIX ASSOCIATED WITH THE
DIMENSIONLESS TEMPERATURE

COEFFICIENT MATRIX ASSOCIATED WITH THE
DIMENSIONLESS "ELASTIC" HEAT FLUX

- COEFFICIENT MATRIX ASSOCIATED WITH THE
DIMENSIONLESS TEMPERATURE


TEMPORARY WORK MATRICES USED
WITHIN THE PROGRAM

PROGRAM NONLINFE
IMPLICIT NONE
PARAMETER MAXN=101
LOGICAL SUCCESS
INTEGER*4 I,J,N,IT,ITMAX,KTEST,KSTOR
REAL*4 ALPH,DELT,H,EPS,UI,BI,T,TMAX,TAU,TPRINT
REAL*4 BJ(MAXN),BJ1(MAXN),UJ(MAXN),UJ1(MAXN),FT(MAXN),
& RHSB(MAXN),RHSU(MAXN),BTST(MAXN),V1(MAXN),V2(MAXN),V3(MAXN),
& UTEST(MAXN),QU(MAXN),QT(MAXN),X(MAXN)
REAL*4 C(MAXN,MAXN),D(MAXN,MAXN),E(MAXN,MAXN),K(MAXN,MAXN),
& L(MAXN,MAXN),NB(MAXN,MAXN),NB(MAXN,MAXN),AA(MAXN,MAXN),
& BB(MAXN,MAXN),CC(MAXN,MAXN),DD(MAXN,MAXN),HH(MAXN,MAXN),
& NN(MAXN,MAXN)
EXTERNAL EM,NM,TRI,SXMMULT,MATADD,MATMUL2

INITIALIZE VARIABLES
IT=0
T=0.0
KTEST=1
KSTOR=1

READ IN THE INPUT DATA
READ(5,*) N,ALPH,DELT,TMAX,TAU,THET,EPS,UI,BI,SUCCESS,ITMAX,
& TPRINT
CALCULATE THE DIMENSIONLESS LENGTH OF THE FINITE ELEMENTS
H=1.0/FLOAT(N-1)

CALCULATE THE POSITION VECTOR
X(1)=0.0
DO 5 I=2,N
   X(I)=X(I-1)+H
5    ENDDO

WRITE OUT THE INPUT DATA
WRITE(6,100) N,H,ALPH,DELT,TMAX,THET,EPS
100  FORMAT('0', 'NUMBER OF ELEMENTS USED - ',I3,/,,
 & ' LENGTH OF EACH ELEMENT  H   ',F7.3,/, ' MATERIAL CONSTANT
 & ' ALPHA = ',F10.5,/, ' TIME MARCHING INTERVAL  DELT - ',F8.4,/,,
 & ' MAXIMUM TIME  TMAX - ',F7.3,/, ' THETA = ',F7.3,/,,
 & ' CONVERGENCE CRITERIA EPSILON - ',E10.3,/)}

INITIALIZE THE SOLUTION VECTORS
CALL INIT(MAXN,N,BJ,UJ,UJ1,BI,UI)

FORM THE [C], [D], [K] AND [L] COEFFICIENT MATRICES
CALL CDKL(MAXN,N,H,C,D,K,L)
L(1,1)=L(1,1)+1.0
L(N,N)=L(N,N)-1.0

BEGIN TIME MARCHING
DO WHILE (T.LE.TMAX)
   T=T+DELT
* ITERATE TO FIND THE SOLUTION AT EACH TIME STEP UNTIL THE SOLUTIONS
* CONVERGE
   DO WHILE (.NOT.SUCCESS)
   * INCREMENT THE ITERATION COUNTER
   IT=IT+1
* FORM THE SET OF ALGEBRAIC EQUATIONS USED TO SOLVE FOR THE
* DIMENSIONLESS "ELASTIC" HEAT FLUX
   CALL BEQ(MAXN,N,D,E,HH,NN,RHSB,BJ,UJ,UJ1,V1,V2,V3,
   & DELT,THET,ALPH)
* SOLVE FOR THE DIMENSIONLESS "ELASTIC" HEAT FLUX
   CALL SOLVE(MAXN,N,D,RHSB,BJ1,V1,V2,V3)
* FORM THE SET OF ALGEBRAIC EQUATIONS USED TO SOLVE FOR THE
* DIMENSIONLESS TEMPERATURE
   CALL UEQ(MAXN,N,C,K,L,NB,KNB,AA,BB,CC,DD,FT,RHSU,BJ,BJ1,UJ,UJ1,
   & V1,V2,V3,DELT,THET,T,TAU)
* SOLVE FOR THE DIMENSIONLESS TEMPERATURE
   CALL SOLVE(MAXN,N,AA,RHSU,UJ1,V1,V2,V3)
* CHECK FOR CONVERGENCE
CALL CONVERGE(MAXN,N,UJ1,UTEST,EPS,SUCCESS)

* PRINT OUT THE SOLUTION AT THE CURRENT TIME INCREMENT *

IF (.NOT.SUCCESS) THEN
DO 10 I=1,N
 UTEST(I)=UJ1(I)
 BTTEST(I)=BJ1(I)
10 ENDDO
ELSE
IF (KTEST/10 .NE. KST0R) THEN
 CALL FLUX(MAXN,N,BJ1,UJ1,QV,QT,H)
 WRITE(6,11) T, IT
11 FORMAT('0', SOLUTION AT TIME, T =',F7.4, ' AFTER',I3, &  ' ITERATION(S)')
 WRITE(6,12) (I,UJ1(I),I=1,N)
 WRITE(6,13) (I,BJ1(I),I=1,N)
 WRITE(6,14) (I,QV(I),I=1,N)
 WRITE(6,15) (I,QT(I),I=1,N)
12 FORMAT( ' ',4(1X,'U(',I2,')=' ,E11.4))
13 FORMAT( ' ',4(1X,'B(',I2,')=' ,E11.4))
14 FORMAT( ' ',4(1X,'QV(',I2,')=' ,E11.4))
15 FORMAT( ' ',4(1X,'QT(',I2,')=' ,E11.4))
 KST0R=KTEST/10
 ENDIF
 DO 20 I=1,N
 BJ(I)=BJ1(I)
 UJ(I)=UJ1(I)
20 ENDDO
ENDIF

* CHECK IF MAXIMUM NUMBER OF ITERATIONS HAS BEEN EXCEEDED *

IF (IT.GT. ITMAX) THEN
 WRITE(6,*),(MAXIMUM NUMBER OF ITERATIONS EXCEEDED), SUCCESS=.TRUE.
 ENDIF
ENDDO

* RESET VARIABLES FOR DETERMINING SOLUTION AT NEXT TIME STEP *

IT=0
 SUCCESS=.FALSE.
 KTEST=KTEST+1
 ENDDO
STOP
END
MODULE NAME : INIT

PROGRAMMER : DOMINIC N. DALO

DATE WRITTEN : APRIL 1986

OBJECTIVE : INITIALIZE THE SOLUTION VECTORS ACCORDING TO THE INITIAL CONDITIONS SELECTED

DESCRIPTION OF VARIABLES :

BI INITIAL VALUE OF THE DIMENSIONLESS "ELASTIC" HEAT FLUX VECTOR
UI INITIAL VALUE OF THE DIMENSIONLESS TEMPERATURE VECTOR
BJ DIMENSIONLESS "ELASTIC" HEAT FLUX VECTOR AT A GIVEN TIME
UJ DIMENSIONLESS TEMPERATURE VECTOR AT A GIVEN TIME
UJ1 DIMENSIONLESS TEMPERATURE VECTOR AT A TIME, DELT (delta t) LATER

SUBROUTINE INIT(MAXN,N,BJ,UJ,UJ1,BI,UI)

IMPLICIT NONE

INTEGER*4 I,N,MAXN

REAL*4 BI,UI,BJ(MAXN),UJ(MAXN),UJ1(MAXN)

INITIALIZE VECTORS

DO 10 I=1,N
BJ(I)=BI
UJ(I)=UI
UJ1(I)=UI
10 ENDDO
RETURN
END
SUBROUTINE CDKL(MAXN,N,H,C,D,K,L)
IMPLICIT NONE
INTEGER*4 I,J,N,MAXN
REAL*4 H,C1,K1,L1,C(MAXN,MAXN),D(MAXN,MAXN),K(MAXN,MAXN),L(MAXN,MAXN)

INITIALIZE ALL THE ELEMENTS IN THE MATRICES TO ZERO
DO 10 I=1,N
DO 10 J=1,N
C(I,J)=0.0
D(I,J)=0.0
K(I,J)=0.0
L(I,J)=0.0
10 ENDDO

CALCULATE THE SCALING FACTORS FOR THE MATRICES
C1=H/6.0
K1=1.0/H
L1=0.5

FORM THE FIRST ROW OF THE MATRICES
C(1,1)=2.0*C1
C(1,2)=C1
D(1,1)=C(1,1)
D(1,2)=C1
K(1,1)=K1
K(1,2)=-K1
L(1,1)=-L1
L(1,2)=L1

FORM THE CENTER ROWS OF THE MATRICES
DO 20 I=2,N-1
C(I,I-1)=C1
C(I,I)=4.0*C1
C(I,I+1)=C1
D(I,I-1)=C1
D(I,I)=C(I,I)
D(I,I+1)=C1
K(I,I-1)=-K1
K(I,I)=2.0*K1
K(I,I+1)=-K1
L(I,I-1)=-L1
L(I,I+1)=L1

20 ENDDO

* FORM THE LAST ROW OF THE MATRICES

C(N,N-1)=C1
C(N,N)=2.0*C1
D(N,N-1)=C1
D(N,N)=C(N,N)
K(N,N-1)=-K1
K(N,N)=K1
L(N,N-1)=-L1
L(N,N)=L1
RETURN
END
****

**MODULE NAME**: EM

**PROGRAMMER**: DOMINIC N. DALO

**DATE WRITTEN**: APRIL 1986

**OBJECTIVE**: FORM THE [E] MATRIX DESCRIBED IN THE MAIN PROGRAM

**DESCRIPTION OF VARIABLES**:

- **A** - SCALING FACTOR FOR [E] MATRIX
- **U** - DIMENSIONLESS TEMPERATURE VECTOR AT CURRENT INCREMENT OF TIME

**SUBROUTINE EM(MAXN,N,A,U,E)**

**IMPLICIT NONE**

**INTEGER** I,J,N,MAXN

**REAL** A,U(MAXN),E(MAXN,MAXN)

**INITIALIZE THE ELEMENTS IN THE MATRIX TO ZERO**

```
DO 10 I=1,N
  DO 10 J=1,N
    E(I,J)=0.0
10  ENDDO
```

**FORM THE FIRST ROW OF THE MATRIX**

```
E(1,1)=-A/(U(1)+U(2))
E(1,2)=-E(1,1)
```

**FORM THE CENTER ROWS OF THE MATRIX**

```
DO 20 I=2,N-1
  E(I,I-1)=-A/(U(I-1)+U(I))
  E(I,I)=A/(U(I)+U(I+1))
  E(I,I+1)=-(E(I,I-1)+E(I,I+1))
20  ENDDO
```

**FORM THE LAST ROW OF THE MATRIX**

```
E(N,N-1)=-A/(U(N-1)+U(N))
E(N,N)=-E(N,N-1)
RETURN
END
```
SUBROUTINE NM(MAXN,N,B,U,NB)

IMPLICIT NONE

INTEGER*4 I,J,N,MAXN
REAL*4 B(MAXN),U(MAXN),NB(MAXN,MAXN)

INITIALIZE THE ELEMENTS OF THE [NB] MATRIX TO ZERO

DO 10 I=1,N
  DO 10 J=1,N
    NB(I,J)=0.0
  10 ENDDO

FORM THE FIRST ROW OF THE MATRIX

NB(1,1)=-(2.0*B(1)+B(2))/(3.0*(U(1)+U(2)))
NB(1,2)=-NB(1,1)

FORM THE CENTER ROWS OF THE MATRIX

DO 20 I=2,N-1
  NB(I,I-1)=-(B(I-1)+2.0*B(I))/(3.0*(U(I-1)+U(I)))
  NB(I,I+1)=(2.0*B(I)+B(I+1))/(3.0*(U(I)+U(I+1)))
  NB(I,I)=-(NB(I,I-1)+NB(I,I+1))
20 ENDDO

FORM THE LAST ROW OF THE MATRIX

NB(N,N-1)=-(B(N-1)+2.0*B(N))/(3.0*(U(N-1)+U(N)))
NB(N,N)=-NB(N,N-1)
RETURN
END
MODULE NAME : BEQ

PROGRAMMER : DOMINIC N. DALO

DATE WRITTEN : APRIL 1986

OBJECTIVE : FORM A SET OF ALGEBRAIC EQUATIONS OF THE FORM

\[ A_j(x) \{ b \} \]

WHICH ARE USED TO SOLVE FOR THE DIMENSIONLESS "ELASTIC" HEAT FLUX VECTOR

DESCRIPTION OF VARIABLES :

CONS SCALING FACTOR INTERNAL TO SUBROUTINE

SUBROUTINE BEQ(MAXN,N,D,E,HH,NN,RHSB,BJ,UJ,UJ1,HHV,DV,NNV,
&DDEL,THE,T,ALPH)

IMPLICIT NONE

INTEGER*4 I,N,MAXN

REAL*4 ALPH,DEL,T,THE,CONS

REAL*4 BJ(MAXN),UJ(MAXN),UJ1(MAXN),RHSB(MAXN),HHV(MAXN),
&DV(MAXN),NNV(MAXN)

REAL*4 D(MAXN,MAXN),E(MAXN,MAXN),HH(MAXN,MAXN),NN(MAXN,MAXN)

CALCULATE THE \( E_ij \) MATRIX FOR THE CURRENT INCREMENT OF TIME

CALL EM(MAXN,N,ALPH,UJ1,E)

FORM THE RIGHT HAND SIDE OF THE SET OF ALGEBRAIC EQUATIONS USED TO SOLVE FOR THE DIMENSIONLESS "ELASTIC" HEAT FLUX

CONS=-DEL*THE
CALL SXMMULT(MAXN,MAXN,N,N,CONS,E,HH)
CONS=-DEL*(1.0-THET)
CALL SXMMULT(MAXN,MAXN,N,N,CONS,E,NN)
CALL MATMUL2(MAXN,MAXN,1,N,1,HH,UJ1,HHV)
CALL MATMUL2(MAXN,MAXN,1,N,1,D,BJ,DV)
CALL MATMUL2(MAXN,MAXN,1,N,1,NN,UJ,NNV)
CALL MATADD(MAXN,1,N,1,HHV,DV,RHSB)
CALL MATADD(MAXN,1,N,1,RHSB,NNV,RHSB)
RETURN
END
MODULE NAME : UEQ
PROGRAMMER : DOMINIC N. DALO
DATE WRITTEN : APRIL 1986

OBJECTIVE : FORM A SET OF ALGEBRAIC EQUATIONS OF THE FORM

\[ \mathbf{A}(x) - \mathbf{b} \]

WHICH WILL BE USED TO SOLVE FOR THE DIMENSIONLESS TEMPERATURE VECTOR FOR THE CURRENT INCREMENT OF TIME

DESCRIPTION OF VARIABLES :

CONS SCALING FACTOR INTERNAL TO SUBROUTINE

SUBROUTINE UEQ(MAXN,N,C,K,L,NB,KNB,AA,BB,CC,DD,FT,RHSU,
& BJ,BJ1,UJ,UJ1,BBV,CCV,DDV,DELT,THET,T,TAU)

IMPLICIT NONE

INTEGER*4 I,N,MAXN
REAL*4 DELT,THET,CONS,T,TAU

REAL*4 FT(MAXN),RHSU(MAXN),BJ(MAXN),BJ1(MAXN),UJ(MAXN),
& UJ1(MAXN),BBV(MAXN),CCV(MAXN),DDV(MAXN)

REAL*4 C(MAXN,MAXN),K(MAXN,MAXN),L(MAXN,MAXN),NB(MAXN,MAXN),
& KNB(MAXN,MAXN),AA(MAXN,MAXN),BB(MAXN,MAXN),CC(MAXN,MAXN),
& DD(MAXN,MAXN)

CALCULATE THE \[ \mathbf{L} \mathbf{B} \mathbf{J} \] MATRIX FOR THE CURRENT INCREMENT OF TIME

CALL NM(MAXN,N,BJ1,UJ1,NB)

FORM THE RIGHT HAND SIDE OF THE SET OF ALGEBRAIC EQUATIONS

CALL MATADD(MAXN,MAXN,N,N,K,NB,KNB)
CONS=DELT*THET
CALL SXMULT(MAXN,MAXN,N,N,CONS,KNB,AA)
CONS=-CONS
CALL SXMULT(MAXN,MAXN,N,N,CONS,L,CC)
CONS=-DELT*(1.0-THET)
CALL SXMULT(MAXN,MAXN,N,N,CONS,KB,CC)
CONS=KS
CONS=-DELT*(1.0-THET)
CALL SXMULT(MAXN,MAXN,N,N,CONS,BB,CC)
CONS=KS
CONS=-DELT*(1.0-THET)
CALL SXMULT(MAXN,MAXN,N,N,CONS,DD,CC)

COMPUTE THE VALUE OF THE FORCING FUNCTION FOR THE CURRENT INCREMENT OF TIME

CALL FTV(MAXN,N,FT,T,DELT,THET,TAU)
CALL MATMUL2(MAXN,MAXN,1,N,1,BB,UBV)
CALL MATMUL2(MAXN, MAXN, 1, N, N, 1, CC, BJ, CCV)
CALL MATMUL2(MAXN, MAXN, 1, N, N, 1, DD, BJ, DDV)
CALL MATADD(MAXN, 1, N, 1, FT, BBV, RHSU)
CALL MATADD(MAXN, 1, N, 1, RHSU, CCV, RHSU)
CALL MATADD(MAXN, 1, N, 1, RHSU, DDV, RHSU)
RETURN
END
OBJECTIVE: COMPUTE THE FORCING FUNCTION FOR THE CURRENT INCREMENT OF TIME

DESCRIPTION OF VARIABLES:

- **AMP**: MAXIMUM AMPLITUDE OF THE FORCING FUNCTION
- **OMEG**: FREQUENCY OF THE FORCING FUNCTION
- **TAU**: DURATION OF THE FORCING FUNCTION
- **FT**: FORCING FUNCTION VECTOR

```fortran
SUBROUTINE FTV(MAXN,N,FT,T,DELT,THET,TAU)

IMPLICIT NONE

INTEGER*4 I,N,MAXN
REAL*4 T,DELT,THET,TAU,OMEG,AMP,FT(MAXN)

AMP=10.0

OMEG=3.14159/TAU

DO 10 I=1,N
  FT(I)=0.0
10 ENDDO

IF (T.LE.TAU.OR.ABS(T-THET).LT.DELT/2.0) THEN
  FT(1)=AMP*DELT*(THET*SIN(OMEG*T)+(1.0-THET)*SIN(OMEG*(T-DELT)))
ENDIF
RETURN
END
```
**MODULE NAME:** SOLVE

**PROGRAMMER:** DOMINIC N. DALO

**DATE WRITTEN:** APRIL 1966

**OBJECTIVE:** DETERMINE THE SOLUTION VECTOR FOR A SET OF ALGEBRAIC EQUATIONS OF THE FORM

\[ [A](x) - (b) \]

WHERE \([A]\) IS A TRIDIAGONAL COEFFICIENT MATRIX

**DESCRIPTION OF VARIABLES:**

\[ A \] - NONSINGULAR TRIDIAGONAL COEFFICIENT MATRIX

\[ AL \] - LOWER DIAGONAL OF \([A]\) MATRIX

\[ AD \] - MAIN DIAGONAL OF \([A]\) MATRIX

\[ AU \] - UPPER DIAGONAL OF \([A]\) MATRIX

\[ X \] - SOLUTION VECTOR

\[ B \] - RIGHT HAND SIDE COLUMN VECTOR

---

SUBROUTINE SOLVE(MAXN,N,A,B,X,AL,AD,AU)

IMPLICIT NONE

INTEGER*4 I,N,MAXN

REAL*4 B(MAXN),X(MAXN),AL(MAXN),AD(MAXN),AU(MAXN),A(MAXN,MAXN)

SEPARATE A TRIDIAGONAL MATRIX INTO UPPER, LOWER, AND MAIN DIAGONALS

DO 10 I=1,N
IF (I.NE.N) THEN
   AL(I)=A(I+1,I)
   AU(I)=A(I,I+1)
ENDIF
10 ENDDO
AL(N)=0.0
AU(N)=0.0

CALCULATE THE SOLUTION VECTOR

CALL TRI(MAXN,N,AL,AD,AU,B)

DO 20 I=1,N
   X(I)=B(I)
20 ENDDO
RETURN
END
MODULE NAME: TRI

PROGRAMMER: DOMINIC N. DALO

DATE WRITTEN: APRIL 1986

OBJECTIVE: CALCULATE THE SOLUTION VECTOR TO A TRIDIAGONAL SYSTEM OF ALGEBRAIC EQUATIONS OF THE FORM

\[
[A\{x\} - \{b\]
\]

DESCRIPTION OF VARIABLES:

A - LOWER DIAGONAL
D - MAIN DIAGONAL
C - UPPER DIAGONAL
B - RIGHT HAND SIDE COLUMN VECTOR AND SOLUTION VECTOR

SUBROUTINE TRI(MAXN,N,A,D,C,B)

IMPLICIT NONE

INTEGER^4 I,N,MAXN

REAL^4 SCALE,A(MAXN),D(MAXN),C(MAXN),B(MAXN)

DO 10 I=2,N
SCALE=A(I-1)/D(I-1)
D(I)=D(I)-SCALE*C(I-1)
B(I)=B(I)-SCALE*B(I-1)
10  ENDDO

B(N)=B(N)/D(N)

DO 20 I=1,N-1
B(N-I)=(B(N-I)-C(N-I)*B(N-I+1))/D(N-I)
20  ENDDO

RETURN

END
**OBJECTIVE:**

DETERMINE WHETHER THE ABSOLUTE NORM OF THE DIFFERENCE BETWEEN TWO VECTORS IS WITHIN A GIVEN TOLERANCE

**DESCRIPTION OF VARIABLES:**

SUCCESS - LOGICAL VARIABLE USED TO INDICATE THAT CONVERGENCE HAS OCCURRED

DIF - DIFFERENCE BETWEEN RESPECTIVE ELEMENTS IN THE TWO VECTORS

ERR - MAXIMUM DIFFERENCE BETWEEN RESPECTIVE ELEMENTS IN THE TWO VECTORS

EPS - CONVERGENCE CRITERIA

XI - FIRST VECTOR

X2 - SECOND VECTOR

**SUBROUTINE CONVERGE(MAXN,N,X1,X2,EPS,SUCCESS)**

*IMPLICIT NONE*

LOGICAL SUCCESS

INTEGER*4 I,N,MAXN

REAL*4 EPS,ERR,DIF,X1(MAXN),X2(MAXN)

INITIALIZE THE MAXIMUM ERROR

ERR=0.0

DETERMINE THE ABSOLUTE NORM OF THE DIFFERENCE BETWEEN THE TWO VECTORS

DO I=1,N
DIF=ABS(X2(I)-X1(I))
IF (DIF.GT.ERR) THEN
ERR=DIF
ENDIF
10 ENDDO

DETERMINE IF THE ABSOLUTE NORM IS WITHIN THE CONVERGENCE CRITERIA

IF (ERR.LT.EPS) THEN
SUCCESS=.TRUE.
ENDIF
RETURN
END
OBJECTIVE: COMPUTE THE VISCOUS AND TOTAL HEAT FLUX FOR AT A GIVEN INCREMENT OF TIME

DESCRIPTION OF VARIABLES:

QV VISCOUS HEAT FLUX VECTOR
QT TOTAL HEAT FLUX VECTOR

COMMENT: THE DIMENSIONLESS VISCOUS HEAT FLUX VECTOR WAS COMPUTED BY IMPLEMENTING A FINITE DIFFERENCE EXPRESSION TO APPROXIMATE THE DIMENSIONLESS FORM OF THE FOURIER EQUATION

\[ Q \text{ viscous} \frac{dU}{dX} \]
WHERE U DIMENSIONLESS TEMPERATURE
X DIMENSIONLESS POSITION.

SUBROUTINE FLUX(MAXN,N,BJ1,UJ1,QV,QT,H)
IMPLICIT NONE
INTEGER*4 I,N,MAXN
REAL*4 H,BJ1(MAXN),UJ1(MAXN),QV(MAXN),QT(MAXN)

QV(1)=(UJ1(1)-UJ1(2))/H
QT(1)=QV(1)+BJ1(1)
DO 10 I=2,N-1
QV(I)=(UJ1(I-1)-UJ1(I+1))/(2.0*H)
QT(I)=QV(I)+BJ1(I)
10 ENDDO
QV(N)=(UJ1(N-1)-UJ1(N))/H
QT(N)=QV(N)+BJ1(N)
RETURN
END
MODULE NAME : SXMMULT
PROGRAMMER : DOMINIC N. DALO
DATE WRITTEN : OCTOBER 25, 1985

OBJECTIVE : MULTIPLY AN ENTIRE MATRIX BY A SCALAR COEFFICIENT.

DESCRIPTION OF VARIABLES :

MROW  - MAXIMUM NUMBER OF ROWS IN MATRIX
MCOL  - MAXIMUM NUMBER OF COLUMNS IN MATRIX
ROW   - ACTUAL NUMBER OF ROWS IN MATRIX
COL   - ACTUAL NUMBER OF ROWS IN MATRIX
SCALE - SCALAR WHICH IS TO MULTIPLY MATRIX
A     - MATRIX WHICH IS TO BE MULTIPLIED BY SCALAR
B     - MATRIX WHICH IS PRODUCT OF SCALAR AND MATRIX

SUBROUTINE SXMMULT(MROW,MCOL,ROW,COL,SCALE,A,B)

IMPLICIT NONE

INTEGER*4 MROW,MCOL,ROW,COL,I,J
REAL*4 SCALE, A(MROW,MCOL), B(MROW,MCOL)

DO 10 I = 1, ROW
    DO 20 J = 1, COL
        B(I,J) = SCALE*A(I,J)
    20 END DO
10 END DO
RETURN
END
OBJECTIVE: FORM THE PRODUCT OF TWO CONFORMABLE MATRICES

DESCRIPTION OF SUBROUTINE CALL PARAMETERS:

IX - MAXIMUM NUMBER OF ROWS IN MATRIX [X]
JX - MAXIMUM NUMBER OF COLUMNS IN MATRIX [X]
IY - MAXIMUM NUMBER OF ROWS IN MATRIX [Y]
JY - MAXIMUM NUMBER OF COLUMNS IN MATRIX [Y]
IZ - MAXIMUM NUMBER OF ROWS IN MATRIX [Z]
JZ - MAXIMUM NUMBER OF COLUMNS IN MATRIX [Z]
N - ACTUAL NUMBER OF ROWS IN MATRIX [X]
L - ACTUAL NUMBER OF COLUMNS IN MATRIX [X]
(MODE: SAME AS NUMBER OF ROWS IN MATRIX [Y])
M - ACTUAL NUMBER OF COLUMNS IN MATRIX [Y]
X - MULTIPLIER MATRIX
Y - MATRIX TO BE PREMULTIPLIED BY MATRIX [X]
Z - PRODUCT MATRIX OF [X] * [Y]

SUBROUTINES AND SUBPROGRAMS REQUIRED: NONE

COMMENTS:

IN ORDER TO BE ABLE TO PERFORM MATRIX MULTIPLICATION, THE TWO
MATRICES MUST BE CONFORMABLE, i.e., THE NUMBER OF COLUMNS IN
THE FIRST MATRIX MUST BE EQUAL TO THE NUMBER OF ROWS IN THE
SECOND MATRIX.

SUBROUTINE MATMUL2( SIZE1, SIZE2, SIZE3, N, L, M, X, Y, Z )

INTEGER*4 SIZE1, SIZE2, SIZE3
REAL*4 X(SIZE1,SIZE2), Y(SIZE2,SIZE3), Z(SIZE1,SIZE3)

PREMULTIPLY MATRIX [Y] BY MATRIX [X]

DO 300 I=1,N
   DO 300 J=1,M
      Z(I,J) = 0.0
   DO 300 K=1,L
      Z(I,J) = Z(I,J) + X(I,K)*Y(K,J)
   300 END DO

RETURN PRODUCT MATRIX [Z] BACK TO CALLING PROGRAM

RETURN
END
MODULE NAME : MATADD

PROGRAMMER : DOMINIC N. DALO

DATE WRITTEN : OCTOBER 25, 1985

OBJECTIVE : ADD TWO MATRICES OF EXACTLY THE SAME SIZE
AND TO RETURN THE SUM OF THE MATRICES AS WELL AS THE
ORIGINAL MATRICES BACK TO THE CALLING PROGRAM.

DESCRIPTION OF VARIABLES :

MROW - MAXIMUM NUMBER OF ROWS IN THE MATRICES
MCOL - MAXIMUM NUMBER OF COLUMNS IN THE MATRICES
ROW - ACTUAL NUMBER OF ROWS IN THE MATRICES
COL - ACTUAL NUMBER OF COLUMNS IN THE MATRICES
A - MATRIX TO BE ADDED TO MATRIX [B]
B - MATRIX TO BE ADDED TO MATRIX [A]
C - SUM OF MATRICES [A] AND [B]

SUBROUTINE MATADD(MROW,MCOL,ROW,COL,A,B,C)

IMPLICIT NONE

INTEGER*4 MROW,MCOL,ROW,COL,I,J

REAL*4 A(MROW,MCOL), B(MROW,MCOL), C(MROW,MCOL)

DO 10 I = 1, ROW
   DO 20 = 1, COL
      C(I,J) = A(I,J) + B(I,J)

20 END DO
10 END

RETURN
END